DATABASE SYSTEMS II
(IS313P)

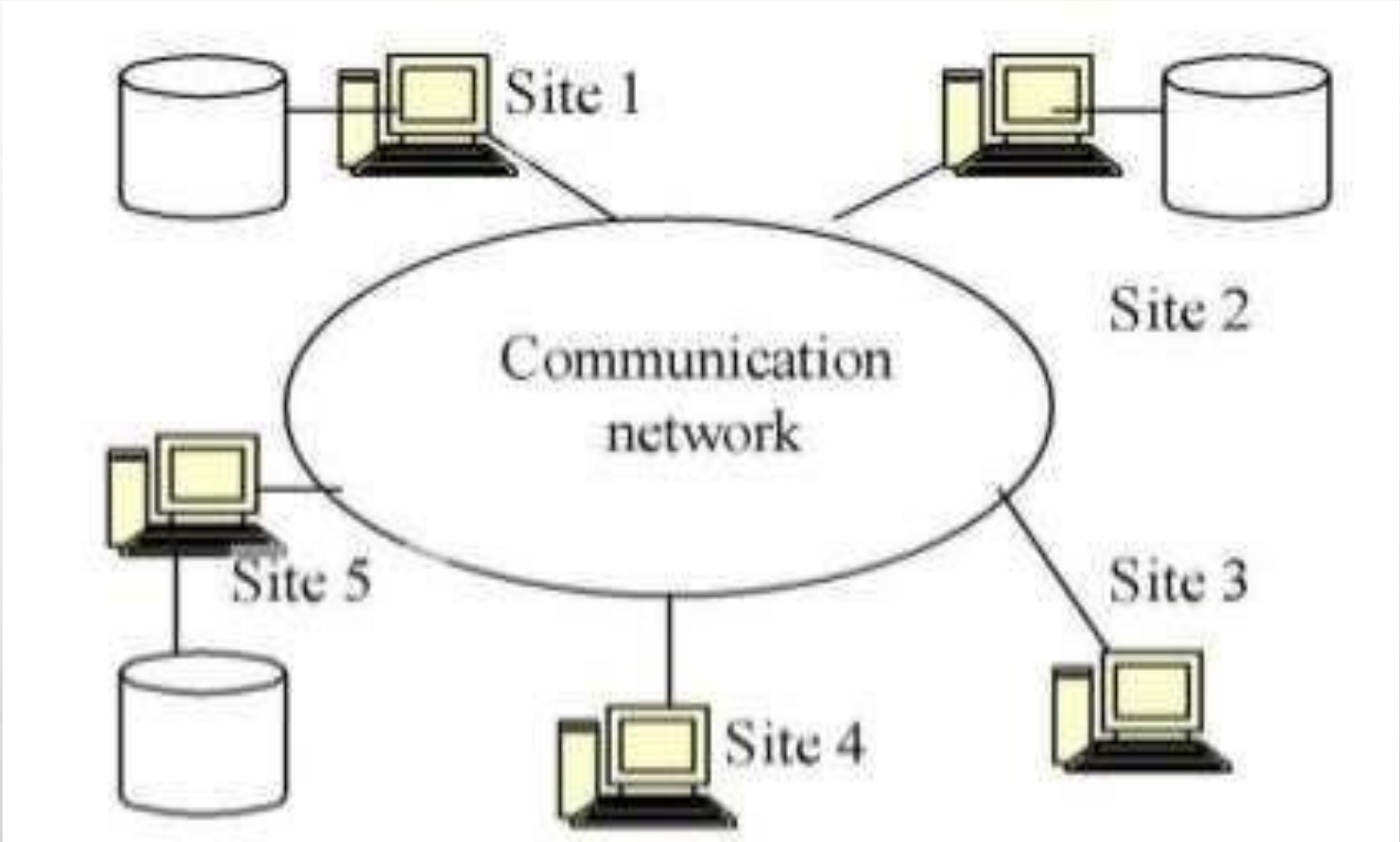
LECTURE NO. 10

13/12/2023

Introduction to Distributed & Object- Oriented Database Systems

Distributed Database

- A distributed database is basically a database that is **not limited to one system, it is spread over different sites**, such as over ***a network of computers***.
- A distributed database system is located on **various sites that don't share physical components.**
- This may be required when a particular database needs to be accessed by various users globally. It needs to be managed such that for the users **it looks like one single database.**


Site 1
Site 2
Communication
network
Site 5
Site 3
Site 4

Types:

1. Homogeneous Database:

In a homogeneous database, all different sites store database identically. The operating system, database management system, and the data structures used – all are the same at all sites. Hence, they're easy to manage.

2. Heterogeneous Database:

In a heterogeneous distributed database, different sites can use different schema and software. Different computers may use a different operating system ,different database application, and they may even use different data models for the database. Also, a particular site might be completely unaware of the other sites. Hence, that can lead to problems in query processing. So, translations are required for different sites to communicate.

Distributed Data Storage

There are 2 ways in which data can be stored on different sites.

1. Replication:

- In this approach, the entire relationship **is stored redundantly at 2 or more sites.**
- If the entire database is available at all sites, it is a fully redundant database. Hence, in replication, **systems maintain copies of data.**
- This is advantageous as it increases **the availability of data at different sites.** Also, now query requests can be processed in parallel.
- However, it has certain disadvantages as well. Data needs to be constantly updated. Any change made at one site needs to be recorded at every site that relation is stored or else it may **lead to inconsistency**. This is a lot of overhead. Also, concurrency control becomes way more complex as concurrent access now needs to be checked over a number of sites.

Distributed Data Storage

There are 2 ways in which data can be stored on different sites.

2. Fragmentation:

- In this approach, the relations are **fragmented** (i.e., they're **divided into smaller parts**) and each of the fragments is stored in different sites where they're required.
- It must be made sure that the fragments are such that **they can be used to reconstruct the original relation** (i.e, there isn't any loss of data).
- Fragmentation is advantageous **as it doesn't create copies of data**, consistency is not a problem.
- Fragmentation of relations can be done in many ways.

- ✓ **Horizontal fragmentation:**
  The relation is fragmented into **groups of tuples** so that each tuple is assigned to at least one fragment.

- ✓ **Vertical fragmentation:**
  The schema of the relation is divided into smaller schemas. Each fragment **must contain a common candidate key** so as to ensure a lossless join.

- ✓ **Hybrid fragmentation:**
  In some situations, the horizontal and the vertical fragmentation isn't enough to distribute data for some applications and in that conditions, we need a fragmentation called a mixed fragmentation. Mixed fragmentation can be done **in two different ways.**

A Distributed Database System

- It is a type of database management system that stores data across multiple computers or sites that are connected by a network.
- In a distributed database system, each site has its own database, and the databases are connected to each other to form a single, integrated system.
- The main advantage of a distributed database system is that it can provide higher availability and reliability than a centralized database system. Because the data is stored across multiple sites, the system can continue to function even if one or more sites fail.
- In addition, a distributed database system can provide better performance by distributing the data and processing load across multiple sites.

A DDBMS therefore has the following characteristics:

- A collection of logically related shared data.
- The data is split into a number of fragments.
- Fragments may be replicated.
- Fragments/replicas are allocated to sites.
- The sites are linked by a communications network.
- The data at each site is under the control of a DBMS.
- The DBMS at each site can handle local applications, autonomously.
- Each DBMS participates in at least one global application.

Different architectures for distributed database systems

- **Client-server architecture:** In this architecture, **clients connect to a central server**, which manages the distributed database system. The server is responsible for coordinating transactions, managing data storage, and providing access control.
- **Peer-to-peer architecture:** In this architecture, **each site in the distributed database system is connected to all other sites.** Each site is responsible for managing its own data and coordinating transactions with other sites.

Different architectures for distributed database systems

- **Federated architecture:** In this architecture, each site in the distributed database system **maintains its own independent database**, but the databases are integrated through **a middleware layer** that provides a common interface for accessing and querying the data.

Note that: Distributed database systems can be used in a variety of applications, including e-commerce, financial services, and telecommunications. However, **designing and managing a distributed database system can be complex** and requires careful consideration of factors such as data distribution, replication, and consistency.

Advantages of Distributed Database System

1) There is fast data processing as several sites participate in request processing.
2) Reliability and availability of this system is high.
3) It possess reduced operating cost.
4) It is easier to expand the system by adding more sites.
5) It has improved sharing ability and local autonomy.

Disadvantages of Distributed Database System

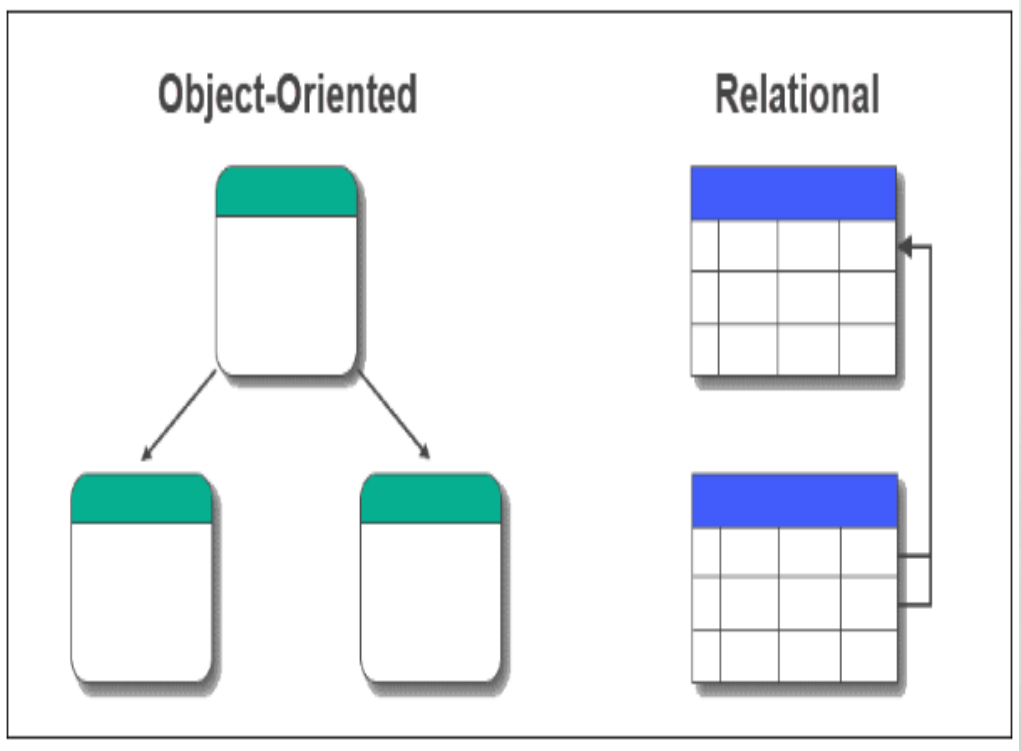
1) The system becomes complex to manage and control.
2) The security issues must be carefully managed.
3) The system require deadlock handling during the transaction processing otherwise the entire system may be in inconsistent state.
4) There is need of some standardization for processing of distributed database system.

Object-Oriented Database

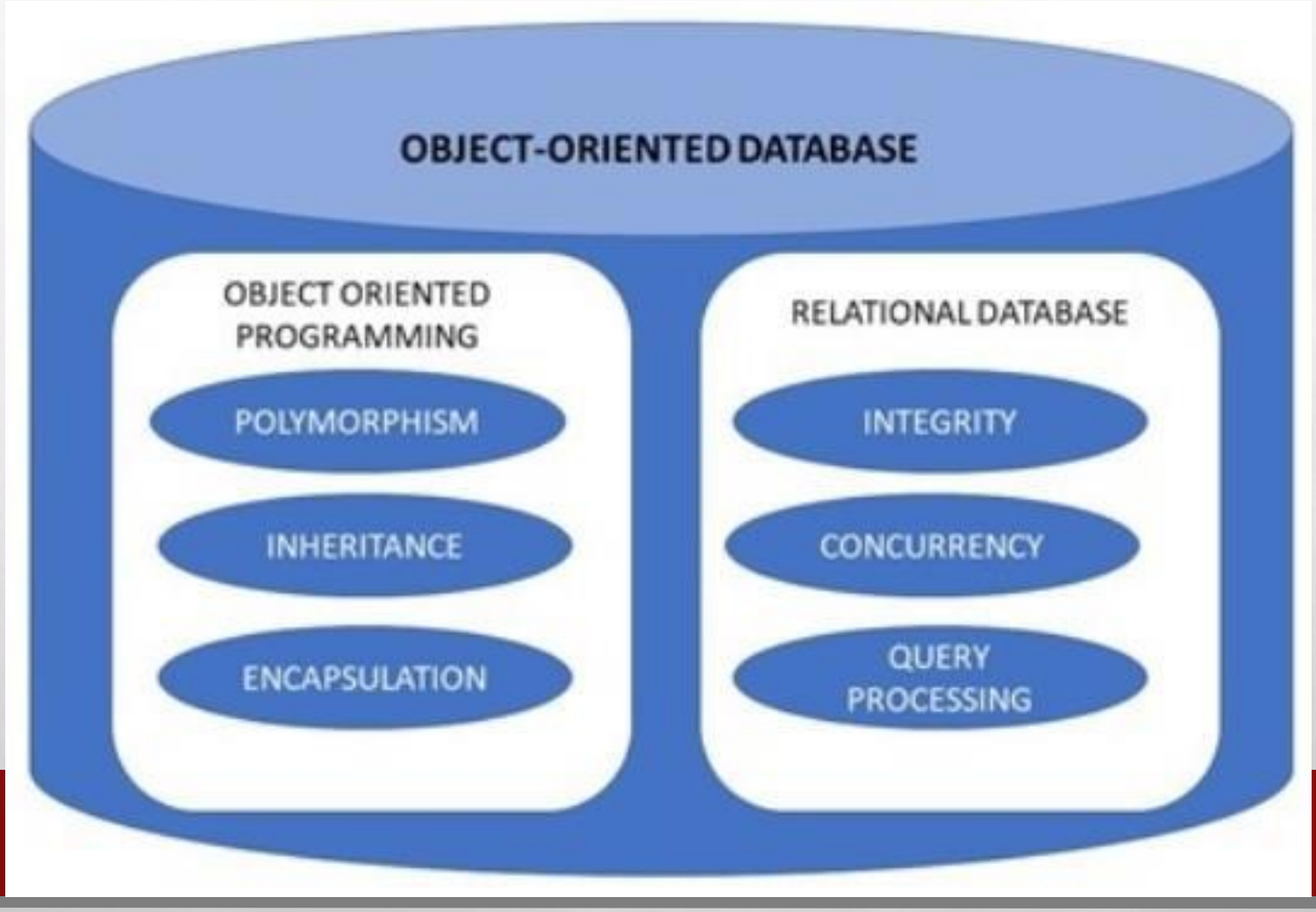
- An object-oriented database (also known as an object database) is a database that **stores information in the form of objects and classes**.
- Similar to object-oriented languages, object-oriented databases follow many object-oriented programming paradigms, such as inheritance, polymorphism, and encapsulation.
- So, object-oriented databases are often built to **interact with certain object-oriented programming languages**, blurring the line between programming language and database solution.

Object-Oriented Database

- Object-oriented databases directly deal with data as complete **objects.** All the information comes in one instantly available object package instead of multiple tables.
- In contrast, the basic building blocks of relational databases are **tables with actions** based on **logical connections** between the table data.



Object-Oriented Database



Elements of Object-Oriented Data Model

- **Objects**

Objects are the basic building block and an instance of a class, where the type is either built-in or user-defined.

- **Attributes and Method**

➢Every object has certain **characteristics.**

➢These are represented using **Attributes.** The **behavior** of the objects is represented using **Methods.**

- **Class**

A class is a collection of similar objects with shared structure i.e. attributes and behavior i.e. methods.

Elements of Object-Oriented Data Model

Polymorphism

- **Polymorphism** is the capability of an object to take multiple forms.
- This ability allows the same program code to work with different data types.
- Both a car and a bike are able to *break*, but the mechanism is different. In this example, the action break is a polymorphism. The defined action is **polymorphic**—the result changes depending on which vehicle performs.

Elements of Object-Oriented Data Model

Inheritance

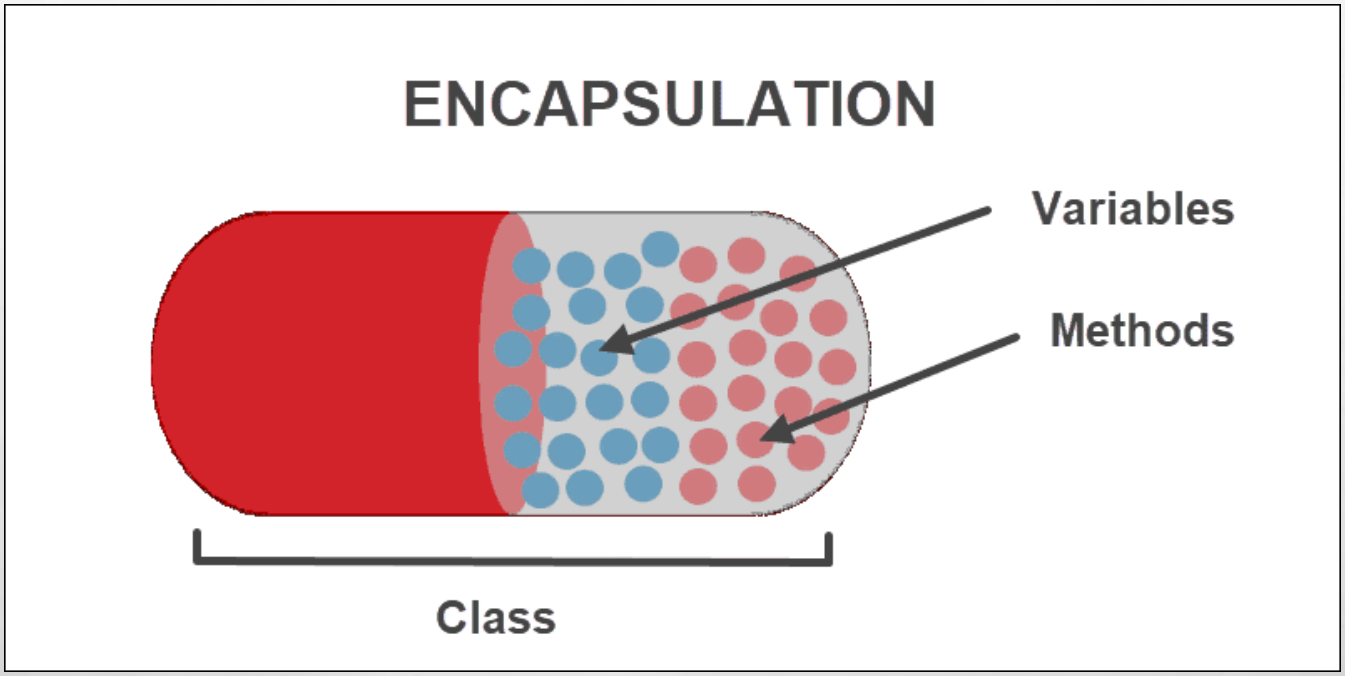
- **Inheritance** creates a hierarchical relationship between related classes while making parts of code reusable.
- Defining new types inherits all the existing class fields and methods plus further extends them.
- The existing class is the **parent** class, while the **child** class extends the parent.

Elements of Object-Oriented Data Model

Encapsulation

- **Encapsulation** is the ability to group data and mechanisms into a single object to provide access protection.
- Through this process, pieces of information and details of how an object works are **hidden**, resulting in data and function security.
- Classes interact with each other through methods without the need to know how particular methods work.

Elements of Object-Oriented Data Model



ANY QUESTIONS ??

I wish you
all the best

THANK YOU & GOODBYE