4. DATA MODELS IN DBMS

A Data Model is a logical structure of Database. It describes the design of database to reflect entities, attributes, relationship among data, constrains etc.

This is the basic structure of the database that organizes the data, defines how the data are stored, accessed and the relationships between the data. Types of Data Models:

Types of Data models:

- Hierarchical Model
- Network Model
- Entity-Relationship(E-R) Model
- Relational Model
- Object-Oriented Model
- Object-Relational Model

1. **Hierarchial Data Model:**

- Data is represented as a tree.
- A record type can belong to only one owner type but a owner type can belong to many record type.
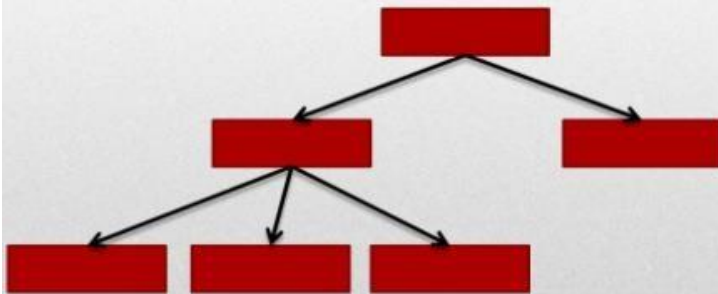
2. **Network Model:**

- It is modified version of Hierarchical Data Model where it allows more general connections among the nodes as well.
- They are difficult to use but are more flexible than hierarchical databases.
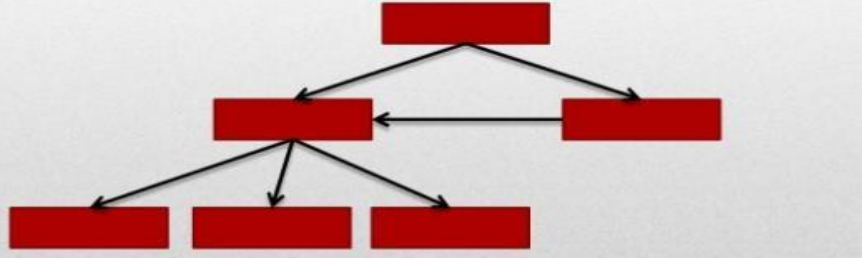
3. Relational Model:

- It is a lower level model that uses a collection of tables to represent both data and relationships among those data.
- Each table has multiple columns, depending on the number of attributes, and each column has a unique name.

Student

| sid	sname	Standard
A-101	Ramesh	11
A-102	Kriti	10
A-103	Laxmi	12

4. Entity relationship model

- It is a high level model based on the need of the organization.
- Its entities are distinguishable from other objects and relationship is an association among several entities.
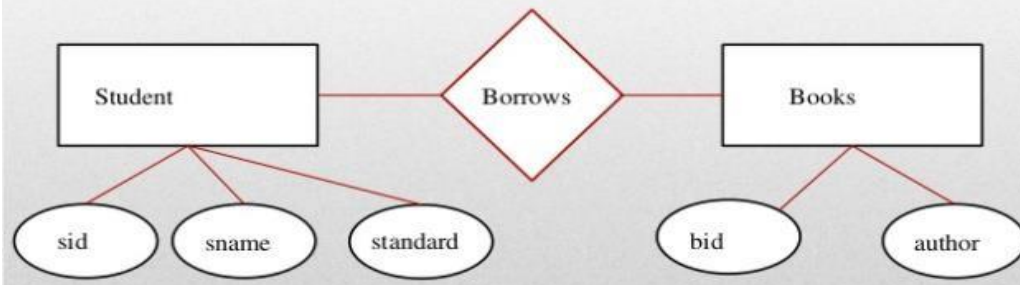
5. Object Oriented Model

- It represents entity sets as class and a class represents both attributes and the behavior of the entity.
- Instance of a class is an object.
- The internal part of the object is not externally visible.
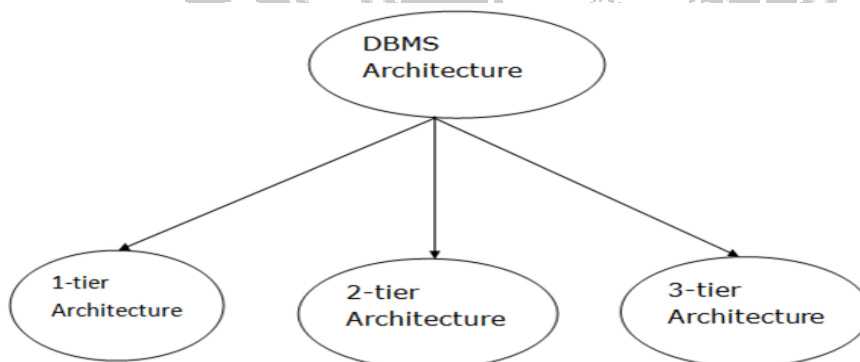- One object communicates with the other by sending messages.
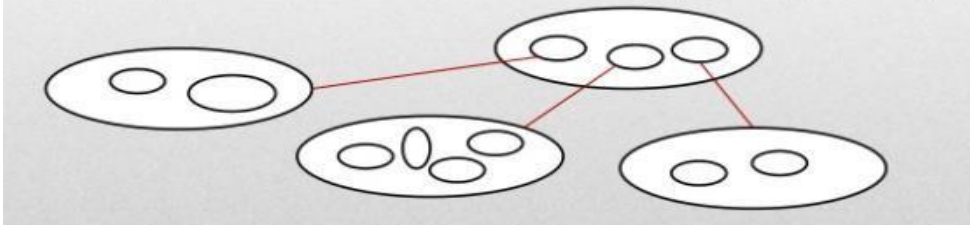
6. **Object Relational model:**

It combines the feature of relational model and object oriented model.

5. DBMS Architecture

- DBMS architecture depends upon how users are connected to the database to get their request done.

Database architecture can be seen as a single tier or multi-tier. But logically, database architecture is of two types like: **2-tier architecture** and **3-tier architecture**.

1-Tier Architecture

- In this architecture, the database is directly available to the user. It means the user can directly sit on the DBMS and uses it.
- Any changes done here will directly be done on the database itself. It doesn't provide a handy tool for end users.
- The 1-Tier architecture is used for development of the local application, where programmers can directly communicate with the database for the quick response.
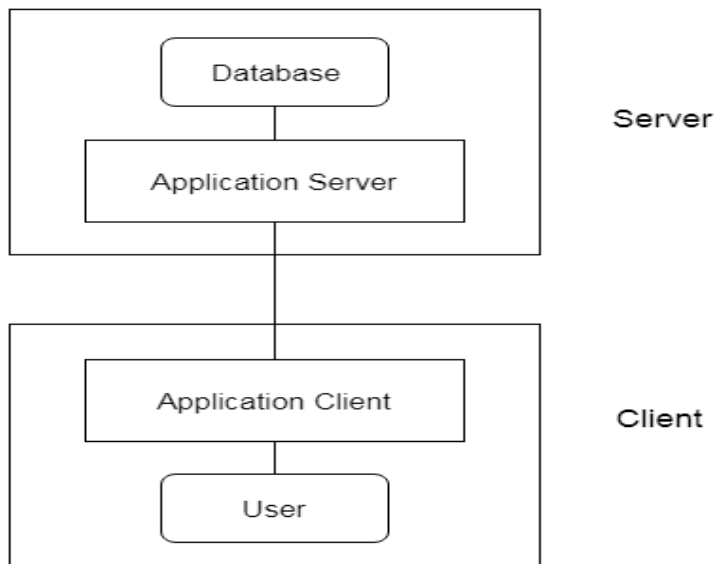
2-Tier Architecture
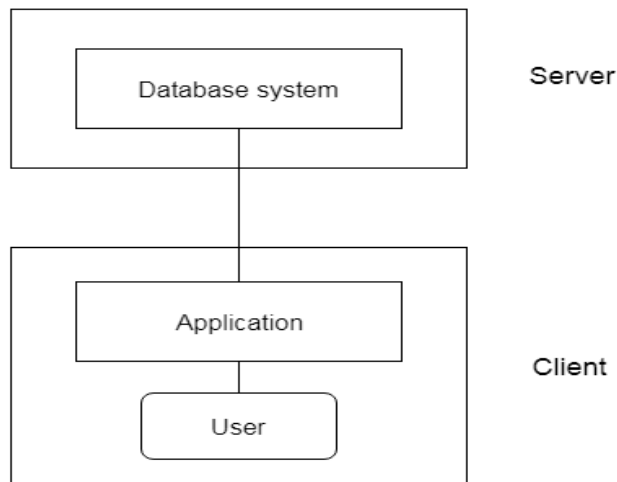
- The 2-Tier architecture is same as basic client-server. In the two-tier architecture, applications on the client end can directly communicate with the database at the server side. For this interaction, API's like: **ODBC**, **JDBC** are used.
- The user interfaces and application programs are run on the client-side.
- The server side is responsible to provide the functionalities like: query processing and transaction management.
- To communicate with the DBMS, client-side application establishes a connection with the server side.

3-Tier Architecture

- The 3-Tier architecture contains another layer between the client and server. In this architecture, client can't directly communicate with the server.
- The application on the client-end interacts with an application server which further communicates with the database system.
- End user has no idea about the existence of the database beyond the application server. The database also has no idea about any other user beyond the application.
- The 3-Tier architecture is used in case of large web application.

Three Schema Architecture

- The three schema architecture is also called ANSI/SPARC architecture or three-level architecture.

- This framework is used to describe the structure of a specific database system.

- The three schema architecture is also used to separate the user applications and physical database. The three schema architecture contains three-levels. It breaks the database down into three different categories.
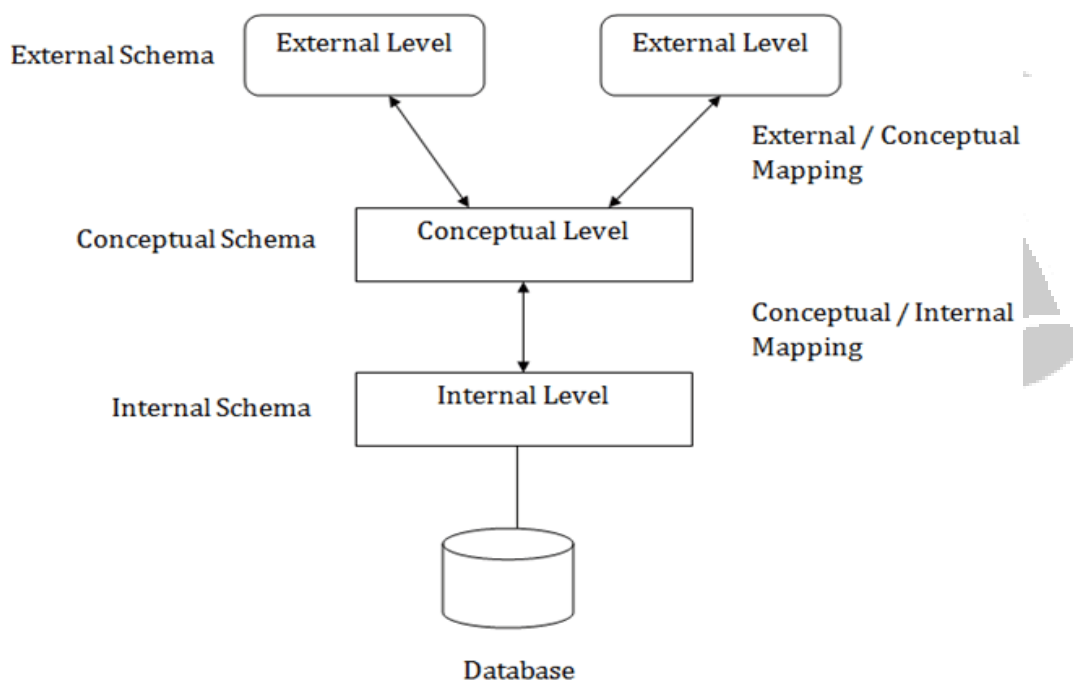
1. **The internal level** – has an internal schema which describes the physical storage structure of the database. Uses a physical data model and describes the complete details of data storage and access paths for the database.

2. **The conceptual level** – has a conceptual schema which describes the structure of the database for users. It hides the details of the physical storage structures, and concentrates on describing entities, data types, relationships, user operations and constraints. Usually a representational data model is used to describe the conceptual schema.

3. **The External or View level** – includes external schemas or user views. Each external schema describes the part of the database that a particular user group is interested in and hides the rest of the database from that user group. Represented using the representational data model.

The three schema architecture is used to visualize the schema levels in a database. The three schemas are only descriptions of data, the data only actually exists is at the physical level.

5.1 COMPONENTS OF DBMS

Database Users are differentiated by the way they expect to interact with the system

- Application programmers
- Sophisticated users
- Naïve users
- Database Administrator
- Specialized users etc..

1. Application programmers: Professionals who write application programs and using these application programs they interact with the database system

2. Sophisticated users : These user interact with the database system without writing programs, But they submit queries to retrieve the information

3. Specialized users: Who write specialized database applications to interact with the database system.

4. Naïve users: Interacts with the database system by invoking some application programs that have been written previously by application programmers

Eg : people accessing database over the web

5. Database Administrator: Coordinates all the activities of the database system; the database administrator has a good understanding of the enterprise's information resources and needs.

- Schema definition

- Access method definition

- Schema and physical organization modification

- Granting user authority to access the database

- Monitoring performance

Storage Manager:The Storage Manager include these following components/modules

- Authorization Manager

- Transaction Manager

- File Manager

- Buffer Manager

Storage manager is a program module that provides the interface between the low-level data stored in the database and the application programs and queries submitted to the system. The storage manager is responsible to the following tasks:

- Interaction with the file manager

- Efficient storing, retrieving and updating of data

Authorization Manager

o Checks whether the user is an authorized person or not

o Test the satisfaction of integrity constraints

Transaction Manager

Responsible for concurrent transaction execution. It ensures that the database remains in a consistent state despite of the system failure