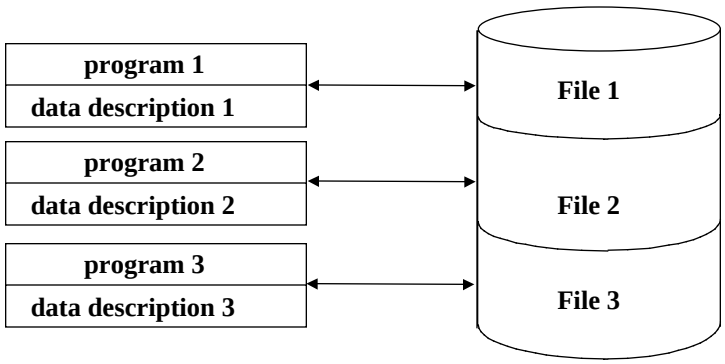
# Outline

- Introduction
  - What is a distributed DBMS
  - Problems
  - Current state-of-affairs
- Background
- Distributed DBMS Architecture
- Distributed Database Design
- Semantic Data Control
- Distributed Query Processing
- Distributed Transaction Management
- Parallel Database Systems
- Distributed Object DBMS
- Database Interoperability
- Current Issues

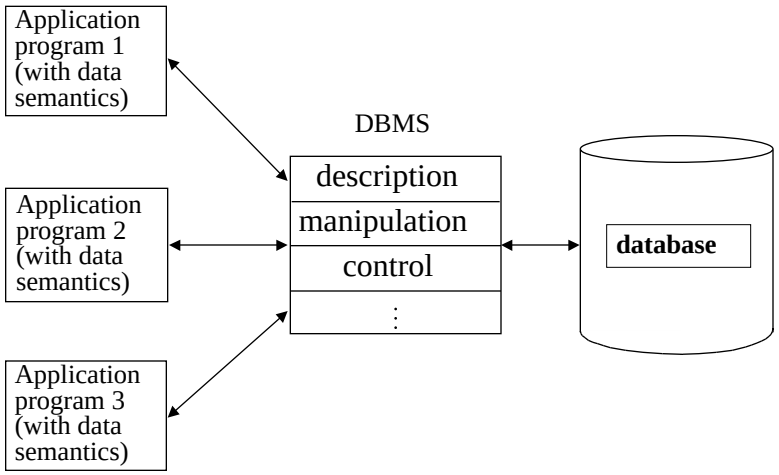




# File Systems

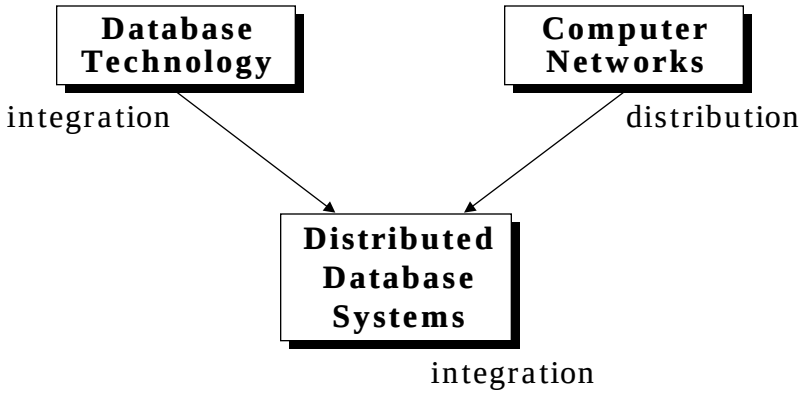


# Database Management






# Motivation




**integration ≠ centralization**

## Distributed Computing

- A concept in search of a definition and a name.
- A number of autonomous processing elements (not necessarily homogeneous) that are interconnected by a computer network and that cooperate in performing their assigned tasks.





## Distributed Computing

- Synonymous terms
  - distributed function
  - distributed data processing
  - multiprocessors/multicomputers
  - satellite processing
  - backend processing
  - dedicated/special purpose computers
  - timeshared systems
  - functionally modular systems

## What is distributed ...

- Processing logic
- Functions
- Data
- Control



## What is a Distributed Database System?

A distributed database (DDB) is a collection of multiple, *logically interrelated* databases distributed over a *computer network.*

A distributed database management system (D–DBMS) is the software that manages the DDB and provides an access mechanism that makes this distribution transparent to the users.

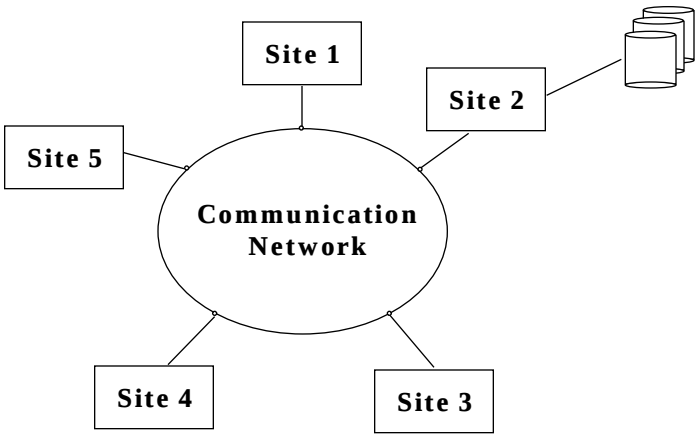
Distributed database system (DDBS) = DDB + D–DBMS

## What is not a DDBS?

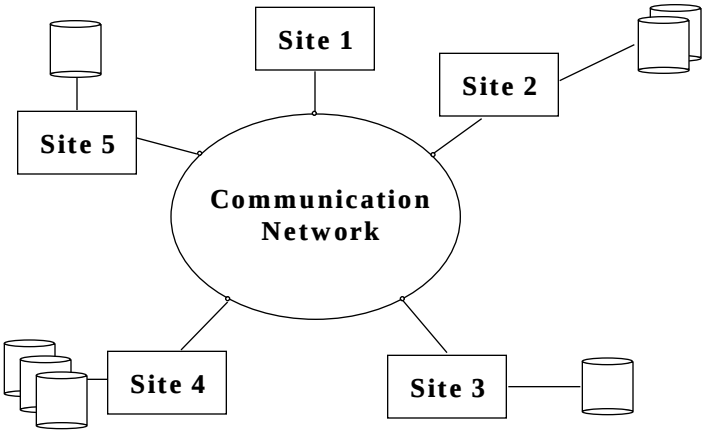
- A timesharing computer system
- A loosely or tightly coupled multiprocessor system
- A database system which resides at one of the nodes of a network of computers - this is a centralized database on a network node



## Centralized DBMS on a Network



# Distributed DBMS Environment



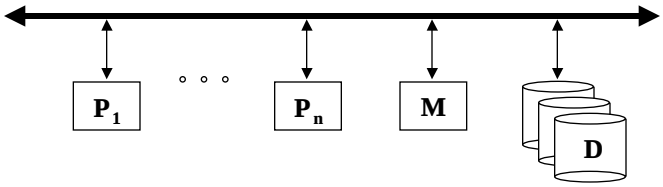





# Implicit Assumptions

- Data stored at a number of sites ⇨ each site *logically* consists of a single processor.
- Processors at different sites are interconnected by a computer network ⇨ no multiprocessors
  - parallel database systems
- Distributed database is a database, not a collection of files ⇨ data logically related as exhibited in the users' access patterns
  - relational data model
- D-DBMS is a full-fledged DBMS
  - not remote file system, not a TP system

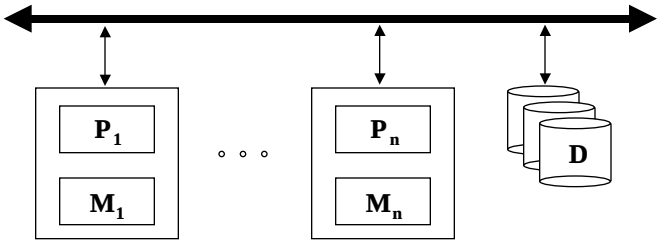
# Shared-Memory Architecture




Examples : symmetric multiprocessors (Sequent, Encore) and some mainframes (IBM3090, Bull's DPS8)

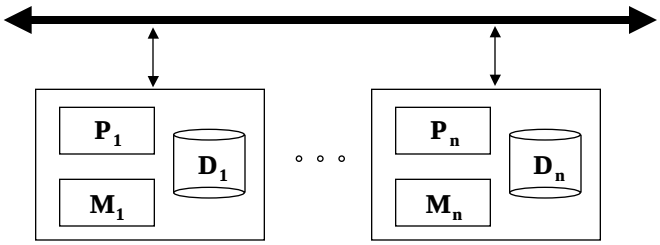


# Shared-Disk Architecture




Examples : DEC's VAXcluster, IBM's IMS/VS Data Sharing

# Shared-Nothing Architecture




Examples : Teradata's DBC, Tandem, Intel's Paragon, NCR's 3600 and 3700



# Applications

- Manufacturing - especially multi-plant manufacturing
- Military command and control
- EFT
- Corporate MIS
- Airlines
- Hotel chains
- Any organization which has a decentralized organization structure

## Distributed DBMS Promises

❶ Transparent management of distributed, fragmented, and replicated data

❷ Improved reliability/availability through distributed transactions

❸ Improved performance

❹ Easier and more economical system expansion



## Transparency

- Transparency is the separation of the higher level semantics of a system from the lower level implementation issues.
- Fundamental issue is to provide

  **data independence**

  in the distributed environment
  - Network (distribution) transparency
  - Replication transparency
  - Fragmentation transparency
    - horizontal fragmentation: selection
    - vertical fragmentation: projection
    - hybrid

# Example

EMP

| ENO | ENAME | TITLE |
|---|---|---|
| E1 | J. Doe | Elect. Eng. |
| E2 | M. Smith | Syst. Anal. |
| E3 | A. Lee | Mech. Eng. |
| E4 | J. Miller | Programmer |
| E5 | B. Casey | Syst. Anal. |
| E6 | L. Chu | Elect. Eng. |
| E7 | R. Davis | Mech. Eng. |
| E8 | J. Jones | Syst. Anal. |

ASG

| ENO | PNO | RESP | DUR |
|---|---|---|---|
| E1 | P1 | Manager | 12 |
| E2 | P1 | Analyst | 24 |
| E2 | P2 | Analyst | 6 |
| E3 | P3 | Consultant | 10 |
| E3 | P4 | Engineer | 48 |
| E4 | P2 | Programmer | 18 |
| E5 | P2 | Manager | 24 |
| E6 | P4 | Manager | 48 |
| E7 | P3 | Engineer | 36 |
| E7 | P5 | Engineer | 23 |
| E8 | P3 | Manager | 40 |

PROJ

| PNO | PNAME | BUDGET | LOC |
|---|---|---|---|
| P1 | Instrumentation | 150000 | Montreal |
| P2 | Database Develop. | 135000 | New York |
| P3 | CAD/CAM | 250000 | New York |
| P4 | Maintenance | 310000 | Paris |
| P5 | CAD/CAM | 500000 | Boston |

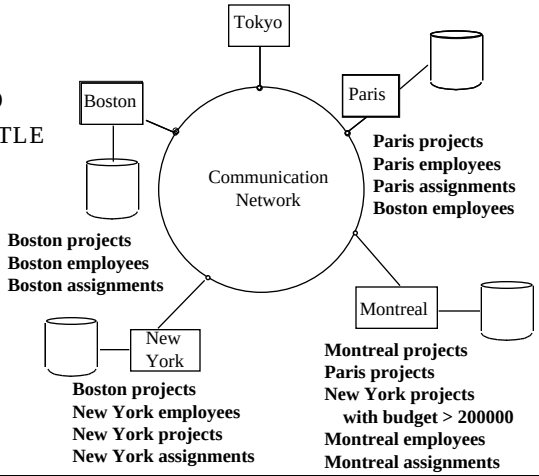
PAY

| TITLE | SAL |
|---|---|
| Elect. Eng. | 40000 |
| Syst. Anal. | 34000 |
| Mech. Eng. | 27000 |
| Programmer | 24000 |

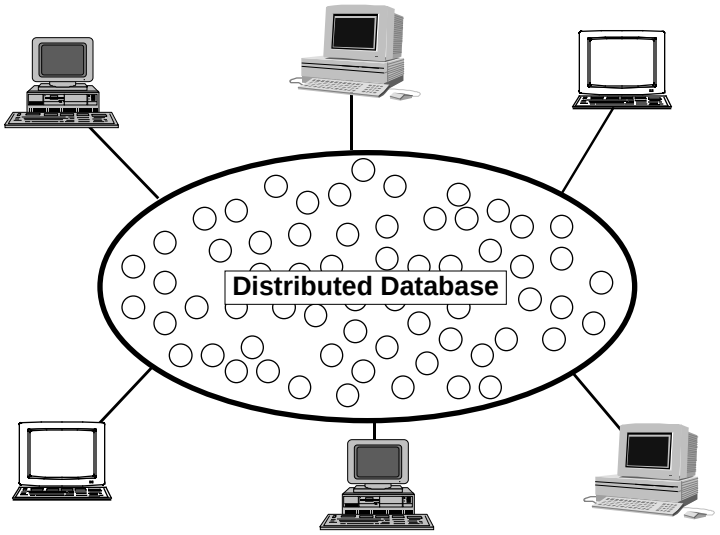


# Transparent Access

**SELECT** ENAME,SAL
**FROM** EMP,ASS,PAY
**WHERE** DUR > 12
**AND** EMP.ENO = ASS.ENO
**AND** EMP.TITLE = PAY.TITLE

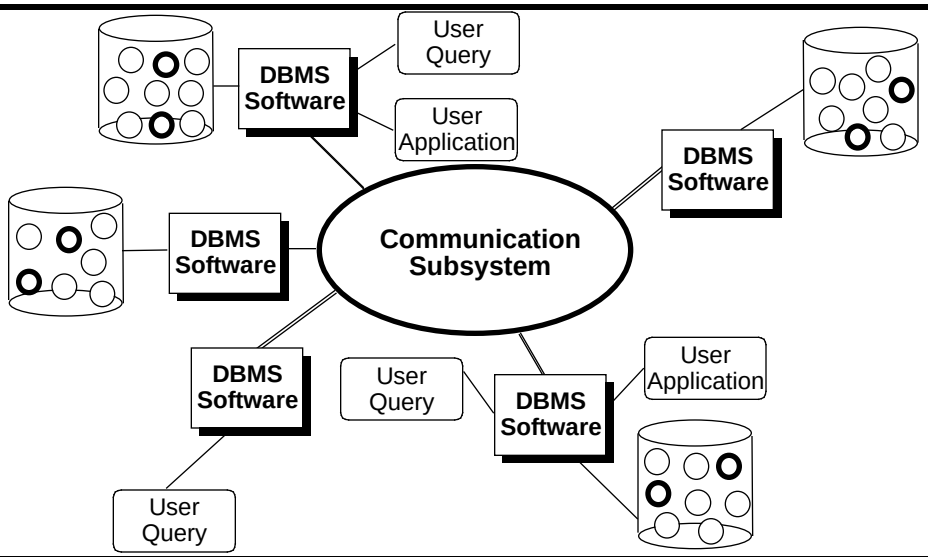


# Distributed Database - User View






# Distributed DBMS - Reality



## Potentially Improved Performance

- Proximity of data to its points of use
  - Requires some support for fragmentation and replication
- Parallelism in execution
  - Inter-query parallelism
  - Intra-query parallelism



## Parallelism Requirements

- Have as much of the data required by *each* application at the site where the application executes
  - Full replication
- How about updates?
  - Updates to replicated data requires implementation of distributed concurrency control and commit protocols

## System Expansion

- Issue is database scaling
- Emergence of microprocessor and workstation technologies
  - Demise of Grosh's law
  - Client-server model of computing
- Data communication cost vs telecommunication cost



## Distributed DBMS Issues

- **Distributed Database Design**
  - how to distribute the database
  - replicated & non-replicated database distribution
  - a related problem in directory management
- **Query Processing**
  - convert user transactions to data manipulation instructions
  - optimization problem
  - min{cost = data transmission + local processing}
  - general formulation is NP-hard

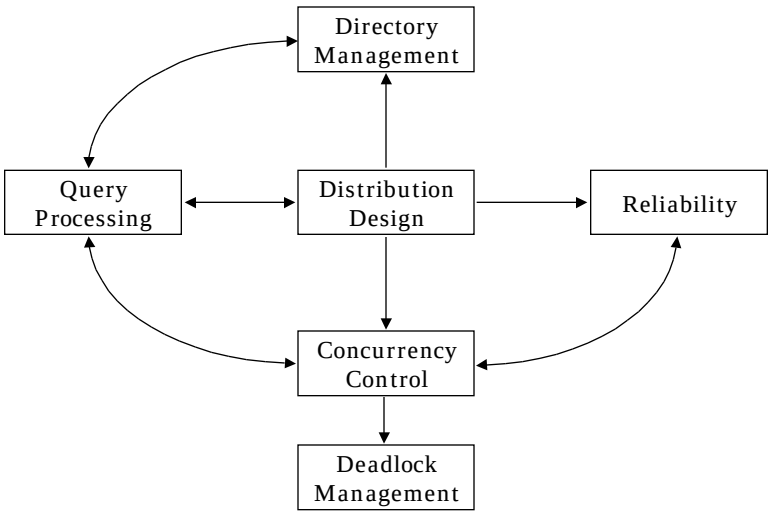
# Distributed DBMS Issues

- **Concurrency Control**
  - synchronization of concurrent accesses
  - consistency and isolation of transactions' effects
  - deadlock management
- **Reliability**
  - how to make the system resilient to failures
  - atomicity and durability





# Relationship Between Issues



# Related Issues

- **Operating System Support**
  - operating system with proper support for database operations
  - dichotomy between general purpose processing requirements and database processing requirements
- **Open Systems and Interoperability**
  - Distributed Multidatabase Systems
  - More probable scenario
  - Parallel issues