Chapter 2: Database Management Systems

- Introduction to DBMS
  - DBMS is a software that enables users to define, create, maintain, and control access to the database.

Introduction…

- What do we do at each of the following stages in a database development?
  - Define the database
  - Create the database
  - Maintain the database
  - Control access to the database

History of Database Management Systems

FIRST GENERATION

- HIERARCHICAL MODEL
  - INFORMATION MANAGEMENT SYSTEM (IMS)
- NETWORK MODEL
  - CONFERENCE ON DATA SYSTEM LANGUAGES (CODASYL)
  - DATA BASE TASK GROUP (DBTG)

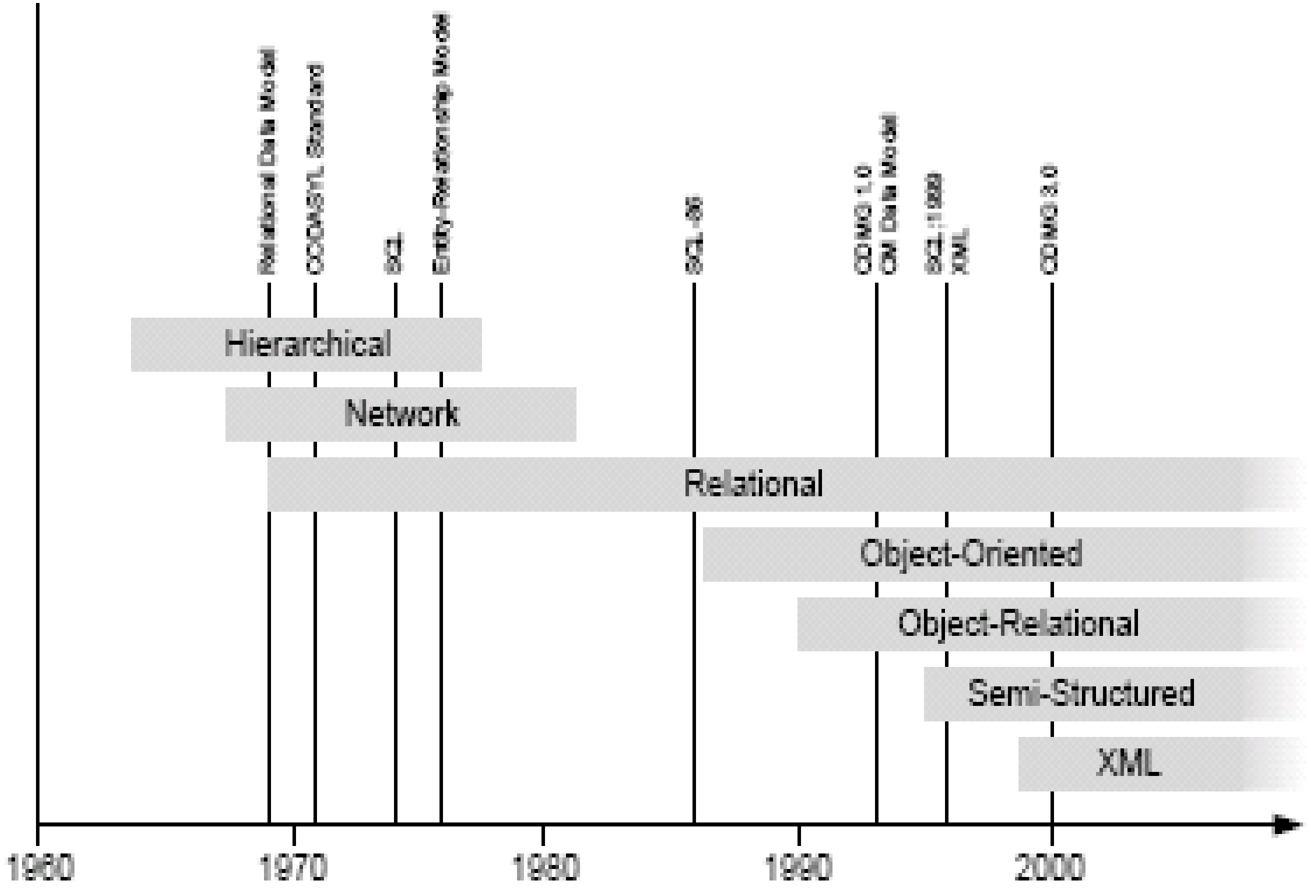
SECOND GENERATION

- RELATIONAL MODEL
  - E. F. CODD
  - DB2, ORACLE

THIRD GENERATION

- EXTENDED RELATIONAL DATA MODEL/OBJECT-RELATIONAL DATA MODEL
- OBJECTED-ORIENTED DATA MODEL

Evolution and History



Database languages

Consists of two parts

- **Data definition language (DDL)**
  - Allow users to specify the data types and structures, and the constraints on the data to be stored in the database
  - In other words, it is used to define the database schema
  - E.g. SQL DDL

- **Data manipulation language (DML)**
  - DML is used for querying, inserting, deleting, and updating database instances
  - E.g. SQL DML

Data models

- Definition
  - Integrated concept for describing data, relationship and constraints

- Types
  - Object-based data models
  - Record-based data models
  - Physical data models (internal structure, ordering, & paths)

Data models…

- Object-Based Data Models
  - Entity-relationship
    - Entity, attribute, relationship
  - Object-oriented
    - Object, class, subclass, inheritance, state (attributes), behavior (methods or actions), encapsulation, message, polymorphism

Data models…

- Record-Based Data Models
  - Relational data model
  - Network model
  - Hierarchical Model
- Physical Data Models
  - Describe how data is stored in the computer representing information such as
    - Record structures
    - Record orderings and
    - Access paths
  - Most common physical data models are
    - Unifying model
    - Frame memory

Types of DBMSs

- Several criteria are used to classify DBMS.
- The first is the data model on which the DBMS is based.
- Based on this DBMS are classified into:-
  - Relational DBMS
  - Object DBMS
  - Object-Relational DBMS
  - Hierarchical DBMS
  - Network DBMS and so on

Types of DBMS…

- Currently the two types of data models used in many commercial DBMSs are the relational data model and the object data model.

- The second criterion used to classify DBMSs is the number of users. According to this DBMS are classified into:-
  - Single user system - supports only one user at a time
  - Multi-user systems - support many users concurrently

Functions of DBMS

- **Define a database** : in terms of data types, structures and constraints
- **Manipulate the database** : querying, generating reports, insertions, deletions and modifications to its content
- **Enforce Security measures** : to prevent unauthorized access
- **Provide Data Independence** - Applications insulated from how data is structured and stored

Functions…

- Enable the user to access database catalog
  - e.g.
    - names, types, and sizes of data items
    - names of relationships
- Provide backup and recovery services
- Enforce integrity constraints on the database
  - Changes on data of DB should follow certain rules

Functions …

- Provide other utility services
  - e.g. import and export facilities
    - Indexing

- Provide transaction support
  - An action, or series of actions, carried out by a single user or application program, which reads or updates the contents of the database should always transform the database from one consistent state to another.

Functions…

- Analog to a “deal”; either it happens or it does not

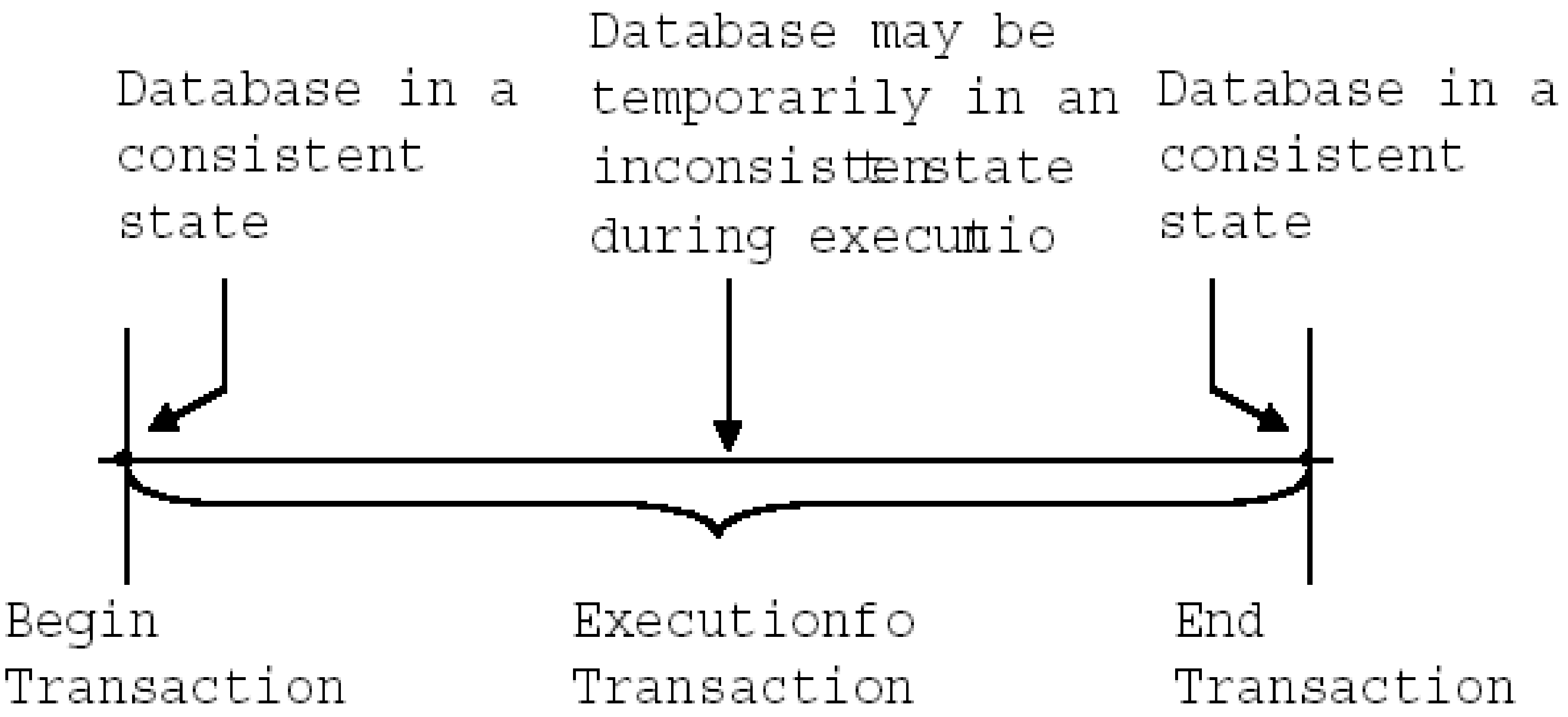
- Allow a set of concurrent users to retrieve and to update the database. Concurrency control within the DBMS guarantees that each **transaction** is correctly executed or completely aborted. OLTP (Online Transaction Processing) is a major part of database applications

Functions…

- outcomes of a transaction

  -> committed

  -> Roll back or undone



Database in a
consistent
state

Database may be
temporarily in an
inconsistenstate
during executio

Database in a
consistent
state

Begin
Transaction

Executionfo
Transaction

End
Transaction

Functions…

- Concurrent Processing and Sharing by a set of users and programs – yet, keeping all data valid and consistent
  - **e.g. problem caused by concurrency**

Time	T_1	T_2	bal_x
t1			100
t2		read(bal_x)	100
t3	read(bal_x)	$bal_x = bal_x + 100$	100
t4	$bal_x = bal_x - 10$	write(bal_x)	200
t5	write(bal_x)	commit	90
t6	commit		90

A DBMS may be unnecessary when :-

- The database and applications are simple, well defined, and not expected to change.

- There are stringent real-time requirements that may not be met because of DBMS overhead.

- Access to data by multiple users is not required.

- The database users need special operations not supported by the DBMS.

Main inhibitors (costs) of using a DBMS:

- High initial investment and possible need for additional hardware.
- Overhead for providing generality, security, concurrency control, recovery, and integrity functions.

Components of DBMS

- Query processor
  - A major DBMS component that transforms queries into a series of low-level instructions directed to the database manager.
- Database manager
  - The DM interfaces with user-submitted application programs and queries.
  - Accepts queries and examines the external and conceptual schemas to determine what conceptual records are required to satisfy the request. The DM then places a call to the file manager to perform the request

Components…

- File manager
  - The file manager manipulates the underlying storage files and manages the allocation of storage space on disk.

- DML preprocessor
  - This module converts DML statements embedded in an application program into standard function calls in the host language.
  - The DML preprocessor must interact with the query processor to generate the appropriate code.

Components…

- DDL compiler
  - The DDL compiler converts DDL statements into a set of tables containing meta-data.
- Catalog manager
  - Manages access to and maintains the system catalog.