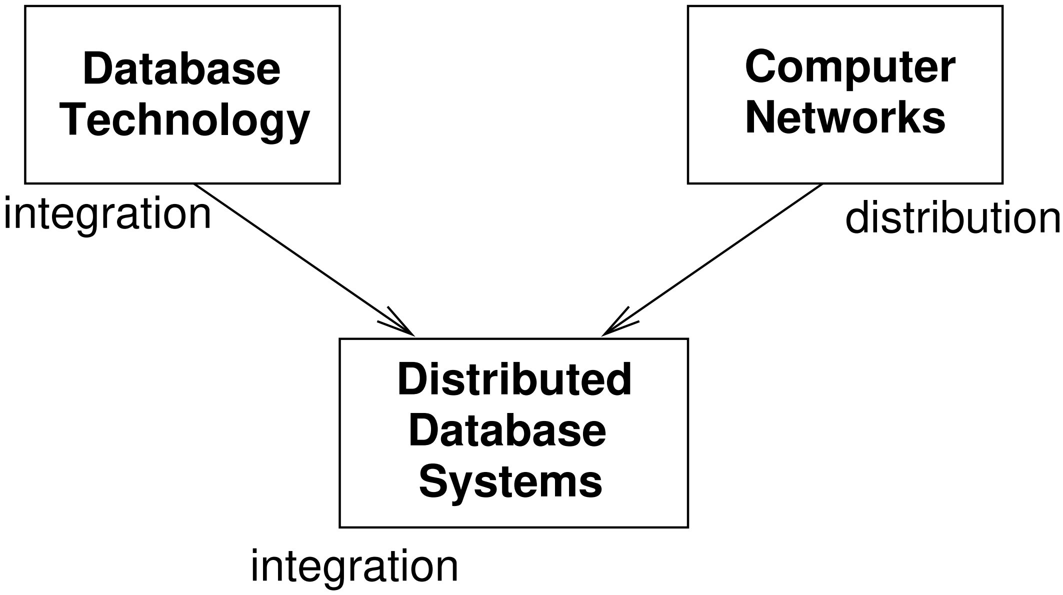
INTRODUCTION

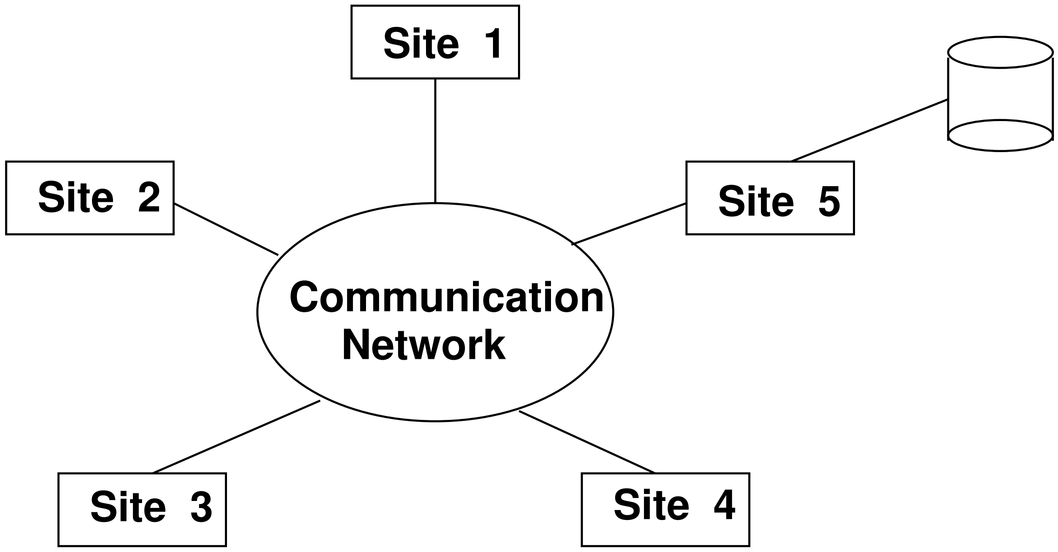
- distributed computing
  - a number of autonomous processing elements (not necessarily homogeneous)
  - interconnected by a computer network
  - and cooperate in performing their assigned tasks
- distribution of
  - processing logic
  - functions
  - data
  - control
- distributed database (DDB)
  - a collection of multiple *logically interrelated* databases
  - distributed over a *computer network*
- distributed database management system (DDBMS)
  - the software that manages the DDB
  - and provides an access mechanism that makes this distribution transparent to the users
- distributed database system (DDBS) = DDB + DDBMS

MOTIVATION

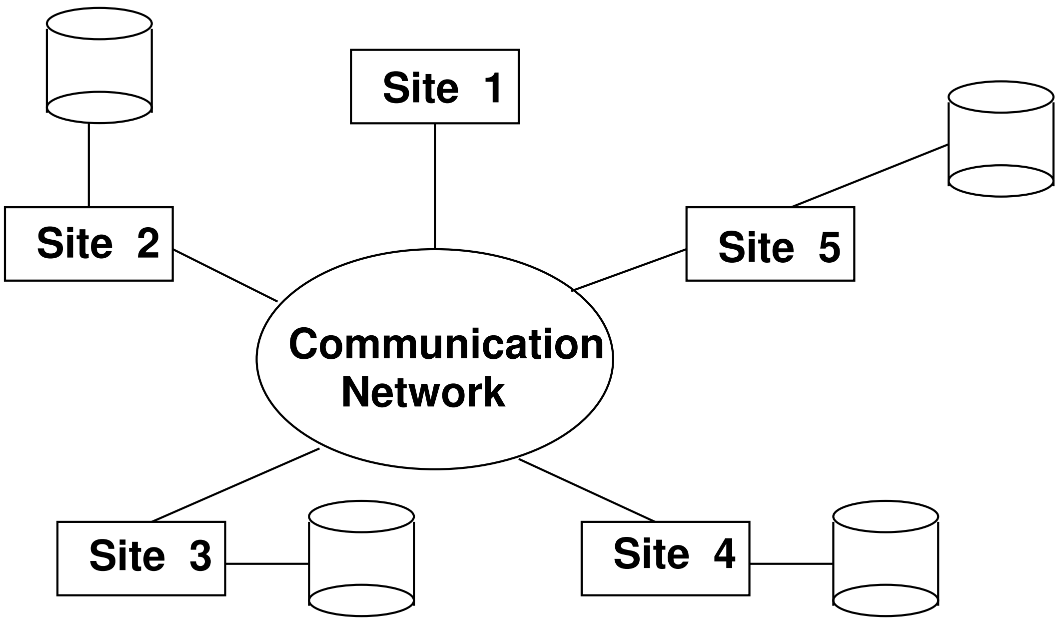


integration ≠ centralization

CENTRALIZED DBMS ON A NETWORK



DISTRIBUTED DBMS ENVIRONMENT

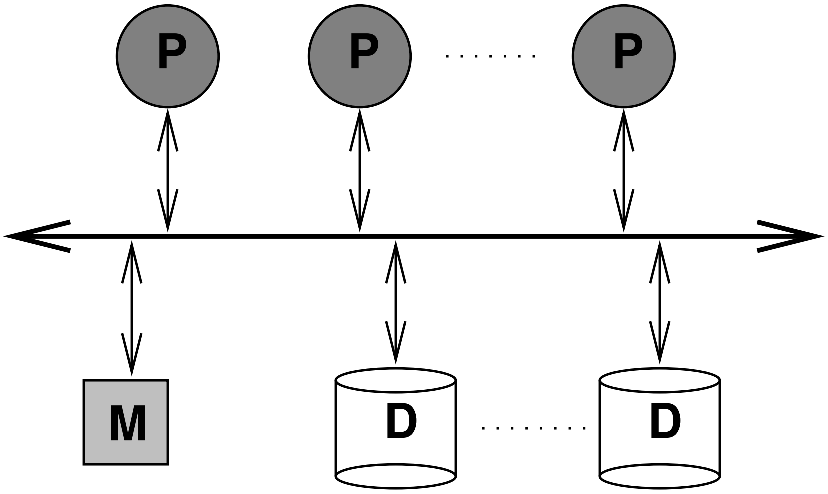


IMPLICIT ASSUMPTIONS

- data stored at a number of sites
- sites are interconnected by a computer network
- distributed database is a database, not a collection of files
  - data logically related as exhibited in users' access patterns
  - relational data model
- DDBMS is a full-fledged DBMS
  - not remote file system

LOOSELY OR TIGHTLY COUPLED MULTIPROCESSOR SYSTEM

Shared-memory Architecture

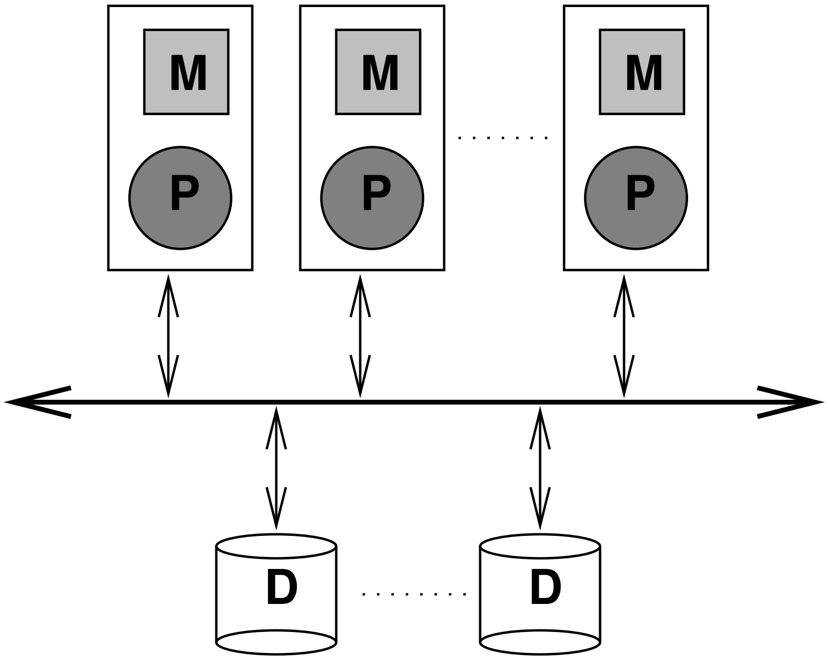


Examples: symmetric multiprocessors (Sequent, Encore) and some mainframes (IBM3090, Bull's DPS8)

We have DEC Alpha 8400 5/350, SUN Enterprise 450, SUN SparcServer 1000

LOOSELY OR TIGHTLY COUPLED MULTIPROCESSOR SYSTEM (CONT)

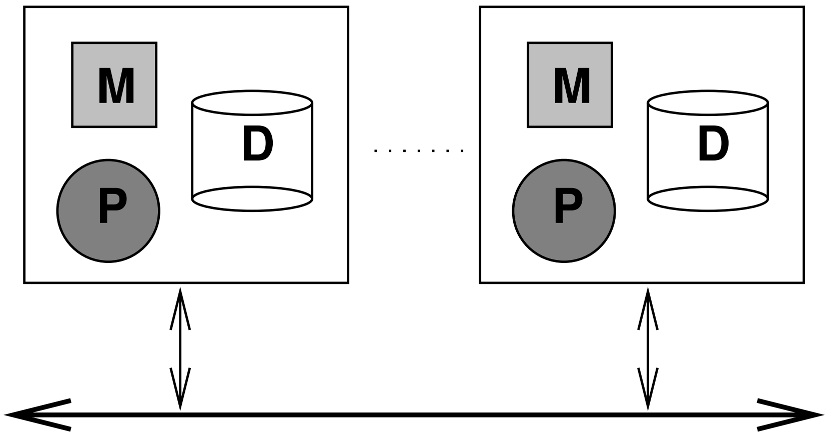
Shared-disk Architecture



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

LOOSELY OR TIGHTLY COUPLED MULTIPROCESSOR SYSTEM (CONT)

Shared-nothing Architecture



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

Based on switch technology.

We have a Fujitsu AP3000. There are several IBM SP2's in Singapore.

APPLICATIONS

any organization which has a decentralized organization structure

- airlines
- banks
- hotel chains
- corporate MIS
- etc.

ADVANTAGES AND DISADVANTAGES OF DDBS

ADVANTAGES

- improved performance
- improved reliability/availability
- economics
- expandability
- shareability
- local autonomy

DISADVANTAGES

- lack of experience
- complexity
- extra network equipment cost
- distribution of control
- difficulty of change (to DDB)

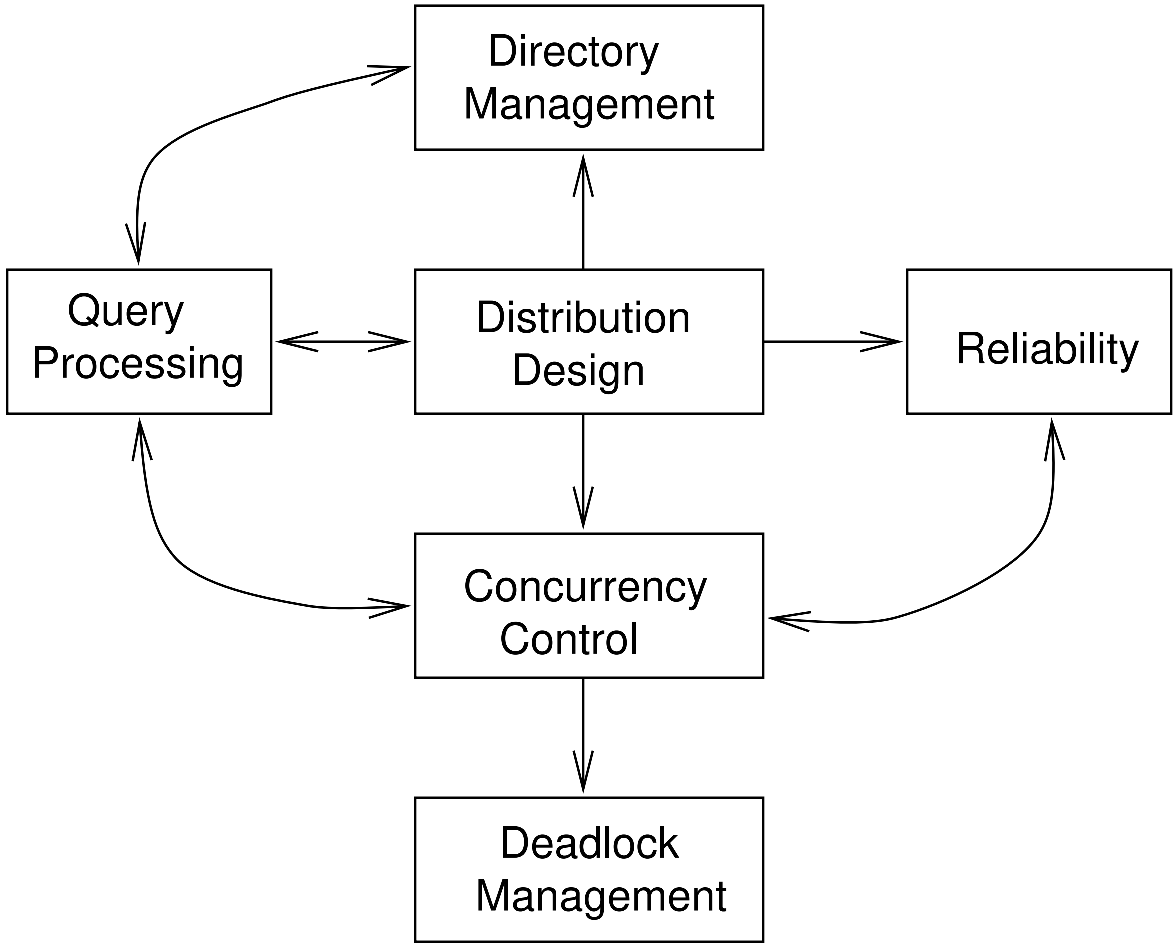
DISTRIBUTED DBMS ISSUES

- **distributed database design**
  - how to distribute the database
  - replicated vs non-replicated database distribution
  - a related problem in directory management
  - general formulation is NP-hard
- **distributed query processing**
  - convert user transactions to data manipulation instructions
  - optimization problem
    - minimize cost = data transmission + local processing
    - general formulation is NP-hard

DISTRIBUTED DBMS ISSUES (CONT)

- **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



TRANSPARENCY SUPPORT

- no replication yet
- generally no fragmentation
  - horizontal fragmentation may come
- limited distribution transparency
  - path specification
  - aliasing
  - remote login to a DBMS
- most are multiple-client/single-server

DISTRIBUTED TRANSACTIONS IN COMMERCIAL SYSTEMS

- some do not provide it at all
  - Oracle 6.x – have to open one database at a time
  - Sybase – primitives are there, applications have to implement distributed transactions themselves
- others do
  - Ingres, NonStop SQL, Oracle V.7
- replication support
  - either nonexistent, or
  - ROWA rule: Read One Write All

POTENTIALLY IMPROVED PERFORMANCE

- proximity of data to its points of use
  - requires some support for fragmentation and replication
- parallelism in execution
  - mode
    - inter-query parallelism
    - intra-query parallelism
  - parallelism requirements
    - put the data where the applications execute
      - full replication
      - updates to replicated data requires implementation of distributed concurrency control and commit protocols
  - commerical system alternatives
    - multiplex time between read and updates
      - open database for reads during regular hours; perform updates after hours, e.g. banks
    - multiplex the database
      - query database
      - production database

SYSTEM EXPANSION

- issue is database scaling
- emergence of microprocessor and workstation technologies
  - demise of Grosh's Law
    - In the early 1970s, it was commonly believed that it would be possible to purchase a fourfold powerful compute if one spent twice as much.
  - client-server model of computing
- data communication cost vs telecommunication cost