Database Management Systems (DBMS)

Lec 1: Purpose and Evalution of DBMS

Ramesh K. Jallu

Overview

- Basic definitions

- Components in a database

- Goals of a database system

- File-processing system and it's disadvantages

- Advantages of using a DBMS

Motivation

- In our daily life, we come across various needs to store data
  - E.g., Contact numbers, electricity bills, salary details, reports, etc.

- How to store the data so that we can get it back when required?

- It should be stored in such a way that we should be able to
  - get the data at any point of time later
  - add details to it whenever required
  - modify stored information as needed
  - delete them

Basic Definitions

- **Data**: Facts that can be stored and have *implicit* meaning
  - E.g., Names, telephone numbers, and addresses, etc.

- **Database**: A well organized collection of interrelated data
  - Source of data: interactions with events in the real world
  - It is designed, built, and populated with data for a specific purpose

Basic Definitions (cont.)

DBMS: A collection of programs that work with the data to create and maintain a database

- A *general-purpose software* system that facilitates the process of ***defining, constructing, manipulating***, and **sharing** databases among various users and applications

Database **+** DBMS software **=** Database system

Basic Definitions (cont.)

- **Query**: A question or an inquiry – it causes some data to be retrieved from the database

- **Application program**: A computer program whose primary purpose is to access data by sending queries

- **Transaction**: Executing program or process that includes one or more database access.
  - It causes some data to be read and/or some data to be written/updated

Components of a database system

1. **Hardware**
   - A computer system used for storing and accessing the database
   - Secondary storage devices together with I/O devices

2. **Software**
   - All requests from users for access to the database are handled
   - Allows useres to communicate with the database

3. **Data**
   - The main component of any database system

Components of a database system (cont.)
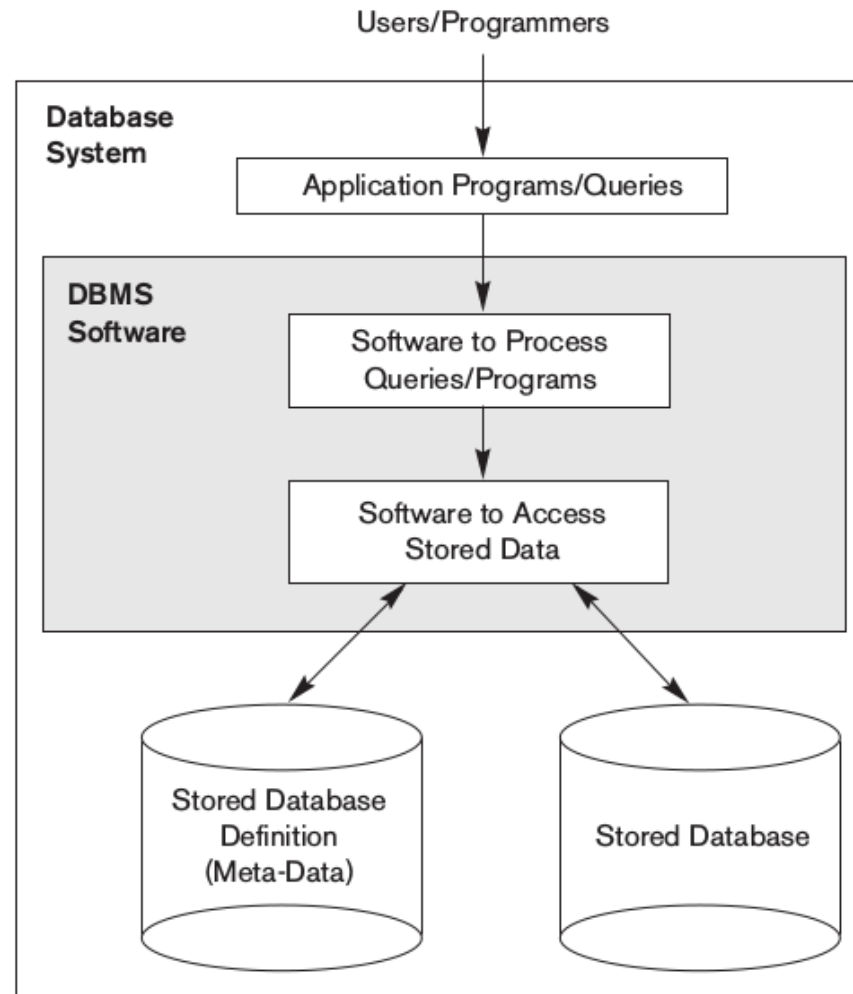
4. **Users**
   - Database administrator
   - Database designer
   - End users: (a) Casual end users, (b) Naïve/parametric end users, (c) sophisticated end users, and (d) standalone users
   - System analyst: determines the requirements and specifications
   - Application programmer

5. **Procedures**
   - A set of instructions and rules that govern the design and use of the database

Environment of a DBMS

Goals of database systems

- Database systems are designed to manage large data
  - Management of data: Defining structures + mechanisms for manipulation

- Must ensure the safety of the data stored despite of system crashes or attempts of an unauthorized access

- Must be able to maintain the database for many years and adopt the changes over time

- Provide a way to store and retrieve data that is both *convenient* and *efficient*

  Our goal: learn the concepts and techniques for managing data

Phases in a database design

1. Requirements collection and analysis
   - This phase involves assessing the informational needs of an organization

2. Conceptual design
   - A description of the structure of database, relationships among data, and rules

3. Logical design
   - Implementaion of the database design: creation

4. Physical design
   - Aims to maximize database efficiency by finding ways to speed up the performance
   - Application programs are designed and implemented as transactions

DBMS applications

- **University**: Students, Courses, Grades, Departments, and Sections, Faculty, etc.

- **Banking**: Customer details, Accounts, Loans, and Transactions, etc.

- **Enterprise**: Customers, Products, Purchases, Payments, Invoices, Salaries, etc.

- **Manufacturing**: Management of production, Inventory, Orders, Supply chain, etc.

- **Airlines**: Passenger details, Reservations, Flight schedules, etc.
- …

An example: University 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

RELATIONS

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

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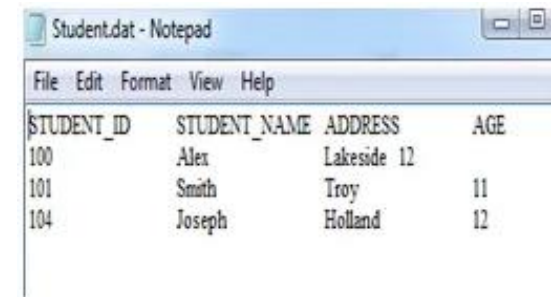
Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
.....	.....	......
.....	.....	......
.....	.....	......
Prerequisite_number	XXXXNNNN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

Evaluation of DBMS

- The ways in which people interact with databases has changed over time

- Early days users interacted via printed reports and paper forms for input

- In 1960's, database systems arose in response to the computerized management of commercial data, e.g., file-processing systems

- Modern database deals with data simple to complex consisting of text, images, videos, website links, etc., e.g., Social media
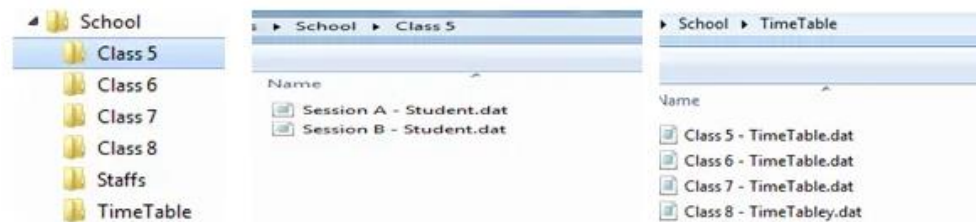
File-processing system

- A **file** is a resource which uniquely records data and is stored in a computer
  - Notepad, word document, etc.

- All the files were grouped based on their categories and each file is named properly

File-processing system (cont.)

- File-processing system is a way of storing, retrieving and manipulating data which is present in various files

- The system has a number of application programs that manipulate the files
  - Add new students, faculty, and courses
  - Assign grades to students, compute CGPA and generate transcripts
  - ...

- New application programs are added to the system as the need arises

Advantages of file-processing system

- Less expensive
  - Uses inbuilt tools and basic programming languages

- Easy to use and maintain
  - No need of high technical person to handle a file

- Fast processing

- Highly scalable
  - Ability to cope with up increased load

- No complex relations among the data

Disadvantages of file-processing systems

- Redundancy and inconsistency

- Difficulty in accessing data

- Data isolation

- Integrity problems

- Atomicity problems

- Congruent access anomalies

- Security problems

1. Redundancy and inconsistency

- Every user group maintains its own files for handling its data-processing

General Office	Library	Hostel	Account Office
Roll No	Roll No	Roll No	Roll No
Name	Name	Name	Name
Class	Class	Class	Class
Father_Name	Address	Father_Name	Address
Date_of_Birth	Date at Birth	Date of Birth	Phone No
Address	Phone No	Address	Fee
Phone No	No of books issued	Phone No	Installments
Previous Record	Fine	Mess bill	Discount
Attendance	etc	RoomNo	Balance
Marks		etc.	Total
etc.			etc.

- Redundancy leads to several problems

1. Redundancy and inconsistency (cont.)

- In the database approach, the data is integrated during database design

General Office	Library	Hostel	Account Office
Rollno Name Class Father_Name Address Phone - No Date_of_birth Previous_Record Attendance Marks etc.	Rollno No_of_books_issued Fine etc.	Rollno RoomNo Mess_Bill etc.	Rollno Fee Installments Discount Balance Total etc.

- Sometimes necessary to use **controlled redundancy**

2. Difficulty in accessing data

- Whenever data has to be retrieved as per the requirements then a new application program has to be written

General Office	Library	Hostel	Account Office
Roll No	Roll No	Roll No	Roll No
Name	Name	Name	Name
Class	Class	Class	Class
Father_Name	Address	Father_Name	Address
Date_of_Birth	Date at Birth	Date of Birth	Phone No
Address	Phone No	Address	Fee
Phone No	No of books issued	Phone No	Installments
Previous Record	Fine	Mess bill	Discount
Attendance	etc	RoomNo	Balance
Marks		etc.	Total
etc.			etc.

- Suppose we need to find the list of students with a particular postal address, and several days later, we need to find those students with grade '**A**' from the list

3. Data isolation

- Writing new application programs to retrieve the appropriate data is difficult as data is scattered in various files, and files may be in different formats

General Office	Library	Hostel	Account Office
Roll No	Roll No	Roll No	Roll No
Name	Name	Name	Name
Class	Class	Class	Class
Father_Name	Address	Father_Name	Address
Date_of_Birth	Date at Birth	Date of Birth	Phone No
Address	Phone No	Address	Fee
Phone No	No of books issued	Phone No	Installments
Previous Record	Fine	Mess bill	Discount
Attendance	etc	RoomNo	Balance
Marks		etc.	Total
etc.			etc.

- Generate a grade report of a student, who is studying in a particular class, his library details, and hostel information

4. Integrity problems

- Incorrect information cannot be stored in database
- In order to maintain the integrity of data, some integrity constraints are enforced on the database.

General Office	Library	Hostel	Account Office
Roll No Name Class Father_Name Date_of_Birth Address Phone No Previous Record Attendance Marks etc.	Roll No Name Class Address Date at Birth Phone No No of books issued Fine etc	Roll No Name Class Father_Name Date of Birth Address Phone No Mess bill RoomNo etc.	Roll No Name Class Address Phone No Fee Installments Discount Balance Total etc.

- This constraint must be enforced on all the applications separately

5. Atomicity problems

- Atomicity refers to the completion of the whole transaction or not completing it at all

- In case of failure, the data should be restored to the consistent state that existed prior to the failure
  - E.g., A bank transaction

- Partial completion of any transaction results in inconsistent database state

- If there is any failure to add, update, or delete in the file system, there is no mechanism to switch back to the previous state

6. Concurrent-access anomalies

- Mutiple users/application programs accessing the same data from the same file almost exactly the same time is called concurrent access
  - E.g., A bank transaction of an account

- When multiple transactions are trying to access the same sharable data, there could arise many problems if the access control is not done properly

- DBMS handles this using the protocols namely **Locks** and **Timestamps**

7. Security

- Database security is a mechanism that protects the database against intentional or accidental threats

- Not every user of the database system should be able to access all the data, but only a (small) part of the data
  - E.g., A customer has to be given access to view only his/her own bank account information.

- Each file can be password protected. But what if we have to give access to only a few records in the file?

- Since application programs are added to the system in ad hoc manner, it is difficult to enforce such security constraints.

Advantages of using the DBMS approach

- It solves the problems with file-processing systems

- Additional implications
  - Potential for enforcing standards

  - Reduced application development time

  - Flexibility

  - Availability of Up-to-Date information

  - Economies of Scale

When not to use a DBMS?

- In situations in which a DBMS may involve unnecessary overhead costs in hardware, software, security, recovery, and training, etc.

- Frequent change in software and hardware requirements

- If the database and applications are simple, well defined, and not expected to change over time

- If access to data by multiple users is not required

- We call it **special-purpose** DBMS

What we learned today?

1. Basic definitions, components in a database system and it's environment

2. Goals and characteristics of a database system

3. Phases in a database design

4. File-processing system, and it's advantages and disadvantages

5. Additional advantages of using the DBMS approach

Thank you!