Lecture 2
Database System Concepts and Architecture

Dr. Mourad YKHLEF

The slides content is derived mainly from Elmasri and Navathe book
Outline

- DBMS architecture
- Data models
- Schema and instance
- DBMS languages
- Database System Environment
- Database System Utilities and Tools
- DBMS classification
DBMS Architecture

- **Centralized system:** All DB functions, user applications, user interfaces, etc. in one system

- **Client-Server:**
  - The clients (workstations or PC’s) have the user application programs, interfaces and (may) have some DB functions.
  - The server has all (or most) of the DB functions
Data Models

- **Data model**
  - Collection of concepts that describe the structure of a database
    - The structure of data is the data types, relationships, and constraints
  - Data Model may include a set of basic operations (update and retrieve)

- **Two models**
  - **Conceptual data models (high level)**
    - Close to the way many users perceive data
  - **Physical data models (low level)**
    - Describe the details of how data is stored on computer storage media
Conceptual data model

- It uses concepts as:
  - **Entities**: real world objects as students, courses, projects, etc.
  - **Attributes**: properties of the entity as names, scores, Tel, etc.
  - **Relationships**: relations among some entities to represent the interaction among them as enrolled-in relation between student and courses entities.

- Categories of Data Models
  - **Relational data model**
    - Used most frequently in traditional commercial DBMSs
  - **Object data model**
    - New family of higher-level implementation data models
    - Closer to conceptual data models
Outline

- DBMS architecture
- Data models
- **Schema and instance**
- DBMS languages
- Database System Environment
- Database System Utilities and Tools
- DBMS classification
Schema

- **Database schema**: Description of a database
  - It is specified during the DB design
  - It is not expected to be changed frequently
  - It is usually represented by diagrams (Schema diagram)
  - The stored description of the schema is called the “meta-data”
Figure 2.1
Schema diagram for the database in Figure 1.2.

<table>
<thead>
<tr>
<th>STUDENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COURSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course_name</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PREREQUISITE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course_number</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section_identifier</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GRADE REPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student_number</td>
</tr>
</tbody>
</table>

---

6. Schema changes are usually needed as the requirements of the database applications change. Newer database systems include operations for allowing schema changes, although the schema change process is more involved than simple database updates.

7. It is customary in database parlance to use schemas as the plural for schema, even though schemata is the proper plural form. The word scheme is also sometimes used to refer to a schema.
Instances

- The data in the DB at a particular moment in time is called
  - a Database State or
  - Snapshot or
  - the Current Set of Occurrence
  - or Instances.

- Its initial state is generated once the DB is populated and then it moves from valid state to another valid state during data manipulation.
Schema Architecture and Data Independence

- **Internal level**
  - Describes physical storage structure of the database

- **Conceptual level**
  - Describes structure of the whole database for a community of users

- **External or view level**
  - Describes part of the database that a particular user group is interested in
Schema Architecture and Data Independence

Figure 2.2
The three-schema architecture.
Data Independence

- Capacity to change the schema at one level of a database system
  - Without having to change the schema at the next higher level

- Types:
  - Logical
  - Physical
- **Logical data independence:**
  is the ability of changing the conceptual Schema without having to change the external schema or the application programs.
  1. Conceptual schema may be changed to expand the DB as adding new data items or reducing it by removing data items.
  2. The applications involved in the modified data items can only changed without any effect on the remaining applications.

- **Physical data independence:**
  is the ability of changing the internal Schema without having to change the conceptual schema.
  1. Internal schema may be changed due to some file reorganization to meet certain system performance.
Outline

- DBMS architecture
- Data models
- Schema and instance
- DBMS languages
- Database System Environment
- Database System Utilities and Tools
- DBMS classification
DBMS Languages

- **Data definition language (DDL)**
  - Defines both schemas

- **Storage definition language (SDL)**
  - Specifies the internal schema

- **View definition language (VDL)**
  - Specifies user views/mappings to conceptual schema

- **Data manipulation language (DML)**
  - Allows retrieval, insertion, deletion, modification
The Database System Environment (for reading)

- **Query Compiler** handles high-level queries by parsing, analyzing and compiling their statements. Then it calls the Run-time processor to execute it.

- **Pre-compiler** extracts the DML commands from the host language.

- **DML compiler** compiles the DML to build the code which is passed to the Run-time processor for execution.
Outline

- DBMS architecture
- Data models
- Schema and instance
- DBMS languages and Interfaces
- Database System Environment
- Database System Utilities and Tools
- DBMS classification
Database System Utilities

- **Loading**: Load existing data files
- **Backup**: Creates a backup copy of the database
- **Database storage reorganization**
  - Reorganize a set of database files into different file organizations
- **Performance monitoring**
  - Monitors database usage and provides statistics to the DBA
- **CASE Tools (example Power Designer)**
- **Data dictionary (data repository) system**
  - Stores design decisions, usage standards, application program descriptions, and user information
- **Application development environments**
- **Communications software**
# Classification of Database Management Systems

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Model</td>
<td>Relational – Object Oriented – Object Relational – Network – Hierarchical</td>
</tr>
<tr>
<td># of users</td>
<td>Single user – Multiuser</td>
</tr>
<tr>
<td># of sites</td>
<td>Centralized – Distributed (Homogeneous – Heterogeneous) – Client-Server</td>
</tr>
</tbody>
</table>