Objectives

In this Lecture, you will learn:

- What are Distributed databases?
- The difference between centralized and decentralized databases.
- When to use the Distributed databases.
- The categories of Distributed databases.

1. Introduction

In recent times, we have seen rapid developments in **network** and **data communication technology,** epitomized by the Internet, mobile and wireless computing, intelligent devices, and grid computing.

Now, with the combination of these two technologies, distributed database technology may change the mode of working from centralized to decentralized.

A centralized Database is a single logical database located at one site under the control of a single DBMS.

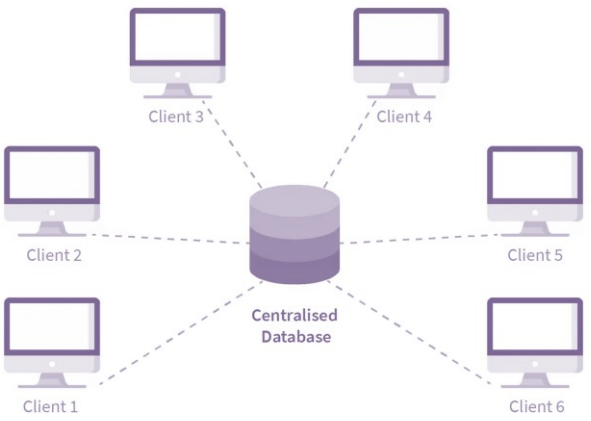


Fig. 1. A centralized DB.

2. Distributed Database & Database management system

- Distributed database A logically interrelated collection of shared data, physically distributed over a computer network.

- A distributed database is a database that is distributed across multiple nodes or servers. Each node or server stores a subset of the data, and the nodes communicate with each other to ensure that the data is consistent across all nodes.

- Distributed DBMS is a software system that permits the management of the distributed database and makes the distribution transparent to users.

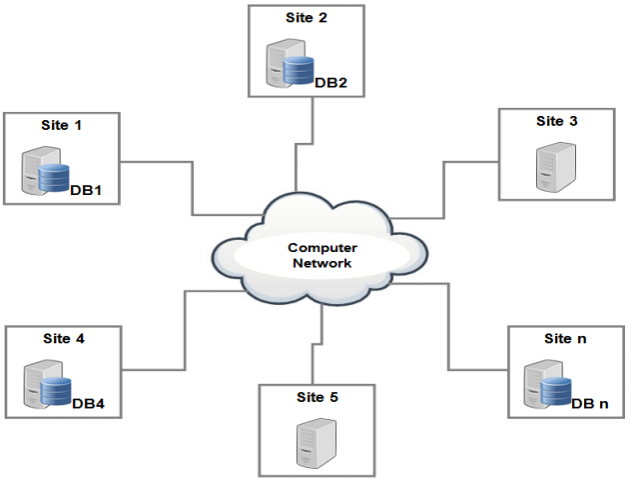


Fig. 2. A decentralized DB.

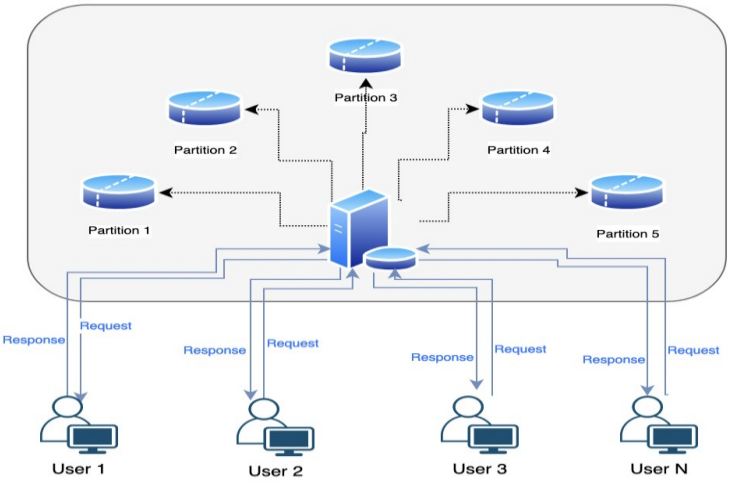


Fig. 3. A distributed DB on several server's partitions.

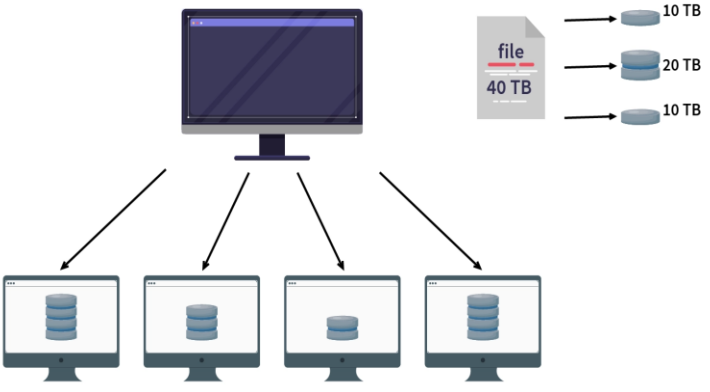


Fig. 4. A distributed database is a database that is not limited to one computer system. It is like a database that consists of two or more files located on different computers or sites either on the same network or on an entirely different network.

3. The purpose and features of DDBMS

Distributed Database Management System (DDBMS) (decentralized) allows users to access not only the data at their site but also data stored at remote sites.

This decentralized approach mirrors the organizational structure of many companies, which are logically distributed into divisions, departments, projects, and so on, and physically distributed into offices, plants, and factories, where each unit maintains its operational data. See Fig. 5.

The shareability of the data and the efficiency of data access should be improved by the development of a distributed database system that reflects this organizational structure, makes the data in all units accessible, and stores data proximate to the location where it is most frequently used.

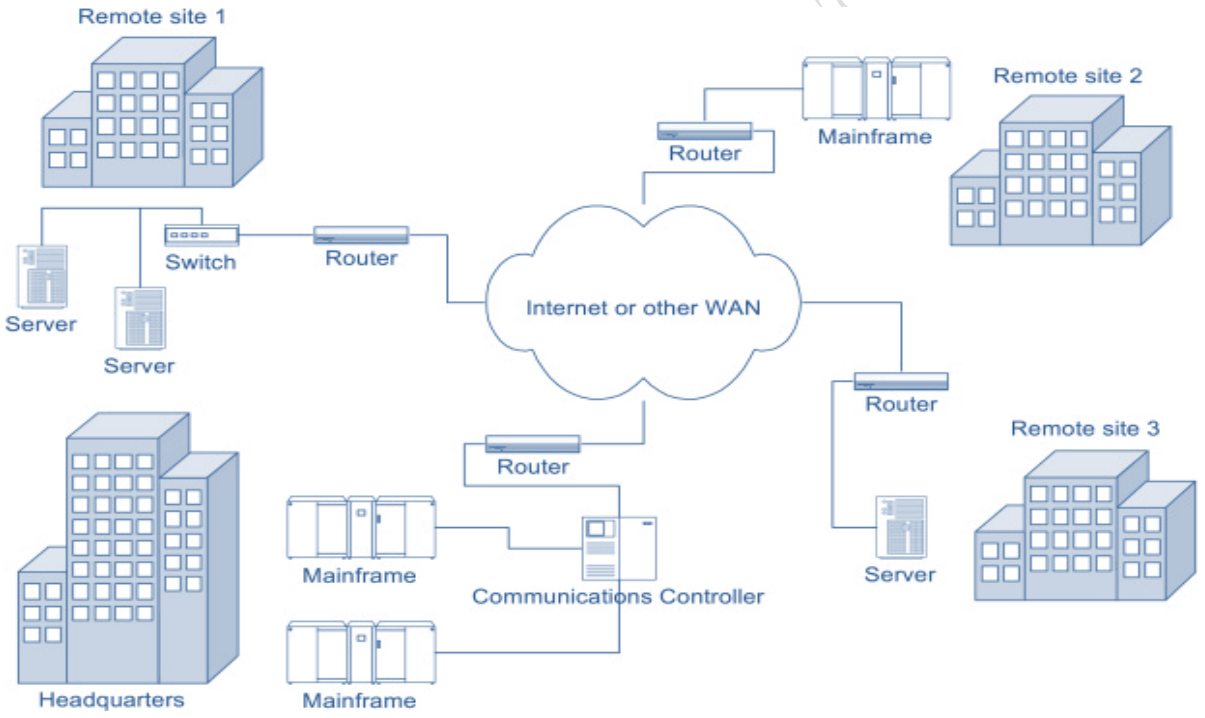
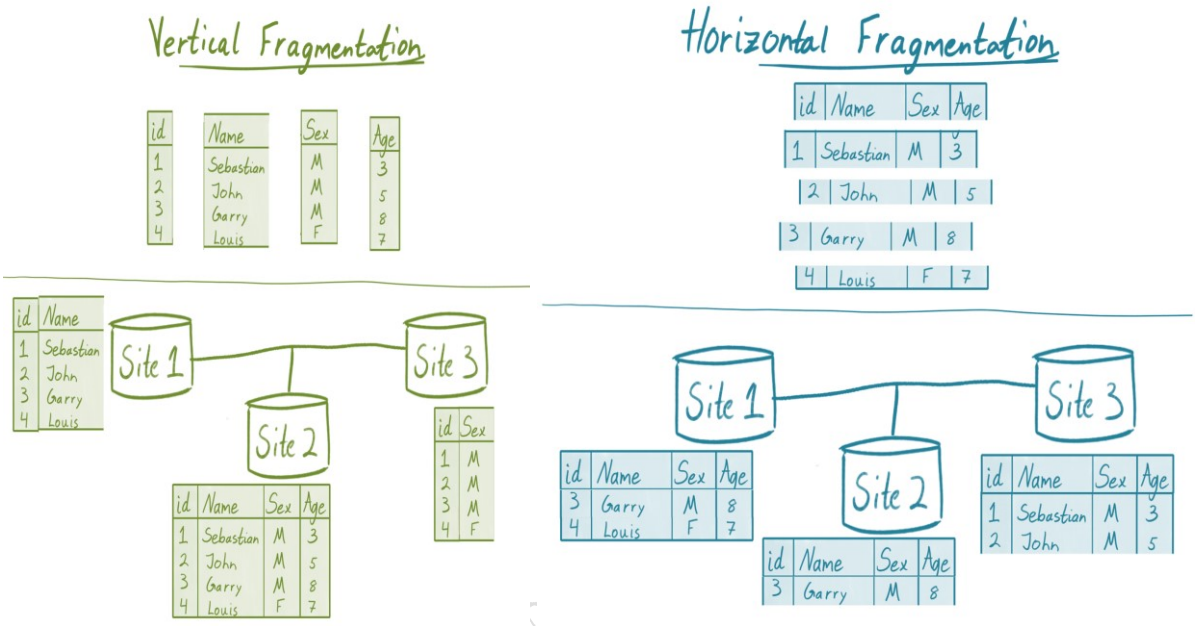


Fig. 5. The distributed DB is guided by organizational structure.

- DDBMS consists of a single logical database that is split into several **fragments,** see Fig.6. Each fragment is stored on one or more computers under the control of a separate DBMS, with:
  - The computers are connected by a communications network.
  - Each site is capable of independently processing user requests that require access to local data (that is, each site has some degree of local autonomy)
  - Also capable of processing data stored on other computers in the network.

- Users access the distributed database via applications, which are classified as :
  - **a.** Those that do not require data from other sites (local applications)
  - **b.** Those that do require data from other sites (global applications).
- We require a DDBMS to have at least one global application.



(A) (B)

Fig. 6. (A): Vertical Fragmention. (B) Horizontal Fragmentation.

4. DDBMS Categories

Different types of distributed database are categorized based on **how data is distributed across multiple nodes**.

4.1 Replicated databases

- data is replicated across multiple nodes.
- So that each node has a copy of the data.
- The data can be replicated across all nodes or a subset of nodes.
- Replication can improve data availability and reduce latency.
- However, it can also increase data inconsistency and storage overhead.
- Commons are Hadoop, Apache Spark, and Cassandra.

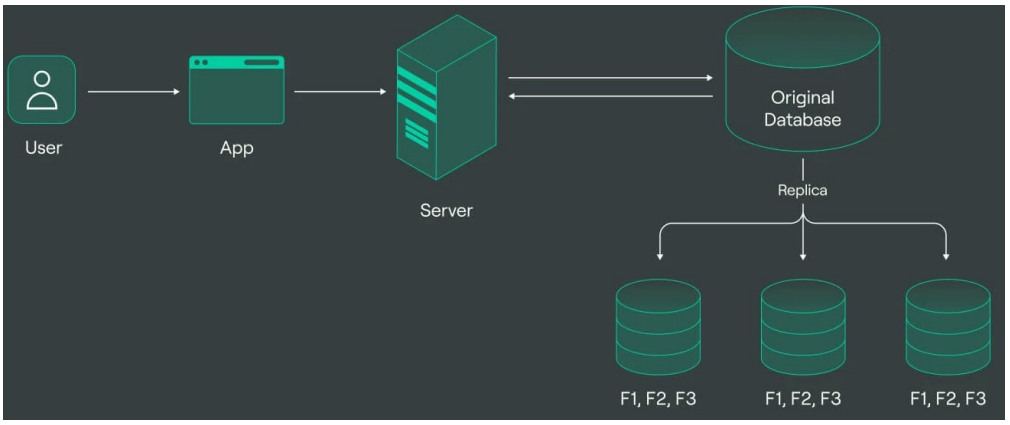


Fig. 7. Replicated DB.

4.2 Partitioned databases

- data is divided into partitions.
- Each partition is stored on a separate node (a server).
- Partitioning can improve query performance and scalability.
- But, it can also increase data inconsistency and complexity.
- Commons are MongoDB, Apache HBase, and Amazon DynamoDB.

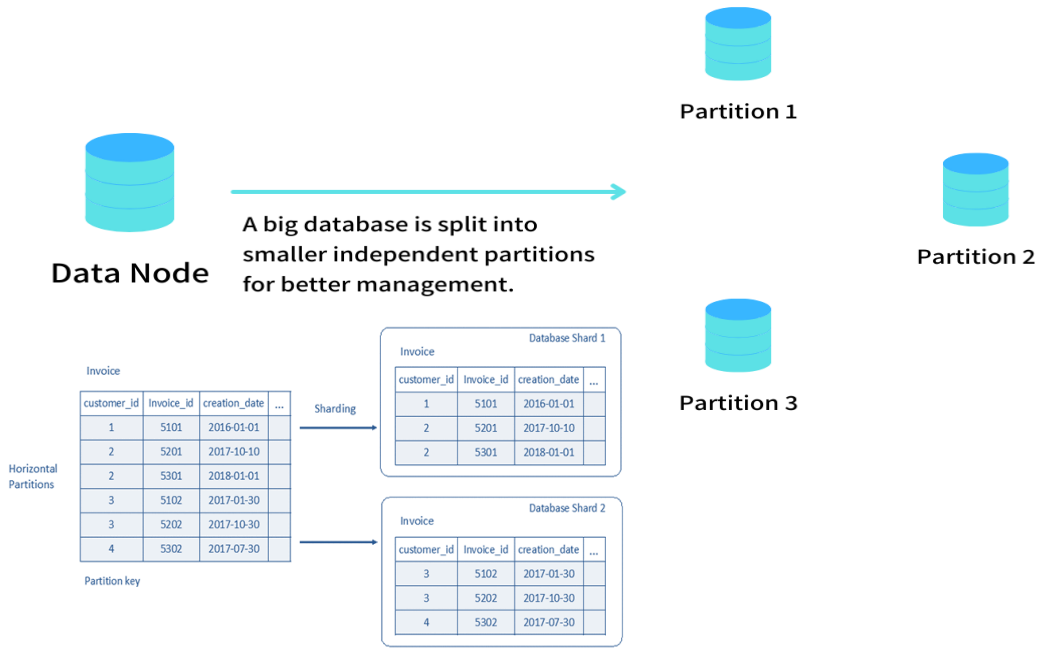


Fig. 8. Partitioned DB.

4.3 Shared-nothing databases:

- Each node has its own processor, memory, and disk storage, and there is no shared memory or disk between nodes.
- Shared-nothing architectures can provide high scalability and availability.
- But they can also increase complexity and cost.
- Commons are Google Spanner, Teradata, and VoltDB.

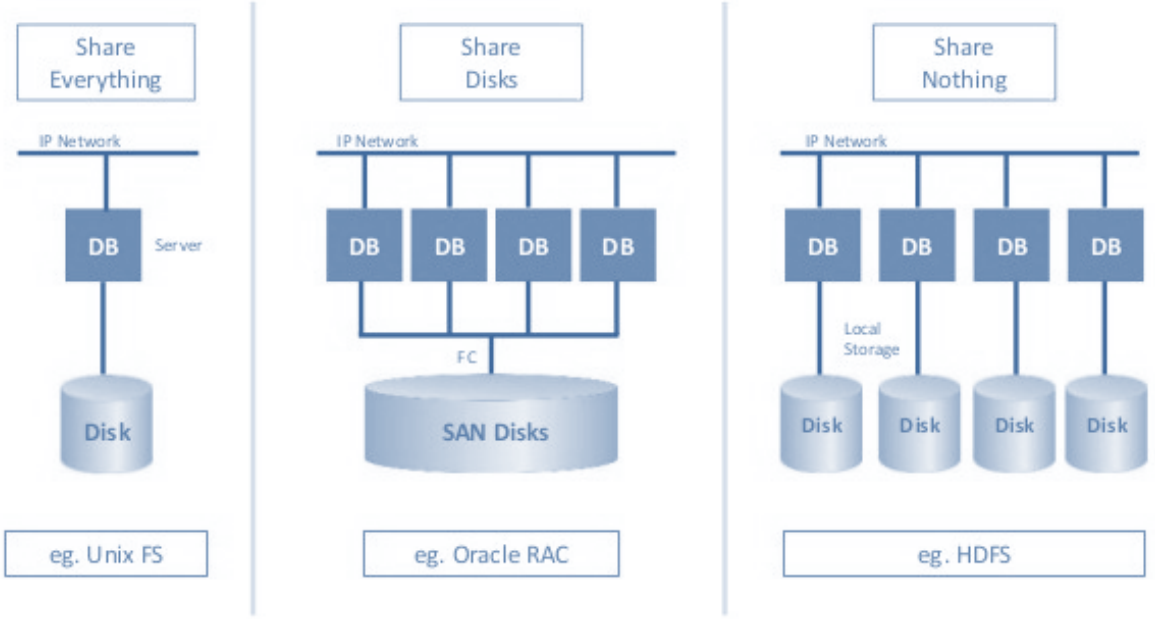


Fig. 9. Share everything, disk, and nothing DBs.

4.4 Hybrid databases

- Combines two or more of the above architectures, such as partitioning and replication.
- Hybrid databases can provide a balance between performance, scalability, and consistency,
- but they can also increase complexity and cost.

Different types of distributed databases can also be categorized based on the internal data structure and the data storage mechanism of different nodes into heterogeneous and homogeneous DB.

A Homogenous distributed database is a network of identical databases stored on multiple sites. All databases store data identically, the operating system, DDBMS, and the data structures used

are all the same at all sites, making them easy to manage. Further, they all have the same DBMS software and data structure

Homogeneous distributed databases are easier to manage and provide better consistency, but they can be less flexible and scalable.

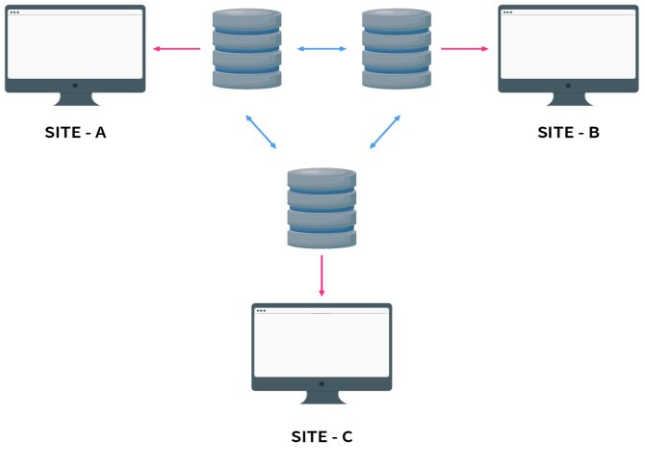


Fig. 10. Homogeneous distributed databases.

A Heterogeneous Distributed Database is the opposite of a homogeneous distributed database. It uses different schemas, operating systems, DDBMS, and different data structures making it difficult to manage. In the case of a heterogeneous distributed database, a particular site can be completely unaware of other sites. This causes limited cooperation in processing user requests, this is why translations are required to establish communication between sites.

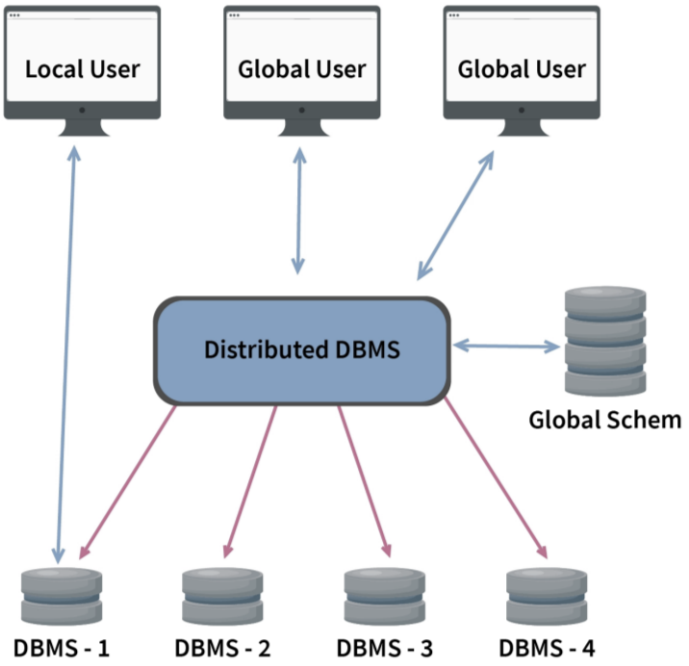


Fig. 11. Heterogeneous distributed databases.