Principles of Distributed Database Systems

M. Tamer Özsu

Patrick Valduriez

Outline

- Introduction
- Distributed and Parallel Database Design
- Distributed Data Control
- Distributed Query Processing
- Distributed Transaction Processing
- Data Replication
- Database Integration – Multidatabase Systems
- Parallel Database Systems
- Peer-to-Peer Data Management
- Big Data Processing
- NoSQL, NewSQL and Polystores
- Web Data Management

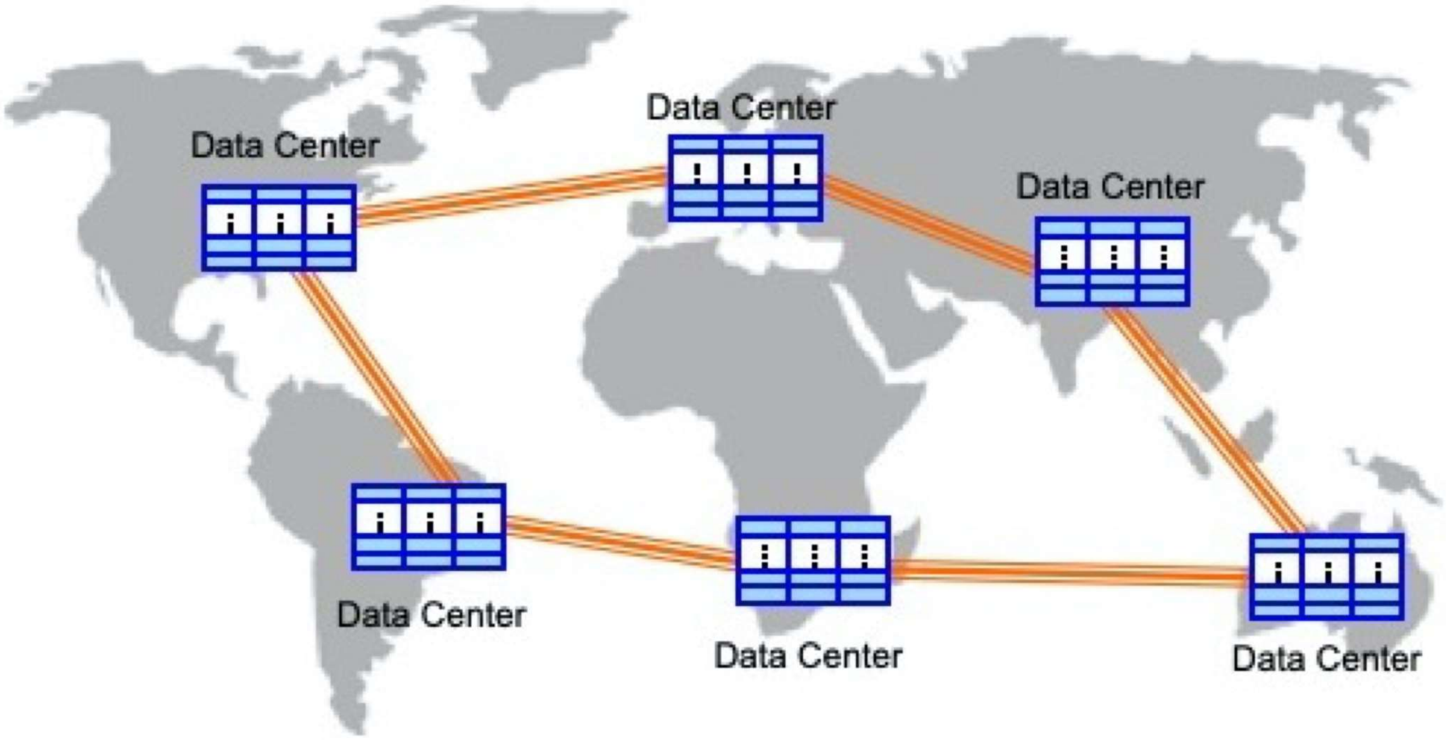
Outline

- Introduction
  - What is a distributed DBMS
  - History
  - Distributed DBMS promises
  - Design issues
  - Distributed DBMS architecture

Distributed Computing

- A number of autonomous processing elements (not necessarily homogeneous) that are interconnected by a computer network and that cooperate in performing their assigned tasks.
- What is being distributed?
  - Processing logic
  - Function
  - Data
  - Control

Current Distribution – Geographically Distributed Data Centers



What is a Distributed Database System?

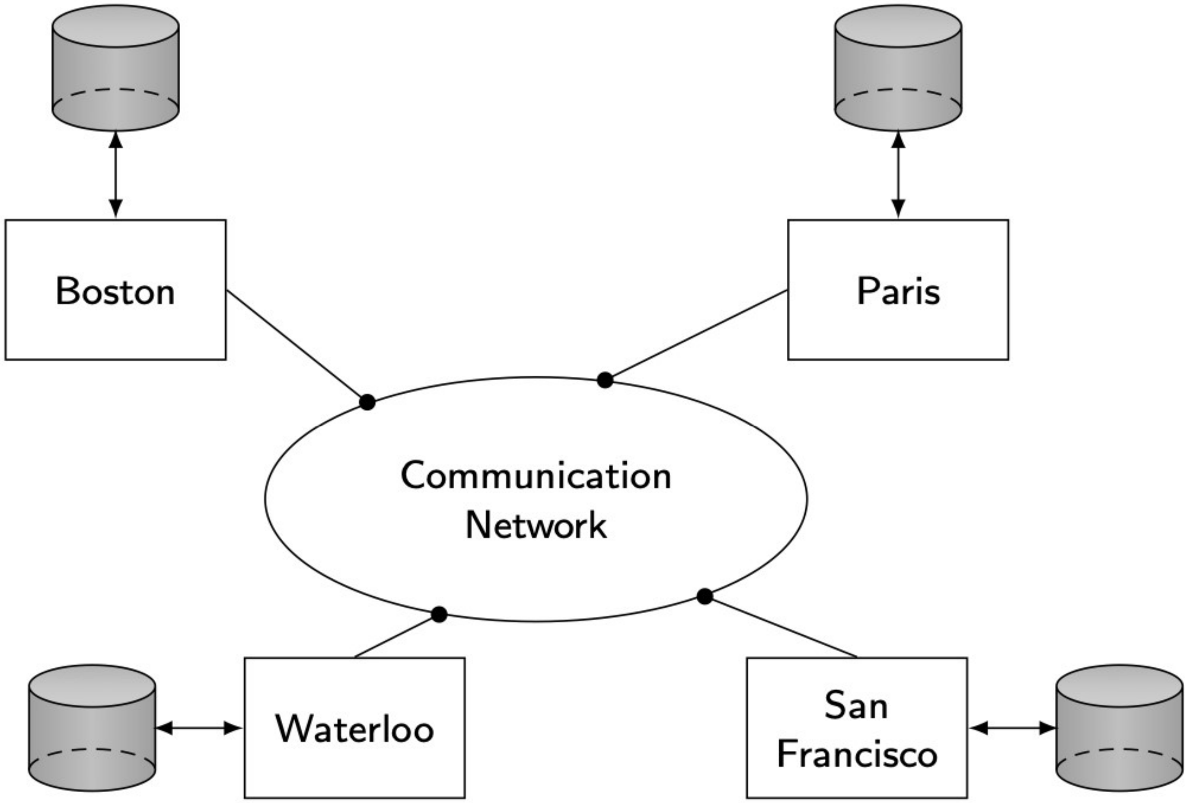
A distributed database is a collection of multiple, logically interrelated databases distributed over a computer network

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

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

Distributed DBMS Environment



Boston employees, Paris employees, Boston projects

Paris employees, Boston employees, Paris projects, Boston projects

Boston

Paris

Communication Network

Waterloo

San Francisco

Waterloo employees, Waterloo projects, Paris projects

San Francisco employees, San Francisco projects

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 → not a multiprocessor system
  - 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
- Distributed DBMS is a full-fledged DBMS
  - Not remote file system, not a TP system

Important Point

Logically integrated

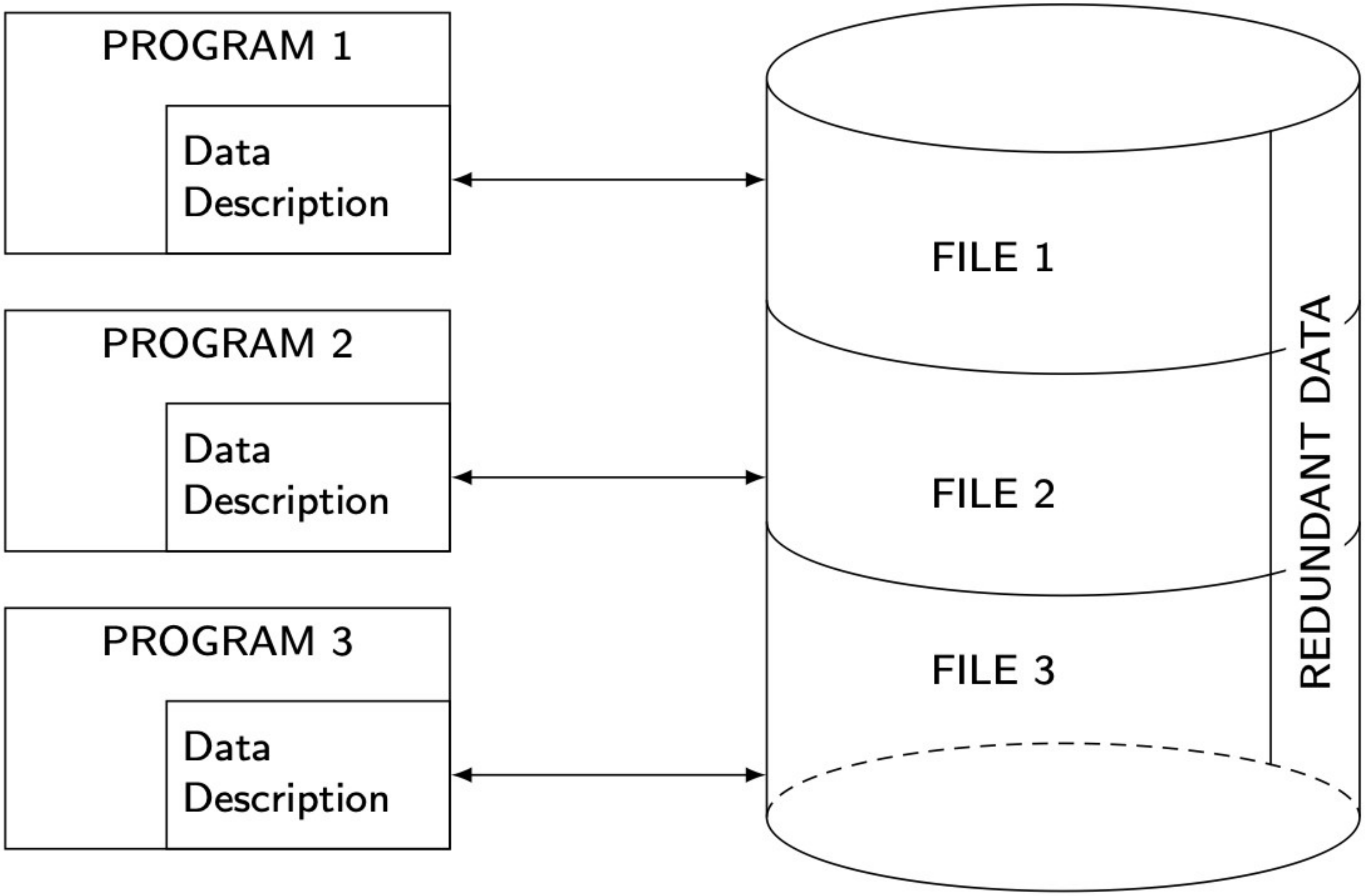
but

Physically distributed

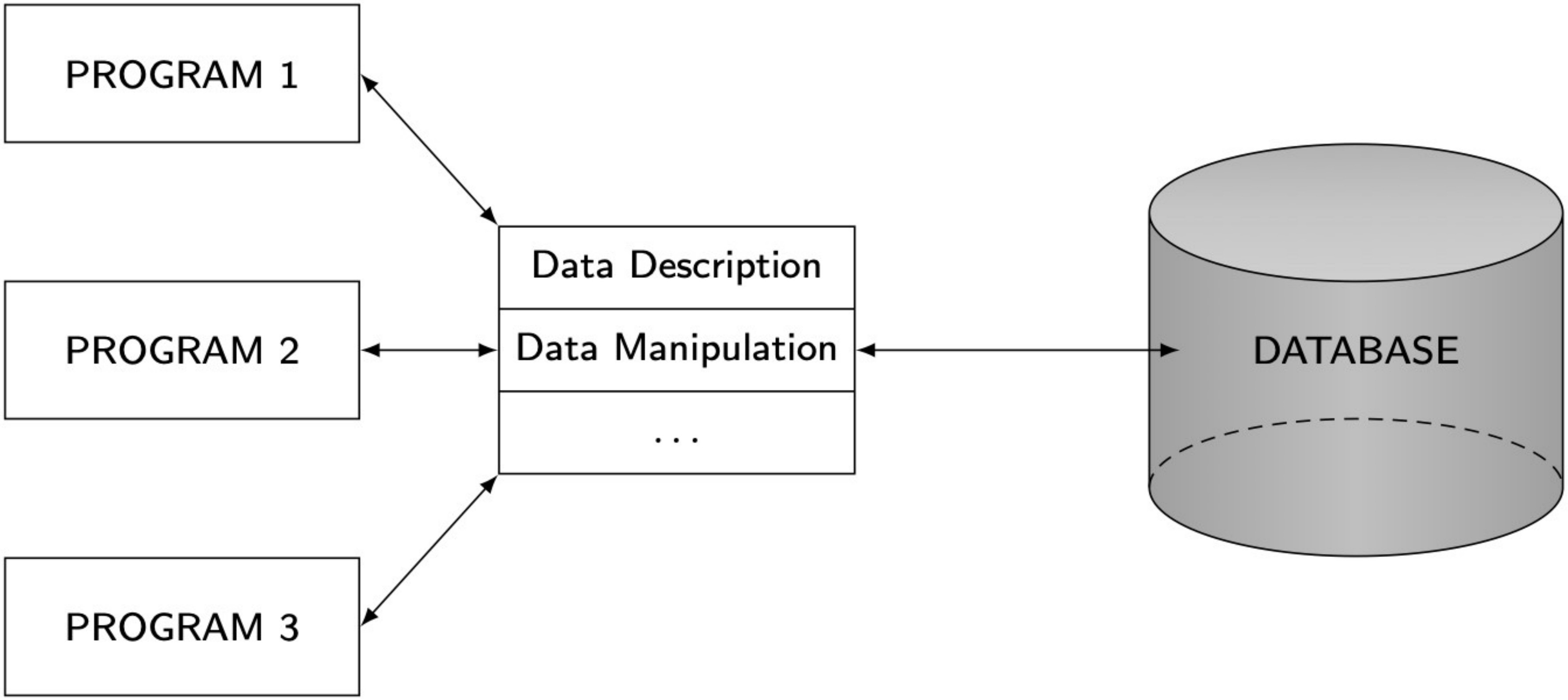
Outline

- Introduction
  - What is a distributed DBMS
  - History
  - Distributed DBMS promises
  - Design issues
  - Distributed DBMS architecture

History – File Systems



History – Database Management

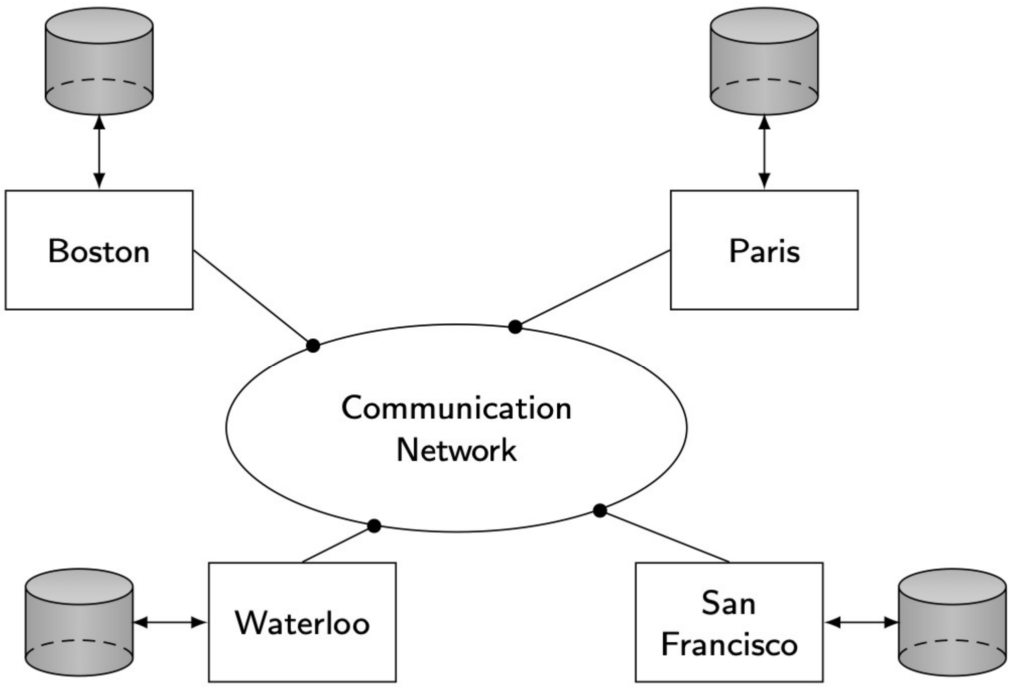


History – Early Distribution

Peer-to-Peer (P2P)

Boston employees, Paris employees, Boston projects

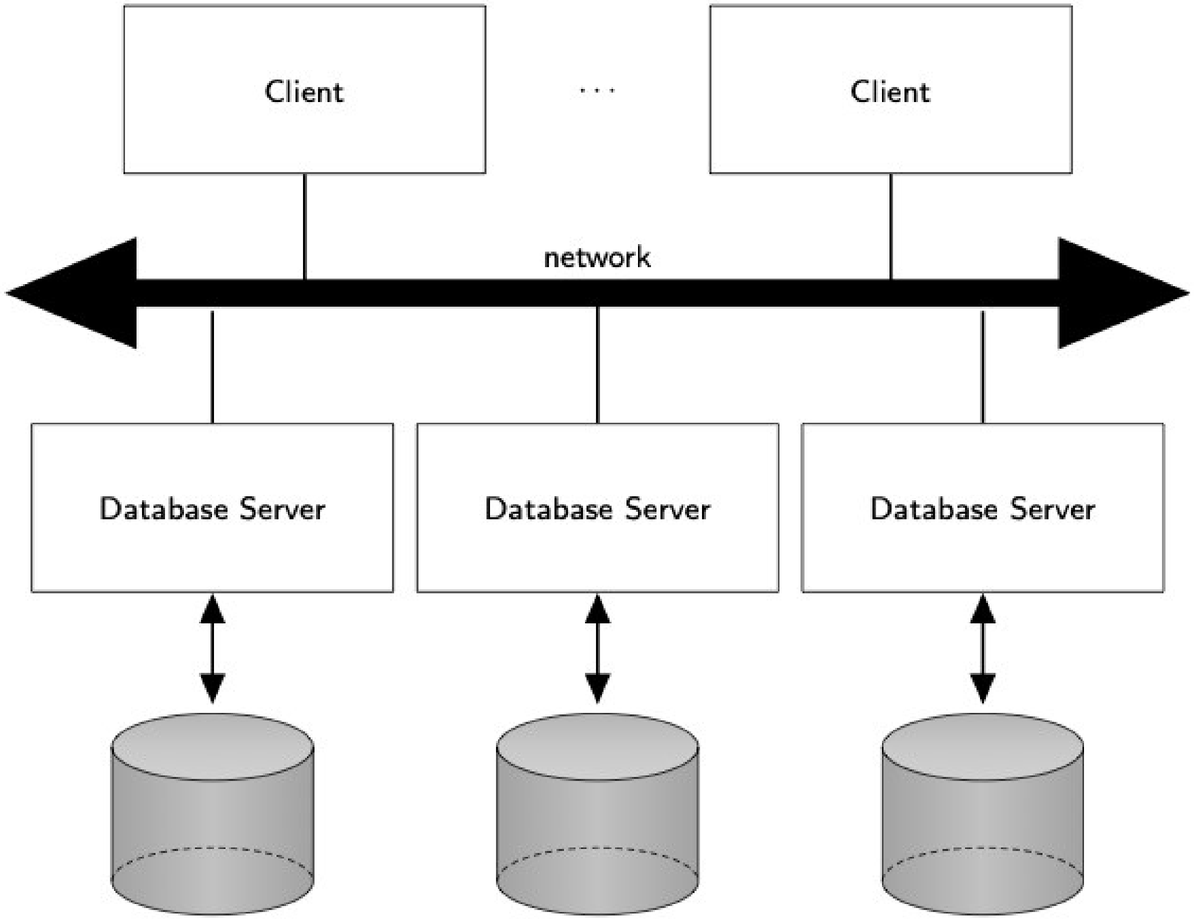
Paris employees, Boston employees, Paris projects, Boston projects



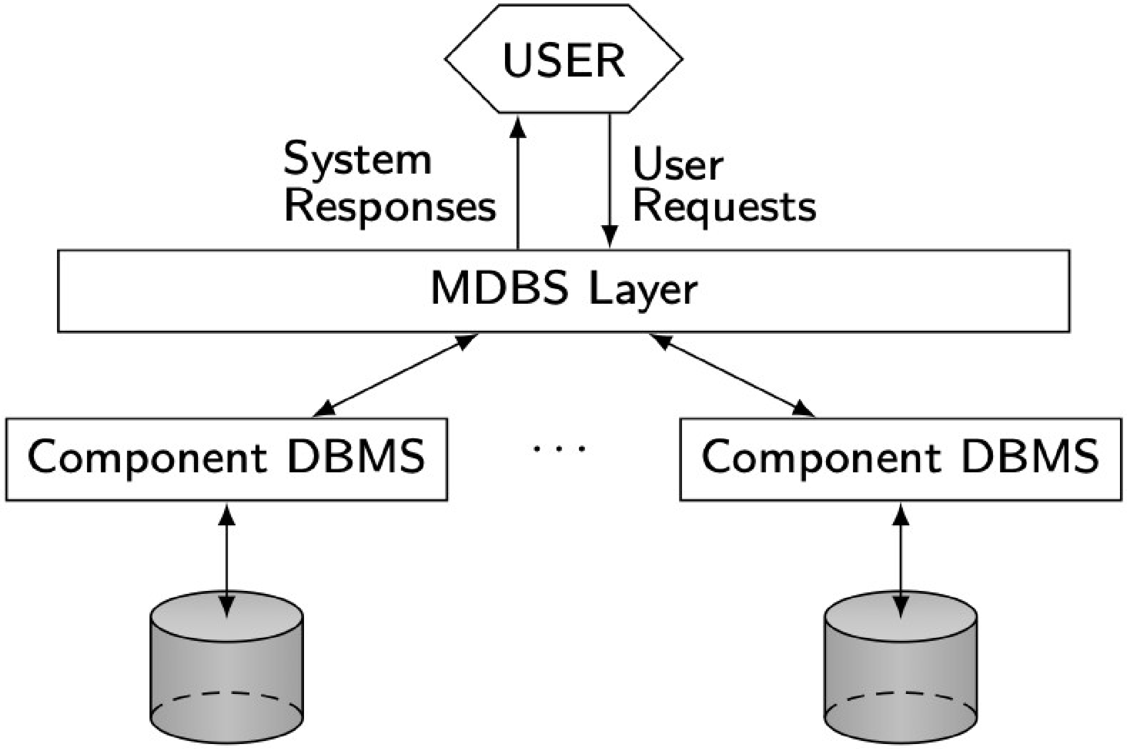
Waterloo employees, Waterloo projects, Paris projects

San Francisco employees, San Francisco projects

History – Client/Server



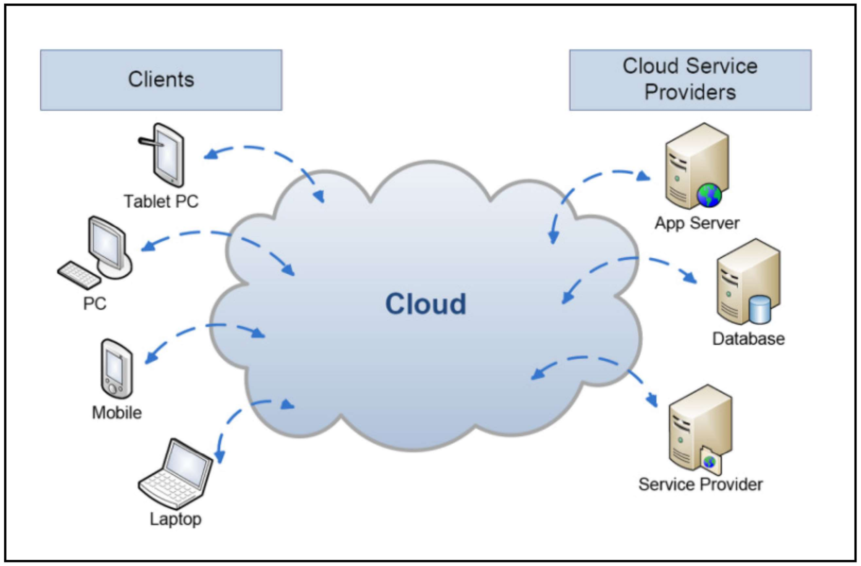
History – Data Integration



History – Cloud Computing

On-demand, reliable services provided over the Internet in a cost-efficient manner

- Cost savings: no need to maintain dedicated compute power
- Elasticity: better adaptivity to changing workload



Data Delivery Alternatives

- Delivery modes
  - Pull-only
  - Push-only
  - Hybrid
- Frequency
  - Periodic
  - Conditional
  - Ad-hoc or irregular
- Communication Methods
  - Unicast
  - One-to-many
- Note: not all combinations make sense

Outline

- Introduction
  - What is a distributed DBMS
  - History
  - Distributed DBMS promises
  - Design issues
  - Distributed DBMS architecture

Distributed DBMS Promises

➊ Transparent management of distributed, fragmented, and replicated data

➋ Improved reliability/availability through distributed transactions

➌ Improved performance

➍ Easier and more economical system expansion

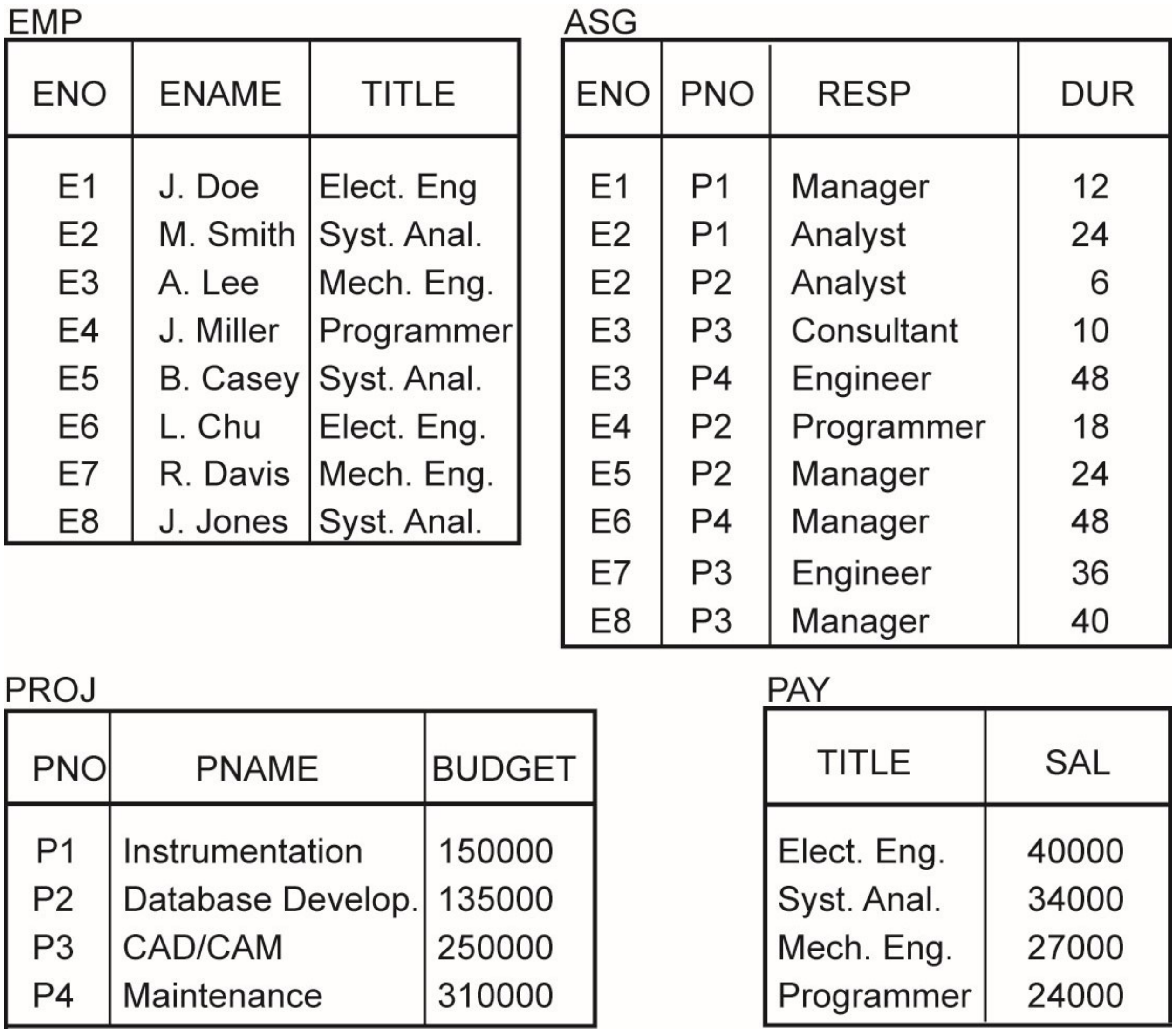
Transparency

- 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
E8	P3	Manager	40

PROJ

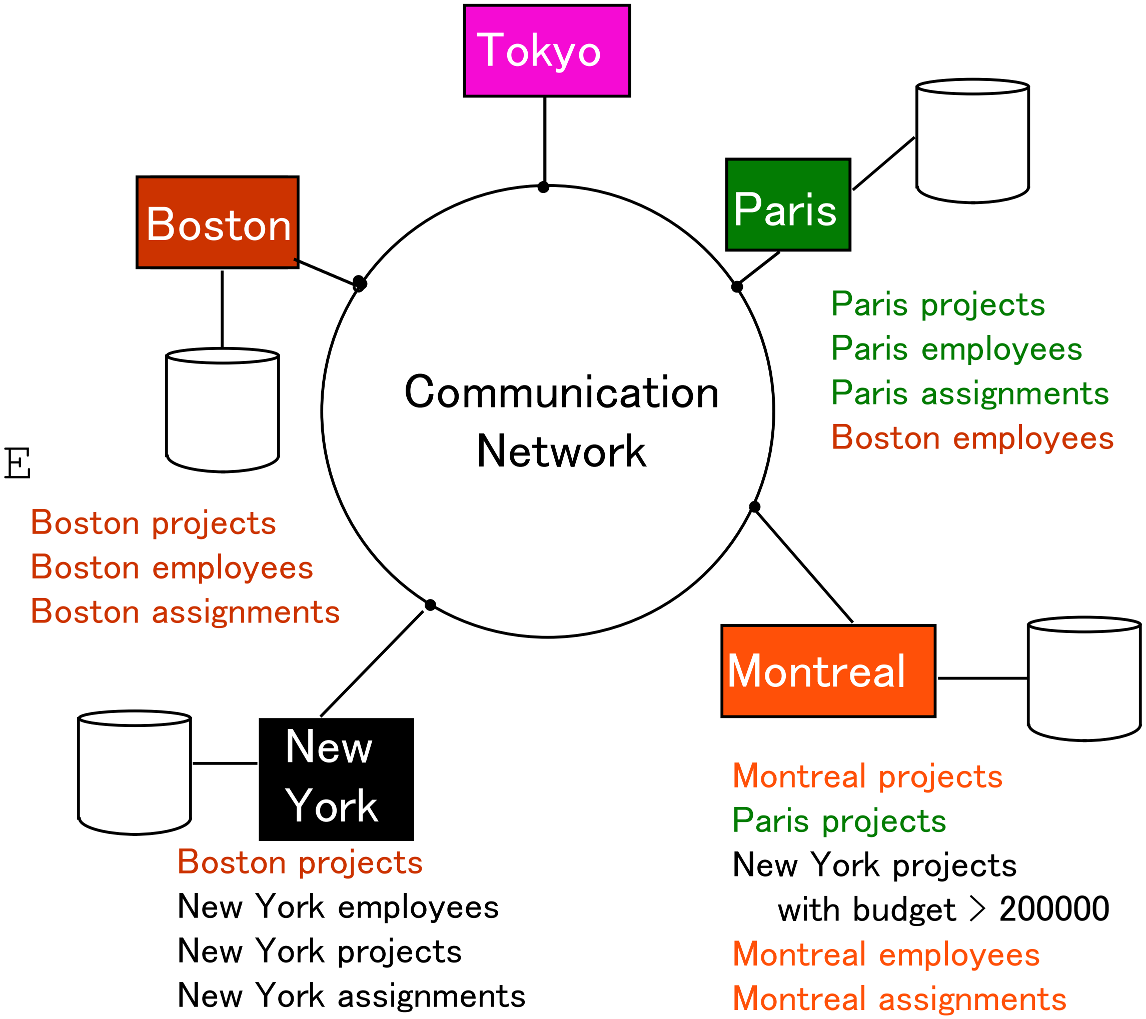
PNO	PNAME	BUDGET
P1	Instrumentation	150000
P2	Database Develop.	135000
P3	CAD/CAM	250000
P4	Maintenance	310000

PAY

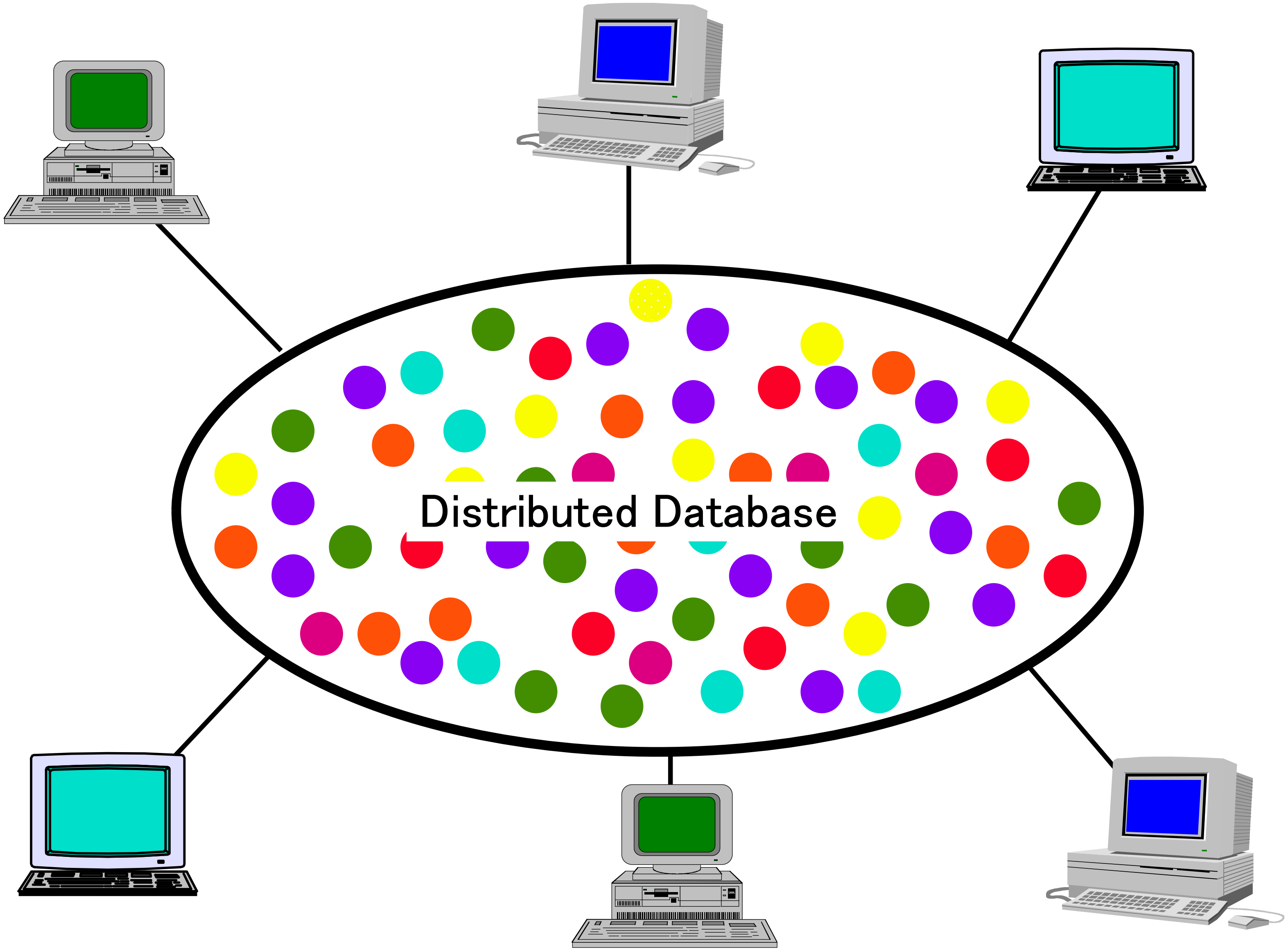
TITLE	SAL
Elect. Eng.	40000
Syst. Anal.	34000
Mech. Eng.	27000
Programmer	24000

Transparent Access

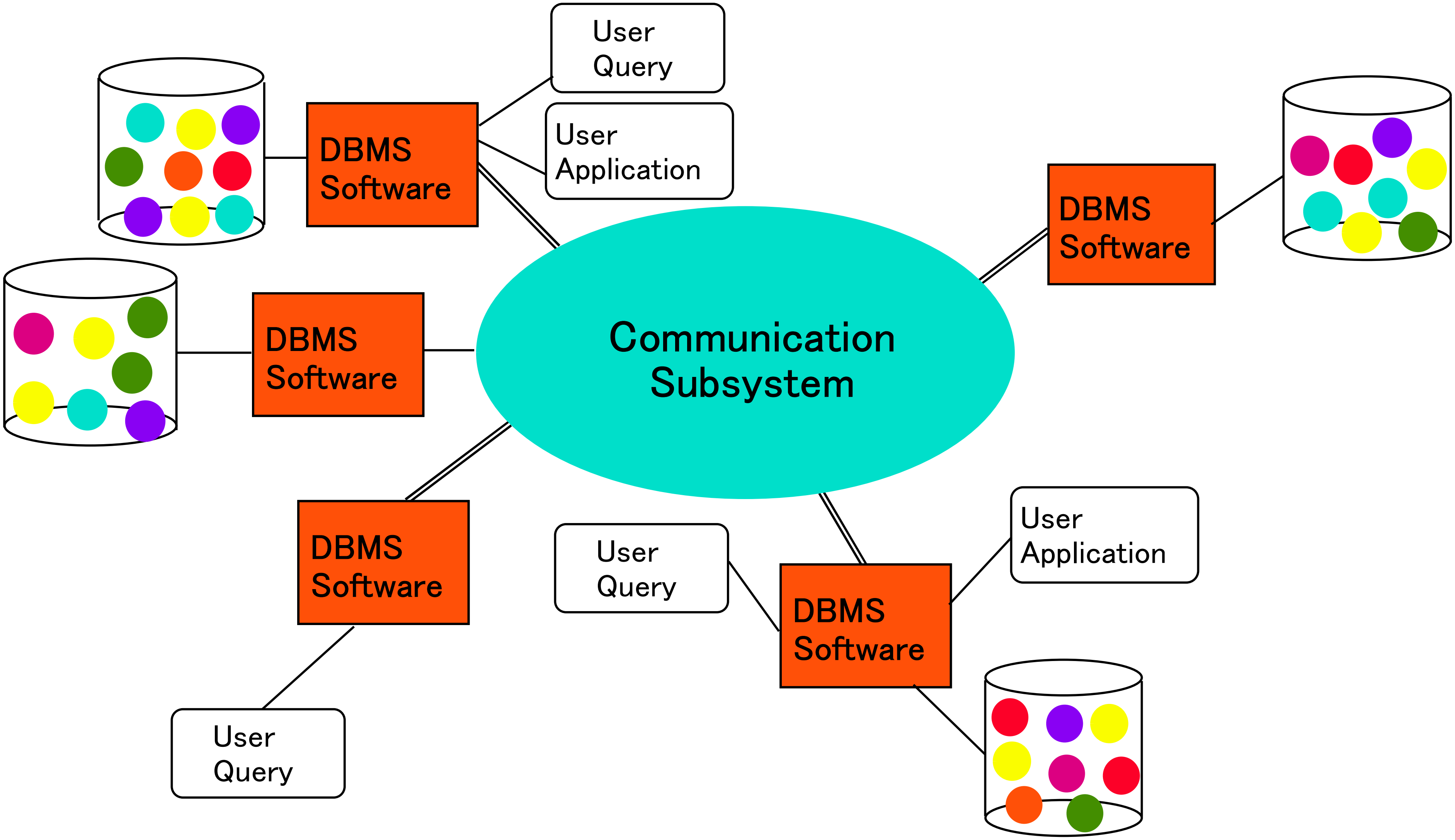
```
SELECT ENAME,SAL
FROM   EMP,ASG,PAY
WHERE  DUR > 12
AND    EMP.ENO = ASG.ENO
AND    PAY.TITLE = EMP.TITLE
```



Distributed Database - User View



Distributed DBMS - Reality



Types of Transparency

- Data independence
- Network transparency (or distribution transparency)
  - Location transparency
  - Fragmentation transparency
- Fragmentation transparency
- Replication transparency