Database Systems

Lecture #01

Sang-Wook Kim

Hanyang University

Objectives

◆ To understand basic concepts of databases and their related terminology

- Database / DBMS / Database system

- Characteristics of a database

- Users of database systems

- DBMS features

Data
And
Knowledge
Engineering

Outline

◆ Overview

◆ An Example of a Database System

◆ Characteristics of the Database

◆ Users of Database Systems

◆ Workers behind the Scene

◆ DBMS Features

◆ Advantages of Using the DBMS

◆ When Not to Use a DBMS

◆ Data

● Known facts that can be *recorded* and that have *implicit meaning*

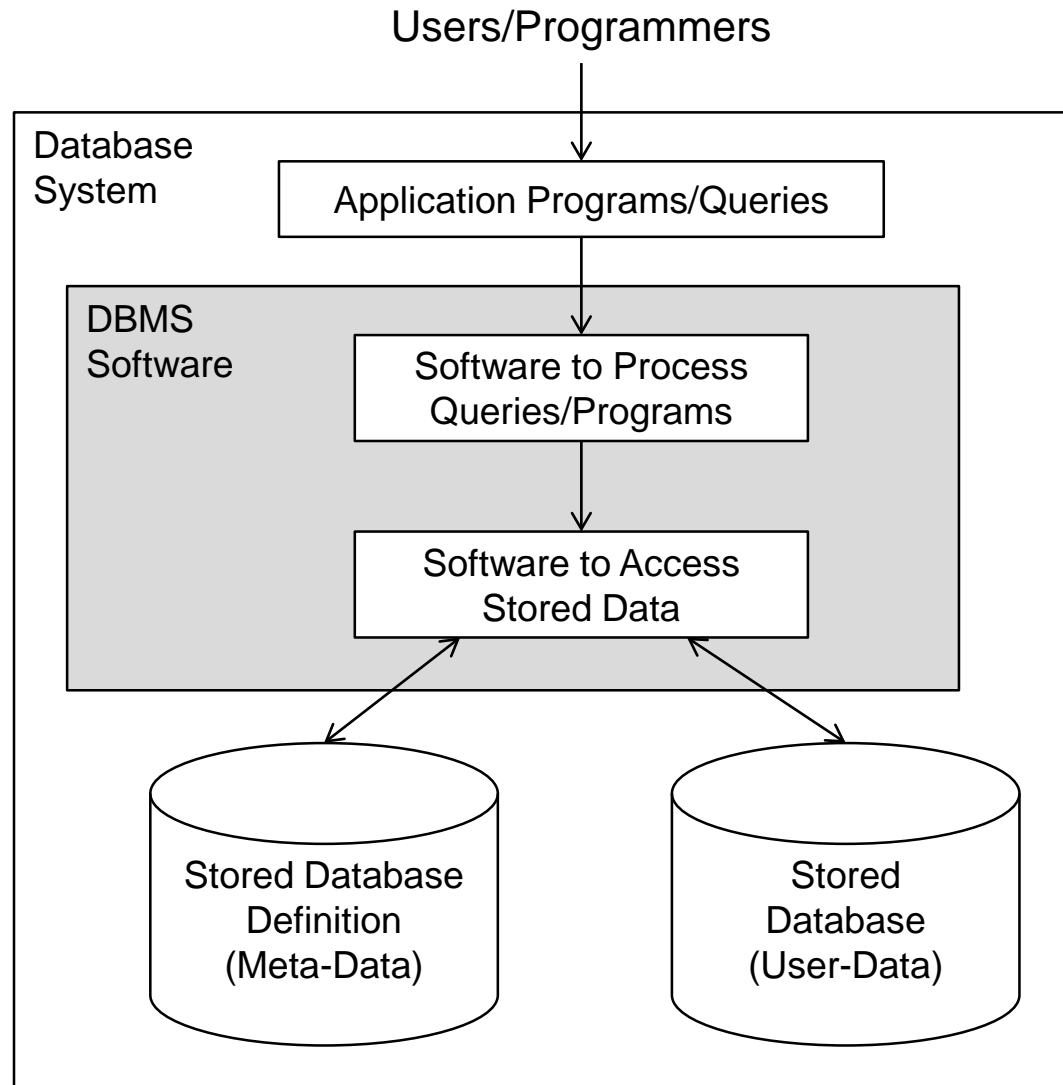▪ Ex. names, addresses, phone numbers, …

◆ Database

- Logically coherent collection of *related data*

  ▪ Ex. Phone book for a company

◆ Mini-world

- Represents some aspect of a real world

- Object of building a database

◆ Database Management System (DBMS)

  ● Collection of programs (software)

  ● Enables user to create and maintain a database (very easily)

◆ Database System

  ● DBMS + Database

Users/Programmers

Database System

Application Programs/Queries

DBMS Software

Software to Process Queries/Programs

Software to Access Stored Data

Stored Database Definition (Meta-Data)

Stored Database (User-Data)

Example of a Database

◆ University database

- Information concerning students, courses, professors, and grades

- Relationships among them

Example of a Database

◆ Data records

- STUDENT
  - Name, StudentNumber, Class, Major

- COURSE
  - CourseNumber, CreditHours, Department

- SECTION
  - SectionIdentifier, CourseNumber, Semester, Year, Instructor

Example of a Database

◆ Data records (cont'd)

- GRADE_REPORT

  - StudentNumber, SectionIdentifier, Grade

- PREREQUISITE

  - CourseNumber, PrerequisiteNumber

Example of a 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

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

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

◆ Self-describing nature of a database system

- **Meta-data**

  - Describes structure of the database

  - Data necessary to maintain the user data

- System Catalog: space for storing meta-data in DBMS

- Database systems store and maintain not only *user data* but also *meta-data*

Characteristics of the Database

◆ Insulation between programs and data

- Owing to the self-describing nature of a database system

- A change of data structure does not affect programs

- Provides higher *program-data independence*

◆ Data abstraction

- Hides the details of a data structure

- **Data model**

  - Provides only a *conceptual view* rather than a physical view

Characteristics of the Database

◆ Support of multiple views of the data

- **View**: *virtual data* derived from the (actual) database

- Different views of the same database for *different perspectives*

Characteristics of the Database

◆ Sharing of data

- Allows multiple users to access the database at the same time

- Provides concurrency control mechanisms

Example of Multiple Views

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

COURSE_PREREQUISITES

Course_name	Course_number	Prerequisites
Database	CS3380	CS3320
		MATH2410
Data Structure	CS3320	CS1310

Data
And
Knowledge
Engineering

◆ System analyst

- Analyzes requirements of end users

◆ Database designer

- Identifies the data to be stored and designs database structures

◆ Application programmer

- Implements the identified specifications as programs

Users of Database Systems

◆ Database administrator (DBA)

- Coordinates and monitors database system resources

◆ End users

- People whose jobs require access to the database via any database application programs

Data
And
Knowledge
Engineering

Classification of End Users

◆ **Casual end users**

- Occasionally accesses the database

◆ **Naïve or parametric users**

- Only uses standard types of queries pre-programmed

- Bank tellers, airline reservation clerks, …

Classification of End Users

◆ **Sophisticated end users**

- **F**acilitates DBMS and implements their applications to meet their complex requirements

- Engineers, scientists, business analysts, ...

◆ **Standalone users**

- Maintains personal databases by using ready-made program packages

Workers behind the Scene

◆ DBMS system designer and implementer

- Design and implement DBMS component modules and interfaces as a software package

◆ Tool developer

- Designs and implements software tools for convenient accesses to databases

◆ Operators and maintenance personnel

- ● Responsible for running and maintenance of hardware and software environment for a database system

◆ Controlling redundancy

- **Redundancy**

  - Identical data stored in multiple locations

- Problem

  - Wastes data storage (disk or SSD)

  - Causes the lack of *data consistency*

DBMS Features

◆ Restricting unauthorized access

- Security and authorization sub-components

- Privileged software

DBMS Features

◆ Persistent storage for program objects

- Make data persistent regardless of the program's termination

DBMS Features

◆ Multiple user interfaces

- Provide convenient database access depending on users' knowledge

- Query language, programming interface, GUI, …

DBMS Features

◆ **Representing complex relationships among data**

- Includes numerous varieties of data that are interrelated in many different ways

DBMS Features

◆ Enforcing integrity constraints

- **Integrity**: a property that the data satisfy its *constraints* given

- DBMS

  - Provides functions for defining integrity constraints

  - Provides functions for making defined integrity constraints satisfied

◆ Backup and recovery

- Makes a database recovered from hardware or software malfunctions

- **Backup**: Makes a copy of a database inside DBMS in normal processing for future recovery

- **Recovery**: Makes a database recovered to a latest state from backup data when malfunctions occur

Advantages of Using the DBMS

◆ Application development time reduced

- Not require programming for data management

◆ Availability of up-to-date information

- A user's database update is shown to every user promptly

◆ Economy of scale

- Maintaining an enterprise-wide database integrates administrative efforts of multiple departments

Data
And
Knowledge
Engineering

When Not to Use a DBMS

◆ Overheads of DBMS

- High initial cost (hardware, software, education, ...)

- Relatively high computational costs

  ▪ Security, concurrency control, integrity constraints, ...

When Not to Use a DBMS

◆ More desirable to use regular files

- ● Simple and well-defined applications not expected to change at all

- ● Very strict real-time requirements

- ● No multiple-user accesses to a database

Data
And
Knowledge
Engineering

◆ Database

- Collection of related data (recorded facts)

◆ DBMS

- Generalized software package for implementing and maintaining a computerized database

Summary

◆ Several categories of database users

◆ Database applications have evolved

- Current trends: IR, Web, data mining

1. http://liinwww.ira.uka.de/bibliography/Database.

2. Albiteboul, S., R. Hull, and V. Vianu, *Foundations of Databases*, Addison-Wesley, Reading, MA, 1995.

3. M. M. Astrahan et al., "System R: a relational approach to database management," *ACM Trans. on Database Systems* 1:2, pp. 97-137, 1976.

4. P. A. Bernstein et al., "The Asilomar report on database research," http://www.acm.org/sigmod/record/issues/9812/asilomar.html.

5. http://www.informatik.uni-trier.de/~ley/db/index.html. A mirror site is found at http://www.acm.org/sigmod/dblp/db/index.html.

References

6. Stonebraker, M. and J. M. Hellerstein (eds.), *Readings in Database Systems*, Morgan-Kaufmann, San Francisco, 1998.

7. M. Stonebraker, E. Wong, P. Kreps, and G. Held, "The design and implementation of INGRES," *ACM Trans. on Database Systems* 1:3, pp. 189-222, 1976.

8. Ullman, J. D., *Principles of Database and Knowledge-Base Systems*, *Volume I*, Computer Science Press, New York, 1988.

9. Ullman, J. D., *Principles of Database and Knowledge-Base Systems*, *Volume II*, Computer Science Press, New York, 1989.

Data
And
Knowledge
Engineering

Have a nice day!

Data
And
Knowledge
Engineering