Class 1

- *Basic data warehousing sessions*
- *Slowly changing dimensions*
- *Data marts*
- *Normalization Denormalization*
- *Primary Key, Foreign Key*

Class 2

1: Introduction - Architecture
2: Retrieving Data Using the SQL SELECT Statement

Class 3

3: Restricting and Sorting Data
4: Using Single-Row Functions to Customize Output

Class 4

5: Reporting Aggregated Data Using the Group Functions
6: Displaying Data from Multiple Tables

Class 5

7: Using Subqueries to Solve Queries
8: Using the Set Operators

Class 6

9: Manipulating Data
10: Using DDL Statements to Create and Manage Tables

Class 7

11: Creating Other Schema Objects
12: Managing Objects with Data Dictionary Views

Class 8

13: Controlling User Access
14: Managing Schema Objects

Class 9

15: Retrieving Data Using Subqueries
16: Introduction to PL/SQL

Class 10

17: Declaring PL/SQL Variables
18: Writing Executable Statements

Class 11

19: Writing Control Structures
20: Handling Exceptions

Class 12

21: Creating Stored Procedures
22: Performance tuning

Class 13

Questions and Interview preparation

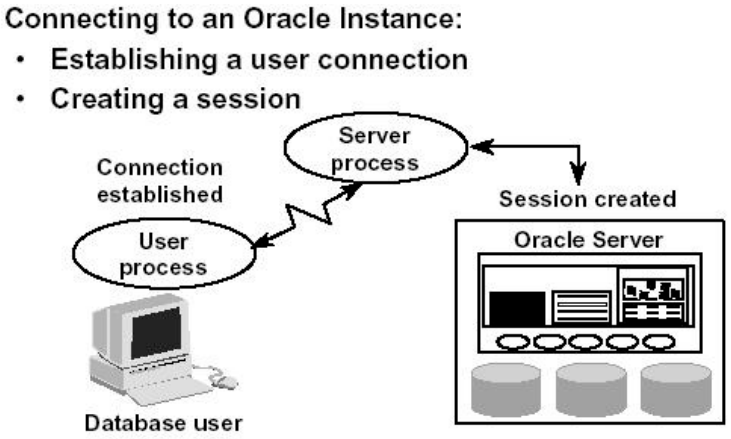
Oracle Content

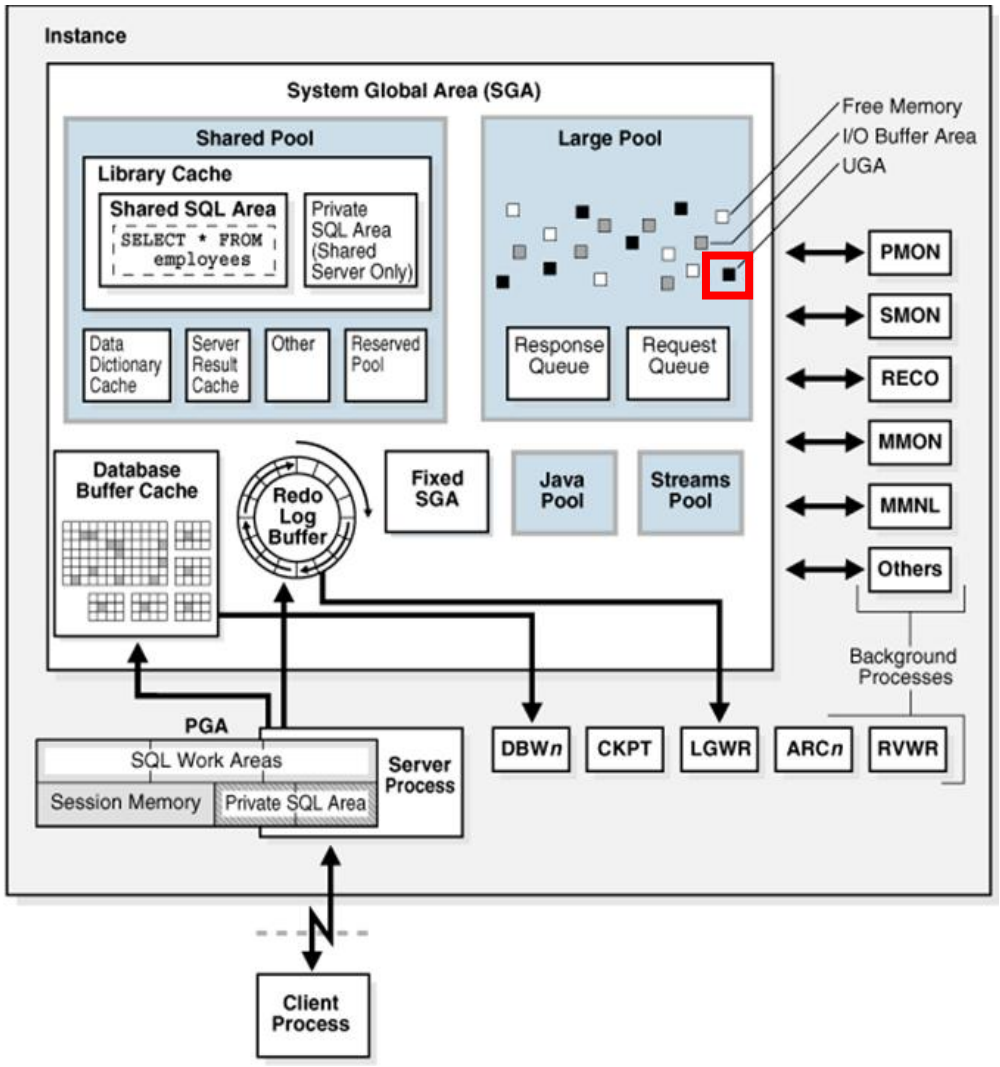
Stay Ahead

Stay Ahead

Oracle PL/SQL – Class 1

Oracle Architecture





Dedicated server (Single user) & Shared Server (Multiple users)

Sessions: When a user connects to an Oracle server, this is termed a session. The **User Global Area** is session memory and these memory structures are described later in this document. The session starts when the Oracle server validates the user for connection. The session ends when the user logs out (disconnects) or if the connection terminates abnormally (network failure or client computer failure).

A user can typically have more than one concurrent session, e.g., the user may connect using SQLPlus and also connect using Internet Developer Suite tools at the same time. The limit of concurrent session connections is controlled by the DBA.

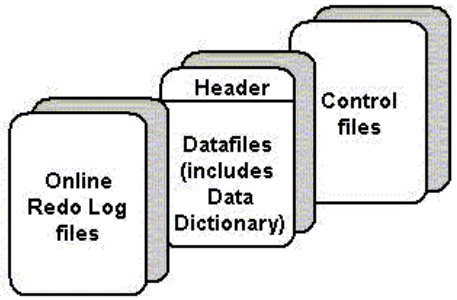
If a system users attempts to connect and the Oracle Server is not running, the system user receives the **Oracle Not Available** error message.

Oracle Architecture

Physical Structure

The physical structure includes three types of files:

- **Control files**
- **Datafiles**
- **Redo log files**



- **Data files** – these contain the organization's actual data.
- **Redo log files** – these contain a chronological record of changes made to the database, and enable recovery when failures occur.
- **Control files** – these are used to synchronize all database activities and are covered in more detail in a later module.

Memory Management and Memory Structure

Automatic memory management:

- DBA specifies the target size for instance memory.
- The database instance automatically tunes to the target memory size.
- Database redistributes memory as needed between the SGA and the instance PGA.

Automatic shared memory management:

- This management mode is partially automated.
- DBA specifies the target size for the SGA.
- DBA can optionally set an aggregate target size for the PGA or managing PGA work areas individually.

Manual memory management:

- Instead of setting the total memory size, the DBA sets many initialization parameters to manage components of the SGA and instance PGA individually.

The memory structures include three areas of memory:

- **System Global Area (SGA)** – this is allocated when an Oracle Instance starts up.

The SGA is a read/write memory area that stores information shared by all database processes and by all users of the database (sometimes it is called the Shared Global Area).

  - This information includes both organizational data
  - The SGA is allocated in memory and virtual memory.
  - The size of the SGA can be established by a DBA by assigning a value to the parameter SGA_MAX_SIZE in the parameter file—this is an optional parameter.

- **Program Global Area (PGA)** – this is allocated when a Server Process starts up.

A PGA is a non shared memory region that contains data and control information exclusively for use by an Oracle process.

  - A PGA is created by Oracle Database when an Oracle process is started.
  - One PGA exists for each Server Process and each Background Process.

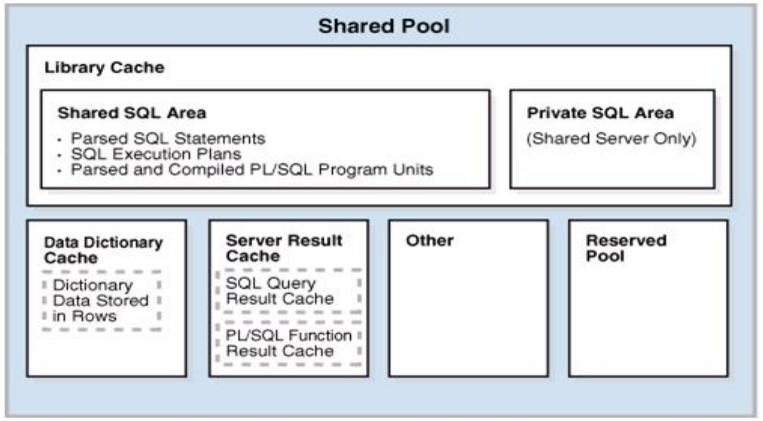
- **User Global Area (UGA)** – this is allocated when a user connects to create a session.

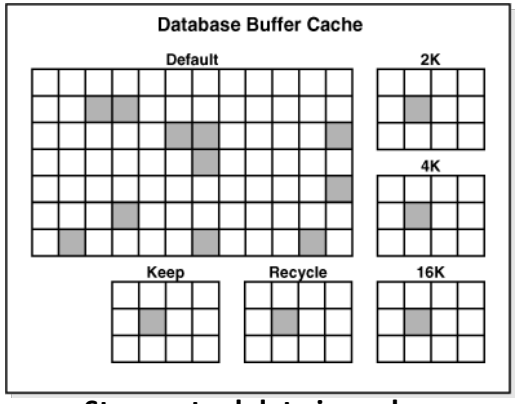
Oracle Architecture

Shared Pool

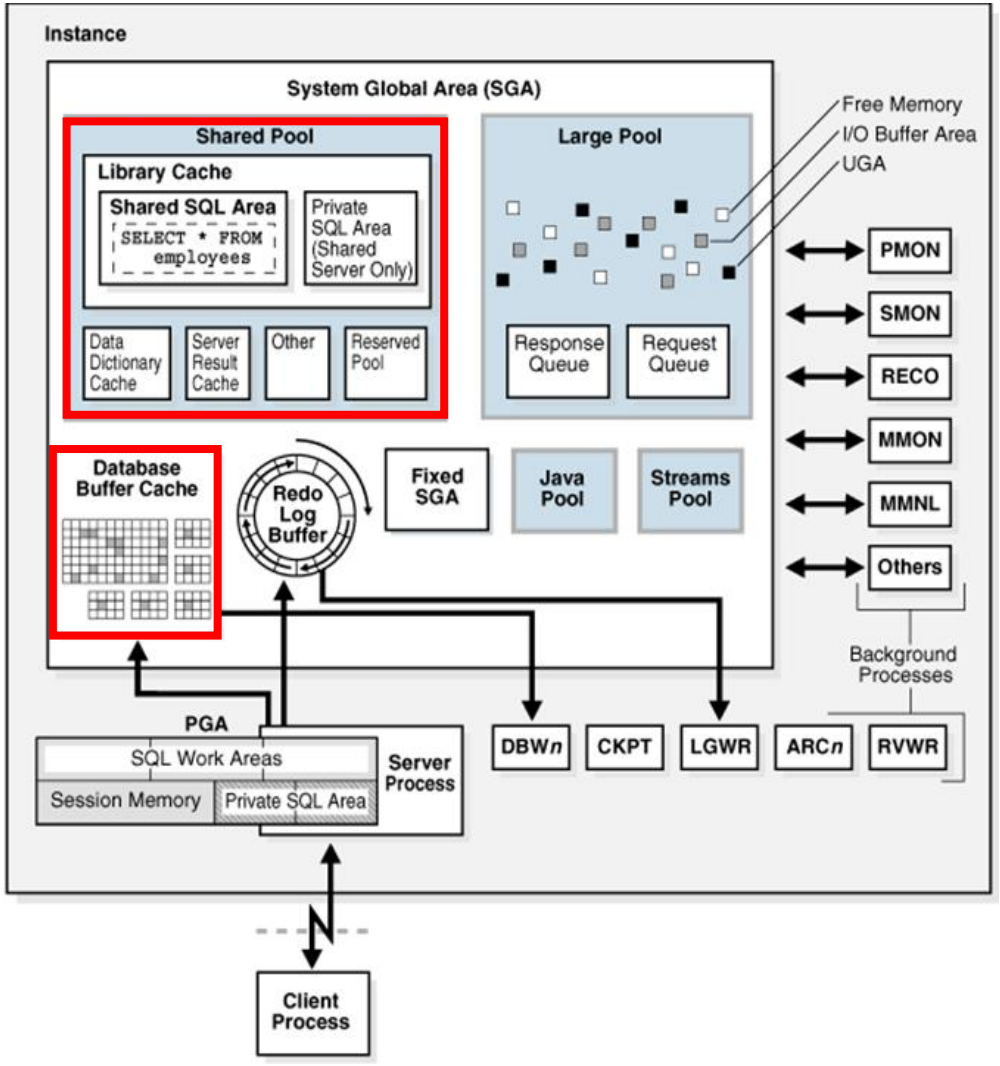
- Used to store:
  - Most recently executed SQL statements
  - Most recently used data definitions
- It consists of two key performance-related memory structures:
  - Library Cache
  - Data Dictionary Cache
- Sized by the parameter SHARED_POOL_SIZE

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





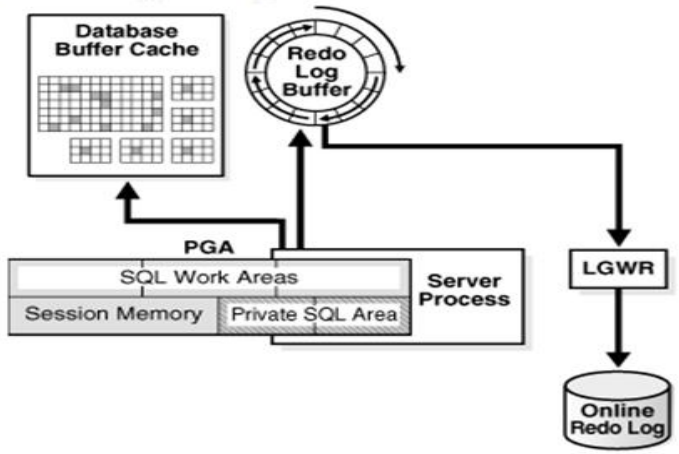
Store actual data in cache

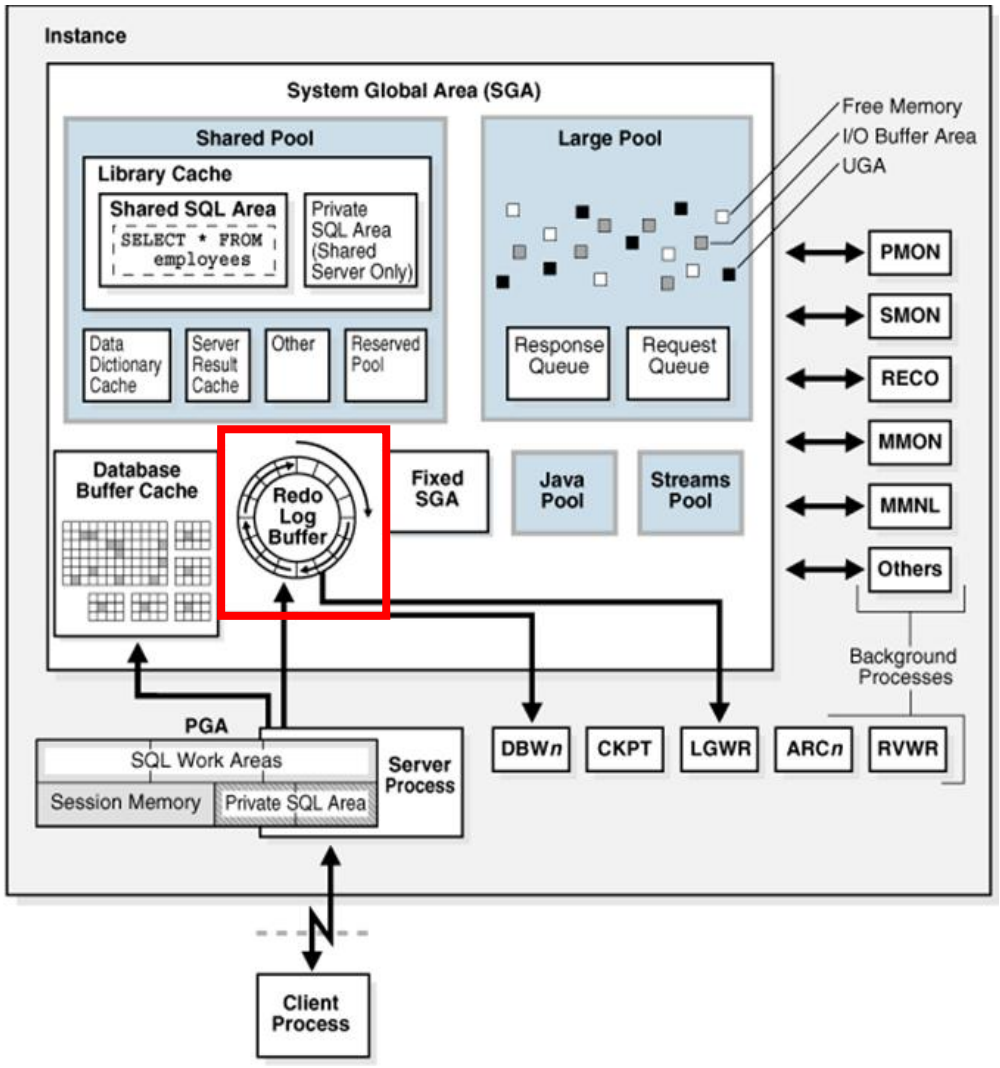


Oracle Architecture

Redo Log Buffer

- Records all changes made to the database data blocks
- Primary purpose is recovery
- Changes recorded within are called redo entries
- Redo entries contain information to reconstruct or redo changes
- Size defined by `LOG_BUFFER`





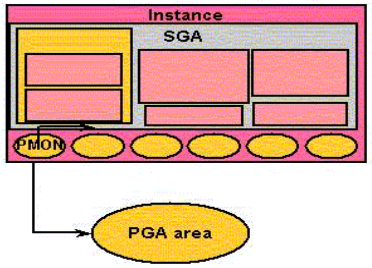
Oracle Architecture

Background Process

Mandatory Background Processes

- Process Monitor Process (PMON)
- System Monitor Process (SMON)
- Database Writer Process (DBWn)
- Log Writer Process (LGWR)
- Checkpoint Process (CKPT)
- Manageability Monitor Processes (MMON and MMNL)
- Recover Process (RECO)

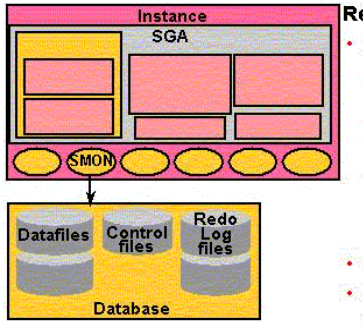
Process Monitor (PMON)



Cleans up after failed processes by:

- **Rolling back the transaction**
- **Releasing locks**
- **Releasing other resources**
- **Restarting dead dispatchers**

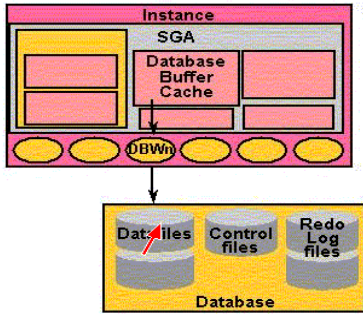
System Monitor (SMON)



Responsibilities:

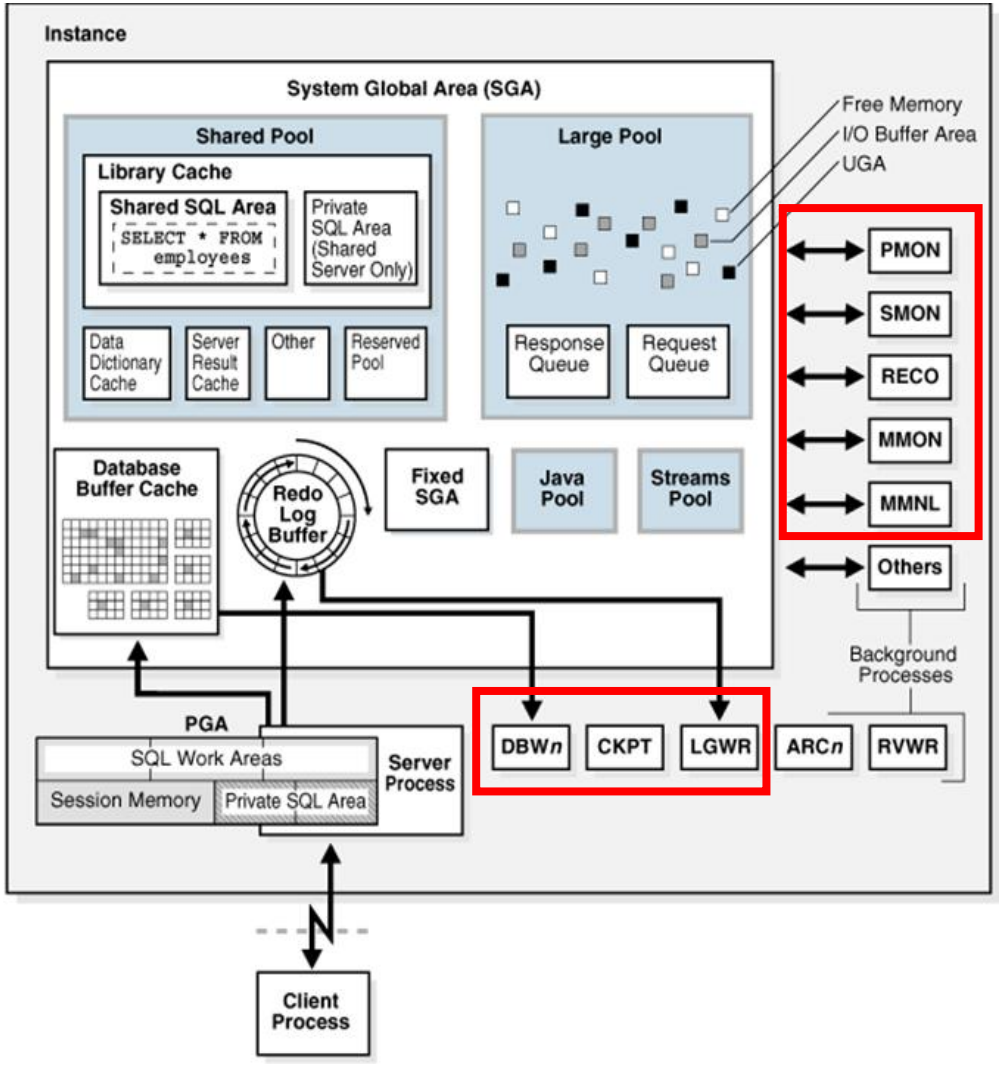
- **Instance recovery**
  - **Rolls forward changes in redo logs**
  - **Opens database for user access**
  - **Rolls back uncommitted transactions**
- **Coalesces free space**
- **Deallocates temporary segments**

Database Writer (DBWn)



DBWn writes when:

- Checkpoint occurs
- Dirty buffers reach threshold
- There are no free buffers
- Timeout occurs
- RAC ping request is made
- Tablespace `OFFLINE`
- Tablespace `READ ONLY`
- Table `DROP` or `TRUNCATE`
- Tablespace `BEGIN BACKUP`



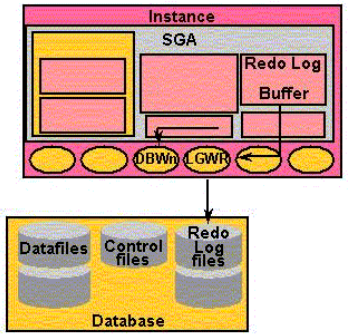
Oracle Architecture

Background Process

Mandatory Background Processes

- Process Monitor Process (PMON)
- System Monitor Process (SMON)
- Database Writer Process (DBWn)
- Log Writer Process (LGWR)
- Checkpoint Process (CKPT)
- Manageability Monitor Processes (MMON and MMNL)
- Recover Process (RECO)

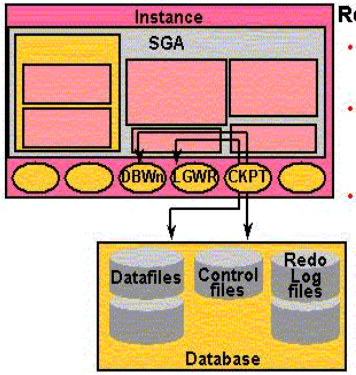
Log Writer (LGWR)



LGWR writes:

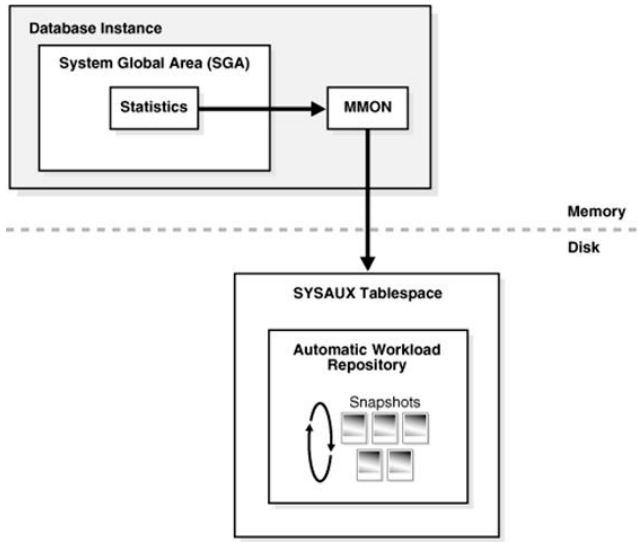
- At commit
- When one-third full
- When there is 1 MB of redo
- Every three seconds
- Before DBWn writes

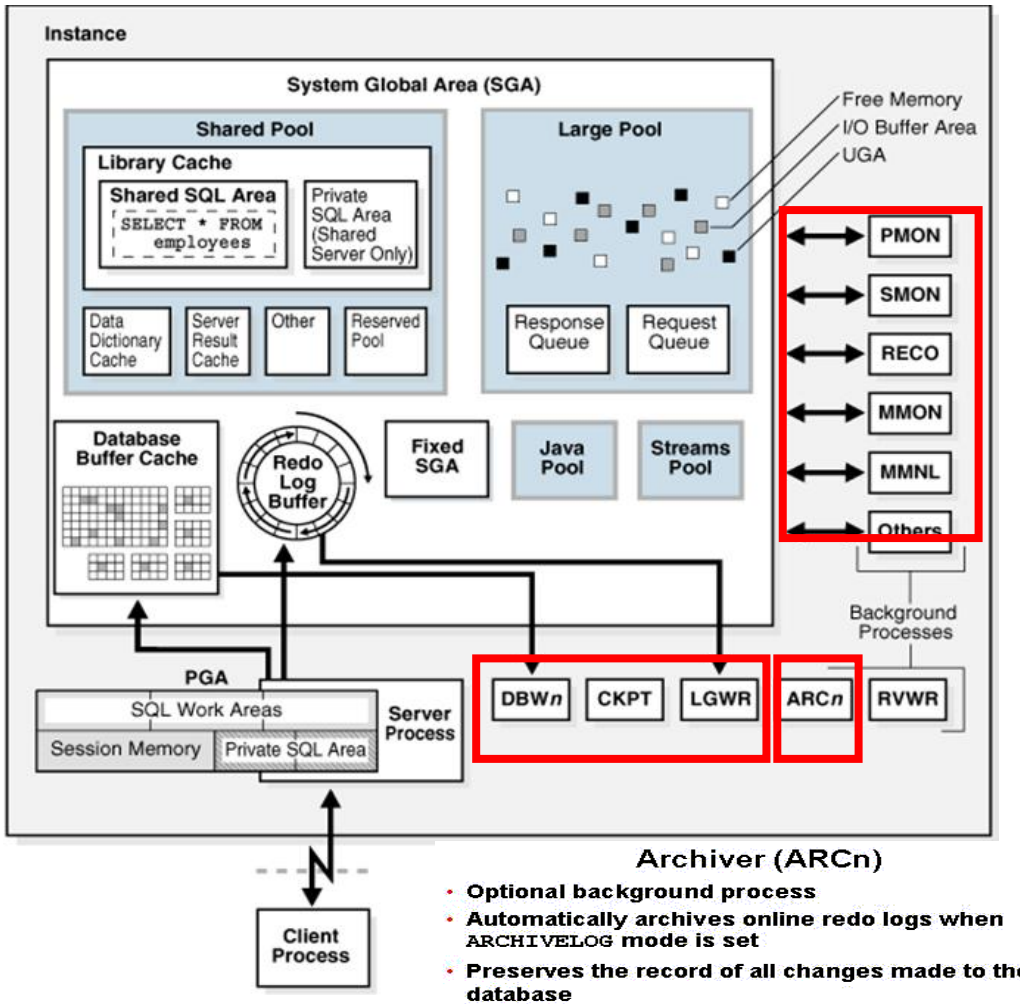
Checkpoint (CKPT)



Responsible for:

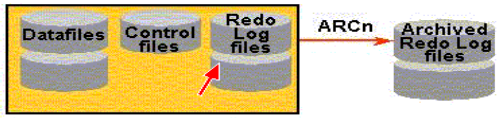
- Signaling DBWn at checkpoints
- Updating datafile headers with checkpoint information
- Updating control files with checkpoint information





Archiver (ARCn)

- Optional background process
- Automatically archives online redo logs when `ARCHIVELOG` mode is set
- Preserves the record of all changes made to the database



Retrieving Data Using the SQL SELECT Statement

To extract data from the database, you need to use the structured query language (SQL) SELECT statement. You may need to restrict the columns that are displayed.

Projection: Choose the columns in a table that are returned by a query. Choose as few or as many of the columns as needed.

Selection: Choose the rows in a table that are returned by a query. Various criteria can we used to restrict the rows that are retrieved.

Joining: Bring together data that is stored in different tables by specifying the link between them.

Various ways to fetch all the columns data from table
SELECT * FROM HR.EMPLOYEES;

SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL, PHONE_NUMBER, HIRE_DATE, JOB_ID, SALARY, COMMISSION_PCT, MANAGER_ID, DEPARTMENT_ID FROM HR.EMPLOYEES;

- SQL statements are not case-sensitive
- SQL statements can be on one or more lines
- Keywords cannot be abbreviated or split across lines
- In SQL Developer, SQL statements can optionally be terminated by a semicolon(;), Semicolons are required if you execute multiple SQL statements

Arithmetic Expressions

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division

SELECT LAST_NAME, SALARY, SALARY+300 FROM HR.EMPLOYEES;

Operator Precedence

SELECT LAST_NAME, SALARY, 12*SALARY+300 FROM HR.EMPLOYEES;

SELECT LAST_NAME, SALARY, 12*(SALARY+300) FROM HR.EMPLOYEES;

Defining a NULL value
- A NULL value is Unknown, Undefined, Unavailable value
- A NULL is not same as ZERO or BLANK space

SELECT LAST_NAME, JOB_ID, SALARY, COMMISSION_PCT FROM HR.EMPLOYEES;

*****NULL value Arithmetic leads to NULL*****
SELECT LAST_NAME, JOB_ID, SALARY,COMMISSION_PCT, 12*SALARY*COMMISSION_PCT FROM HR.EMPLOYEES;

Defining column Alias
SELECT LAST_NAME, JOB_ID, SALARY,COMMISSION_PCT, 12*SALARY*COMMISSION_PCT AS YEAR_COMM FROM HR.EMPLOYEES;

SELECT LAST_NAME, JOB_ID, SALARY,COMMISSION_PCT, 12*SALARY*COMMISSION_PCT YEAR_COMM FROM HR.EMPLOYEES;

Retrieving Data Using the SQL SELECT Statement

Concatenation Operator

SELECT LAST_NAME || JOB_ID as EMPLOYEES FROM HR.EMPLOYEES;

If you concatenate NULL value with LAST_NAME the output will be LAST_NAME

SELECT LAST_NAME|| COMMISSION_PCT FROM HR.EMPLOYEES;

Literal characters

SELECT LAST_NAME || ' is a ' || JOB_ID as EMPLOYEES FROM HR.EMPLOYEES;

Duplicate Rows
SELECT DEPARTMENT_ID FROM HR.EMPLOYEES;

SELECT DISTINCT DEPARTMENT_ID FROM HR.EMPLOYEES;

DESCRIBE TABLE

Use describe command to see the structure of the table

DESCRIBE HR.EMPLOYEES;

Data Type	Description
NUMBER(*p*, *s*)	Number value having a maximum number of digits *p*, with *s* digits to the right of the decimal point
VARCHAR2(*s*)	Variable-length character value of maximum size *s*
DATE	Date and time value between January 1, 4712 B.C. and December 31, A.D. 9999.
CHAR(*s*)	Fixed-length character value of size *s*

Retrieving Data Using the SQL SELECT Statement

Test your knowledge

The following SELECT statement executes successfully:
SELECT LAST_NAME, JOB_ID, SALARY AS SAL FROM HR.EMPLOYEES;

TRUE/FALSE?

The following SELECT statement executes successfully:
SELECT * FROM HR.JOB_GRADES;

TRUE/FALSE?

There are three coding errors in the following statement. Can you identify them?
SELECT EMPLOYEE_ID, LAST_NAME, SAL X 12 ANNUAL SALARY FROM HR.EMPLOYEES;

Exercise

1) Your first task is to determine the structure of the DEPARTMENTS table and its content

```
Name            Null     Type
--------------- -------- ------------
DEPARTMENT_ID   NOT NULL NUMBER(4)
DEPARTMENT_NAME NOT NULL VARCHAR2(30)
MANAGER_ID               NUMBER(6)
LOCATION_ID              NUMBER(4)
```

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700

2) The HR department wants a query to display the last name, job code, hire date and employee number for each employee, with the employee number appearing first. Provide an alias STARTDATE for the hire date column. **(Shown below)**

EMPLOYEE_ID	LAST_NAME	JOB_ID	STARTDATE
100	King	AD_PRES	17-JUN-03
101	Kochhar	AD_VP	21-SEP-05
102	De Haan	AD_VP	13-JAN-01
103	Hunold	IT_PROG	03-JAN-06
104	Ernst	IT_PROG	21-MAY-07
105	Austin	IT_PROG	25-JUN-05

3) The HR department needs a query to display all unique job codes from the EMPLOYEES table

4) The HR department has requested a report of all employees and their job ids. Display the last name concatenated with the job id. Name the column as DETAILS. **(Shown below)**

DETAILS
Abel,SA_REP
Ande,SA_REP
Atkinson,ST_CLERK
Austin,IT_PROG
Baer,PR_REP
Baida,PU_CLERK
Banda,SA_REP