Advanced Database Topics
A Little History

- Prior to the 1980s → hierarchical and network databases.
- Hardware → dumb terminals using private networks
- Database → centralized and stored on the disk packs
- End user terminals → simply input/output devices → Processing at the mainframe
- Data → text data
- Networks had to handle text data
- No access from outside to the organization's private network.
New Needs

- Microcomputer enabled workstation processing power.
- Satellite and network technology provided for very high speed, high traffic, and low cost long distance communications networks.
- Internet in the late 1990s and the corresponding phenomenal growth in electronic commerce (E-commerce) necessitated public access to data in people's homes.
- The volume of data needed to be transmitted increased greatly.
New Needs

- Business environment changed during the last two decades
- Information stored at different locations, on different hardware and operating systems, with different commercial DBMS products, and with different underlying data models had to be combined
- The centralized database was no longer feasible to handle these new demands
There are many advantages to using a distributed database rather than a centralized database. They are:

- Improved performance, because high traffic data are stored locally.
- More efficient data management, because the DBA workload is shared.
- Better network integrity, because the whole system does not stop if one computer goes down.
- Expansion of the database is facilitated when the organization grows, since new data does not have to be centralized. It can remain and be administered in the original location.
- Data for the whole organization can still be accessed from any location.
Distributed Database

- Data administration is improved (??)
- In a distributed database system even a simple task like creating a backup copy of the database can take a considerable amount of time.
- If the database is divided among several locations the time and workload for this task can be shared.
Replication of Data

- System failure in one location should not stop processing in other locations
- Replicate all or parts of the database in more than one location.
- Database replication improves performance and provides a fail-safe option, but it involves considerable complexity
- Replication of frequently used data improves response time and reduces network traffic
- If the data changes at one location it must be changed at all locations
C. J. Date established rules for the ideal distributed DBMS system. Rules are a goal that distributed systems strive toward, but have not yet reached. According to Date's rules:

- Each site is responsible for its own portion of the distributed database, including security, backup, and recovery.
- Each site has equal capabilities and does not rely on any other site.
- The system should work regardless of the computer hardware, operating system, or network installed at any site.
Date's Rules of Distributed Databases:

1. Local site independence
2. Central site independence
3. Failure independence
4. Location transparency
5. Fragmentation transparency
6. Replication transparency
7. Distributed query processing
8. Distributed transaction processing
9. Hardware independence
10. Operating system independence
11. Network independence
12. Database independence
Complexities of Distributed Databases

- There also are many complications involved in the management of distributed database systems.
- The distributed database must be carefully designed to insure the following:
  - Store data as close as possible to where it is used most often.
  - Make the location of the data transparent to the end user.
  - Make the system easy to expand.
  - Optimize queries to improve response time in the distributed environment.
Database Design

- The designer must analyze the organization's needs and business processes to determine the best way to distribute the database.

- There are several possibilities for storing the data in more than one location:
  - Centralized master database
  - Replication of the entire or part of the database in several locations
  - Horizontal partitions
  - Vertical partitions
  - Mixture of the above
Fragmentation

- Horizontal fragmentation of the database means that rows of a table(s) may be stored in different locations.
  - Similar to the separation of the customer table in the retailing example above.

- Vertical fragmentation means that columns of a table (i.e., attributes or groups of attributes of an entity) are stored in different locations.
Query Formulation

- Distributed databases require a considerable amount of network overhead.
- Poorly formulated query it may cause unnecessary data retrieval from the database.
- Query optimization is ideally performed by the distributed database management system.
In traditional relational databases, E-R Modeling and normalization focuses on identifying entities, their attributes, and the relationships between entities. This works well for most organizational data, especially business data.

The advent of the microcomputer and processing power on the desktop:
- Computer aided design, CAD, became the norm for engineering work, so it became necessary to store drawings.
- Powerful multimedia PCs with sound cards and color monitors enabled the manipulation of sound and video files.
- Many other applications were developed that required more than just text and numeric processing.

OODB
Why??

- These new applications were facilitated by the development of Object-Oriented Programming
- Still evolving development of object-oriented data modeling, object-oriented databases, and object-oriented database management systems
  - OODBMS and O/R DBMS are two types of database management systems that are currently available
  - O/R DBMS uses the basic theory of relational database management systems with object-oriented features added
  - OODBMS is more object-oriented and was developed separately from the relational products
  - OODMBS suffers from a lack of standardization that is available with relational database systems