

# CS3743 Introduction to Database Systems Overview of the Course

Weining Zhang

Department of Computer Science  
 University of Texas at San Antonio

August 15, 2009

## Contact Information

Office	SB 4.01.19
Phone	458-5557
Email	wzhang@cs.utsa.edu
Office Hour	MWF 10:00-11:00am
	or by appointment

## Outline

- 1 Logistics
  - Logistics
- 2 Syllabus
  - Syllabus
- 3 Basic Database Concepts
  - Database Systems
  - Data Models
  - Database Applications
  - Database Management Systems

## Resources

- Web page: <http://www.cs.utsa.edu/~wzhang/cs3743/home>
  - Contains everything about the course: syllabus, announcement, project, lecture notes, etc.
  - You should check the pages frequently.
- Mailing list: cs3743@cs.utsa.edu
  - You may want to set up email forwarding
- Computer account is available for registered students
- Oracle database client/server and web server are provided for course project



## Course Project

- Goal
  - The project is to develop a realistic database application and to gain experience in team work
- Topics
  - You choose your own application domain and design your own database
  - Be creative
- Progress
  - The project has 5 milestones, each has a due date and a separate grade

## Database System = Databases + DBMS

- A database is large, integrated collection of data that models (represents) a real-world enterprise.
  - Entities (e.g., students, courses)
  - Relationships (e.g., Mary takes CS1234)
  - Constraints (e.g., each course must have at least 10 students)
- A Database Management System (DBMS) is a software package designed to store and manage databases effectively and efficiently.
  - Support a data model (data structures and operations)
- A database application is a program that performs its functions by using an DBMS to manage data

## Project Team

- Each team consists of 3-4 members and elects a team leader
- Teams will be formed among yourselves. Statistically, teams formed early in the class performs better. So, act soon.
- Members must collaborate with each other and overcome any difference and difficulty
- If requested, different grades can be assigned to members according to individual contribution to the project. A detailed documentation of participation is required by the instructor

## Characteristics of Database Approach

- Self-describing of database system
  - Description of data structure and constraints are stored in the same system
- Insulation between program and data
  - System hides data structures and data operations from programs
  - Programs are independent of implementation of data structures and operations
- Multiple views of data
  - Different types of users can view data differently through view definitions
- Sharing data among multiple users and programs
  - Multiple users can query and update data concurrently

# Database Systems

## Advantage of Using DMBS

- Efficient data storage
- Abstract data model
- High-level database language
- Different views of the data
- Data integrity & security
- Support for database application
- Concurrent access by multiple users
- Crash recovery
- Data analysis, data mining, visualization,

# Data Models

## Schema vs Instances

- A database schema describes the structure and constraints of data
  - Static (seldom changes)
- A database instance is the actual data stored in the system at an instance of time.
  - Dynamic (frequently changes)

# Database Systems

## People Involved With Database Systems

- Database Administrators
- Database Designers
- End Users
- Application Programmers
- DBMS designers and implementers
- DB tool developers
- System operators

# Data Models

## Data Model

**Definition**  
 A data model is a set of concepts and a language used to describe the structure and operations of data.

In terms of databases, a data model typically describes

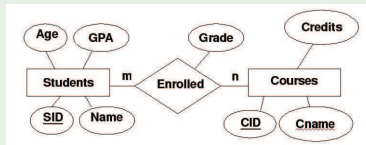
- Types of objects
- Relationships among objects
- Constraints on objects & relationships
- Operations on objects & relationships

# Conceptual Data Models

- Conceptual Models use concepts that are familiar to application domain experts
  - Entity/Relationship (E/R) Model uses entity type, relationship type, and constraints
  - Uniform Modeling Language (UML) uses graphical constructs to model data, process, use case, ...

## Example

In ER model, students, courses, and their relationships can be represented by an ER diagram



# Logic Data Models

- Logic Models use concepts that are implementation-oriented
  - Relational Model. Concepts are table, attribute, domain, relational algebra, calculus, ...
  - Object-Oriented Model. Concepts include class, object, inheritance, encapsulation, ...
  - Object-Relational Model.

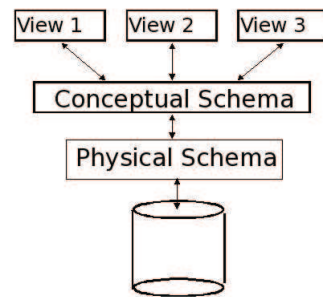
## Example

In relational model, students, courses, and their relationships can be represented by tables with schema

```
Students(sid: string, name: string, login: string, age: integer, gpa:real)
Courses(cid: string, cname:string, credits:integer)
Enrolled(sid:string, cid:string, grade:string)
```

# Three-Level Database Schema

- 1 Views describe how users see the data
- 2 Conceptual schema defines logical structure using a data model
- 3 Physical schema describes files and indexes used



# Data Independence

## Definition

The data independence is the ability of a DBMS such that changes to lower level schema are hidden from users of upper level schema.

## Example

- Logical data independence allows changes logical (conceptual) structure of data without changing views
- Physical data independence allows changes in physical structure of data without changing conceptual schema.

Data independence is one of the most important benefits of using a DBMS.

Data Models

# Database Languages

- Data Definition Language (DDL). Used to specify database schema
- Data Manipulation Language (DML). Used to insert, delete, change, and retrieve data
  - Declarative Language. Specify WHAT to retrieve. Ex: SQL, relational calculus
  - Procedural Language. Specify HOW to retrieve. Ex. relational algebra
- DBMS Interfaces can be interactive or programming
  - Web-based, menu-based, form-based, graphical, natural language based, or parametric
  - Application Programming Interface (API), e.g., Java/JDBC

Database Applications

# Database Applications

**Definition**  
 A database application is a software in an application domain that uses a DBMS to store and manipulate data

- Example**
- Retail stores, e.g., Amazon.com, HEB, etc.
  - Social networks, e.g., Facebook.com, YouTube.com, etc.
  - Educational system, e.g., UTSA ASAP
  - Banking systems
  - Library systems
  - many others

Database Management Systems

# A DBMS Architecture

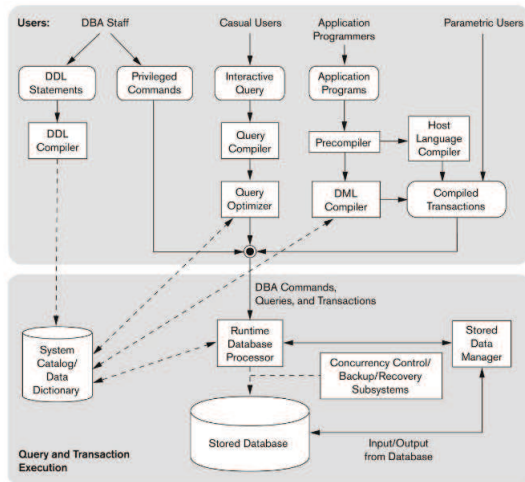


Figure 2.3 Component modules of a DBMS and their interactions.

Database Management Systems

# Classifications of DBMSs

DBMSs can be classified according to

- Data models.
  - Relational, Object, Relational-Object, Logic, Spatial, Temporal, Multimedia, Probabilistic, Fuzzy, Stream, Text, XML
- System architecture.
  - Centralized, Client/Server, Distributed, Federated, Parallel, Embedded, Real-time, Mobile, Web
- Applications supported:
  - Transactional, Data Warehousing, Data mining, Scientific,