Oracle Database Admin Guide 11Gr2

* Cơ chế hoạt động của instance
* Cấu trúc bộ nhớ, tác dụng của từng thành phần
* Background process, tác dụng và các thức hoạt động của từng process
* Khi thực thi một câu lệnh, Oracle thực thi theo tuần tự như thế nào
* Khi khởi động database thì Oracle làm những gì?
* Có bao nhiều cách backup CSDL, với mỗi cách thì Oracle làm như thế nào
* Khi restore CSDL từ bản backup, Oracle làm những gì
* Tại sao phải có redo, archive. Oracle ghi những dữ liệu gì vào redo
* Thế nào là logging và nologging. Khi nào thì nologging có tác dụng
* Direct path và conventional path
* Oracle SCN được Oracle sinh ra để làm gì, nếu SCN không đồng nhất giữa các thành phần của CSDL sẽ nảy sinh vấn đề gì
* Chính sách backup nghĩa là gì? Tại sao nên có incremental backup? Oracle thực hiện incremental backup thế nào?
* Thế nào là chính sách thực thi câu lệnh, Oracle lựa chọn chính sách thực thi câu lệnh dựa trên cơ chế nào?
* Cơ chế index của Oracle. Bạn có một bảng trong CSDL, bạn phân vân có nên tạo index cho bảng hay không. Điều gì sẽ chi phối quyết định của bạn
* Giới thiệu các view cơ bản của Oracle
* Kiến trúc lưu trữ của Oracle. Thế nào là pct used, pct free. Thế nào là row change. Thế nào là row migration
* Tempory tablespace & temporary table

# Oracle Database Concept

## Oracle SCN

SCN (System Change Number): đây là một số gia tăng duy nhất trong cơ sở dữ liệu (như đồng hồ thời gian của bạn). Số SCN được tăng lên mỗi 3 giây.

* + SCN xác định một phiên bản đã commit của cơ sở dữ liệu tại một thời điểm. Oracle gán mọi transaction đã commit một SCN.
  + SCN là một dấu thời gian nội bộ cho một commited version của cơ sở dữ liệu . Các máy chủ cơ sở dữ liệu Oracle sử dụng SCN clock để đảm bảo tính nhất quán SCN transaction. Ví dụ, khi một người sử dụng commit một transaction, cơ sở dữ liệu ghi lại một SCN này và commit ghi trên redo log.
  + SCNs là quan trọng đối với transaction bởi vì nó có chức năng như một dấu thời gian đồng bộ commit transaction , ngay cả khi transaction không thành công. Nếu một transaction thay đổi dữ liệu sai hoặc không phù hợp, một quản trị viên có thể sử dụng SCN này để phối hợp thay đổi trên cơ sở dữ liệu. SCN cho transaction commit cũng có thể được sử dụng để xác định các transaction sau đó.
  + Nếu SCN không đồng nhất giữa các thành phần CSDL sẽ nảy sinh vấn đề. Database sẽ lấy SCN của control file và apply cho tất cả các thành phần khác
  + Control file lưu 1 thông tin rất quan trọng là SCN. Con số này được phát sinh và tăng liên tục theo thời gian. Oracle dựa vào con số này đề đồng bộ tất cả các file trong database như: datafile, control file, redo log file v.v...

## Oracle Service

An automatic workload management facility, called database services. Database services (services) are logical abstractions for managing workloads in Oracle Database. Services divide workloads into mutually disjoint groupings. Each service represents a workload with common attributes, service-level thresholds, and priorities.

A single service can represent an application, multiple applications or a subset of a single application. For example, the Oracle E-Business suite defines a service for each responsibility, such as general ledger, accounts receivable, order entry, and so on. A single service can be associated with one or more instances of an Oracle RAC database, and a single instance can support multiple services.

# II. Oracle Database Architecture

## Tổng quan về các thành phần trong kiến trúc Oracle

**Oracle server :**

* + Là tập hợp những file, những tiến trình và cấu trúc bộ nhớ
  + Oracle server bao gồm Oracle Instance và Oracle Database

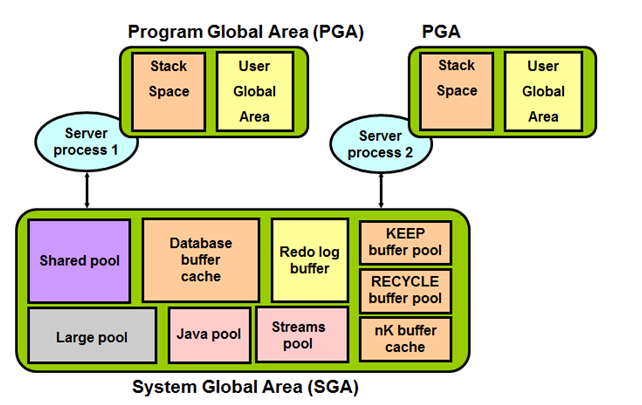
**Oracle Instance**

* + Là tập hợp của các tiến trình ngầm (background processes) và cấu trúc bộ nhớ (memory structure). Một Instance được khởi động để truy cập dữ liệu trong Oracle Database.
  + Mỗi khi một Instance được khởi động, một System Global Area (SGA) được cấp phát và các tiến trình ngầm của Oracle cũng sẽ được khởi động. Các tiến trình ngầm này sẽ thực hiện tác vụ vào/ra và quản lý các tiến trình khác của Oracle nhằm cung cấp khả năng chạy song song để thi hành tốt hơn và tin cậy hơn:
    - Pmon
    - Smon
    - DBWR
    - LogWR
    - …

**Oracle Database**

* + Là một tập hợp các file hệ thống hay còn gọi là các file Database, cung cấp các thông tin về những thiết bị lưu trữ vật lý và thông tin và Database. Những file Database được sử dụng để đảm bảo rằng dữ liệu được lưu giữ ở trạng thái nhất quán, và có thể được khôi phục lại tại thời điểm một Instance bị lỗi:
    - Spfile : memory, vị trí control file, các setting
    - Control file : vị trí datafile,redo log thông tin backup, SCN
    - Datafile
    - Redolog file
    - Archivelog file

**Kiến trúc Instance**



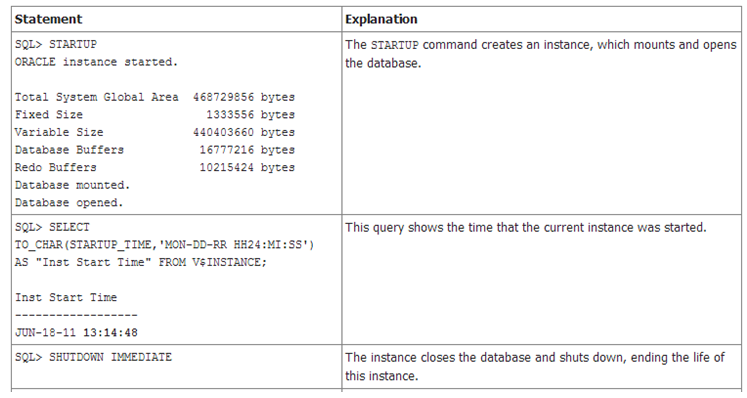
Mục đích của SGA :

* + Quản lý hoạt động của các tiến trình truy cập tới Oracle Database
  + Đọc những block dữ liệu từ đĩa
  + Làm bộ đệm cho dữ liệu \Lưu trữ kế hoạch thực thi các câu truy vấn

Mô hình Database instance

* + Có 2 dạng mô hình chính của database instance:
    - Single instance : 1 database – 1 instance
    - Rac : 1 database – n instance

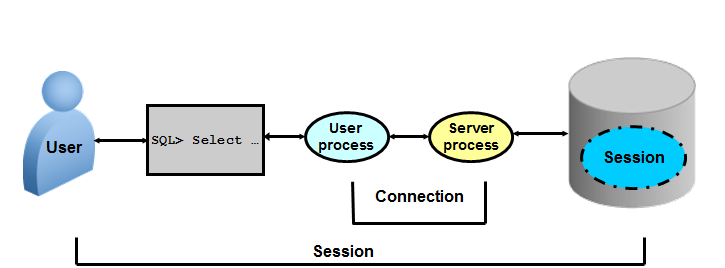
Quá trình hoạt động của instance



Oracle System Identifier (SID)

* + Là định danh duy nhất của Oracle database **instance** trên 1 host
  + Được set bởi biến môi trường ORACLE\_SID

Quá trình kết nối tới instance

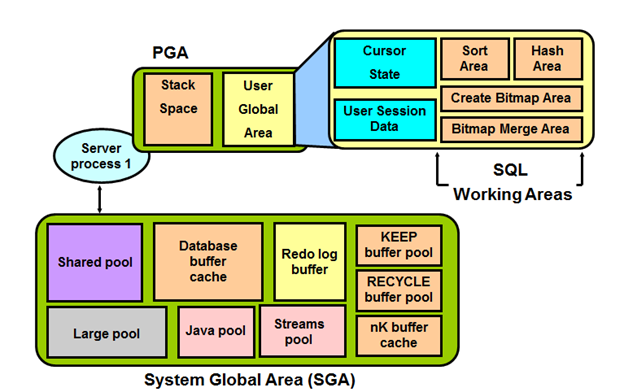


* + Khi user muốn thao tác với database thì phải tạo ra các yêu cầu ( user process), các yêu cầu này được gửi tới server và server sẽ thực hiện các yêu cầu này để thao tác với db.

## Kiến trúc bộ nhớ

Có hai cấu trúc bộ nhớ cơ bản trong một Instance:

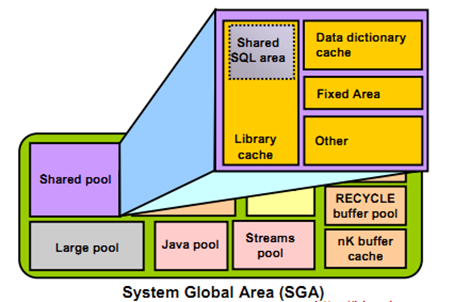
* + System Global Area (SGA): Là vùng bộ nhớ chia sẻ được sử dụng để lưu trữ dữ liệu và các thông tin điều khiển của Oracle server. Được chỉ định khi một Instance được khởi động, và là thành phần cơ bản của một Oralce Instance.
  + Program Global Area (PGA): Là một phần cấu trúc bộ nhớ lưu trữ dành cho 1 User process kết nối tới 1 Instance bao gồm dữ liệu và thông tin điều khiển cho một Server hoặc một Background process. Được chỉ định khi một Server Process được khởi động.



### Kiến trúc SGA

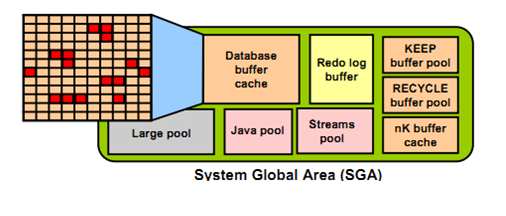
**Shared pool**

* + **Library cache**: Lưu trữ những định nghĩa về những đoạn lệnh SQL và PL/SQL vừa được thực thi gần đây nhất theo thuật giải Least Recently Used (LRU). Library cache bao gồm 2 cấu trúc là Shared SQL area và Shared PL/SQL area. Kích thước của vùng này được xác định bởi Shared pool sizing.
  + **Data dictionary cache**: Thu thập những định nghĩa được dùng gần đây nhất trên cơ sỡ dữ liệu bao gồm các thông tin về Database file, tables, indexes, columns, user, privileges,…Trong quá trình phân tích cú pháp đoạn lệnh, Server Process sẽ đọc các thông tin định nghĩa ở Data dictionary cache để lấy tên các đối tượng, xác nhận truy cập,…Kích thước của vùng này được xác định bởi Shared pool sizing.
  + **Fixed Area**: được khởi động đồng thời vs SGA, có tác dụng tính toán kích thước của shared pool hoặc là SGA



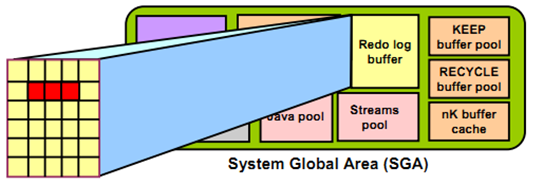
**Database Buffer Cache**

* + Lưu trữ những bản copy của **Block** dữ liệu đã được đọc từ **Data File.** Khi một đoạn SQL được thực thi, thì **Server Process** sẽ đọc các thông tin từ **Database buffer cache** để lấy các **block** dữ liệu cần thiết, điều này giúp cho tốc độ hoạt động của hệ thống sẽ cao hơn vì đọc trên cache sẽ nhanh hơn là đọc trên đĩa cứng . Nếu các block dữ liệu không có trong Database buffer cache thì Server process mới đọc dữ liệu từ data file. Database buffer cache cũng sử dụng thuật giải LRU như ở Shared Pool.



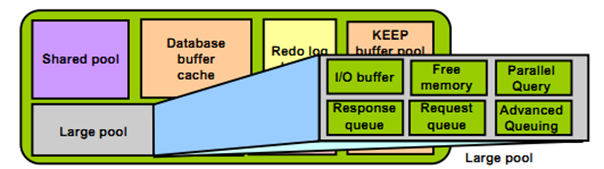
**Redo log buffer**

* + Là một bản ghi tạm thời, ghi lại tất cả những thay đổi trên các **Data Block** với mục đích chính là để phục hồi dữ liệu.
  + Được thực hiện bởi các background process.



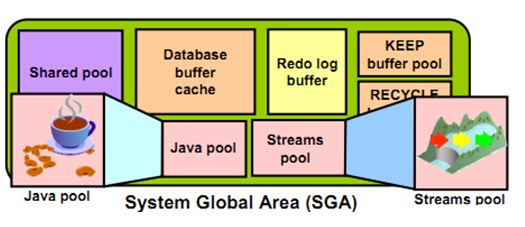
**Large Pool**

* + Cung cấp một vùng nhớ lớn, được cấp phát cho các trường hợp như:
    - Vùng bộ nhớ cho các giao dịch lớn
    - Xử lý I/O
    - Backup và recovery hệ thống



**Java Pool và Streams Pool**

* + Java Pool là vùng dùng cho các Procedure viết bằng Java.
  + Stream Pool dùng cho Oracle Stream.

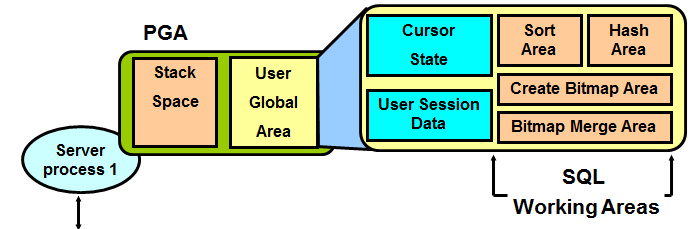


**KEEP buffer pool:** 1 thành phần database buffer cache đặc biệt để lưu trữ block dữ liệu trên bộ nhớ 1 thời gian dài.

**RECYCLE buffer pool:** 1 thành phần database buffer cache đặc biệt để tái sử dụng hoặc loại bỏ những block dữ liệu trên memory 1 cách nhanh chóng..

**nK buffer cache:** 1 hoặc nhiều database buffer cache đặc biệt được thiết kế để lưu trữ những block đặc biệt có kích thước lớn hơn kích thước mặc định

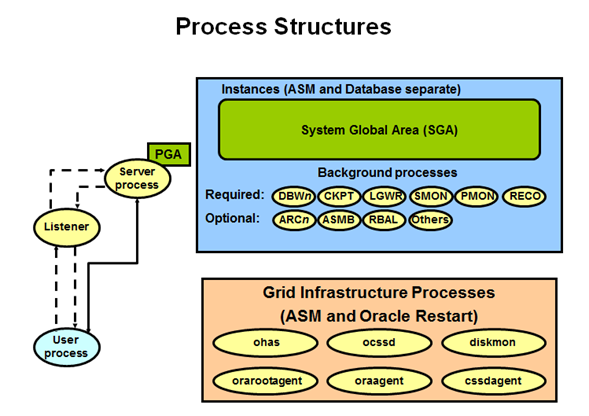
### Kiến trúc PGA



**Program Global Area (PGA)** là vùng nhớ riêng bao gồm dữ liệu và thông tin điều khiển của một Server Process. Mỗi Server Process đều có một PGA riêng biệt. PGA bao gồm:

* + Stack space : lưu trữ các biến và các mảng được xử lý trên PGA
  + User Global Area (UGA):
    - Cursor State : lưu trữ thông tin con trỏ
    - User session data : Lưu trữ thông tin điều khiển của 1 session
    - SQL working Areas : thực thi các câu lệnh truy vấn SQL :
      * Hash area: kết nối giữa các bảng bằng hash
      * Create bitmap area: sử dụng để tạo index bitmap
      * Bitmap merge area: sử dụng để tạo các kế hoạch cho index bitmap
      * Sort area: sử dụng cho các hàm gom nhóm dữ liệu như: ORDER BY and GROUP BY

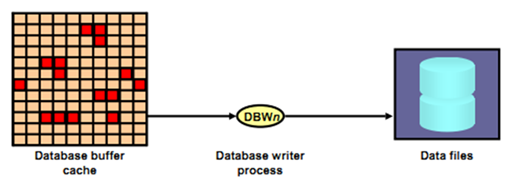
### Background process



Background process (các tiến trình nền) thực hiện các chức năng thay cho lời gọi tiến trình xử lý tương ứng. Nó điều khiển vào ra, cung cấp các cơ chế xử lý song song nâng cao hiệu quả và độ tin cậy. Tùy theo từng cấu hình mà Oracle instance có các Background process như:

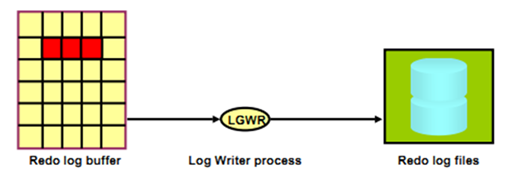
* + Database writer process (DBWn)
  + Log Writer process (LGWR)
  + Checkpoint process (CKPT)
  + System monitor process (SMON)
  + Archiver processes (ARCn)
  + Process Monitor Process ( PMON)

**Database Writer (DBWn)**



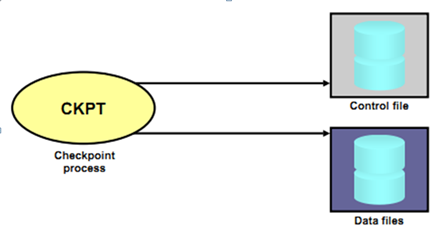
* + **Database Writer** **(DBWn)** sẽ có nhiệm vụ ghi các modified (dirty) buffer từ Database buffer xuống đĩa (Data files).
  + Mục đích: đồng bộ dữ liệu giữa database buffer cache và data files
  + Những trường hợp mà **DBWn** thực hiện công việc ghi trên:
    - Đã hết các free buffers.
    - Có quá nhiều dirty buffers.
    - Cứ mỗi 3 giây một lần.
    - Xảy ra Checkpoint.

**Log Writer Process (LGWR)**



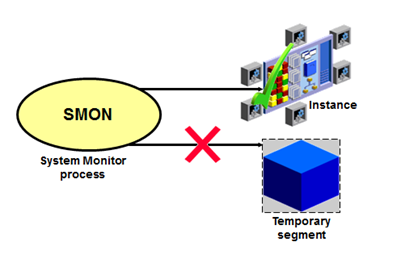
* + **Log Writer Process (LGWR)**: Có nhiệm vụ ghi các Redo log buffer xuống một Redo log file.
  + Mục đíchĐồng bộ dữ liệu, giải phóng bộ nhớ
  + Các trường hợp **LGWR** thực hiện công việc ghi trên:
    - Khi User Process thực hiện **commit** một transaction.
    - Khi Redo log buffer đã đầy.
    - Trước khi DBWn ghi dữ liệu xuống đĩa.
    - Cứ mỗi 3 giây một lần.

**Checkpoint (CKPT)**



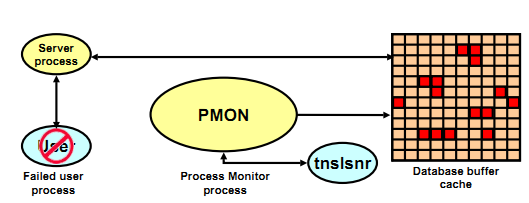
* + Checkpoint: có nhiệm vụ kiểm tra sự đồng bộ giữa các datafile và controlfile
  + Mục đích: Xác định SCN cho mỗi thao tác được thực hiện
  + Các trường hợp CKPT thực hiện công việc :
    - Mỗi khi switch log,
    - Khi Instance shutdown ( ngoài abort)
    - Default 3s checkpoint sẽ nhắc nhở việc ghi dữ liệu xuống

**System Monitor Process (SMON)**



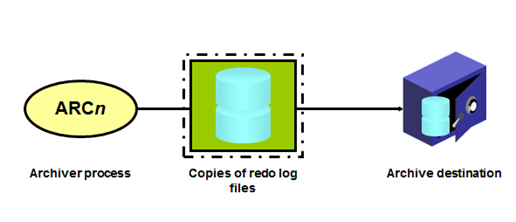
* + **SMON** có các nhiệm vụ:
    - Recovery instance khi startup nếu cần thiết.
    - Xóa các dữ liệu tạm (Clean temporary segment)

**Process Monitor Process (PMON)**



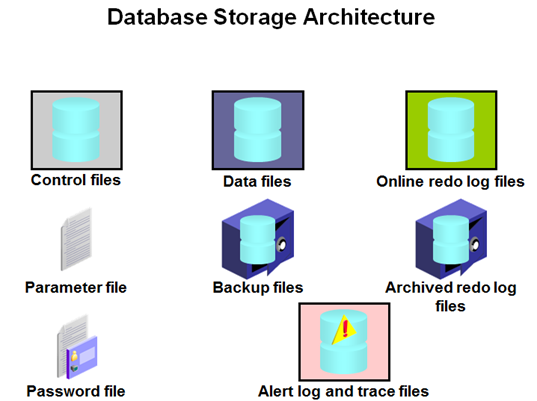
* + **PMON** có nhiệm vụ recovery 1 process khi thất bại:
    - Rollback lại transaction.
    - Tự động đăng ký Database services với Listeners
    - Giải phóng blocks và các tài nguyên khác.
    - Ví dụ : nó thiết lập lại (reset) trạng thái của các bảng đang thực hiện trong transaction, giải phóng các locks trên bảng này, và huỷ bỏ process ID của nó ra khỏi danh sách các active processes.

**Archiver Processes (ARC*n*)**



* + Sử dụng để lưu trữ redolog để phòng khi cần recover database
  + Thực hiện: Copy redo log file xuống archive destination mỗi khi switch log xảy ra

## **Kiến trúc lưu trữ của Oracle**



### Parameter file

* + File chứa những thông tin cấu hình của 1 db và địa chỉ của controlfile.
  + Ví dụ như: Các vùng nhớ khởi tạo được cấp phát bao nhiêu G, các control\_file nằm ở đâu, bao nhiêu process, session v.v…
  + Parameter có 2 loại là SPFile (file Binary) và Pfile (file text), nhiệm vụ của 2 file này là giống nhau. Mặc định, lúc khởi động DB, Oracle sẽ ưu tiên sử dụng spfile trước, nếu không có thì sẽ chuyển sang dùng pfile ( pfile có thể lưu trữ địa chỉ của spfile)

### Control Files

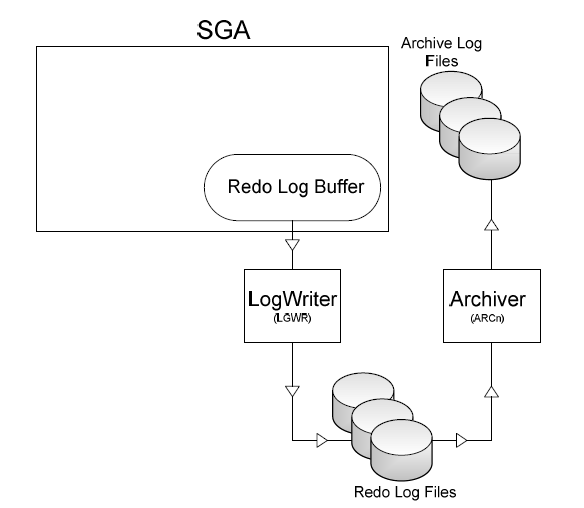
* + Mỗi Oracle database đều có ít nhất một control file. Control file chứa các mục thông tin quy định cấu trúc vật lý của database như:
    - Tên của database.
    - Tên và nơi lưu trữ các datafiles hay redo log files.
    - Các thông tin về backup của database
  + Control file cũng được sử dụng đến khi thực hiện khôi phục lại dữ liệu.

### Datafiles

* + Mỗi một Oracle database đều có thể có một hay nhiều datafiles. Các database datafiles chứa toàn bộ dữ liệu trong database. Các dữ liệu thuộc cấu trúc logic của database như tables hay indexes đều được lưu trữ dưới dạng vật lý trong các datafiles của database.
  + Một số tính chất của datafiles:
    - Mỗi datafile chỉ có thể được sử dụng trong một database.
    - Datafiles cũng còn có một số tính chất cho phép tự động mở rộng kích thước mỗi khi database hết chỗ lưu trữ dữ liệu.
    - Một hay nhiều datafiles tạo nên một đơn vị lưu trữ logic của database gọi là tablespace. Một datafile chỉ thuộc về một tablespace.
    - Dữ liệu trong một datafile có thể đọc ra và lưu vào vùng nhớ bộ đệm của Oracle. Ví dụ: khi một user muốn truy cập dữ liệu trong một table, trong trường hợp thông tin yêu cầu không có trong cache memory hiện thời, nó sẽ được đọc trực tiếp từ các datafiles ra và lưu trữ vào trong bộ nhớ.

### Redo Log Files - archivelog

* + REDO LOG file là một phần quan trọng của quá trình Oracle recovery. Nếu không có nó thì 1 instance ko thể recovery bị crash hoặc không thể khởi động 1 cách nhất quán
  + Chức năng chính của redo log là ghi lại tất cả các thay đổi đối với dữ liệu trong database. Redo log files được sử dụng để bảo vệ database khỏi những sự cố. Oracle cho phép sử dụng cùng một lúc nhiều redo log gọi là multiplexed redo log để cùng lưu trữ các bản sao của redo log trên các ổ đĩa khác nhau
  + Khi mỗi Oracle block thay đổi, Oracle tạo ra các vector thay đổi cho nó. Mỗi vetor này được cọi là REDO entry hoặc REDO records. Những sự thay đổi này sẽ được server process ghi vào redo log buffer trên PGA. Redo log buffer ghi xuống online redo log vào 1 thời điểm nào đó bởi LGWR
  + Redolog thường gồm ít nhất 2 đơn vị, ghi lại những thao tác tác động lên DB, khi 1 redolog đầy sẽ switch sang redo log bên cạnh. Khi switch sang 1 redolog đã có dữ liệu, dữ liệu này sẽ được ghi xuống archivelog để lưu trữ



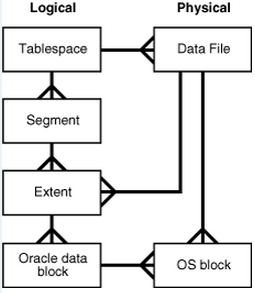
### Backup file

* + Lưu trữ các file backup của DB

**Password file**

* + Lưu trữ thông tin account – password của DB

## Cấu trúc logic databse và các vấn đề liên quan



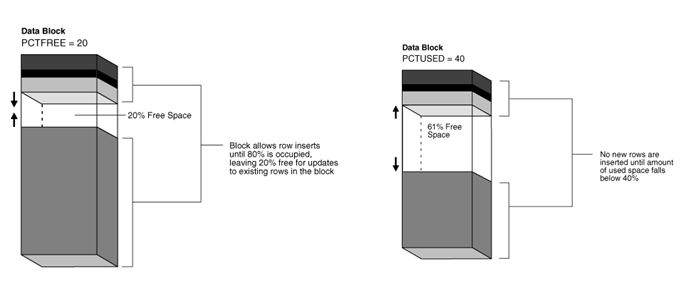
Cấu trúc logic của Oracle database bao gồm các đối tượng tablespaces, schema objects, data blocks, extents, và segments.

### Tablespaces

* + Một database có thể được phân chia về mặt logic thành các đơn vị gọi là các tablespaces. Tablespaces thường bao gồm một nhóm các thành phần có quan hệ logic với nhau.
  + Mỗi tablespace có thể được tạo nên, bởi một hoặc nhiều datafiles. Kích thước của database cũng có thể xác định được bằng tổng kích thước của các tablespaces của nó.
  + **Phân loại:** Một database gồm có ít nhất một tablespace SYSTEM, đây là nơi lưu trữ thông tin hệ thống,ngoài ra còn có các tablespace khác nơi chứa dữ liệu của các user Non-SYSTEM tablespace.
    - Tablespace SYSTEM
      * Sẽ tự động được tạo khi database tạo.
      * Có trong tất cả các database dùng cho hoạt động của database
      * Chứa thông tin về các data dictionary view,các định nghĩa của store procedure,pakage và các database trigger.
      * Chứa SYSTEM **Undo segment**.
      * Chứa SYSTEM **rollback segment**
  + Non – SystemTablespace
    - Có thể lưu trữ rollback segment, temporary segment, data segment, index segment
    - Giúp cho quản trị database linh hoạt hơn.

### Pct used, pct free

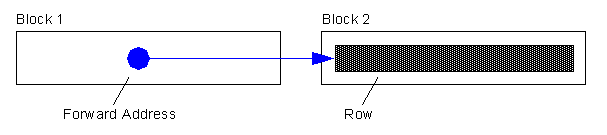
* Freespace: Không gian cấp phát cho việc insert/update trong tương lai, ảnh hưởng bởi giá trị của 2 tham số là PCTUSED và PCTFREE
* Data Row chứa dữ liệu
* Khi create/alter bất kỳ table/index nào, Oracle sẽ sử dụng 2 tham số đề điều khiển không gian
  + PCTFREE: Số % dành riêng cho việc update các dữ liệu đã có trong tương lai
  + PCTUSED: Số % của không gian nhỏ nhất đã được sử dụng cho việc insert data mới, giá trị này xác định khi nào thì các blocks sẽ được đưa trở lại vào trong FREELIST
  + FREELIST: Cấu trúc xác định mà Oracle sử dụng để maitains 1 danh sách các block free hiện có.



* Tham số PCTFREE xác định số % nhỏ nhất (không gian) của 1 data block được dành riêng cho việc update những row đã có trong block đó.
* Ví dụ:
  + Ta xác định 20% là giá trị của PCTFREE trong câu lệnh CREATE TABLE, thì có nghĩa, 20% của từng data block trong table segment sẽ được dành riêng cho việc update các row đã có bên trong từng block.
  + Block chỉ được phép insert dữ liệu khi dữ liệu khi % block giảm xuống nhỏ hơn PCTUSED

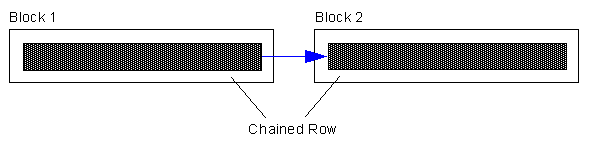
### Row Migration

* Trong trường hợp : 1 row đã được lưu overhead vào trong 1 data block, tuy nhiên, khi insert dữ liệu vào thì free space đã đầy. Vì thế, Oracle sẽ (dịch chuyển) data của toàn bộ row này sang 1 data block mới, với data block cũ, Oracle sẽ giữ lại 1 row piece – cũng chính là rowID để trỏ tới block mới.
* Đối với trường hợp Row Migration, việc Full Table Scan không bị ảnh hưởng (I/O increase), là bởi vì Forward address sẽ bị bỏ qua, tuy nhiên, Row Migration lại làm ảnh hưởng đến việc đọc Index. Bởi vì, Index sẽ nói rằng : “Đến file X, block Y, slot Z…để tìm row này”, và vì thế, khi nhận được thông điệp trên, ta lại mất thêm 1 I/O physical hoặc logical để tìm row này. (Physical I/O, Logical I/O đề cập ở phần khác).



### Row Chaining

* Trong nhiều trường hợp, data cho 1 row quá lớn để lưu trong 1 single data block, do vậy, Oracle sẽ lưu data của row này vào trong 1 hoặc nhiều chained data block (các data block móc nối). Lờy ví dụ, nếu ta sử dụng 1 data block size 4KB cho Database, và cần insert 1 row với size 8KB, Oracle sẽ sử dụng 3 blocks để lưu lại data trong row. Row chaining xảy ra với các trường hợp:
  + Rowsize vượt quá data block size
  + Table có LONG và LONGRAW columns
  + Table có trên 255 column



# III. Hoạt động của Database

## Các view cơ bản của Oracle

### Instance/Database

* V$DATABASE:  hiển thị thông tin về cơ sở dữ liệu từ các tập tin control file.
* V$INSTANCE:  hiển thị trạng thái của instance hiện tại.
* V$PARAMETER:  hiển thị thông tin về các tham số khởi tạo hiện đang có hiệu lực trong phiên làm việc. Một phiên làm việc mới được thừa hưởng giá trị tham số từ các giá trị sử dụng hoàn toàn hiển thị bởi V $ SYSTEM\_PARAMETER.
* V$SPPARAMETER: hiển thị thông tin nội dung của các tập tin tham số máy chủ.
* V$SYSTEM\_PARAMETER:  hiển thị thông tin về các tham số parameter hiện đang có hiệu lực. Một session mới được thừa hưởng giá trị tham số từ các giá trị instance mở rộng.
* V$PROCESS: chứa thông tin về các tiến trình đang hoạt động
* V$BGPROCESS: hiển thị thông tin về các tiến trình nền(background processes)
* V$PX\_PROCESS\_SYSSTAT: chứa thông tin về các session chạy thực hiện song song
* V$SYSTEM\_EVENT: chứa thông tin về tổng số chờ đợi cho một sự kiện.

### Disk

* + V$DATAFILE:  chứa thông tin datafile từ các tập tin control file.
  + V$FILESTAT: hiển thị số lượng vật lý đọc và ghi được thực hiện và tổng số single-block và multi-block I / O được thực hiện.
  + V$LOG\_FILE: chứa thông tin về redo log file.
  + V$LOG\_HISTORY: chứa log thông tin đăng nhập lịch sử từ control file.
  + V$DBFILE: chứa thông tin datafile từ các tập tin control file.
  + V$TEMPFILE: hiển thị thông tin tempfile.
  + V$TEMPSEG\_USAGE: mô tả chi tiết sử dụng temporary segment (Bảng chứa dữ liệu tạm thời)
  + V$SEGMENT\_STATISTICS:  hiển thị thông tin về số liệu thống kê segment-level statistics. Tham số tĩnh.

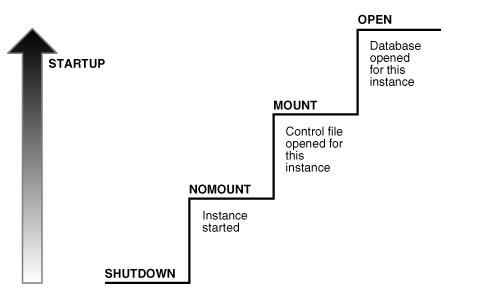
### Memory

* + V$BUFFER\_POOL\_STATISTICS:  hiển thị số liệu thống kê về tất cả tham số về vùng nhớ đệm buffer pools.
  + V$LIBRARYCACHE: chứa số liệu thống kê tham số về hiệu suất bộ nhớ cache thư viện và hoạt động.
  + V$SGAINFO: hiển thị thông tin về kích thước SGA, bao gồm các kích thước của các thành phần khác nhau SGA, kích thước hạt, và bộ nhớ miễn phí.
  + V$PGASTAT:  cung cấp thông tin về PGA thống kê sử dụng bộ nhớ cũng như thống kê về quản lý bộ nhớ PGA tự động khi nó được kích hoạt.

### Contention

* + V$LOCK: liệt kê các lock hiện đang nắm giữ cơ sở dữ liệu Oracle và yêu cầu requests lock
  + V$UNDOSTAT: hiển thị một biểu đồ số liệu thống kê cho thấy hệ thống làm việc như thế nào. Số liệu thống kê bao gồm tiêu thụ undo space, các transaction đồng thời , và chiều dài của các truy vấn được thực hiện trong mọi trường hợp.
  + V$WAITSTAT: thông số block contention. Bảng này chỉ được cập nhật khi time statistic được kích hoạt.

## Startup và Shutdown Database



Database chỉ có thể được khởi động bởi SYSDBA hoặc SYSOPER

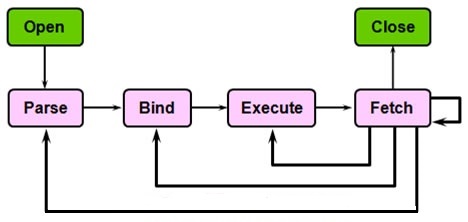
Bao gồm 3 bước :

* + Nomount
  + Instance đã được khởi động nhưng chưa được kết nối vs db
    - Oracle sẽ tìm đến spfile ( hoặc pfile nếu được chỉ định)
    - Đọc parameter file để cài đặt các tham số
    - Khởi động các background process cần thiết
    - Mở alertlog và trace file để ghi lại log về các tham số đã khởi động
  + Mount
    - Instance đã được kết nối với db bằng cách đọc control file
    - DB chưa được mở, chỉ có DBA mới có quyền truy cập
    - Oracle sau khi đọc spfile ( hoặc pfile) xác định được vị trí control file và đọc nó để có tên và vị trí của các datafile và redolog file
  + Open
  + DB đã mở, người dùng hợp lệ có thể thao tác vs dữ liệu trên db thông qua instance
  + Oracle open datafile và online redo logfile

## **Query SQL**

### **Kế hoạch thực thi 1 câu lệnh trong Oracle**

Oracle Server xử lý câu lệnh SQL theo trình tự chính sách thực thi các bước sau:



* **Open**
  + Ngầm định khai báo và khởi tạo Cursor cho câu lệnh SQL
* **Parse**
  + Syntatic: Kiểm tra cú pháp
  + Semantic : Kiểm tra đối tượng (object)
  + View merging: Rewrite lại câu lệnh dựa vào các based table thay vì sử dụng view
  + Statement Transformation : Rewirte lại sự biến đổi của câu lệnh để phân tích thành những câu đơn giản hơn.
  + Optmization : Tối ưu hóa câu lệnh
  + QEP Generation : Query Evulation Plan : đánh giá kế hoạch cho câu lệnh
* **Bind**
  + Tìm và gán giá trị cho các [**bind-variable**](http://www.vietpace.com/kienthuc/VietPace_toiuu_caulenh_Oracle_SQL_Phan1.html#_Giải_nghĩa_từ) nếu có
* **Execute**
  + Thực thi các bước mô tả trong “sơ đồ thực thi câu lệnh SQL”
* **Fetch**
  + Chuyển kết quả về nơi gọi thực thi lệnh
  + Fetch ở đây có thể lặp lại nhiều lần do tham số limit của nó ( giới hạn xử lý mỗi lần)
* **Close**
  + Ngầm định đóng Cursor cho câu lệnh

### Excution plan

EXPLAIN PLAN

SET statement\_id = 'ex\_plan2' FOR

SELECT last\_name FROM employees

WHERE last\_name LIKE 'Pe%';

SELECT PLAN\_TABLE\_OUTPUT

FROM TABLE(DBMS\_XPLAN.DISPLAY(NULL, 'ex\_plan2','BASIC'));

----------------------------------------

| Id | Operation | Name |

----------------------------------------

| 0 | SELECT STATEMENT | |

| 1 | INDEX RANGE SCAN| EMP\_NAME\_IX |

----------------------------------------

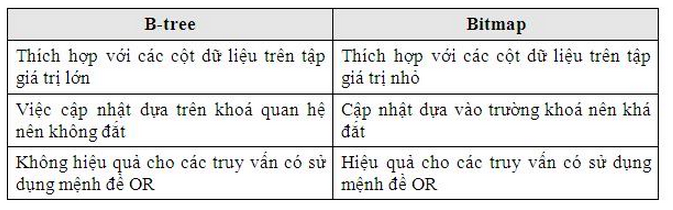
### Cơ chế index của Oracle.

Index là những cấu trúc tùy chọn liên quan đến bảng và cột.

* + Có thể tạo index trên một hoặc nhiều cột của một bảng để tăng tốc độ thực hiện câu lệnh SQL trên bảng đó.
  + Các index này sẽ giúp bạn xác định vị trí thông tin nhanh hơn. Index là cách thức chính của việc giảm đĩa I / O khi được sử dụng đúng cách.
  + Quyết định index đối với 1 bảng dựa vào yêu tố
    - Tạo Index nếu kết quả nhỏ hơn 15% tổng số row của 1 bảng lớn
    - Tăng hiệu năng cho phép join bảng ( bằng cách index cột dùng để join)
    - Không index cho bảng nhỏ
  + Không index tại cột kiểu LONG và LONGRAW

Các loại index chính:

* + B-Tree indexes
    - Btree Index được dùng để giúp truy vấn các câu truy vấn dạng Insert, Update, delete.
    - B-tree là loại index mặc định - nếu tạo ra một index mà không xác định bất cứ điều gì, thì đó là một index B-tree.
  + Bitmap indexes
    - Là 1 index được dùng để làm việc với những trường có dữ liệu rời rạc, với số lượng ít các giá trị khác nhau (tức mức độ lặp lại ở trong trường này thường là lớn)
    - Được sử dụng khI Mức độ dữ liệu trùng lặp lớn: trong oracle thì quyết định là: distinct val/ total val < 1% thì dùng bitmap index : Tức nếu số giá trị rời rạc của 1 cột trong bảng trên tổng số dòng của bảng mà nhỏ hơn 1% thì ta dùngBit map index.
    - Ví dụ trường giới tính: distinct val = 2 <số giá trị riêng biệt> và Total val = 1000 dòng. 2/1000 < 1% dùng bitmap.
    - Không hoặc là ít thao tác update hoặc insert lên bảng dữ liệu.
    - Bảng có rất nhiều cột



* + Bitmap join indexes
    - Bitmap join index cho tham gia giữa các bảng (2+). Tạo Bitmap join index được xác định trên một bảng duy nhất. Nó lưu trữ các kết quả của sự join.
  + Function-based indexes
    - Có thể tạo fuction-based indexes cho những truy vấn sử dụng các function đối với các trường:
      * Upper(abc)
      * 12 \*salary

## Backup & Recover Database

### Backup

Data backup là việc tạo ra các bản sao của dữ liệu gốc, cất giữ ở một nơi an toàn. Và lấy ra sử dụng (restore) khi hệ thống gặp sự cố. Sao lưu (backup) dữ liệu là cách tốt nhất hiện nay để bảo vệ dữ liệu

Mục địch của việc backup-restore dữ liệu này là để đưa hệ thống trở lại trạng thái trước khi gặp sự cố. Nguyên nhân của sự cố gây ảnh hưởng đến dữ liệu có thể thuộc một trong 2 dạng chính sau:

* + - Nguyên nhân khách quan: Sự cố xảy ra ngoài ý muốn, con người không thể biết trước được, thường là các thảm họa (VD: thiên tai, cháy nổ,…). Do đó cần cất giữ bản sao ở xa bản chính.
    - Nguyên nhân chủ quan: Sự cố xảy ra do những thao tác không chính xác của con người (ví dụ: lỗi phần cứng, lỗi phần mềm, thao tác nhầm…). Do đó cần cất giữ bản sao ở vị trí sao cho thuận lợi cho việc phục hồi dữ liệu, không nhất thiết phải lưu trữ ở nơi xa bản chính.

Incremental backup là chỉ backup những block thay đổi so với bản backup trước. Backup kiểu incremental chia làm 2 loại:

* + - Cumulative - thay đổi so với bản backup level 0 bản full
    - Differential - thay đổi so với bản backup trước đó

Mục đích chính của incremental backup:

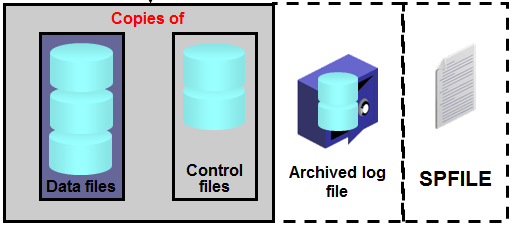
* + - Tiết kiệm thời gian và dung lượng backup.
    - Restore nhanh hơn so với Full backup. Tuy nhiên khi Restore cần đủ file:1 File Full backup lần gần nhất và tất cả các File Incremental backup từ thời điểm Full backup đến thời điểm cần restore.

Thực hiện incremental backup:

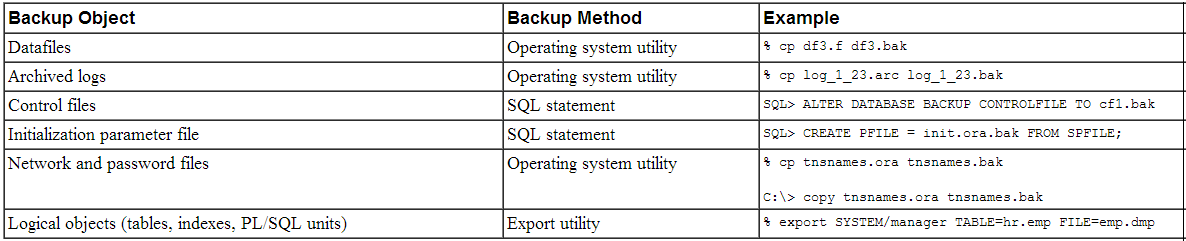
* + - Đầu tiên ta phải có một bản backup full level 0.
    - Ngày tiếp theo tạo bản backup incremental backup tất cả các block thay đổi so với bản backup level 0 ban đầu.
    - Ngày thứ 2 tạo bản backup incremental backup tất cả block thay đổi so với bản backup level 1 của ngày thứ 1.

**Các cách backup DB:**

* + Export và import : data pump
    - Oracle export dữ liệu ra file .dmp
    - File .dmp có thể được lưu trữ như 1 file backup cho db
  + Backup bằng cách sử dụng RMAN
    - * Oracle copy toàn bộ các data file và control file, có thể bao gồm archivelog file và spfile thành 1 hoặc nhiều file, lưu trữ ở vùng FRA. Khi cần recover thì mang ra sử dụng



* + User-manged backup
    - Phương thức này tương đối thủ công, thực hiện do DBA là chính.
    - Chủ yếu thực hiện cho những DB có dung lượng rất lớn



* + - Phương án hay sử dụng nhất : đóng băng DB ( suspended mode)
      * Nguyên lý hoạt động : đóng băng DB ở suspended mode, sau đó, di chuyển con trỏ dữ liệu sang 1 vùng mới và giải băng cho DB. DB hiện trạng = bản đóng băng + vùng mới.

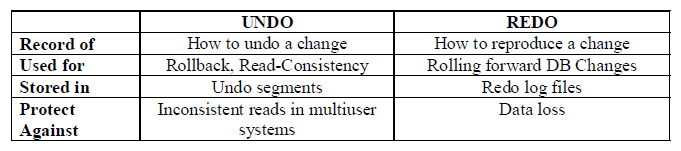
### Restore DB

Khi muốn restore DB thì DB phải ở chế độ mount

* + Ở chế độ nomount : DB sẽ đọc spfile. Nếu spfile cần restore, thì cần chỉ định file spfile cần restore. DB sẽ đọc file này và tìm tới control file
  + Ở chế độ mount: DB sẽ đọc control file, nếu cần thì restore control file chỉ định.
  + Sau đó : khi được chỉ định file( hoặc thời gian) cần restore : DB sẽ kiểm tra lại SCN của file đó,

### Undo

* + Khi các thao tác DML như: insert/ update / delete thực sự tạo thay đổi cho data block, mặc dù chưa commit thao tác. Để chắc chắn tính nguyên vẹn của database Oracle ghi dữ liệu cũ vào UNDO để có thể rollback lại. Còn các thao tác với dữ liệu ghi vào REDO log để re-play lại thao tác đó nếu như nó thất bại



### Automatic Recovery

Nếu database bị crash hoặc shutdown đột ngột thì khi khởi động lại thì Oracle thực hiện công việc recovery. Công việc này gồm 2 pha chính :

* + Rollforward phase:
    - Trong pha này, Oracle sẽ kiểm tra lại toàn bộ những transaction ở online redolog với điểm xuất phát là checkpoint gần nhất.
    - Checkpoint này là thời điểm nhất quán của dữ liệu gần nhất
    - Điều này xảy ra vs tất cả các transaction:
      * Transaction đã được commit thì tạo checkpoint cho nó
      * Transaction chưa được commit thì chuyển tới pha 2
  + Rollback phase
    - Sử dụng undo segment ( hoặc undo tablespace) để đưa các transaction chưa được commit về thời điểm nó bắt đầu

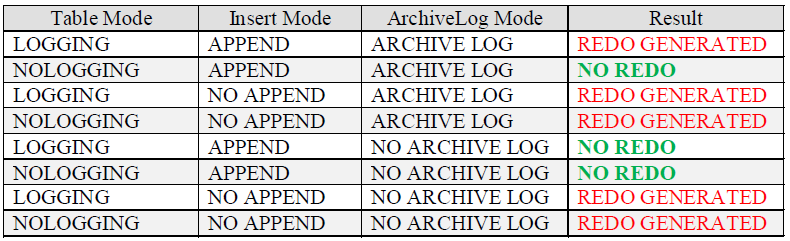
### Logging và nologging

Tuy việc tạo nên REDO LOG file là một phần quan trọng của quá trình Oracle recovery nhưng vẫn có khả năng có những redo log file được tạo nên từ những thao tác không mong muốn

Luật quan trọng nhất với dữ liệu là không được phép đặt nó trong trình trạng không thể recovery. Nhưng điều luật này đồng nghĩa với việc không thể sử dụng các option để rút ngắn thời gian hoặc gia tăng hiệu năng của hệ thống

Oracle cung cấp cho người sử dụng khả năng giới hạn ghi log vào bảng và indexes bằng NOLOGGING mode.

* + NOLOGGING có tác động mạnh đến việc recovery.
  + Đặc điểm chính cần chú ý:
    - NOLOGGING được thiết kế để insert 1 lượng dữ liệu lớn(bulk data)
    - Nếu dữ liệu không được ghi log thì nó không thể được recover, dữ liệu nên được backup sau khi chỉnh sửa.
  + Các trường hợp:



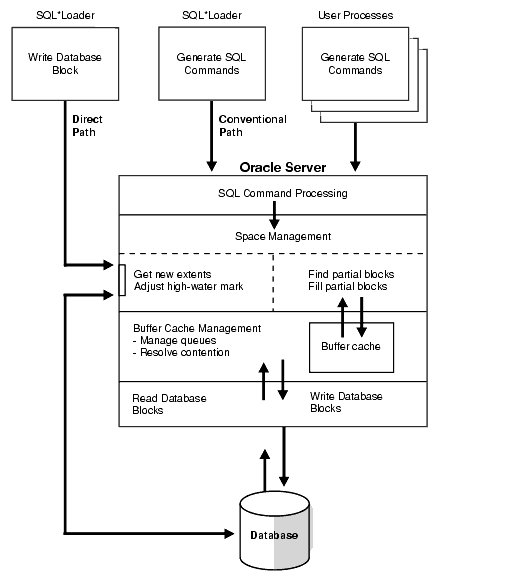
Trong một số trường hợp khi ở chế độ NOLOGGING những câu lệnh vẫn ghi log:

* + CREATE TABLE ... AS SELECT
  + CREATE INDEX.
  + UPDATE/INSERRT/DELETE

# IV. Oracle Database Feature

## Direct path và conventional path

SQL loader cung cấp 2 phương thức để cập nhật dữ liệu

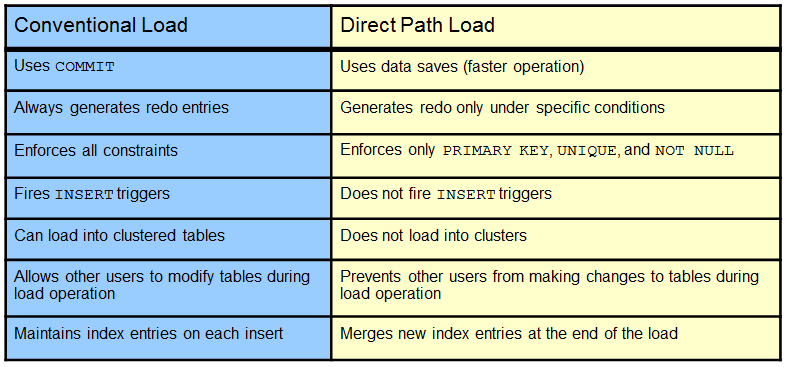


* + Conventional path thực thi câu lệnh insert tới các bảng trong 1 Oracle DB:

INSERT INTO TABLE T PARTITION (P) VALUES ...

* + Direct path ghi trực tiếp lên các datablock trên datafile

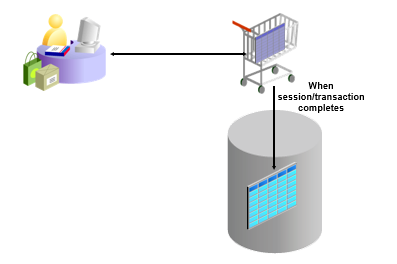
LOAD INTO TABLE T PARTITION (P) VALUES ...



## Tempory table & temporary tablespace

### Temporary table

* + Temporary Table là một bảng chứa dữ liệu mà chỉ tồn tại trong thời gian của một giao dịch hoặc phiên . Dữ liệu trong một bảng tạm thời là trong phiên giao dịch, có nghĩa là mỗi phiên chỉ có thể xem và sửa đổi dữ liệu riêng của mình .
  + Temporary table rất hữu ích trong các ứng dụng một tập hợp kết quả được đếm.
  + Ví dụ : một giỏ mua hàng trong một ứng dụng trực tuyến có thể là một bảng tạm thời. Mỗi mục được đại diện bởi một hàng trong bảng tạm thời. Bởi vì bảng tạm thời được định nghĩa cố định , nên có thể tạo index cho Chỉ số tạo ra trên bảng tạm thời cũng chỉ là tạm thời . Các dữ liệu trong các chỉ số có cùng một phiên , phạm vi giao dịch như các dữ liệu trong bảng tạm thời .



### Temporary tablespace

* + Tempory Tablespace được sử dụng để chứa các tempory table trong thời gian của các phiên giao dịch.
  + Temporary tablespaces sử dụng lưu trữ:
    - Intermediate sort results: nếu dữ liệu truy vấn trong PGA quá nhiều thì nó có thể được lưu trữ tại đây và lấy dần ra
    - Temporary tables and temporary indexes
    - Temporary LOBs
    - Temporary B-trees
  + Mặc định, chỉ có Temp tablespace duy nhất được tạo ra khi cài đặt một Oracle mới. Nhưng có thể tạo thêm tablespace bổ sung

## Cluster table

* + Ý tưởng: ghép các table thường xuyên được sử dụng để truy vấn cùng nhau thành 1 cluster
  + Sử dụng cột dùng để join các bảng thành cluster key
  + Đọc thêm dữ trong Tài liệu của hãng

## Quản lý Oracle Resources

* + Ý tưởng chính : cấp quota cho các resource consumer group
  + Đọc thêm trong tài liệu

## Database Link

Ý tưởng: Sử dụng 1 đường link để kết nối từ database này sang database khác

SQL> CREATE [PUBLIC] DATABASE LINK <link\_name>

CONNECT TO <user\_name>

IDENTIFIED BY <password>

USING '<service\_name>';