TOPIC NAME:

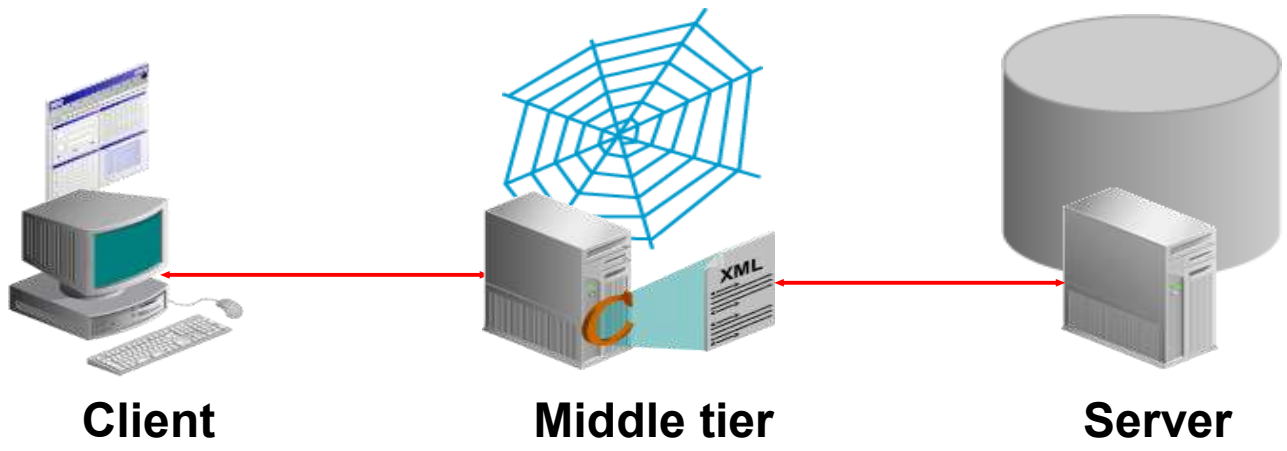
- **EXPLORING THE ARCHITECTURE OF ORACLE**
- **FEATURES OF ORACLE**

The **Oracle relational database management system (RDBMS)** provides an open, comprehensive, integrated approach to information management.

Oracle Database

- A database is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information.
- Oracle Database reliably manages a large amount of data in a multiuser environment so that many users can concurrently access the same data. This is accomplished while delivering high performance. At the same time, it prevents unauthorized access and provides efficient solutions for failure recovery.

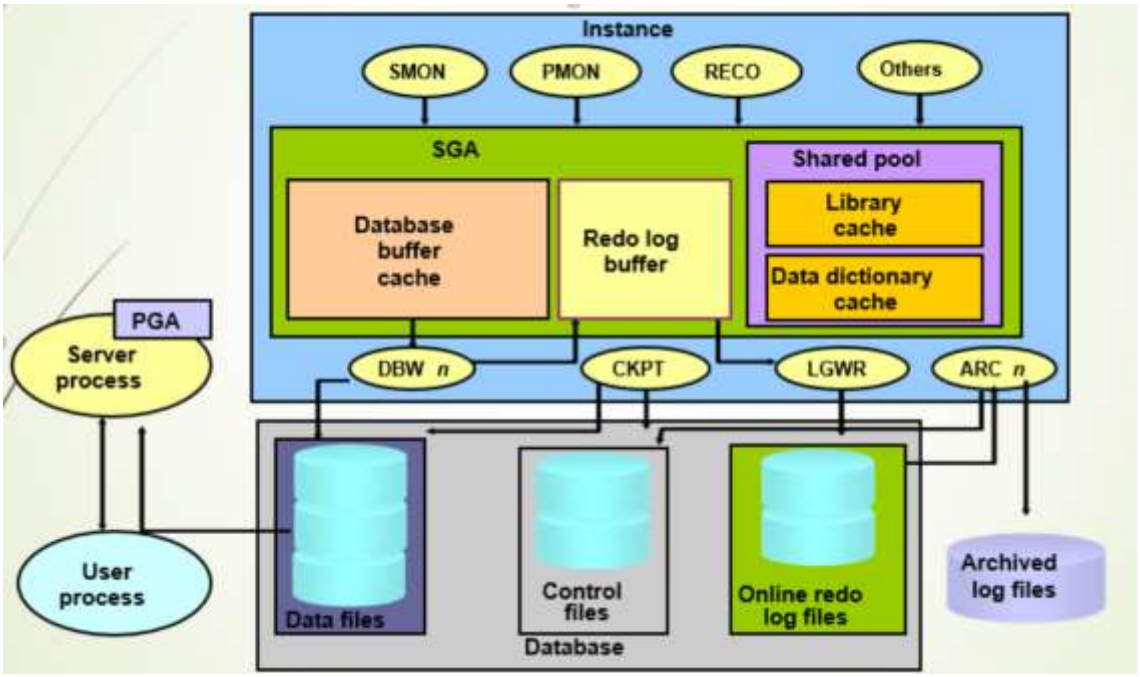
Connecting to Server



A database user can connect to an Oracle server in one of three ways:

- The user logs on to the operating system running the Oracle instance and starts an application or tool that accesses the database on that system. The communication pathway is established using the inter-process communication mechanisms available on the host operating system.
- The user starts the application or tool on a local computer and connects over a network to the computer running the Oracle database. In this configuration (called client/server), network software is used to communicate between the user and the back-end server.

DATABSE ARCHITECTURE OVERVIEW:



An **Oracle database** consists of an **instance** and its associated databases. The instance consists of **memory structures** and **background processes**. Every time an instance is started, a shared

memory area called the **System Global Area (SGA**) is allocated and the background processes are started.

The database consists of both physical structures and logical structures. Because the physical and logical structures are separate, the physical storage of data can be managed without affecting access to logical storage structures.

A DB instance is a set of memory structures that manage database files.

DETAIL EXPLANATION OF ABOVE DIAGRAM:

This diagram represents the architecture of an Oracle Database system. It consists of three main components:

1. **User and Server Processes**:
    - The **User Process** initiates a request (e.g., running a query).
    - The **Server Process** executes the request and communicates with the database.
    - The **PGA (Program Global Area)** stores data and control information specific to a user session.
2. **Instance (SGA and Background Processes)**:
    - The **SGA (System Global Area)** is shared memory that stores frequently used data and SQL execution plans.
        - **Database Buffer Cache**: Temporarily stores retrieved or modified data.
        - **Redo Log Buffer**: Stores changes before they are permanently written to disk.
        - **Shared Pool**: Contains the **Library Cache** (for SQL execution plans) and **Data Dictionary Cache** (for metadata).
    - **Background Processes**:

- **SMON, PMON, RECO, and Others**: Manage system recovery, session management, and process coordination.
- **DBW (Database Writer)**: Writes modified data from the buffer cache to **Data Files**.
- **CKPT (Checkpoint Process)**: Updates **Control Files** to track database changes.
- **LGWR (Log Writer)**: Saves changes from the redo log buffer to **Online Redo Log Files**.
- **ARC (Archiver Process)**: Moves redo logs to **Archived Log Files** for backup and recovery.

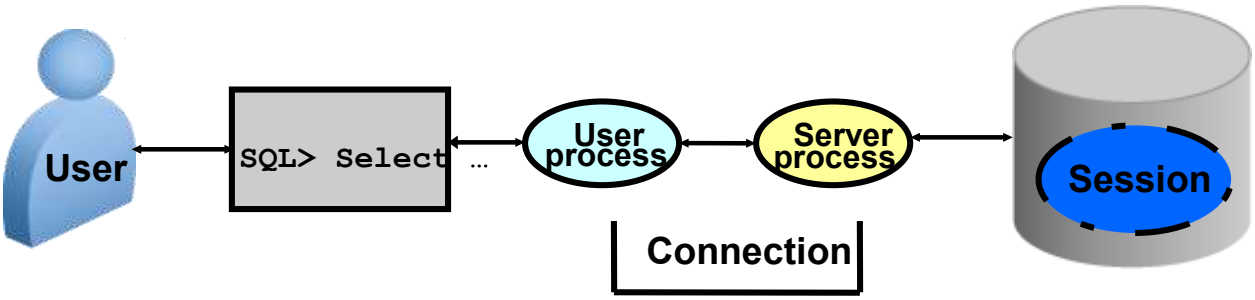
3. **Database Storage**:
    - **Data Files**: Store actual tables and indexes.
    - **Control Files**: Store metadata about database structure.
    - **Online Redo Log Files**: Keep track of changes to help with recovery in case of failure.
    - **Archived Log Files**: Backup of redo logs for historical tracking and recovery.

HOW IT WORKS IN SIMPLE TERMS

When a user executes a query:

1. The request goes to the **Server Process**, which interacts with the **SGA**.
2. Data is first retrieved from the **Buffer Cache**; if not found, it is fetched from **Data Files**.
3. Any changes are stored in the **Redo Log Buffer** before being written to **Online Redo Log Files**.
4. The **Database Writer** periodically saves changes to **Data Files**.
5. The **Log Writer** ensures transaction logs are saved for recovery.
6. The **Checkpoint Process** updates **Control Files** to track progress.
7. If necessary, the **Archiver** saves old log files for backup.

CONNECTING TO THE DATABASE:



Session

Connections and sessions are closely related to user processes but are very different in meaning.

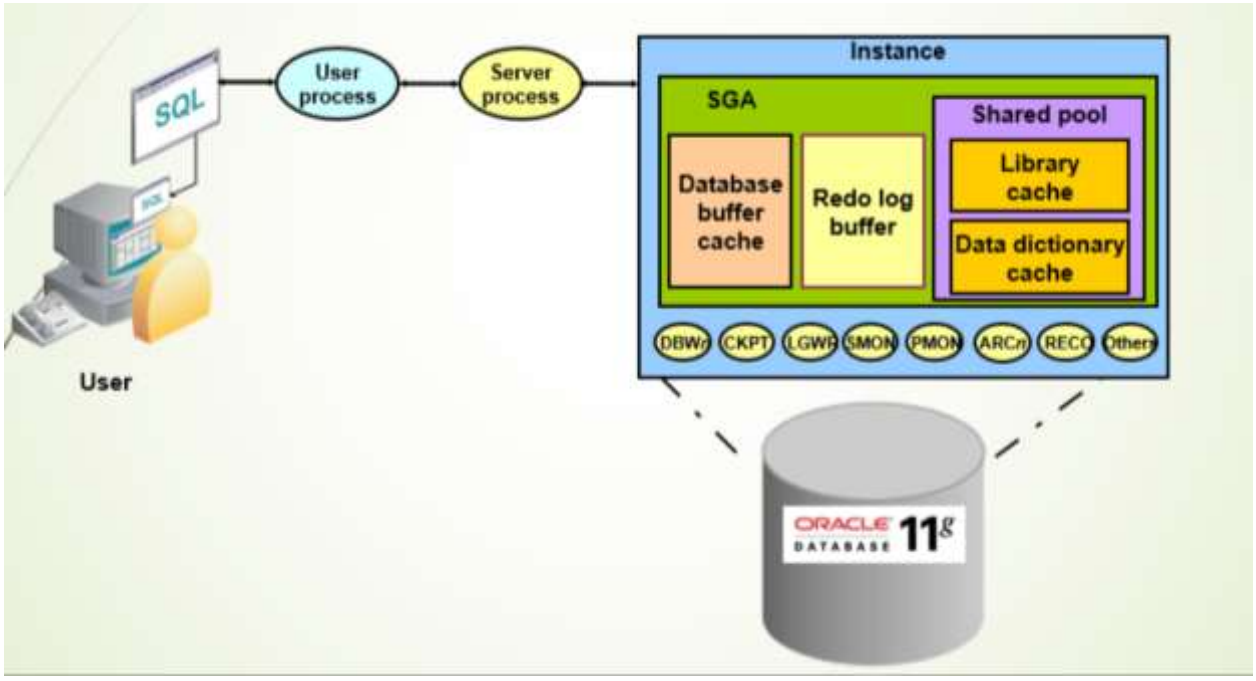
A **connection** is a communication pathway between a user process and an Oracle Database instance. A communication pathway is established using available inter-process communication mechanisms (on a computer that runs both the user process and Oracle Database) or network software (when different computers run the database application and Oracle Database, and communicate through a network).

A **session** represents the state of a current user login to the database instance. For example, when a user starts SQL*Plus, the user must provide a valid username and password, and then a session is established for that user. A session lasts from the time a user connects until the user disconnects or exits the database application.

In the case of a **dedicated connection**, the session is serviced by a permanent dedicated process. The session is serviced by an available server process selected from a pool, either by the middle tier or by Oracle shared server architecture.

Multiple sessions can be created and exist concurrently for a single Oracle database user using the same username. For example, a user with the username/password of HR/HR can connect to the same Oracle Database instance several times.

INTERACTING WITH ORACLE DATABSAE:



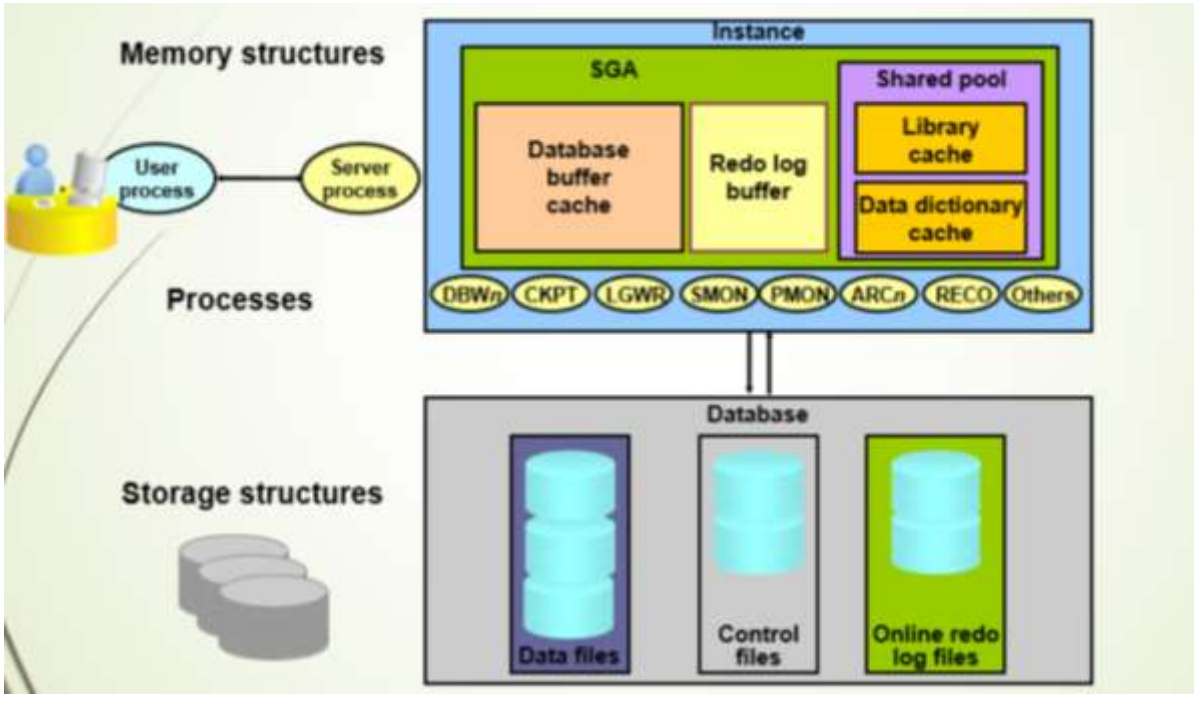
The following example describes Oracle database operations at the most basic level. It illustrates an Oracle database configuration in which the user and associated server process are on separate computers, connected through a network.

1. An **instance** has started on a node where **Oracle Database** is installed, often called the host or database server.
2. A user starts an application spawning a user process. The application attempts to establish a connection to the server. (The connection may be local, client/server, or a three-tier connection from a middle tier.)
3. The server runs a listener that has the appropriate Oracle Net Services handler. The server detects the connection request from the application and creates a dedicated server process on behalf of the user process.
4. The user runs a DML-type SQL statement and commits the transaction. For example, the user changes the address of a customer in a table and commits the change.
5. The server process receives the statement and checks the shared pool (an SGA component) for any shared SQL area that contains a similar SQL statement. If a shared SQL area is found, the server process checks the user's access privileges to the requested data, and the existing shared SQL area is used to process the statement. If a shared SQL area is not found, a new shared SQL area is allocated for the statement so that it can be parsed and processed.
6. The server process retrieves any necessary data values, either from the actual data file (table) or from values stored in the SGA.
7. The server process modifies data in the SGA. Because the transaction is committed, the LogWriter process (LGWR) immediately records the transaction in the redo log file. The

Database Writer process (DBWn) writes modified blocks permanently to disk when it is efficient to do so.

8. If the transaction is successful, the server process sends a message across the network to the application. If it is not successful, an error message is transmitted.
9. Throughout this entire procedure, the other background processes run, watching for conditions that require intervention. In addition, the database server manages other users' transactions and prevents contention between transactions that request the same data.

ORACLE DATABASE SERVER STRUCTURES:



After starting an instance, the Oracle software associates the instance with a specific database. This is called **mounting the database**. The database is then ready to be opened, which makes it accessible to authorized users. Multiple instances can execute concurrently on the same computer, each accessing its own physical database.

You can look at the Oracle Database architecture as various interrelated structural components.

An Oracle instance uses **memory structures and processes** to manage and access the database. All memory structures exist in the main memory of the computers that constitute the database server. Processes are jobs that work in the memory of these computers. A process is defined as a "**thread of control**" or a mechanism in an operating system that can run a series of steps.