UNIT - V

Distributed object Database Management Systems: Fundamental object concepts and models, object distributed design, architectural issues, object management, distributed object storage, object query Processing.

Object Oriented Data Model: Inheritance, object identity, persistent programming languages, persistence of objects, comparison OODBMS and ORDBMS

Distributed object Database Management Systems:

A **database** is an ordered collection of related data that is built for a specific purpose. A database may be organized as a collection of multiple tables, where a table represents a real world element or entity. Each table has several different fields that represent the characteristic features of the entity.

For example, a company database may include tables for projects, employees, departments, products and financial records. The fields in the Employee table may be Name, Company_Id, Date_of_Joining, and so forth.

A **database management system** is a collection of programs that enables creation and maintenance of a database. DBMS is available as a software package that facilitates definition, construction, manipulation and sharing of data in a database. Definition of a database includes description of the structure of a database. Construction of a database involves actual storing of the data in any storage medium. Manipulation refers to the retrieving information from the database, updating the database and generating reports. Sharing of data facilitates data to be accessed by different users or programs.

Examples of DBMS Application Areas

- Automatic Teller Machines
- Train Reservation System
- Employee Management System
- Student Information System

Examples of DBMS Packages

- MySQL
- Oracle
- SQL Server
- dBASE
- FoxPro
- PostgreSQL, etc.

Database Schemas

A database schema is a description of the database which is specified during database design and subject to infrequent alterations. It defines the organization of the data, the relationships among them, and the constraints associated with them.

Databases are often represented through the **three-schema architecture** or **ANSISPARC architecture**. The goal of this architecture is to separate the user application from the physical database. The three levels are −

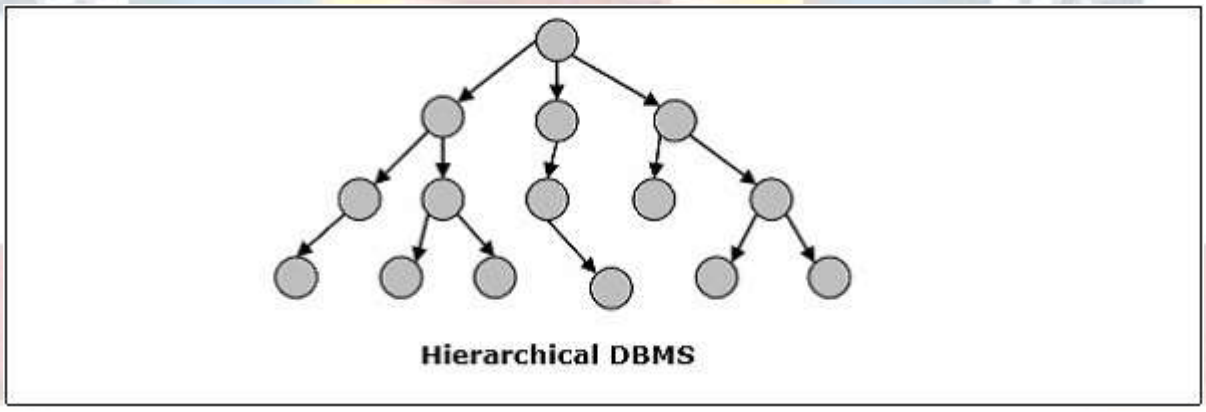
- **Internal Level having Internal Schema** − It describes the physical structure, details of internal storage and access paths for the database.
- **Conceptual Level having Conceptual Schema** − It describes the structure of the whole database while hiding the details of physical storage of data. This illustrates the entities, attributes with their data types and constraints, user operations and relationships.
- **External or View Level having External Schemas or Views** − It describes the portion of a database relevant to a particular user or a group of users while hiding the rest of database.

Types of DBMS

There are four types of DBMS.

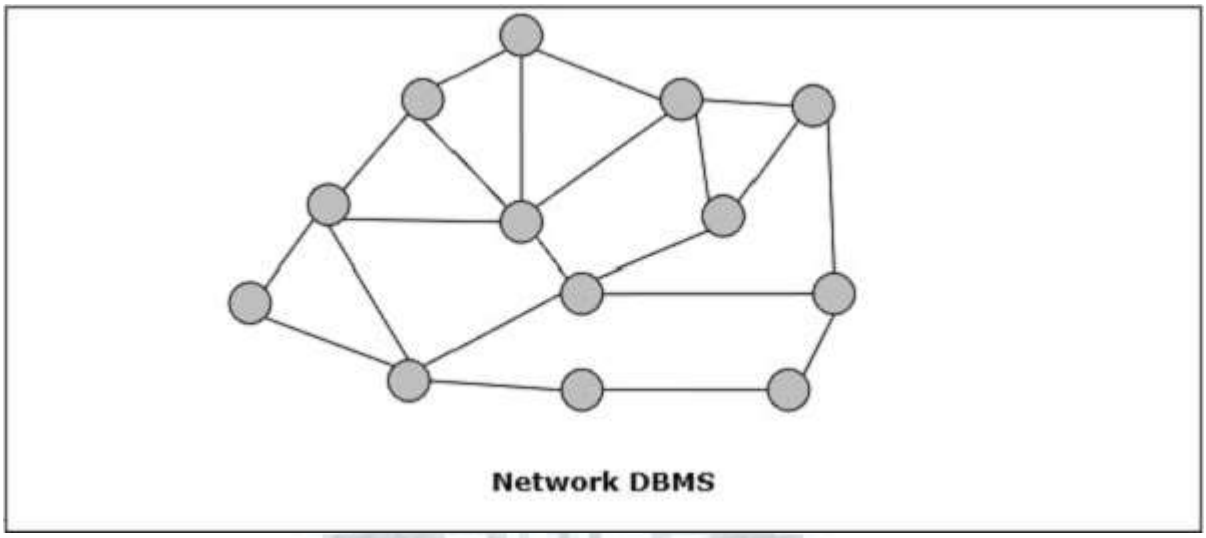
Hierarchical DBMS

In hierarchical DBMS, the relationships among data in the database are established so that one data element exists as a subordinate of another. The data elements have parent-child relationships and are modelled using the "tree" data structure. These are very fast and simple.



Network DBMS

Network DBMS in one where the relationships among data in the database are of type many-to-many in the form of a network. The structure is generally complicated due to the existence of numerous many-to-many relationships. Network DBMS is modelled using "graph" data structure.



Network DBMS

Relational DBMS

In relational databases, the database is represented in the form of relations. Each relation models an entity and is represented as a table of values. In the relation or table, a row is called a tuple and denotes a single record. A column is called a field or an attribute and denotes a characteristic property of the entity. RDBMS is the most popular database management system.

For example − A Student Relation −

Field

Tuple →

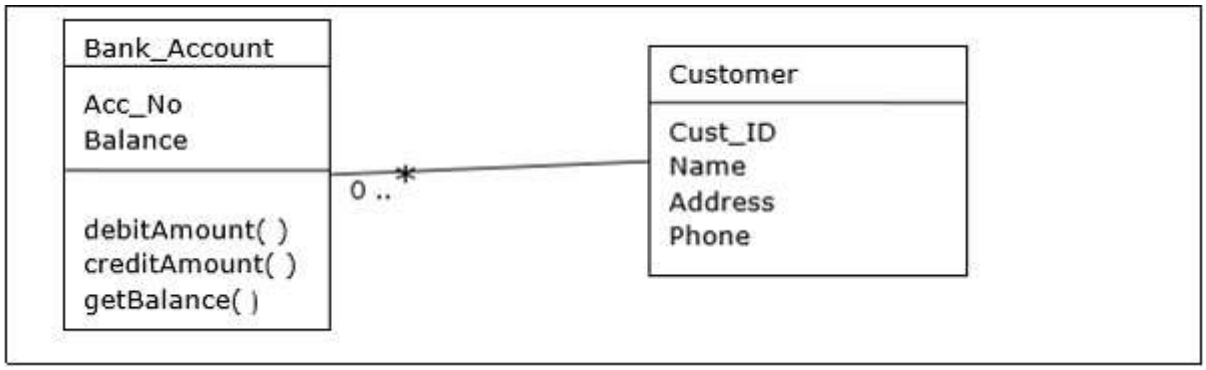
S_Id	Name	Year	Stream
1	Ankit Jha	1	Computer Science
2	Pushpa Mishra	2	Electronics
5	Ranjini Iyer	2	Computer Science

Fundamental object concepts and models:

Object Oriented DBMS

Object-oriented DBMS is derived from the model of the object-oriented programming paradigm. They are helpful in representing both consistent data as stored in databases, as well as transient data, as found in executing programs. They use small, reusable elements called objects. Each object contains a data part and a set of operations which works upon the data. The object and its attributes are accessed through pointers instead of being stored in relational table models.

For example − A simplified Bank Account object-oriented database −



Distributed DBMS

A distributed database is a set of interconnected databases that is distributed over the computer network or internet. A Distributed Database Management System (DDBMS) manages the distributed database and provides mechanisms so as to make the databases transparent to the users. In these systems, data is intentionally distributed among multiple nodes so that all computing resources of the organization can be optimally used.

Operations on DBMS

The four basic operations on a database are Create, Retrieve, Update and Delete.

- **CREATE** database structure and populate it with data − Creation of a database relation involves specifying the data structures, data types and the constraints of the data to be stored.

Example − SQL command to create a student table −

```
CREATE TABLE STUDENT (
  ROLL INTEGER PRIMARY KEY,
  NAME VARCHAR2(25),
  YEAR INTEGER,
  STREAM VARCHAR2(10)
);
```

- Once the data format is defined, the actual data is stored in accordance with the format in some storage medium.

Example SQL command to insert a single tuple into the student table −

```
INSERT INTO STUDENT ( ROLL, NAME, YEAR, STREAM)
VALUES ( 1, 'ANKIT JHA', 1, 'COMPUTER SCIENCE');
```

- **RETRIEVE** information from the database – Retrieving information generally involves selecting a subset of a table or displaying data from the table after some computations have been done. It is done by querying upon the table.

Example − To retrieve the names of all students of the Computer Science stream, the following SQL query needs to be executed −

```
SELECT NAME FROM STUDENT

WHERE STREAM = 'COMPUTER SCIENCE';
```

- **UPDATE** information stored and modify database structure – Updating a table involves changing old values in the existing table's rows with new values.

Example − SQL command to change stream from Electronics to Electronics and Communications −

```
UPDATE STUDENT

SET STREAM = 'ELECTRONICS AND COMMUNICATIONS'

WHERE STREAM = 'ELECTRONICS';
```

- Modifying database means to change the structure of the table. However, modification of the table is subject to a number of restrictions.

Example − To add a new field or column, say address to the Student table, we use the following SQL command −

```
ALTER TABLE STUDENT

ADD (ADDRESS VARCHAR2 (50));
```

- **DELETE** information stored or delete a table as a whole – Deletion of specific information involves removal of selected rows from the table that satisfies certain conditions.

Example − To delete all students who are in 4^{th} year currently when they are passing out, we use the SQL command −

```
DELETE FROM STUDENT

WHERE YEAR = 4;
```

- Alternatively, the whole table may be removed from the database.

Example − To remove the student table completely, the SQL command used is −

```
DROP TABLE STUDENT;
```

Object distributed design:

In distributed computing, distributed objects[citation needed] are objects (in the sense of object-oriented programming) that are distributed across different address spaces, either in different processes on the same computer, or even in multiple computers connected via a network, but which work together by sharing data and invoking methods. This often involves location transparency, where remote objects appear the same as local objects. The main method of distributed object communication is with remote method invocation, generally by

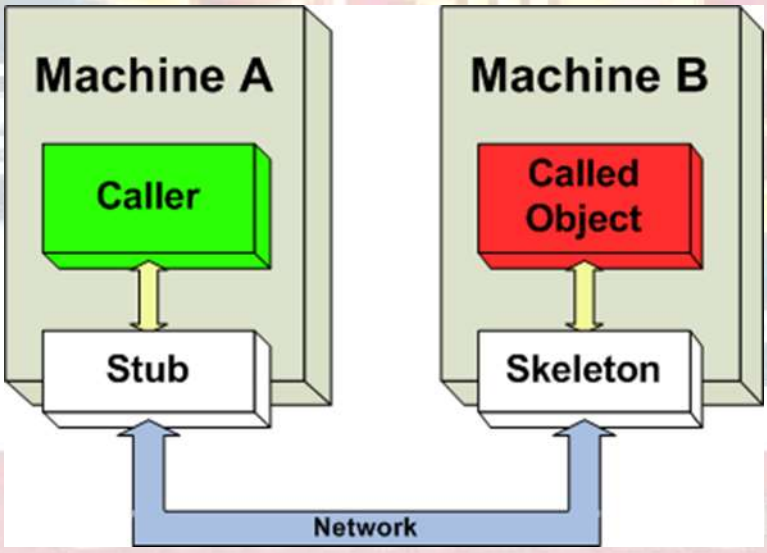
message-passing: one object sends a message to another object in a remote machine or process to perform some task. The results are sent back to the calling object.

Distributed objects were popular in the late 1990s and early 2000s, but have since fallen out of favour.[1]

The term may also generally refer to one of the extensions of the basic object concept used in the context of distributed computing, such as replicated objects or live distributed objects.

Replicated objects are groups of software components (replicas) that run a distributed multi-party protocol to achieve a high degree of consistency between their internal states, and that respond to requests in a coordinated manner. Referring to the group of replicas jointly as an object reflects the fact that interacting with any of them exposes the same externally visible state and behavior.

Live distributed objects (or simply live objects)[2] generalize the replicated object concept to groups of replicas that might internally use any distributed protocol, perhaps resulting in only a weak consistency between their local states. Live distributed objects can also be defined as running instances of distributed multi-party protocols, viewed from the object-oriented perspective as entities that have distinct identity, and that can encapsulate distributed state and behavior.



Architectural issues:

Architectural issues in distributed data base systems
Information systems. Data management systems. Database design and models. Database management system engines. Parallel and distributed DBMSs.
Software and its engineering. Software organization and properties. Software system structures. Software architectures.

Object management:

A Object management system is a collection of programs that enables creation and maintenance of a database. DBMS is available as a software package that facilitates

definition, construction, manipulation and sharing of data in a database. Definition of a database includes description of the structure of a database. Construction of a database involves actual storing of the data in any storage medium. Manipulation refers to the retrieving information from the database, updating the database and generating reports. Sharing of data facilitates data to be accessed by different users or programs.

Distributed object storage:

Object storage, also known as object-based storage, is a strategy that manages and manipulates data storage as distinct units, called objects. These objects are kept in a single storehouse and are not ingrained in files inside other folders. Instead, object storage combines the pieces of data that make up a file, adds all its relevant metadata to that file, and attaches a custom identifier.

Object storage adds comprehensive metadata to the file, eliminating the tiered file structure used in file storage, and places everything into a flat address space, called a storage pool. This metadata is key to the success of object storage in that it provides deep analysis of the use and function of data in the storage pool.

Object query Processing:

The main application of distributed query processing are classic distributed databases allowing to store and query data at different sites transparently. ... Typical examples of such systems are object-oriented database systems but also middleware technologies such as application servers.