Databases and Database Users

Lecture 1

Dr. Marwa Hussien

About the Course

- Text book: Fundamentals of Database Systems.
- Course has 100 points:
  - 50 points (final)
  - 15 points (mide-term exam)
  - 15 points (lab work)
  - 20 points (lab exam)

Outline

- Basic Definitions
- Types of Databases and Database Applications
- Typical DBMS Functionality
- Example of a Database (UNIVERSITY)
- Main Characteristics of the Database Approach
- Types of Database Users
- Advantages of Using the Database Approach
- Centralized and Client-Server Architectures

Basic Definitions

- **Database:** a collection of related data.
- **Data:** is a collection of facts and figures that can be processed to produce information.
- **Database Management System (DBMS):** a software package/ system to facilitate the creation and maintenance of a computerized database. It stores data in such a way that it becomes easier to retrieve, manipulate, and produce information.
- **Database System:** the DBMS software together with the data itself. Sometimes, the applications are also included.

Database Implicit Properties

- A database represents some aspect of the real world, sometimes called the miniworld or the universe of discourse (UoD).
- A database is a logically coherent collection of data with some inherent meaning.
- A random assortment of data cannot correctly be referred to as a database.
- A database is designed, built, and populated with data for a specific purpose. It has an intended group of users and some preconceived applications in which these users are interested.

Types of Databases and Database Applications

- Traditional Applications:
  - Numeric and Textual Databases
- More Recent Applications:
  - Multimedia Databases
  - Geographic Information Systems (GIS)
  - Biological and Genome Databases
  - Data Warehouses
  - Mobile databases
  - Real-time and Active Databases

Impact of Databases and Database Technology

- **Businesses:** Banking, Insurance, Retail, Transportation, Healthcare, Manufacturing
- **Service Industries:** Financial, Real-estate, Legal, Electronic Commerce, Small businesses
- **Education :** Resources for content and Delivery
- **More recently:** Social Networks, Environmental and Scientific Applications, Medicine and Genetics
- **Personalized Applications:** based on smart mobile devices

Recent Types of Database Developments

Social Networks started capturing a lot of information about people and about communications among people-posts, tweets, photos, videos in systems such as:

Facebook - Twitter - Linked-In: all constitutes data)

Search Engines

Google, Bing, Yahoo : collect their own repository of web pages for searching purposes

Big Data storage systems involving large clusters of distributed computers

NOSQL (Not Only SQL) systems

A large amount of data now resides on the "cloud" which means it is in huge data centers using thousands of machines.

Simplified Database System Environment

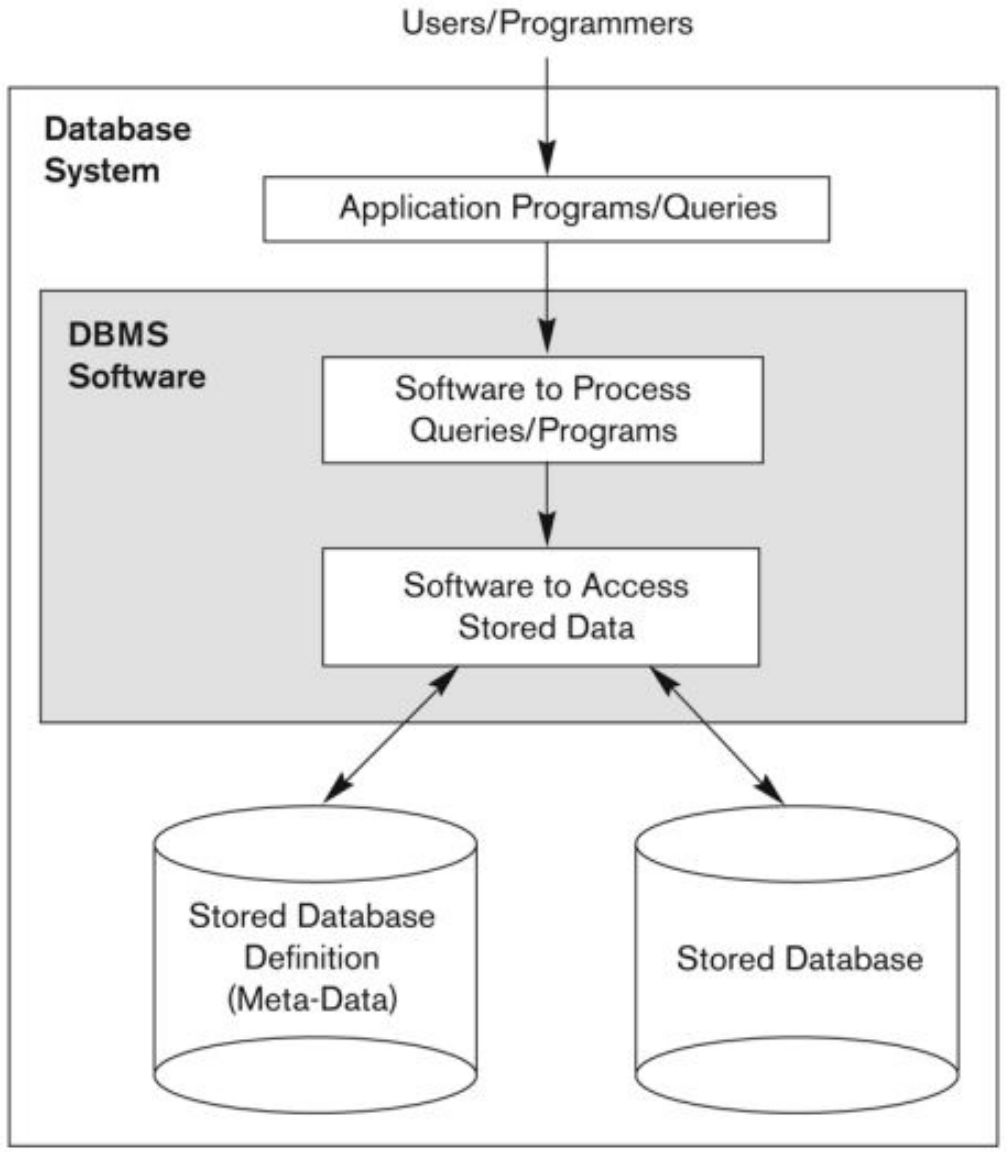

Figure 1.1
A simplified database system environment.

Typical DBMS Functionality

1. Define a particular database in terms of its data types, structures, and constraints
2. Construct or Load the initial database contents on a secondary storage medium
3. Manipulating the database:
   - Retrieval: Querying, generating reports
   - Modification: Insertions, deletions and updates to its content
   - Accessing the database through Web applications
4. Processing and Sharing by a set of concurrent users and application programs – yet, keeping all data valid and consistent

Typical DBMS Functionality

5. Security measures to prevent unauthorized access
6. “Active” processing to take internal actions on data
7. Presentation and Visualization of data
8. Maintenance of the database and associated programs over the lifetime of the database application
9. Database Protection includes system protection against hardware or software malfunction (or crashes) and security protection against unauthorized or malicious access.

Application Activities Against a Database

- Applications interact with a database by generating:
  - Queries: that access different parts of data and formulate the result of a request
  - Transactions: that may read some data and “update” certain values or generate new data and store that in the database
- Applications must not allow unauthorized users to access data
- Applications must keep up with changing user requirements against the database

Example of a Simple Database

- Mini-world for the example: Part of a UNIVERSITY environment.
- Some mini-world entities:
  - STUDENTs
  - COURSEs
  - SECTIONs (of COURSEs)
  - DEPARTMENTs
  - INSTRUCTORs
- Some mini-world relationships:
  - SECTIONs are of specific COURSEs
  - STUDENTs take SECTIONs
  - COURSEs have prerequisite COURSEs
  - INSTRUCTORs teach SECTIONs
  - COURSEs are offered by DEPARTMENTs
  - STUDENTs major in DEPARTMENTs

Example of a Simple Database

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
92	CS1310	Fall	04	Anderson
102	CS3320	Spring	05	Knuth
112	MATH2410	Fall	05	Chang
119	CS1310	Fall	05	Anderson
135	CS3380	Fall	05	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Figure 1.2
A database that stores student and course information.

Main Characteristics of the Database Approach

- **DBMS catalog:** stores the description of a particular database (e.g. data structures, types, and constraints)
- **Meta-data:** The description of the data stored in a database (Data about data)
- **A data model:** is used to hide storage details and present the users with a conceptual view of the database
- **Database views:** Each user may see a different view of the database, which describes only the data of interest to that user

Example of a Database Catalog

RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
….	….	…..
….	….	…..
….	….	…..
Prerequisite_number	XXXXNNNN	PREREQUISITE

Figure 1.3
An example of a database catalog for the database in Figure 1.2.

Main Characteristics of the Database Approach

- Allowing a set of concurrent users (multiple users) to retrieve from and to update the database.
- **Concurrency control:** within the DBMS guarantees that each transaction is correctly executed or aborted
- **Recovery subsystem:** ensures each completed transaction has its effect permanently recorded in the database
- **OLTP (Online Transaction Processing):** is a major part of database applications. This allows hundreds of concurrent transactions to execute per second.

Database Users

- Users may be divided into:
  - **Actors on the Scene:** Those who actually use and control the database content, and those who design, develop and maintain database applications
  - **Workers Behind the Scene:** Those who design and develop the DBMS software and related tools, and the computer systems operators

Database Users – Actors on the Scene

1. **Database administrators:** Responsible for authorizing access to the database, for coordinating and monitoring its use, acquiring software and hardware resources, controlling its use and monitoring efficiency of operations
2. **Database Designers:** Responsible to define the content, the structure, the constraints, and functions or transactions against the database. They must communicate with the end-users and understand their needs
3. **End-users:** They use the data for queries, reports and some of them update the database content.
   - Casual: access database occasionally when needed
   - Naïve or Parametric: they make up a large section of the end-user population.
     - Users of Mobile Apps, bank-tellers, reservation clerks, social media users

Database Users – Actors on the Scene

4. **System Analysts and Application Developers:** This category currently accounts for a very large proportion of the IT work force.
   - System Analysts: They understand the user requirements of naïve and sophisticated users and design applications including canned transactions to meet those requirements.
   - Application Programmers: Implement the specifications developed by analysts and test and debug them before deployment.
   - Business Analysts: There is an increasing need for such people who can analyze vast amounts of business data and real-time data (“Big Data”) for better decision making related to planning, advertising, marketing etc.

Database Users – Actors behind the Scene

1. **System Designers and Implementors:** Design and implement DBMS packages in the form of modules and interfaces and test and debug them. The DBMS must interface with applications, language compilers, operating system components, etc.
2. **Tool Developers:** Design and implement software systems called tools for modeling and designing databases, performance monitoring, prototyping, test data generation, user interface creation, simulation etc. that facilitate building of applications and allow using database effectively.
3. **Operators and Maintenance Personnel:** They manage the actual running and maintenance of the database system hardware and software environment.

Advantages of Using the Database Approach

- Controlling redundancy in data storage and in development and maintenance efforts.
- Sharing of data among multiple users.
- Restricting unauthorized access to data. Only the DBA staff uses privileged commands and facilities.
- Providing persistent storage for program Objects
- Providing Storage Structures (e.g. indexes) for efficient Query Processing
- Providing optimization of queries for efficient processing.
- Providing backup and recovery services.
- Providing multiple interfaces to different classes of users.
- Representing complex relationships among data.
- Enforcing integrity constraints on the database.

Centralized and Client-Server DBMS Architectures

- Centralized DBMS:
  - Combines everything into single system including DBMS software, hardware, application programs, and user interface processing software.
  - User can still connect through a remote terminal – however, all processing is done at centralized site.

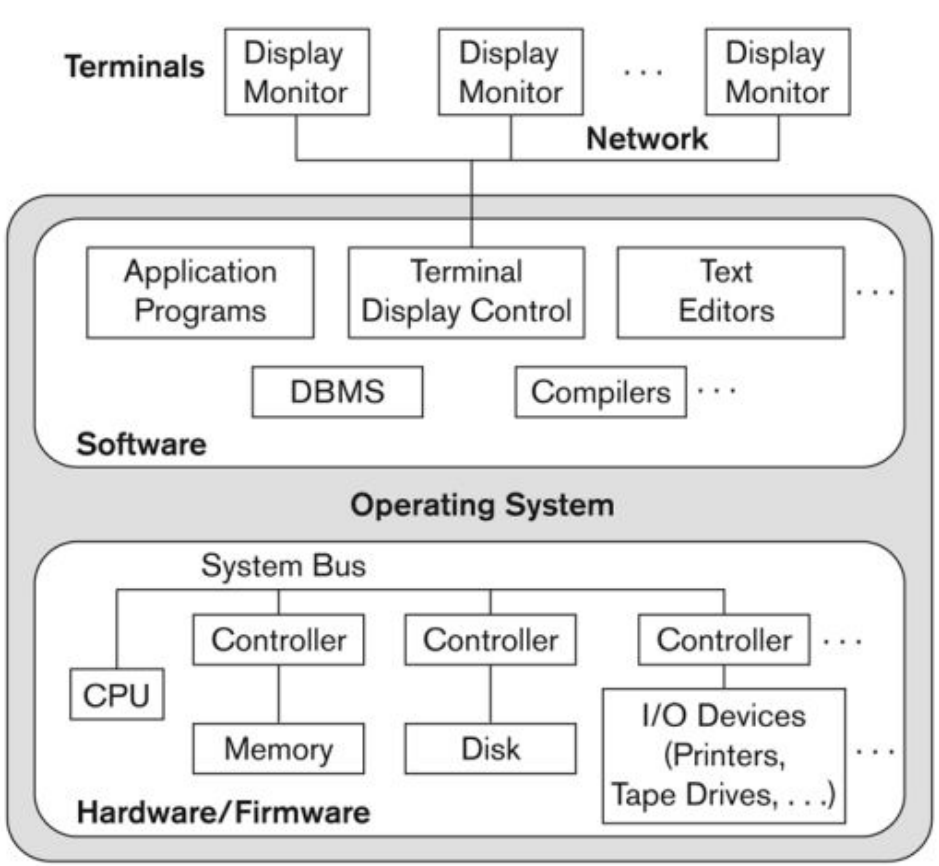

Figure 2.4
A physical centralized architecture.

Client-server Architecture

- Clients:
  - Provide appropriate interfaces through a client software module to access and utilize the various server resources
  - Connected to the servers via some form of a network.
    - (LAN: local area network, wireless network, etc.)
- DBMS Server:
  - Provides database query and transaction services to the clients
  - Relational DBMS servers are often called SQL servers, query servers, or transaction servers
  - Applications running on clients utilize an Application Program Interface (API) to access server databases via standard interface such as: JDBC: for Java programming access

Three Tier Client-Server Architecture

- Common for Web applications
- Intermediate Layer called Application Server or Web Server:
  - Stores the web connectivity software and the business logic part of the application used to access the corresponding data from the database server
  - Acts like a conduit for sending partially processed data between the database server and the client.
- Three-tier Architecture Can Enhance Security:
  - Database server only accessible via middle tier
  - Clients cannot directly access database server
  - Clients contain user interfaces and Web browsers
  - The client is typically a PC or a mobile device connected to the Web

Three Tier Client-Server Architecture

Figure 2.7
Logical three-tier client/server architecture, with a couple of commonly used nomenclatures.