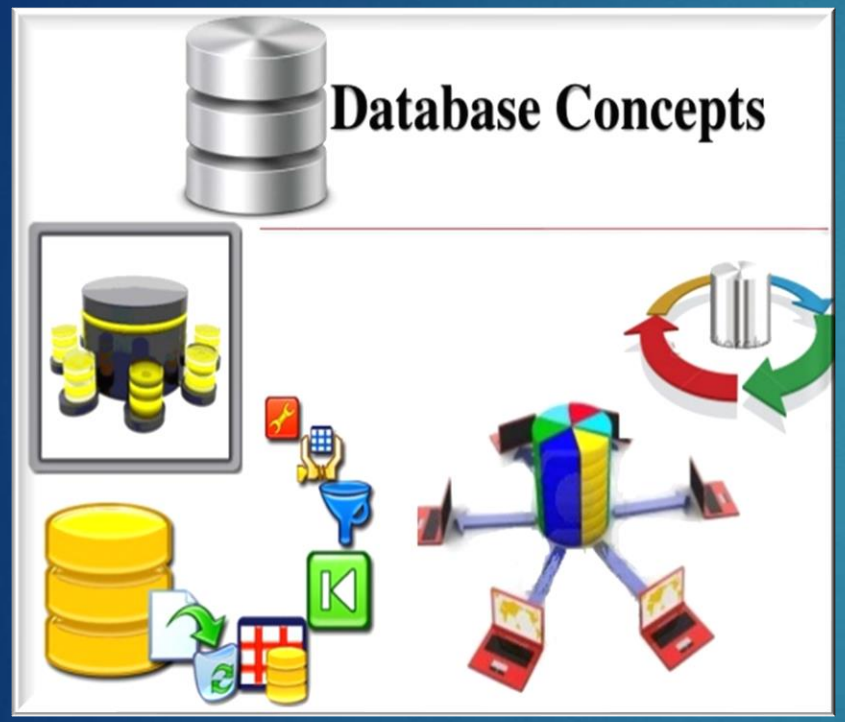
Database Management System

Unit-1

Dr. K ADISESHA

DATABASE CONCEPTS



- Introduction
- Data Abstraction
- Architecture of DBMS
- Data Models
- Database Language

Prof. K. Adisesha

Introduction

Definition:

- ***Data:*** Data is a collection of facts, numbers, letters or symbols that the computer process into meaningful information.
- ***Information:*** Information is processed data, stored, or transmitted by a computer.
- ***Database:*** A Database is a collection of logically related data organized in a way that data can be easily accessed, managed and updated.
- ***Record:*** a record (sometimes called a row) is a group of fields within a table that are relevant to a specific entity. It contains fields such as: ID number, name, street address, city, telephone number and so on.

Introduction

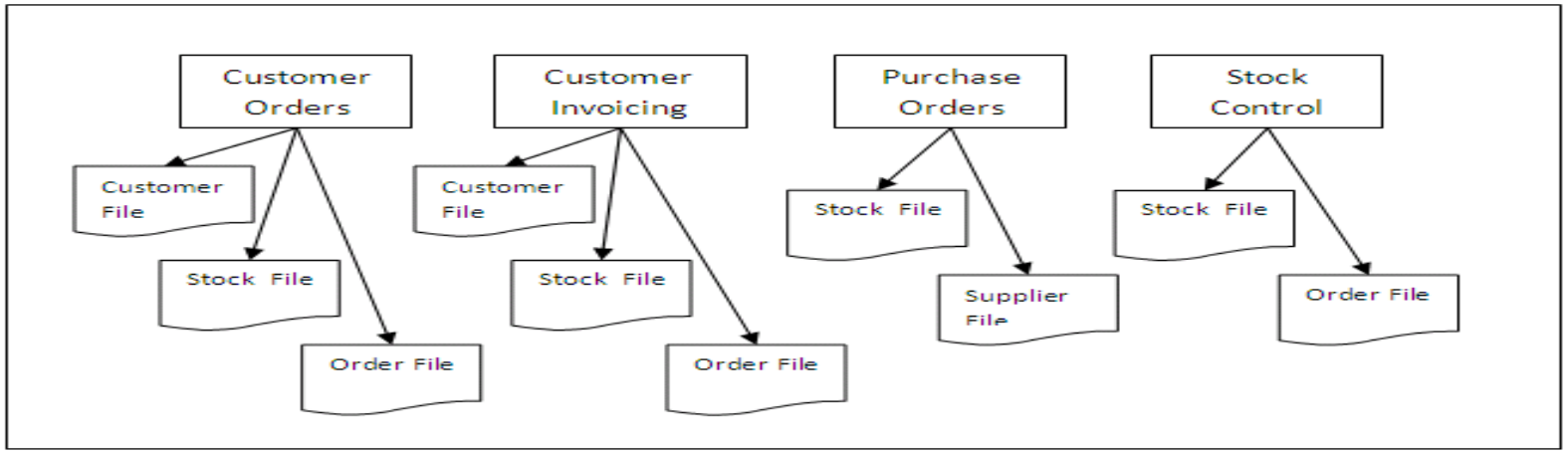
Difference between Manual and Computerized data processing:

Manual Data Processing	Computerized Data Processing
• The volume of data, which can be processed, is limited.	• The volume of data, which can be processed is large
• Requires large quantity of paper	• Requires less quantity of paper
• Speed and accuracy is executed is limited	• Execution is Faster and Accurate
• Labor cost is high	• Labor cost is low
• Storage medium is paper.	• Storage medium is Hard disk etc.

Introduction

Traditional file-based approach:

- **File-based approach** refers to the situation where data is stored in one or more separate computer files defined and managed by different application programs.
- Example, the details of customers may be stored in one file, orders in another, etc.
- The file-based approach might have application programs that deal with purchase orders, invoices, sales and marketing, suppliers, customers, employees, and so on.



Introduction

Traditional file-based approach:

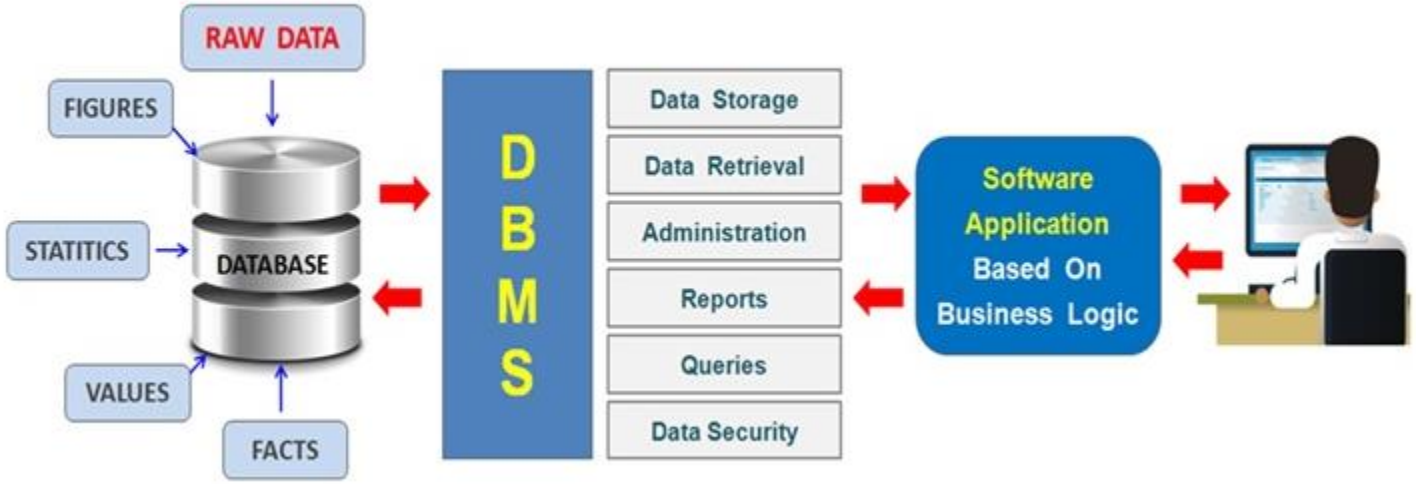
- **Limitations:**
- ***Data duplication:*** Each program stores its own separate files. If the same data is to be accessed by different programs, then each program must store its own copy of the same data.
- ***Data inconsistency:*** If the data is kept in different files, there could be problems when an item of data needs updating, as it will need to be updated in all the relevant files; if this is not done, the data will be inconsistent, and this could lead to errors.
- ***Difficult to implement data security:*** Data is stored in different files by different application programs. This makes it difficult and expensive to implement organisation-wide security procedures on the data.

Introduction

DBMS:

Database management systems are software systems used to manage and manipulate data in a database.

- As most application performance issues originate in the database, knowing how to monitor and optimize your database is essential to your operations..



Introduction

History of DBMS:

Charles Bachman's Integrated Data Store (IDS) is said to be the first DBMS in history.

- History of Database Management System

- ***1960 – Charles Bachman designed first DBMS system.***
- ***1970 – Codd introduced IBM'S Information Management System (IMS).***
- ***1976 – Peter Chen coined and defined the Entity-relationship model also know as the ER model.***
- ***1980 – Relational model becomes a widely accepted database component.***
- ***1985 – Object-oriented DBMS develops.***
- ***1990 – Incorporation of object-orientation in relational DBMS.***
- ***1991 – Microsoft ships MS access, a personal DBMS and that displaces all other personal DBMS products.***
- ***1995 – First Internet database applications.***
- ***1997 – XML applied to database processing. Many vendors begin to integrate XML into DBMS products.***

Introduction

Applications of Database:

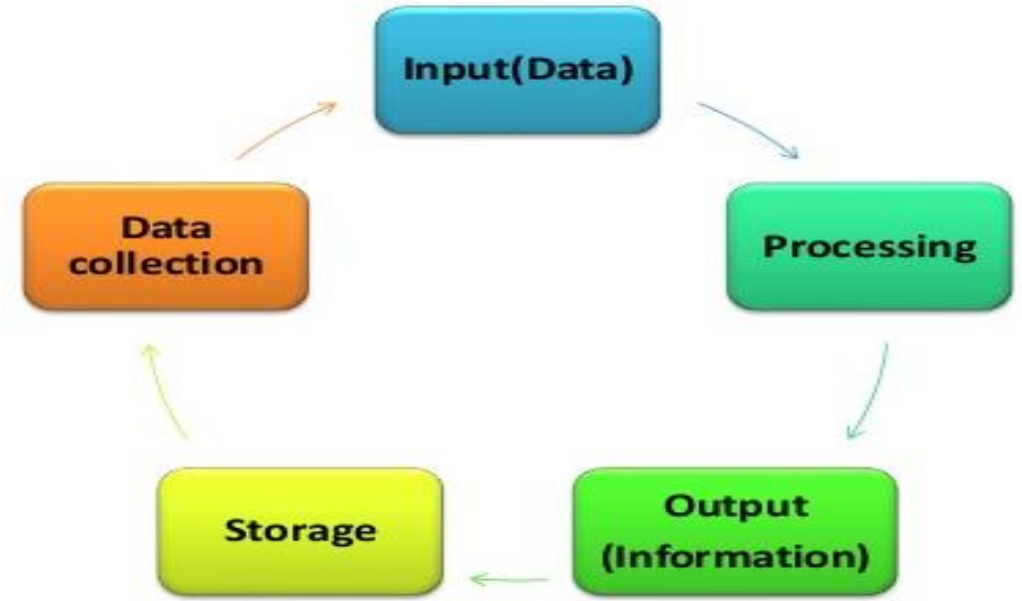
- ***Banking:*** For customer information, accounts and loans, and banking transactions.
- ***Colleges:*** For student information, course registrations and grades.
- ***Credit card transactions:*** For purchases on credit cards and generation of monthly statements.
- ***Finance:*** For storing information about holdings, sales and purchases of financial
- instruments such as stocks and bonds.
- ***Telecommunication:*** For keeping records of call made, generating monthly bills, and storing information about the communication networks.
- ***Voter id/Aadhar database:*** This is the biggest database in the world storing a data about 60 million people residing in India.
- **Sales:** For customer, product, and purchase information.

Data processing cycle

Data processing cycle:

The order in which information is processed in a computer information management system is called data process cycle.

- To design, use and maintain the database, Data processing cycle involves.
  - Data Collection
  - Data Input
  - Data Processing
  - Data storage
  - Output
  - Communication



Data processing cycle

Data processing cycle:

To design, use and maintain the database, many peoples are involved.

- **Data Collection:** It is the process of systematic gathering of data from various sources that has been systematically observed, recorded and organized.
- **Data Input:** The raw data is put into the computer using a keyboard, mouse or other devices such as the scanner, microphone and the digital camera.
- **Data Processing:** Processing is the series of actions or operations on the input data to generate outputs.
- **Data storage:** Data and information should be stored in memory so that it can be accessed later.
- **Output:** The result obtained after processing the data must be presented to the user in user understandable form.
- **Communication:** Computers have communication ability in communication connections, data may be transmitted as an e-mail or posted to the website where the online services are rendered.

Features of Database

Features or advantages of Database:

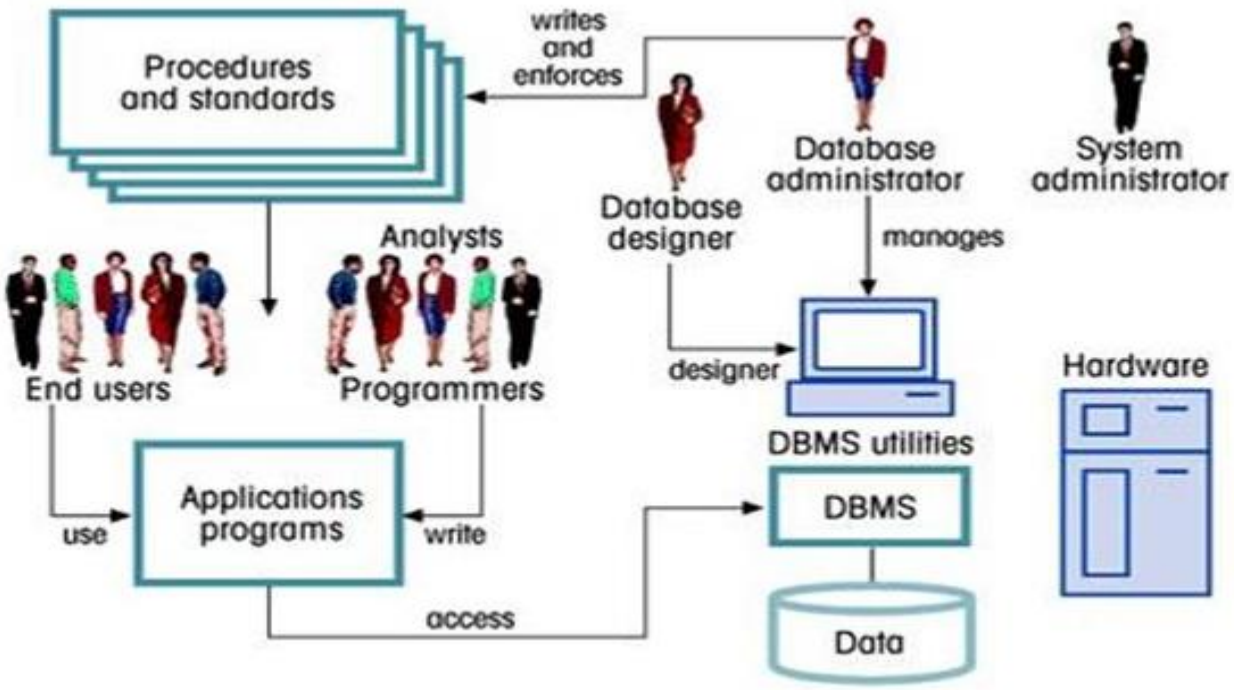
- ***Redundancy can be minimized or controlled:*** In DBMS environment if redundancy is present, then it can be controlled by propagating updates in all the places where ever redundant data is present.
- ***Data Integrity:*** Data Integrity refers to the correctness of the data in the database. In other words, the data available in the database is reliable data.
- ***Data Sharing:*** In DBMS, data is stored in the centralized database and all the permitted users can access the same piece of information required at the same time.
- ***Database Security:*** DBMS provides a variety of security mechanisms for the user to protect his or her data stored in the database.
- ***Supports Concurrent access:*** DBMS supports concurrent access to the same data stored in the database by applying locking and time stamp mechanisms.

Database users

Database users:

To design, use and maintain the database, many peoples are involved.

- The people who work with the database include:
  - **System Analysts**
  - **Application programmers**
  - **Database Administrators (DBA)**
  - **End Users (Database Users)**



Database users

Database users:

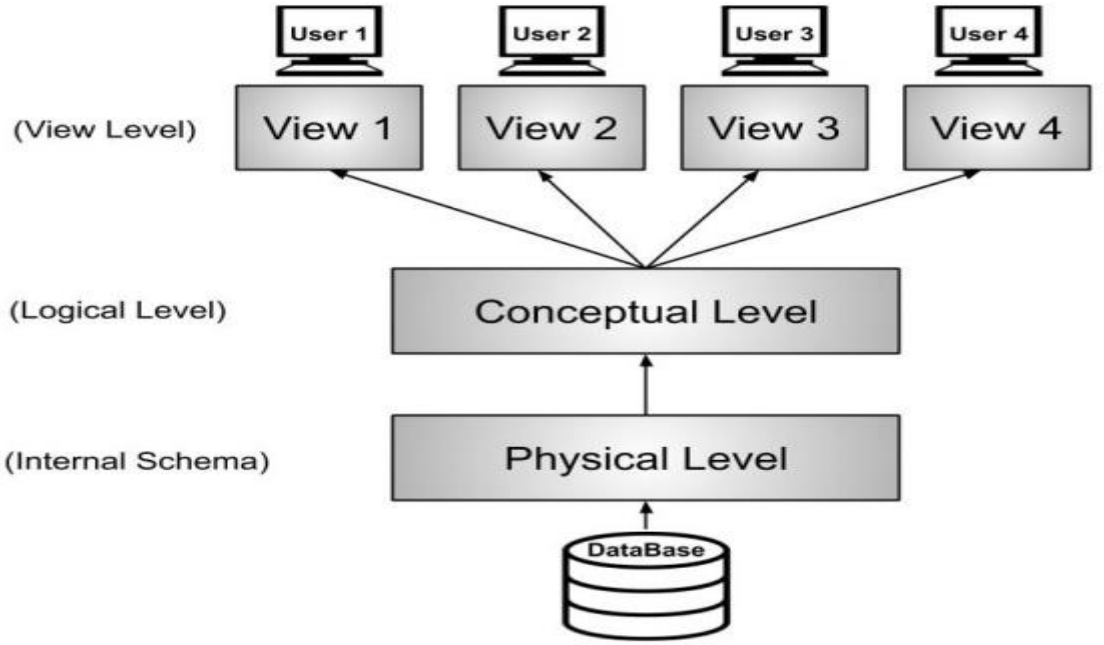
- ***System Analysts:*** System analysts determine the requirement of end users; (especially end users), to create a solution for their business need and focus on non-technical and technical aspects.
- ***Application programmers:*** These are the computer professionals who implement the specifications given by the system analysts and develop the application programs.
- ***Database Administrators (DBA):*** DBA is a person who has central control over both data and application. The responsibilities of DBA are authorization access, schema definition and modification, software installation and security enforcement and administration.
- ***Database users:*** Are those who interact with the database in order to query and update the database, and generate reports.

Data Abstraction

Data Abstraction:

A major purpose of a database system is to provide users with an abstract view of the data.

- That is the system hides certain details of how the data are stored and maintained.
- There are three level of data abstraction.
  - Physical Level(Internal level)
  - Conceptual Level (Logical level)
  - View Level(External level)



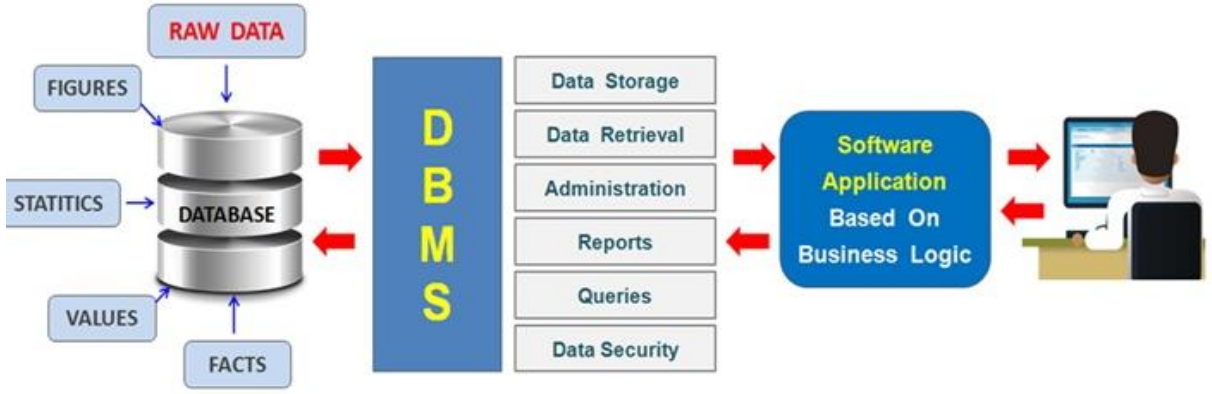
Levels of Data Abstraction

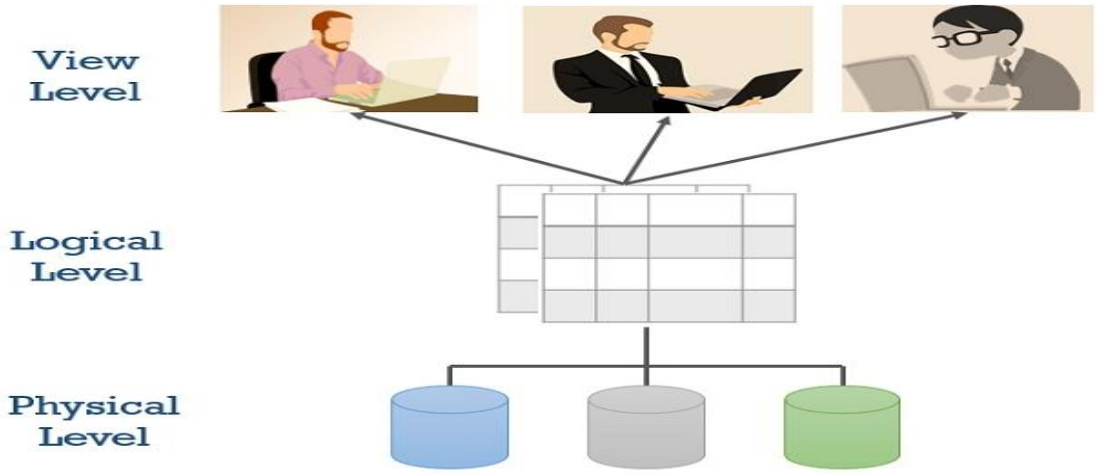
Data Abstraction

Data Abstraction:

Physical Level:

- It is the lowest level of abstraction that describes how the data are actually stored.
- The physical level describes complex low-level data structures in detail.
- It contains the definition of stored record and method of representing the data fields and access aid used.



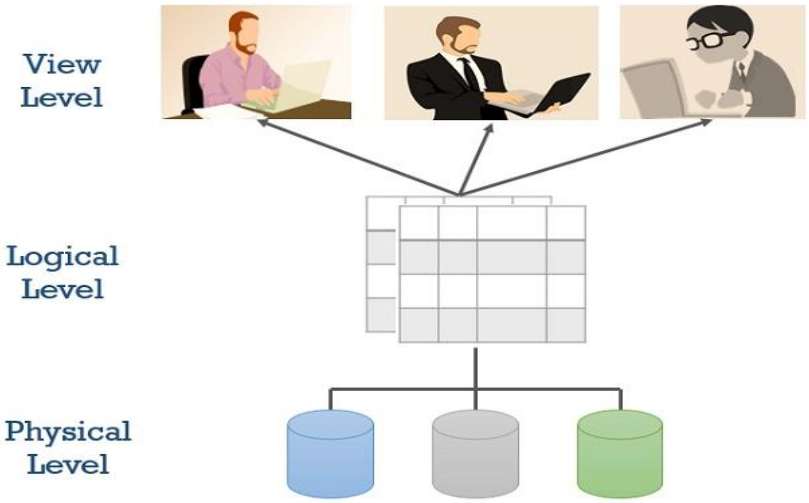


Data Abstraction

Data Abstraction:

Conceptual Level:

- It is the next higher level of abstraction that describes what data are stored in the database and what relationships exist among those data.
- It also contains the method of deriving the objects in the conceptual view from the objects in the internal view.

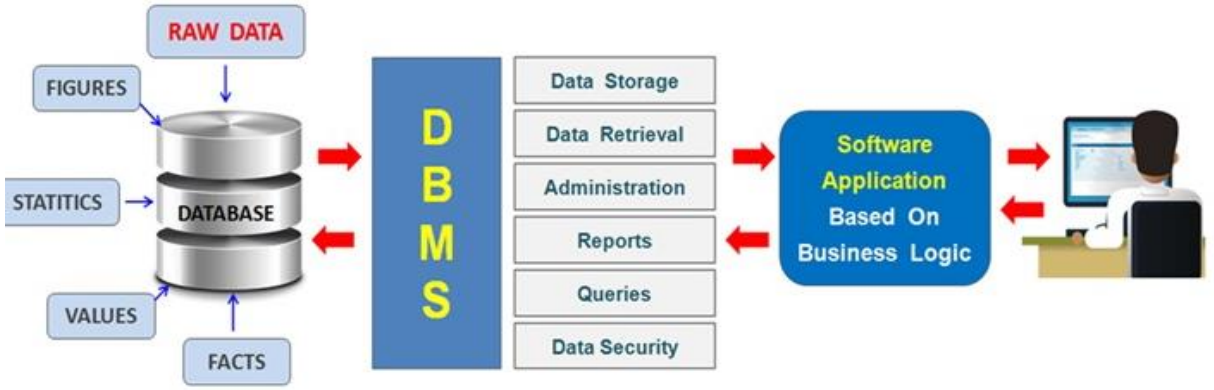


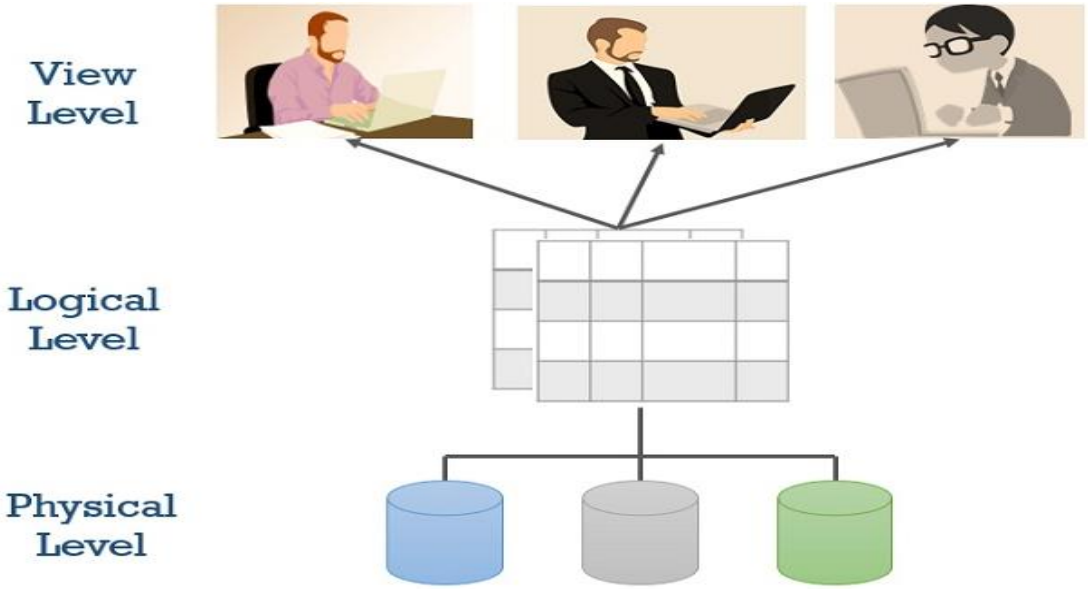
Data Abstraction

Data Abstraction:

View Level:

- It is the highest level of abstraction that describes only part of the entire database.
- It also contains the method of deriving the objects in the external view from the objects in the conceptual view.



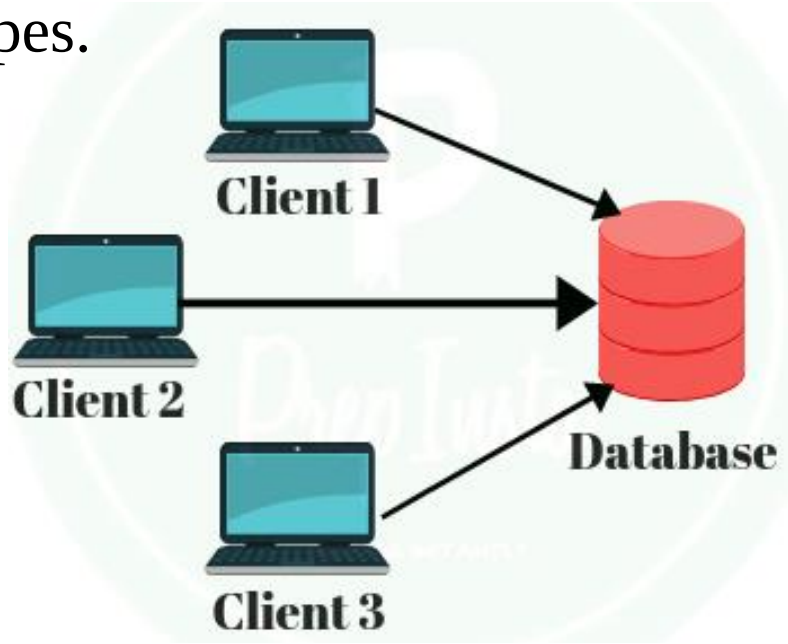


DBMS Architecture

DBMS Architecture:

The design of Database Management System highly depends on its architecture:

- It can be centralized or decentralized or hierarchical.
- Database architecture is logically divided into three types.
  - ***Logical one-tier in 1-tier Architecture***
  - ***Logical two-tier Client/Server Architecture.***
  - ***Logical three-tier Client/Server Architecture.***

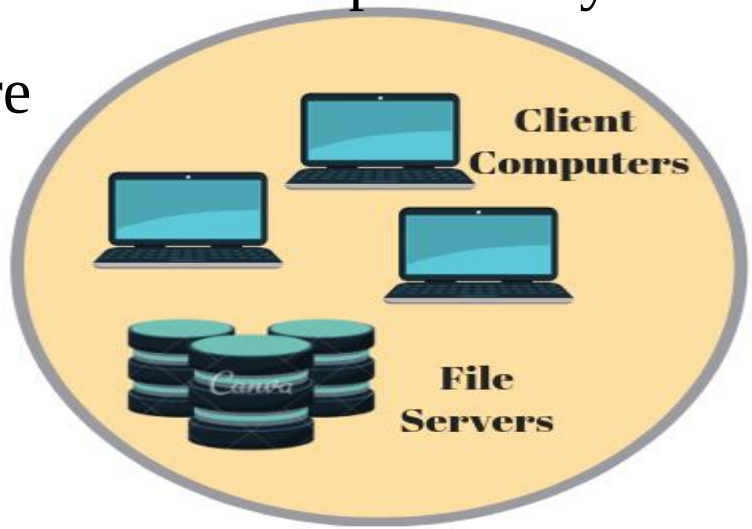


DBMS Architecture

Logical one-tier in 1-tier Architecture:

DBMS is the only entity where user directly sits on DBMS and uses it.

- Any changes done here will directly be on DBMS itself.
- It does not provide handy tools for end users and preferably database designers and programmers use single tier architecture



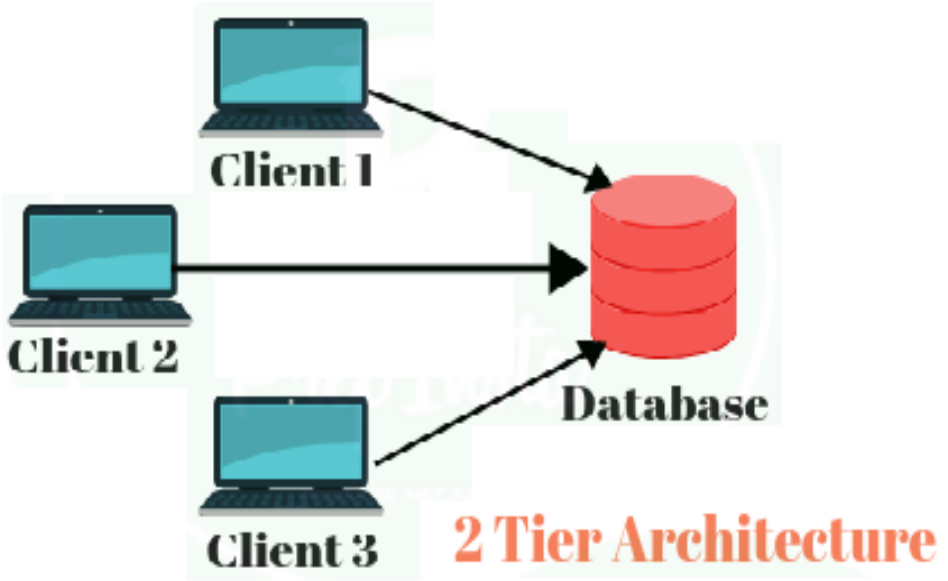
Single Tier Architecture

DBMS Architecture

Logical two-tier Client/Server Architecture:

Two-tier Client / Server architecture is used for User Interface program and Application Programs that runs on client side.

- An interface called ODBC (Open Database Connectivity) provides an API that allows client side program to call the DBMS.
- Most DBMS vendors provide ODBC drivers.
- A client program may connect to several DBMS's.

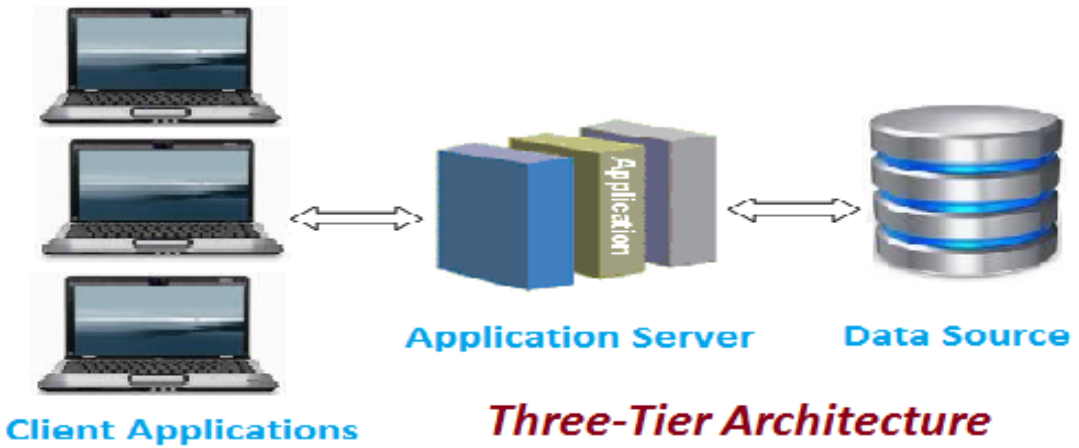


DBMS Architecture

Logical three-tier Client/Server Architecture:

Three-tier Client / Server database architecture is commonly used architecture for web applications. Intermediate layer called Application server or Web Server stores .

- The web connectivity software and the business logic (constraints) part of application used to access the right amount of data from the database server.
- This layer acts like medium for sending partially processed data between the database server and the client.



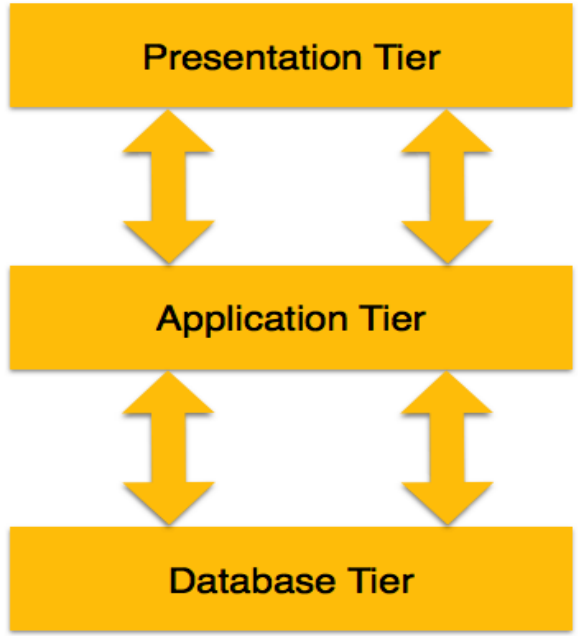
Three-Tier Architecture

DBMS Architecture

Logical three-tier Client/Server Architecture:

A 3-tier architecture separates its tiers from each other based on the complexity of the users and how they use the data present in the database.

- ***Database (Data) Tier*** – At this tier, the database resides along with its query processing languages
- ***Application (Middle) Tier*** – At this tier reside the application server and the programs that access the database. End-users are unaware of any existence of the database beyond the application.
- ***User (Presentation) Tier*** – End-users operate on this tier and they know nothing about any existence of the database beyond this layer. At this layer, multiple views of the database can be provided by the application.



Database Model

Database Model:

Data model is a collection of conceptual tools for describing data, data relationship, data semantics and constraints.

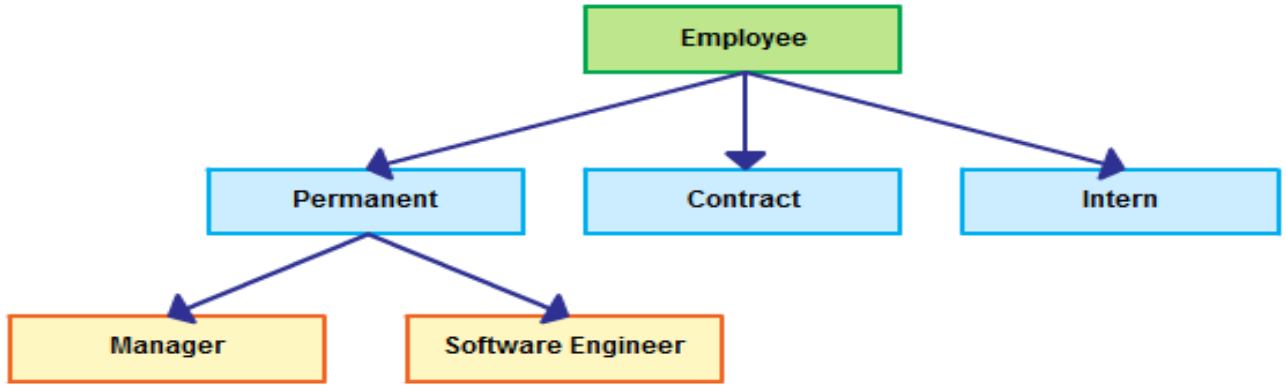
- Data model theory, which is a formal description of how data may be structured and used.
- Data model instance, which is a practical data model designed for a particular application.
- In history of database design, three models have been in use.
  - ***Hierarchical Model***
  - ***Network Model***
  - ***Relational Model***
  - ***Object-Oriented data model***
  - ***Object-Relational data Model***

Database Model

Hierarchical data model:

The Hierarchical data model organizes data in a tree structure. In this data model, data is represented by a collection of records and the relationships are represented by links.

- In this model, each entity has only one parent but can have several children.
- At the top of hierarchy, there is only one entity, which is called Root node.



Hierarchical database model

Database Model

Hierarchical data model:

Advantages:

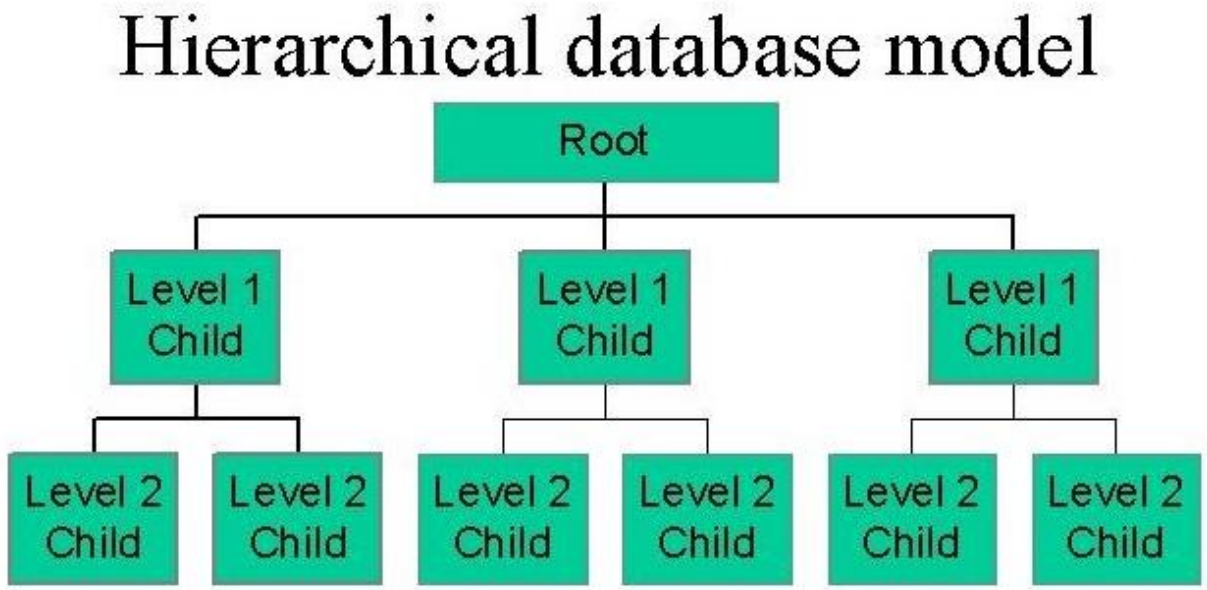
- **Simplicity:** The relationship between the various layers is logically simple.
- **Data Security:** The data security is provided by the DBMS.
- **Data Integrity:** There is always link between the parent segment and the child segment under it.
- **Efficiency:** It is very efficient because when the database contains a large number of one to many relationships and when the user requires large number of transaction.

Database Model

Hierarchical data model:

Disadvantages:

- Implementation complexity
- Database management problem
- Lack of structural Independence.
- Operational Anomalies

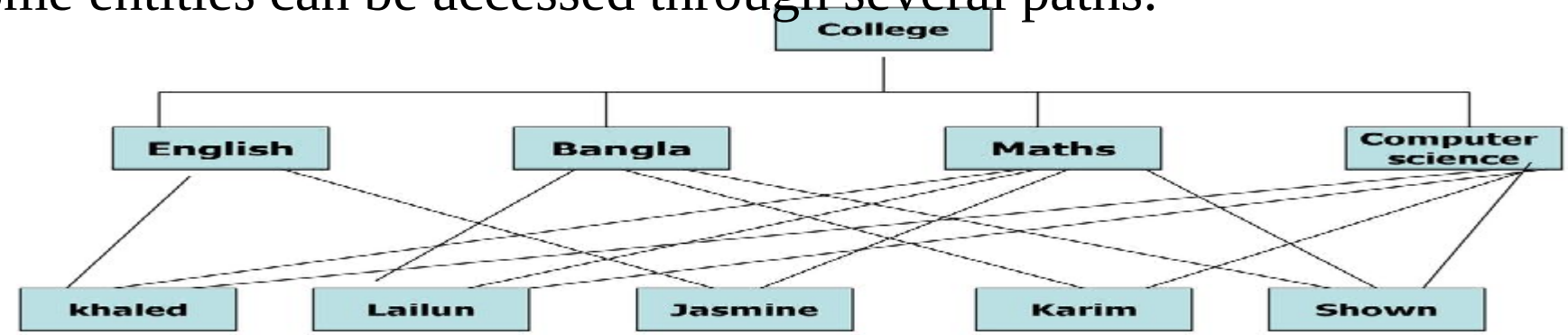


Database Model

Network data model:

In 1971, the Conference on Data Systems Languages (CODASYL) formally defined the network models. In this model, data is represented by a collection of records and the relationships are represented by links.

- Each record is collection of fields, which contains only one data value. A link is an association between two records. In the network model, entities are organized in a graph, in which some entities can be accessed through several paths.



The Network Database Model

Database Model

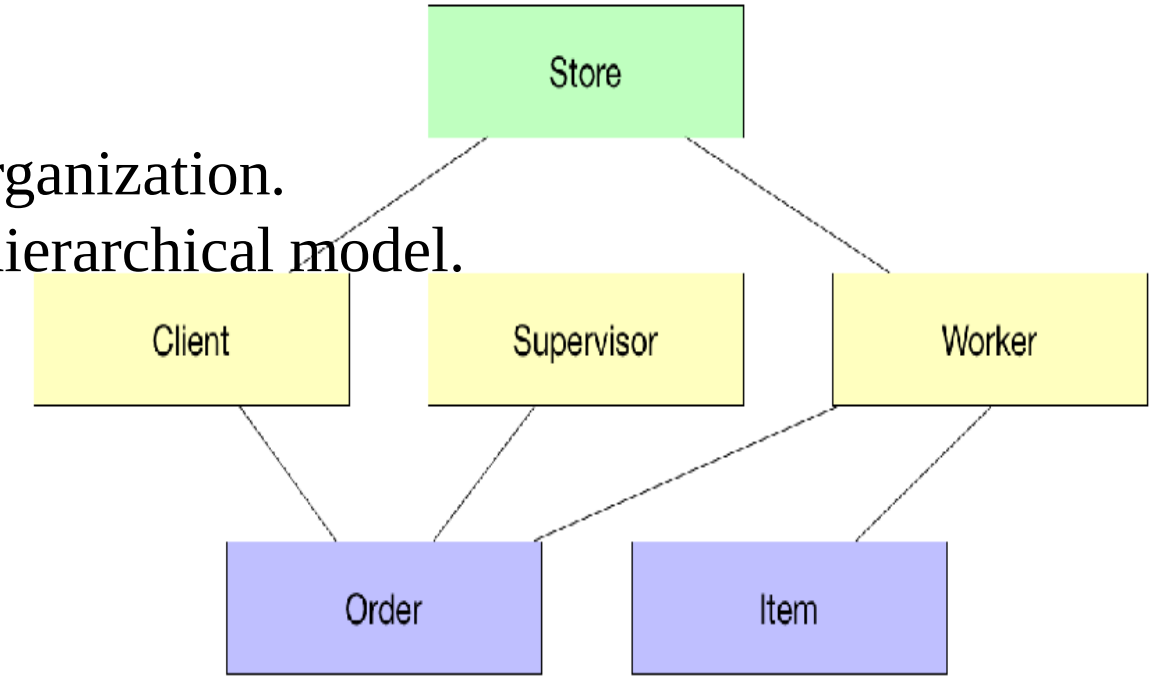
Network data model:

Advantages:

- It is simple and easy to implement.
- It can handle many relationships within the organization.
- It has better data independence compared to hierarchical model.

Disadvantages:

- More complex system of database structure
- Lack of structural dependence.



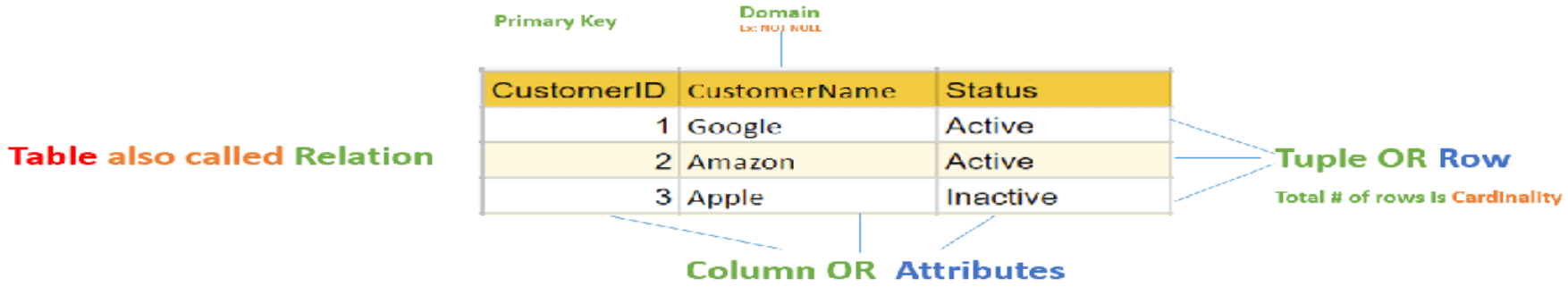
The Network Database Model

Database Model

Relation Data Model:

E.F Codd developed the relation data model in 1970. Unlike, hierarchical and network model, there are no physical links. All data is maintained in the form of tables consisting of rows and columns.

- Each row (record) represents an entity and a column (field) represents an attribute of the entity.
- In this model, data is organized in two-dimensional tables called relations.
- The tables or relations are related to each other.



Database Model

Object oriented data model:

Object oriented data model is based upon real world situations. These situations are represented as objects, with different attributes. All these object have multiple relationships between them.

➢ Elements of Object oriented data model

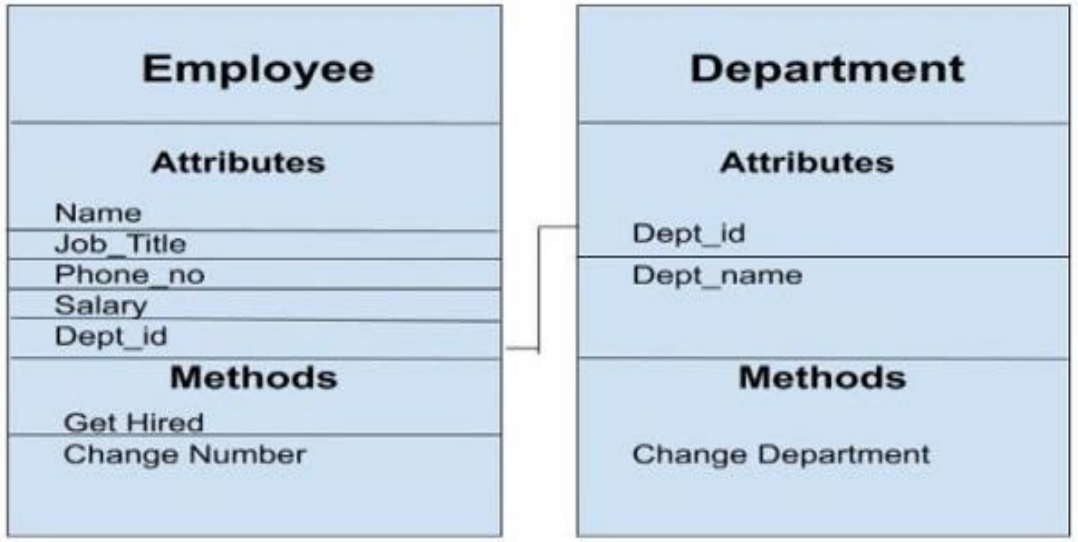
- ***Objects:*** The real world entities and situations are represented as objects in the Object oriented database model.
- ***Attributes and Method:*** Every object has certain characteristics. These are represented using Attributes. The behavior of the objects is represented using Methods.
- **Class:** Similar attributes and methods are grouped together using a class. An object can be called as an instance of the class.
- ***Inheritance:*** A new class can be derived from the original class. The derived class contains attributes and methods of the original class as well as its own.

Database Model

Object oriented data model:

Each object data and relationships are contained in a single unit. The attributes are Name, job_title. Methods are used to perform the operation with the help of attributes.

- ***Advantages***
  - Semantic content is added.
  - Support for complex objects.
  - Inheritance promotes data integrity.
  - Visual representation includes semantic content.
- ***Disadvantages***
  - It is a complex navigational system.
  - Slow development of standards.
  - High system overheads.
  - Slow transactions.



Database Model

Object-Relational data Model:

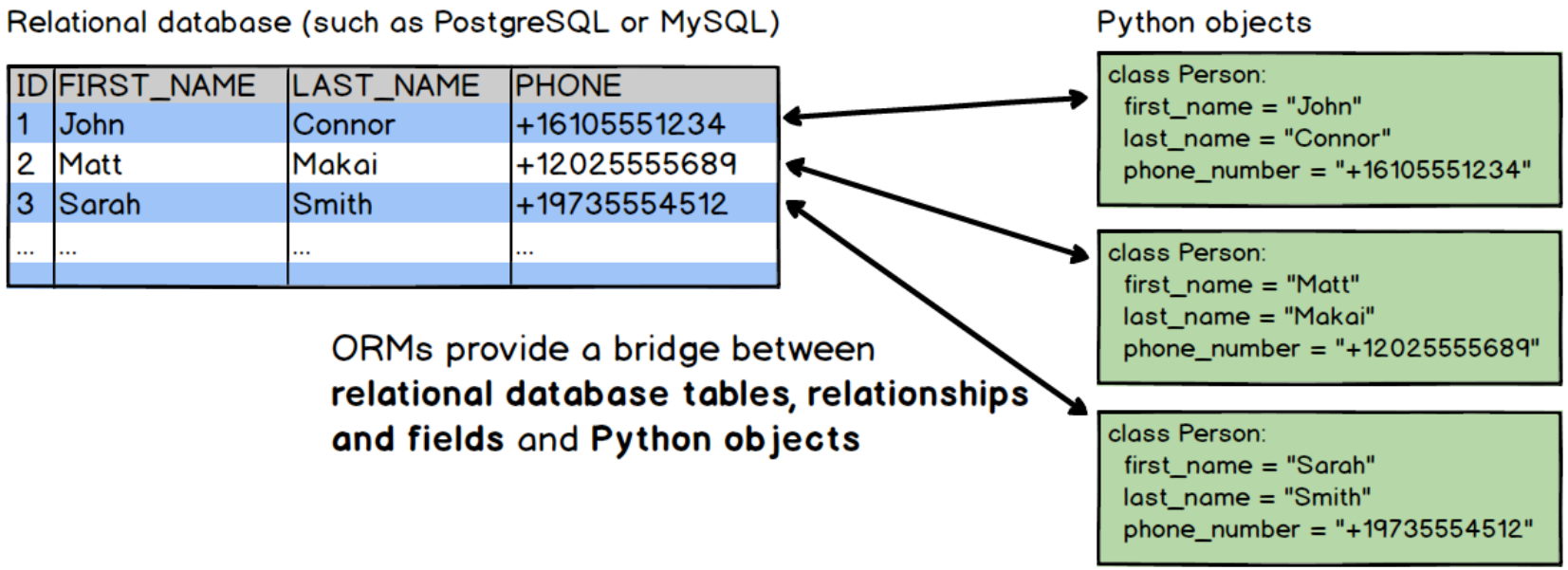
An object-relational database (ORD) is a database management system (DBMS) that's composed of both a relational database (RDBMS) and an object-oriented database (OODBMS).

- Elements of ORD supports the basic components of any object-oriented database model in its schemas and the query language used, such as objects, classes and inheritance.
- One of ORD's aims is to bridge the gap between conceptual data modeling techniques for relational and object-oriented databases like the entity-relationship diagram (ERD) and object-relational mapping (ORM).
- It also aims to connect the divide between relational databases and the object-oriented modeling techniques that are usually used in programming languages like Java, C#, Python and C++.

Database Model

Object-Relational data Model:

An object-relational database (ORD) is a database management system (DBMS) that's composed of both a relational database (RDBMS) and an object-oriented database (OODBMS).



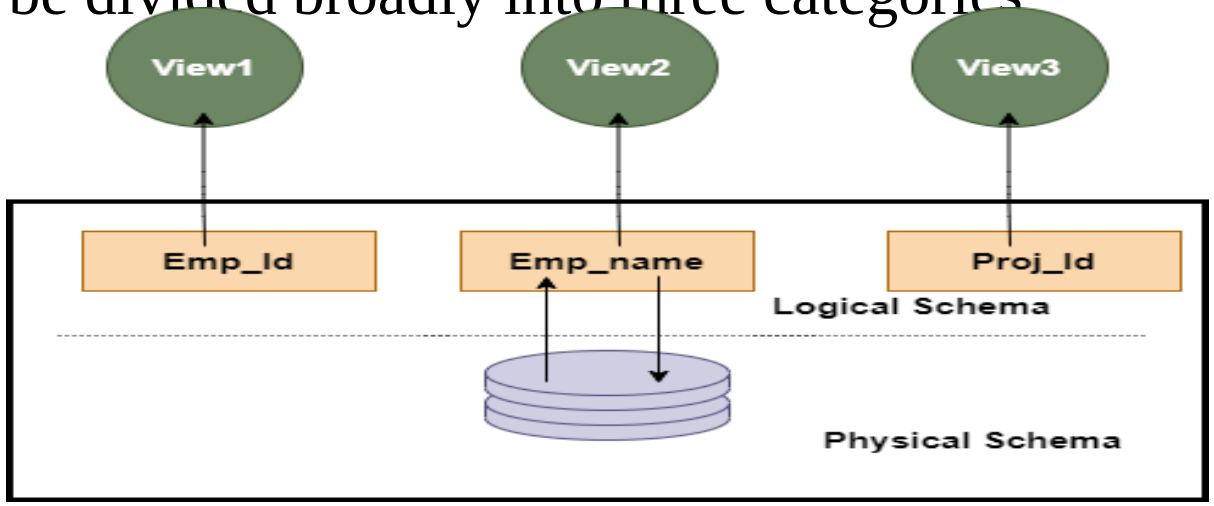
Database Schema

Database Schema:

A database schema is the skeleton structure that represents the logical view of the entire database.

- It defines how the data is organized and how the relations among them are associated.
- It formulates all the constraints that are to be applied on the data.
- A database schema can be divided broadly into three categories –

  - ***Logical Schema***
  - ***Physical Schema***
  - ***View Schema***



Database Schema

Database Schema:

It contains a descriptive detail of the database, which can be depicted by means of schema diagrams.

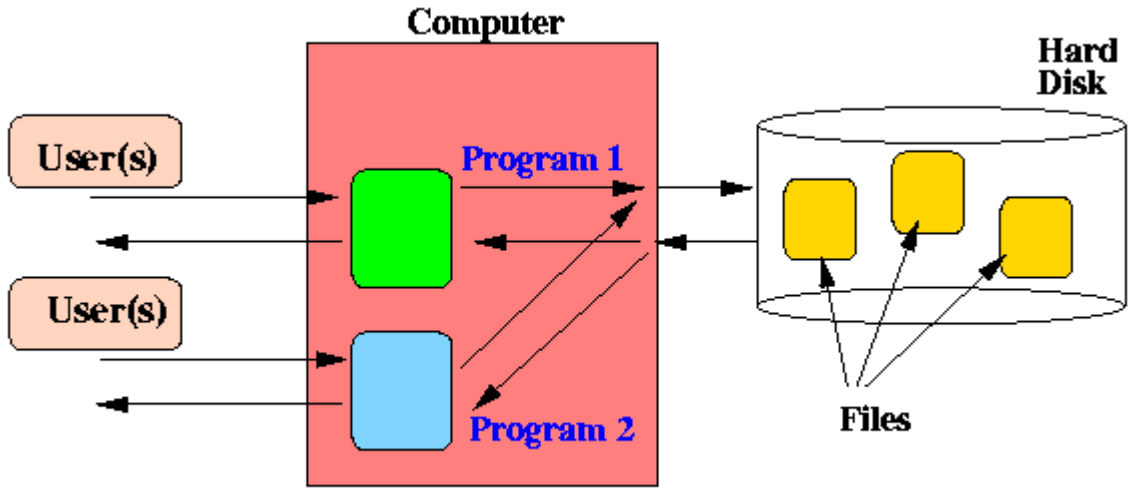
- ***Physical Database Schema*** – This schema pertains to the actual storage of data and its form of storage like files, indices, etc. It defines how the data will be stored in a secondary storage.
- ***Logical Database Schema*** – This schema defines all the logical constraints that need to be applied on the data stored. It defines tables, views, and integrity constraints.
- ***View Schema-*** The view level design of a database is known as view schema. This schema generally describes the end-user interaction with the database systems.

Data Independence

Data Independence:

The capacity to change data at one layer does not affect the data at another layer is called data independence.

- Two types of data independence are
  - Physical Data Independence
    - ✓ File Organization
    - ✓ Data Model
  - Logical Data Independence
    - ✓ Relational Data Model
    - ✓ Entity Relationship

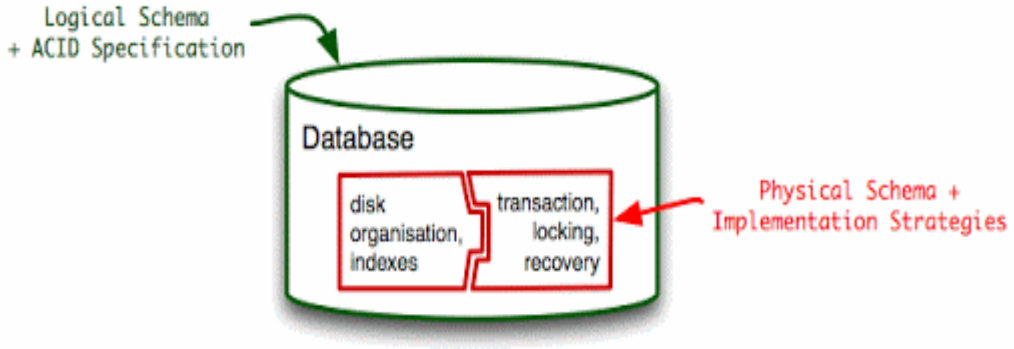


Data Independence

Physical data independence :

It is the capacity to change the internal level without having to change either the schemas at the conceptual or external level.

- Changes to the internal schema may be needed because some physical files had to be reorganized.
- Physical data independence refers to the data insulation of an application from the physical storage structure only, it is easier to achieve than logical data independence.
- The physical data independence are:
  - ***File Organization***
  - ***Database Architecture***
  - ***Database Models***

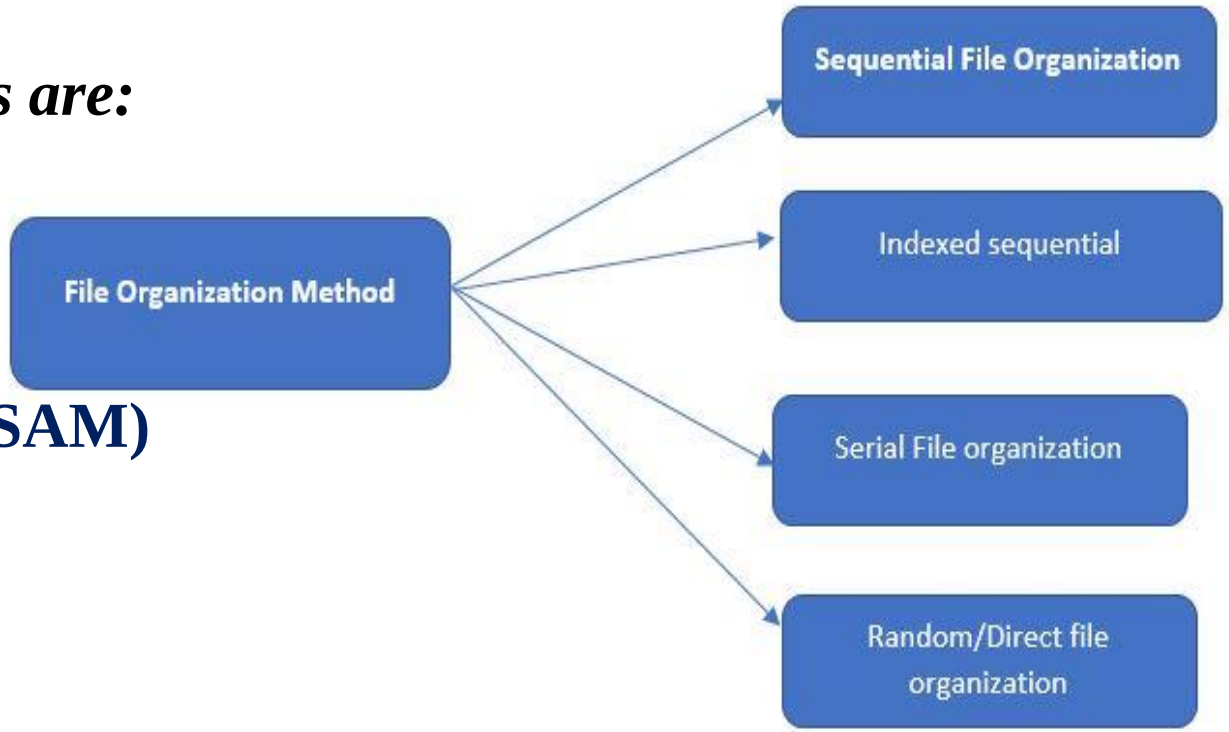


File organization

File organization Methods:

The difference file organization methods are:

- **Serial File Organization:**
- **Direct Access File Organization**
- **Index sequential file organization (ISAM)**



File organization

File organization Methods:

The difference between serial and direct access file organization.

- **Serial File Organization:**
  - Organization is continuous and simple.
  - Data processing, which requires the use of all records, is best suited to use this method.
- **Direct Access File Organization**
  - The type of storage device used is comparatively expensive.
  - It is less efficient in the usage of storage space compared to the sequential organization.

File organization

Index sequential file organization (ISAM):

The index sequential file organization is a combination of Sequential file organization and an Index file. It is also referred as ISAM (indexed sequential access method).

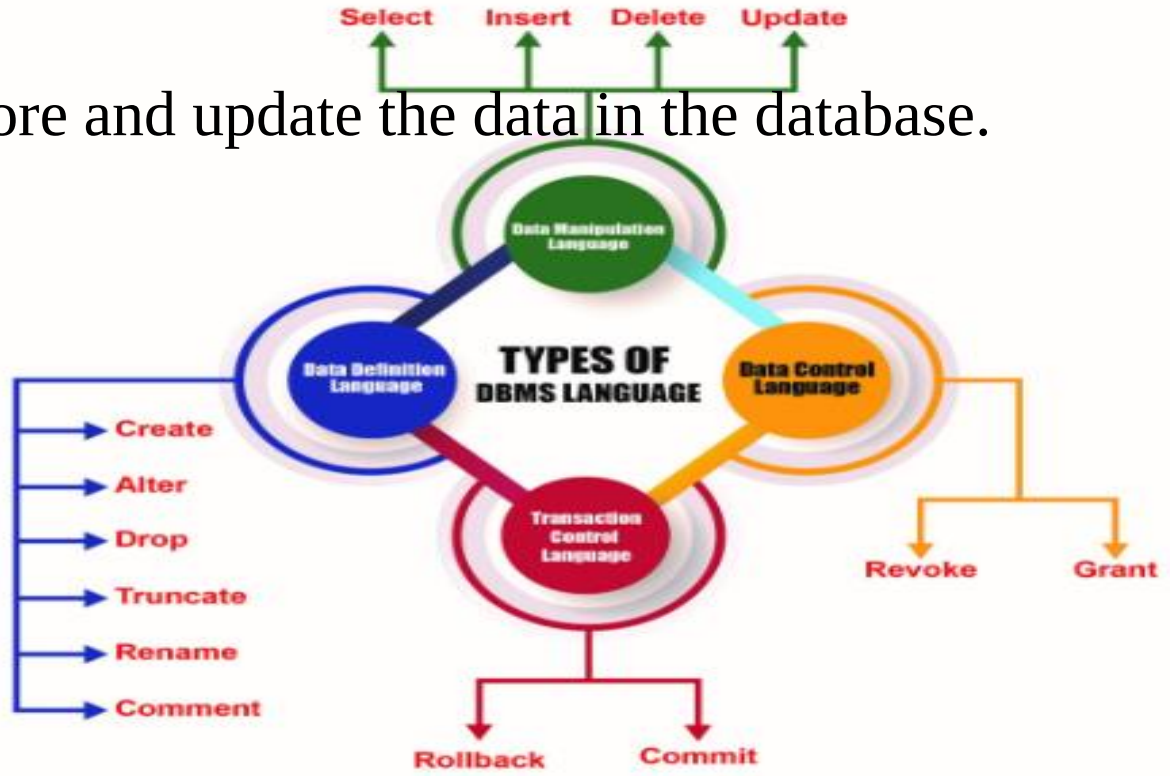
- Data is stored physically in adjacent storage locations and there exists a logical relationship among the data stored by using ordering field. An additional file called as Index file would be created, which contains n number of records.
- Each record of index file has two fields:
  - The field is of the same data type as the ordering key field and
  - The second field is a pointer to a disk block (a block address).

Database Language

Database Language:

A DBMS has appropriate languages and interfaces to express database queries and updates.

- Database languages can be used to read, store and update the data in the database.
- ***Types of Database Language***
  - ***Data Definition Language (DDL)***
  - ***Data Manipulation Language (DML)***
  - ***Data Control Language (DCL)***
  - ***Transaction Control Language (TCL)***



Database Language

Data Definition Language:

DDL stands for Data Definition Language. It is used to define database structure or pattern.

- It is used to create schema, tables, indexes, constraints, etc. in the database.
- ***Some tasks that come under DDL:***
  - ***Create:*** It is used to create objects in the database.
  - ***Alter:*** It is used to alter the structure of the database.
  - ***Drop:*** It is used to delete objects from the database.
  - ***Truncate:*** It is used to remove all records from a table.
  - ***Rename:*** It is used to rename an object.
  - ***Comment:*** It is used to comment on the data dictionary.

ColumnN datatype);
ColumnN datatype);

Database Language

Data Definition Language:

***Create:* It is used to create a table or a database.**

- Syntax: ***CREATE DATABASE DatabaseName;***
- Example: *CREATE DATABASE Employee;*
- The 'CREATE TABLE' Statement is used to create a table.
- Syntax

CREATE TABLE TableName (
Column1 datatype,
Column2 datatype,
Column3 datatype,
.... ColumnN datatype);

Example:

CREATE TABLE Employee_Info
(
EmployeeID int,
EmployeeName varchar(255),
Emergency ContactName varchar(255),
PhoneNumber int,
Address varchar(255),
);

ColumnN datatype);
ColumnN datatype);

Database Language

Data Definition Language:

***Drop:* It is used to delete a table or a database.**

- ***'DROP DATABASE' Statement:*** *Is used to drop an existing database. When you use this statement, complete information present in the database will be lost.*
  - **Syntax : *DROP DATABASE DatabaseName;***
  - **Example : *DROP DATABASE Employee;***
- ***'DROP TABLE' Statement* :** *This statement is used to drop an existing table. When you use this statement, complete information present in the table will be lost.*
  - ***Syntax : DROP TABLE TableName;***
  - ***Example : DROP Table Employee_Info;***

ColumnN datatype);
ColumnN datatype);

Database Language

Data Definition Language:

- ***TRUNCATE :*** *This command is used to delete the information present in the table but does not delete the table. So, once you use this command, your information will be lost, but not the table.*
  - ***Syntax : TRUNCATE TABLE TableName;***
  - ***Example : TRUNCATE Table Employee_Info;***
- ***ALTER :*** *This command is used to delete, modify or add constraints or columns in an existing table.*
  - ***Syntax : ALTER TABLE TableName***
    ADD ColumnName Datatype;
    Example: ALTER TABLE Employee_Info
    ADD BloodGroup varchar(255);

    ALTER TABLE TableName
    DROP COLUMN ColumnName;
    Example: ALTER TABLE Employee_Info
    DROP COLUMN BloodGroup ;

Database Language

Data Manipulation Language:

DML stands for Data Manipulation Language. It is used for accessing and manipulating data in a database. It handles user requests in the database.

- ***some tasks that come under DML:***
  - ***Select:*** It is used to retrieve data from a database.
  - ***Insert:*** It is used to insert data into a table.
  - ***Update:*** It is used to update existing data within a table.
  - ***Delete:*** It is used to delete all records from a table.
  - ***Merge:*** It performs UPSERT operation, i.e., insert or update operations.
  - ***Call:*** It is used to call a structured query language or a Java subprogram.
  - ***Explain Plan:*** It has the parameter of explaining data.
  - ***Lock Table:*** It controls concurrency.

Database Language

Data Manipulation Language:

DML stands for Data Manipulation Language. It is used for accessing and manipulating data in a database. It handles user requests in the database.

- ***some tasks that come under DML:***
  - ***Select:*** It is used to retrieve data from a database.

    *SQL> Select * from students;*
  - ***Insert:*** It is used to insert data into a table.

    SQL> Insert into students values(116, 'SUNNY',82,85,85,83,82,93);
  - ***Update:*** It is used to update existing data within a table.

    SQL> UPDATE Students SET Perc_marks=Total/6.0 WHERE student_id>0;
  - ***Delete:*** It is used to delete all records from a table.

    SQL>DELETE from students WHERE student_id=10;

Database Language

Data Control Language:

DCL stands for Data Control Language. It is used to retrieve the stored or saved data.

- The DCL execution is transactional. It also has rollback parameters.
- ***Some tasks that come under DCL:***
  - ***Grant:*** It is used to give user access privileges to a database.
  - ***Revoke:*** It is used to take back permissions from the user.

- There are the following operations which have the authorization of Revoke:
  - *CONNECT, INSERT, USAGE, EXECUTE, DELETE, UPDATE and SELECT.*

Database Language

Transaction Control Language:

TCL is used to run the changes made by the DML statement.

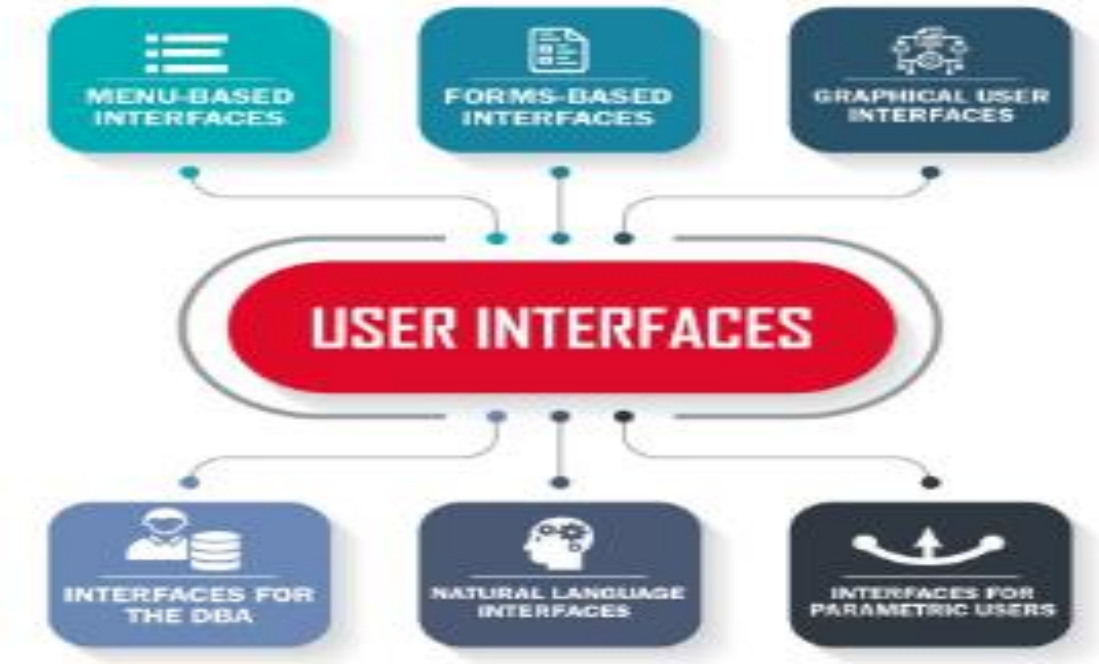
- TCL can be grouped into a logical transaction..
- ***Some tasks that come under TCL:***
  - ***Commit:*** It is used to save the transaction on the database.
  - ***Rollback:*** It is used to restore the database to original since the last Commit.

User-Interfaces

User-Interfaces:

DBMS is essentially a set of applications which support access to the databases

- DBMS allows its user to access the data from database using various user-interfaces provided by DBMS:

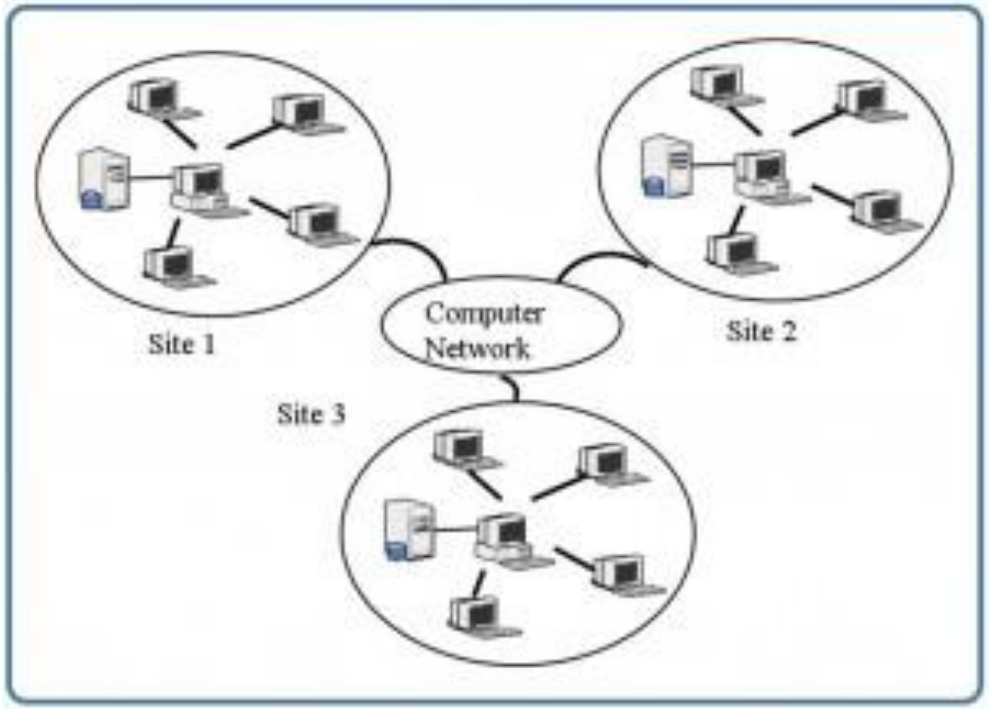


DBMS Classification

Classification of DBMS:

Database management systems can be classified based on several criteria, such as

- **The data model**
  - *Traditional models*
  - *Object-oriented data model*
  - *Relational model*
- **Numbers of Users**
  - *Single-user database system*
  - *Multiuser database system*
- **Database distribution**
  - *Centralized systems*
  - *Distributed database system*



Discussion

Queries ?

Prof. K. Adisesha

9449081542

Thank you