Read Chapter 1

Databases and Database Users

Outlines

1. Introduction
2. Characteristics of the Database Approach
3. Database users
4. Advantages of Using the DBMS Approach
5. Brief History of Database Applications
6. When Not to Use a DBMS

1 Introduction

Databases and database systems are an essential component of life in modern society.

- We encounter several activities every day that involve some interaction with a database:
  - Deposit or withdraw funds in the bank
  - Make a hotel or airline reservation
  - Access a computerized library catalog to search for a bibliographic item
  - Purchase something online (a book, toy, computer, ..)

Types of Databases and Database Applications

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

Recent Developments (1)

- **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 of the above constitutes data
- **Search Engines**, Google, Bing, Yahoo: collect their own repository of web pages for searching purposes

Recent Developments (2)

- New Technologies are emerging from the so-called non-database software vendors to manage vast amounts of data generated on the web:
  - **Big Data** storage systems involving large clusters of distributed computers (Chapter 25)
  - **NOSQL** (Originally referring to "non-Structured Query Language" or "non-relational") systems (Chapter 24)
- A large amount of data now resides on the “cloud” which means it is in huge data centers using thousands of machines.

Basic Definitions

- **Database:**
  - A collection of related data.
- **Data:**
  - Known facts that can be recorded and have an implicit meaning.
- **Mini-world:**
  - Some part of the real world about which data is stored in a database. For example, student grades and transcripts at a university.
- **Database Management System (DBMS):**
  - A software package/system to facilitate **the creation and maintenance** of a **computerized** database.
- **Database System:**
  - The DBMS software together with the data itself. Sometimes, the applications are also included.

Simplified database system environment

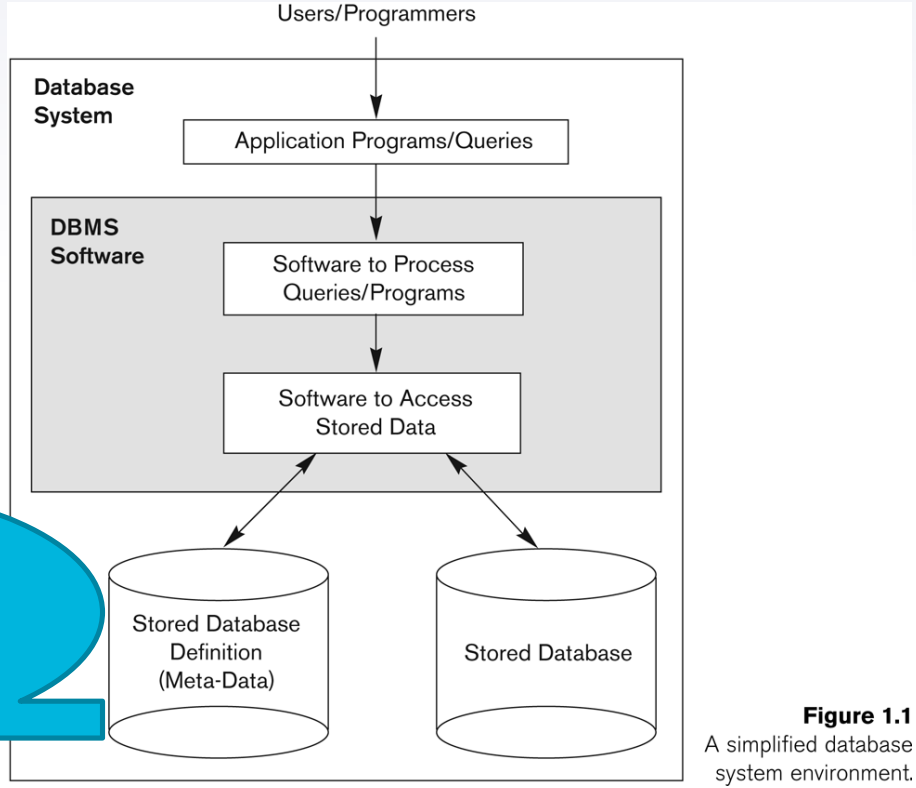


Figure 1.1
A simplified database system environment.

Descriptive information stored by the DBMS in the form of a database catalog or dictionary

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

What a DBMS Facilitates

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

Other DBMS Functionalities

- DBMS may additionally provide:
  - **Protection** or **Security** measures to prevent unauthorized access
  - **"Active" processing** to take internal actions on data
  - **Presentation** and **visualization** of data
  - **Maintenance** of the database and associated programs over the lifetime of the database application

Application Programs and DBMS

- 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

Example of a Database (with a Conceptual Data Model)

- **Mini-world for the example:**
  - Part of a UNIVERSITY environment
- **Some mini-world *entities*:**
  - STUDENTs
  - COURSEs
  - SECTIONs (of COURSEs)
  - (Academic) DEPARTMENTs
  - INSTRUCTORs

Example of a Database (with a Conceptual Data Model)

- **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

STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

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	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	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.

- Database manipulation involves querying and updating.
- Examples of queries are as follows:
  - Retrieve the transcript—a list of all courses and grades—of 'Smith'
  - List the names of students who took the section of the 'Database' course offered in fall 2008 and their grades in that section
  - List the prerequisites of the 'Database' course
- Examples of updates include the following:
  - Change the class of 'Smith' to sophomore
  - Create a new section for the 'Database' course for this semester
  - Enter a grade of 'A' for 'Smith' in the 'Database' section of last semester

2 Characteristics of the Database Approach

Characteristics of the Database Approach

- **1. Self-describing nature of a database system:**
  - A DBMS **catalog** stores the description of a particular database (e.g. data structures, types, and constraints)
  - The description is called **meta-data***.
  - This allows the DBMS software to work with different database applications.

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

Note: Major_type is defined as an enumerared type with all known majors. XXXXNNNN is used to define a type with four alpha characters followed by four digits

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

Characteristics of the Database Approach (cont.)

- **2. Insulation between programs and data:**
  - Called **program-data independence**.
  - Allows changing data structures and storage organization without having to change the DBMS access programs
    - E.g., ADTs

Data Item Name	Starting Position in Record	Length in Characters (bytes)
Name	1	30
Student_number	31	4
Class	35	1
Major	36	4

Figure 1.4
Internal storage format for a STUDENT record, based on the database catalog in Figure 1.3.

Characteristics of the Database Approach (cont.)

- **3.Data Abstraction:**
  - A **data model** is used to hide storage details and present the users with a **conceptual view** of the database.
  - Programs refer to the data model constructs rather than data storage details
- **4.Support of multiple views of the data:**
  - Each user may see a different view of the database, which describes **only** the data of interest to that user.

Characteristics of the Database Approach (continued)

(a)

TRANSCRIPT

Student_name	Student_transcript				
	Course_number	Grade	Semester	Year	Section_id
Smith	CS1310	C	Fall	08	119
	MATH2410	B	Fall	08	112
Brown	MATH2410	A	Fall	07	85
	CS1310	A	Fall	07	92
	CS3320	B	Spring	08	102
	CS3380	A	Fall	08	135

(b)

COURSE_PREREQUISITES

Course_name	Course_number	Prerequisites
Database	CS3380	CS3320
		MATH2410
Data Structures	CS3320	CS1310

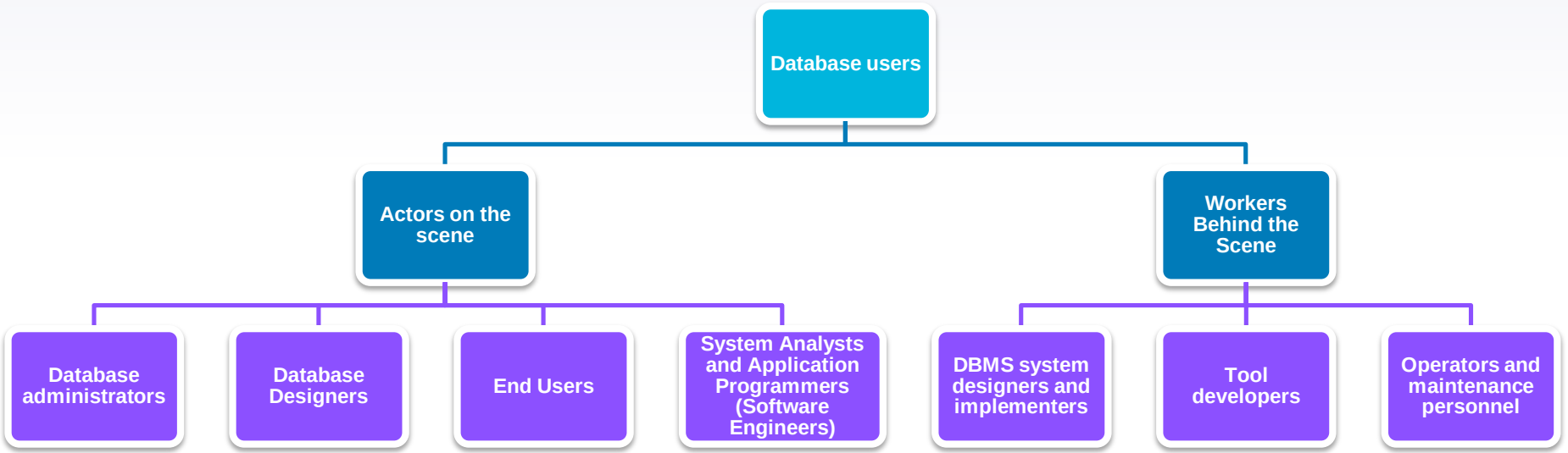
Figure 1.5
Two views derived from the database in Figure 1.2. (a) The TRANSCRIPT view. (b) The COURSE_PREREQUISITES view.

Characteristics of the Database Approach (cont.)

- **5.Sharing of data and multi-user transaction processing:**
  - Allowing a set of **concurrent 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.
  - Ex. assign a seat on an airline flight

3 Database users

Database Users



Database Users

- Users may be divided into
  - Those who actually **use and control** the database content, and those who **design, develop and maintain** database applications (called "Actors on the Scene")
    - People whose jobs involve the day-to-day use of a large database
  - Those **who design and develop the DBMS software** and related tools, and the computer systems operators (called "Workers Behind the Scene").
    - who work to **maintain the database system environment** but who are **not actively interested** in the database contents as part of their daily job.

Database Users – Actors on the Scene

- 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.
    - The DBA is accountable for problems such as security breaches and poor system response time.
  - **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.

Database Users - Actors on the Scene (continued)

- **3.End-users:** They use the data for queries, reports and some of them update the database content. End-users can be categorized into:
  - **Casual**: access database occasionally when needed
  - **Naïve** or Parametric: they make up a large section of the end-user population.
    - They use **previously well-defined functions** in the form of "**canned transactions**" against the database.
    - Users of Mobile Apps mostly fall in this category
    - Bank-tellers or reservation clerks are parametric users who do this activity for an entire shift of operations.
    - Social Media Users post and read information on social media websites

Database Users - Actors on the Scene (cont.)

3.End-users:

- **Sophisticated:**
  - These include business analysts, scientists, engineers, others thoroughly familiar with the system capabilities.
  - Many use tools in the form of software packages that work closely with the stored database.
- **Stand-alone:**
  - Mostly maintain personal databases using ready-to-use packaged applications.
  - An example is the user of a tax program that creates its own internal database.
  - Another example is a user that maintains a database of personal photos and videos.

Database Users-Actors on the Scene (continued)

- **4.System Analysts and Application Developers (software engineers)**

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 Workers behind the Scene

- **Workers behind the scenes are associated with the design, development, and operation of the DBMS software and system environment.**
  - **System Designers and Implementors:** Design and implement DBMS packages in the form of **modules** and **interfaces** and test and debug them (modules for implementing the catalog, query language processing, interface processing). The DBMS must interface with applications, language compilers, operating system components, etc.
  - **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.
  - **Operators and maintenance personnel** (system administration personnel) are responsible for the actual **running and maintenance** of the **hardware and software** environment for the database system.

4 Advantages of Using the DBMS Approach

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 storage structures (e.g. indexes) for efficient query processing .

Advantages of Using the Database Approach (continued)

- Providing optimization of queries for efficient processing (provide capabilities for *efficiently executing queries and updates*)
- Providing **backup and recovery services** from hardware or software failures.
- Providing **multiple interfaces** to different classes of users
- Representing complex relationships among data
- Enforcing **integrity constraints** on the database
- Drawing inferences and actions from the stored data using deductive and active rules and **triggers**

5

Historical Development of Database Technology

Historical Development of Database Technology

- Early database applications:
  - The **Hierarchical and Network Models** were introduced in mid 1960s and dominated during the seventies.
  - A bulk of the worldwide database processing still occurs using these models, particularly, the hierarchical model using IBM's IMS system.
- Relational model based systems:
  - Relational model was originally introduced in 1970, was heavily researched and experimented within IBM Research and several universities.
  - **Relational DBMS** Products emerged in the early 1980s.

Historical Development of Database Technology (cont.)

- Object-oriented and emerging applications:
  - **Object-Oriented** Database Management Systems (OODBMSs) were introduced in late 1980s and early 1990s to cater to the need of **complex data processing** in engineering design and other applications.
    - Their use has not taken off much
  - Many relational DBMSs have incorporated object database concepts, leading to a new category called ***object-relational*** DBMSs (ORDBMSs)
  - *Extended relational* systems add further capabilities (e.g. for multimedia data, text, XML, and other data types)

6

When not to use a DBMS

When not to use a 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
- When a DBMS may be **unnecessary**:
  - If the database and applications are simple, well defined, and not expected to change
  - If access to data by multiple users is not required
- When a DBMS may be **infeasible**
  - In embedded systems where a general purpose DBMS may not fit in available storage

Summary

- In this chapter we defined a database as a collection of related data, where *data* means recorded facts.
- A typical database represents some aspect of the real world and is used for specific purposes by one or more groups of users. A DBMS is a generalized software package for implementing and maintaining a computerized
- database.
- The database and software together form a database system.
- We identified several characteristics that distinguish the database approach from traditional file-processing applications, and we discussed the main categories of database users, or the *actors on the scene*.

Summary (continued)

- We noted that in addition to database users, there are several categories of support personnel, or *workers behind the scene*, in a database environment.
- We presented a list of capabilities that should be provided by the DBMS software to the DBA, database designers, and end users to help them design, administer, and use a database.
- A brief historical perspective on the evolution of database applications was given.
- We pointed out the recent rapid growth of the amounts and types of data that must be stored in databases, and we discussed the emergence of new systems for handling "big data" applications.
- Finally, we discussed the overhead costs of using a DBMS and discussed some situations in which it may not be advantageous to use one.