

# Distributed Databases

# Overview

- Distributed vs. decentralized
- Why distributed databases
- Distributed database architecture and environment
- Explain advantages and risks of distributed databases
- Explain strategies and options for distributed database design

# Distributed vs. Decentralized

- **Distributed Database:** A single logical database that is spread physically across computers in multiple locations that are connected by a data communications link
- **Decentralized Database:** A collection of independent databases

# Why Distributed Database

- Business unit autonomy and distribution
- Data sharing
- Data communication costs
- Data communication reliability and costs
- Database recovery

# Distributed Database Options

- Homogeneous - Same DBMS at each node
  - Autonomous - Independent DBMSs
  - Non-autonomous - Central, coordinating DBMS
  - Easy to manage, difficult to enforce
- Heterogeneous - Different DBMSs at different nodes
  - Systems – With full or partial DBMS functionality
  - Difficult to manage, preferred by independent organizations

# Homogeneous, Non-Autonomous Database

- Data is distributed across all the nodes
- Same DBMS at each node
- All data is managed by the distributed DBMS

# Typical Heterogeneous Environment

- Data distributed across all the nodes
- Different DBMSs may be used at each node
- Local access is done using the local DBMS and schema

# Major Objectives

- Location Transparency
  - User does not have to know the location of the data
  - Data requests automatically forwarded to appropriate sites
- Local Autonomy
  - Local site can operate with its database when network connections fail
  - Each site controls its own data, security, logging, recovery



# Significant Trade-Offs

- **Synchronous** Distributed Database
  - All copies of the same data are always identical
  - Data updates are immediately applied to all copies throughout network
  - Good for data integrity
  - High overhead → slow response times
- **Asynchronous** Distributed Database
  - Some data inconsistency is tolerated
  - Data update propagation is delayed
  - Lower data integrity
  - Less overhead → faster response time

# Advantages of Distributed Database over Centralized Databases

- Increased reliability/availability
- Local control over data
- Modular growth
- Lower communication costs
- Faster response for certain queries

# Disadvantages of Distributed Database Compared to Centralized Databases

- Software cost and complexity
- Processing overhead
- Slower response for certain queries

# Options for Distributing a Database

- Data replication
  - Copies of data distributed to different sites
- Horizontal partitioning
  - Different rows of a table distributed to different sites
- Vertical partitioning
  - Different columns of a table distributed to different sites
- Combinations of the above

# Distributed DBMS

- ***Distributed database*** requires ***distributed DBMS***
- Functions of a distributed DBMS:
  - Locate data with a *distributed data dictionary*
  - Determine location from which to retrieve data and process query components
  - DBMS translation between nodes with different local DBMSs
  - Data consistency
  - Scalability
  - Security, concurrency, query optimization, failure recovery