Database Management Systems

Fall 2021

Database Management Systems

- **Instructor:** Dr. Choupani – Roya
- **Office:** 205G
- **Email:** rchoupani@angelo.edu
- **Office hours:** by appointment

Topics for Today

- **Get to know each other**
- **Information about the course, books, evaluation**
- **Course objectives**
- **What you will learn in this course**
- **Introduction to databases, and database management systems**

Course Objectives

- To provide the basis for data management and storage optimization techniques.
- To provide the basis for data handling and management, and the related algorithms and techniques.
- To make students able to apply data modeling techniques.
- To write queries to manipulate data

Data Management

- Handling large data requires using permanent storage such as hard disks. Data is stored in files on these storage types. However, these storage types are slow and we need to optimize file access.
- In addition using files we may face many problems such as
  - Redundancy problem
  - Inconsistency problem
  - Security problem
  - Concurrency problem

Drawbacks of using file systems to store data

- Data redundancy:
  - Assume a file is created to store the employee data of a company
  - Data is:

Employee Name	Department	Location	Department Phone	Project	Project Start Date	Duration
John	Engineering	Dallas	12345	Construction	2010	3
Hilton	Engineering	Dallas	12345	Construction	2010	3
David	Engineering	Dallas	12345	Construction	2010	3
Richard	Engineering	Dallas	12345	Construction	2010	3
Lisa	Engineering	Dallas	12345	Construction	2010	3

Drawbacks of using file systems to store data

- Data redundancy:

- Some data items are repeated. (if the department has 10,000 employees, **location** and **department phone** will be repeated 10,000 times)
- If a data item changes (phone number for example), all records should be updated
- Data access (input/output) becomes very slow

Drawbacks of using file systems to store data

- Data inconsistency:

- Now assume we have two employee files: One maintained by the Human Resource, and the second by the department at which the employee works.
- One employee changes his/her phone number. He/she informs the department but forgets to inform HR

Drawbacks of using file systems to store data

- **Data Integrity**

- Generally we want to impose some restrictions on data

- For example,
  - Age should be greater than or equal to zero and less than 200.
  - Account balance should be greater than or equa to zero

Drawbacks of using file systems to store data

- Atomicity of updates
  - Updating data should not be interrupted.
  - Failures may cause inconsistency with partial updates carried out
  - E.g. transfer of funds from one account to another should either complete or not happen at all

Drawbacks of using file systems to store data

- Concurrent access by multiple users
  - Concurrent accessed needed for performance
  - Uncontrolled concurrent accesses can lead to inconsistencies
    - E.g. two people reading a balance and updating it at the same time

Drawbacks of using file systems to store data

- Security problems

- The access to data should be restricted to authorized people only.
- In addition, each authorized user may have different access rights to data.
  - Ex. Students can view the course information but cannot modify it.

Performance Issue

- Many applications store a huge amount of data
- The data should be stored on non-volatile devices
- Access to data should be fast

Performance Issue

- Some applications that store data are:
  - Banking: all transactions
  - Airlines: reservations, schedules
  - Universities: registration, grades
  - Sales: customers, products, purchases
  - Manufacturing: production, inventory, orders, supply chain
  - Human resources: employee records, salaries, tax deductions

Requirements of a Solution

- Capable of:
  - Storing data on non-volatile storage
  - Handling large amount of data
  - Providing fast access to data
  - Avoiding redundancy
  - Preserving data integrity
  - Providing concurrent access to data
  - Providing data security
  - Performing atomic operations
  - More ?

Database

- A database is:
  - A collection of inter-related data and information
    - Example data:
      - List of students in a university
      - List of courses
      - List of instructors
    - Example information
      - Course CS123 is offered by instructor ABC
      - Student S234 is taking course CS123

Database Management System

- A database management system (DBMS) is a program that enables
  - storing,
  - modifying,
  - extracting

  data and information from a ***database***.
- It also provides tools to analyze the stored data
  - Example:
    - What was the average grade in cs123?
    - How many students passed CS123?
- Example DBMS: ORACLE, MS-SQL, MySQL,…

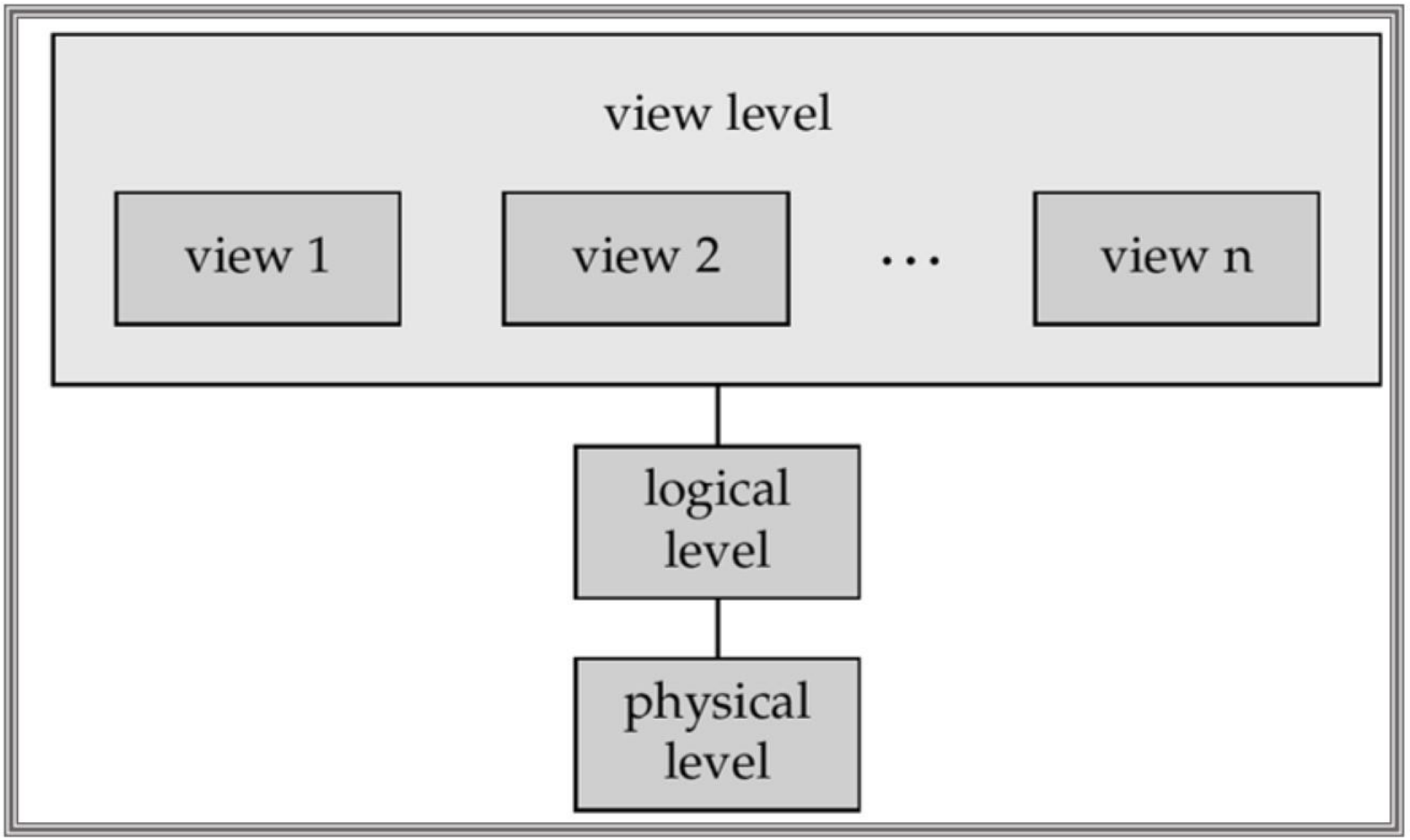
Abstraction

- A database is too complex to be handled easily
- There are lots of details to be considered.
- But each database user needs only part of these details at any given time.
- Therefore, we can have a simplified model by hiding some of those details
- **Abstraction** is **hiding unnecessary details** through **modeling**

Levels of Abstraction

- Physical level: describes how a record (e.g., customer) is stored.
- Logical level: describes data stored in database, and the relationships among the data.
  - e.g. Student details, and the courses he/she takes (the relationship between student and course)
- View level: application programs hide details of data types. Views can also hide information (e.g., salary) for security purposes.

Abstraction Levels in Databases



Data Modeling

- A **data model** is an abstract representation of data that
  - Organizes data elements
  - Defines standards
  - Show how data items are related to one another

- Example:
  - A university data model can group data items, and show their relationships.

Example Data Model

Student Model				
Student ID	Student First Name	Student Last Name	Date of Birth	Department

Department Model		
Department Name	Location	Phone

Course Model			
Course code	Course Name	Credits	Department

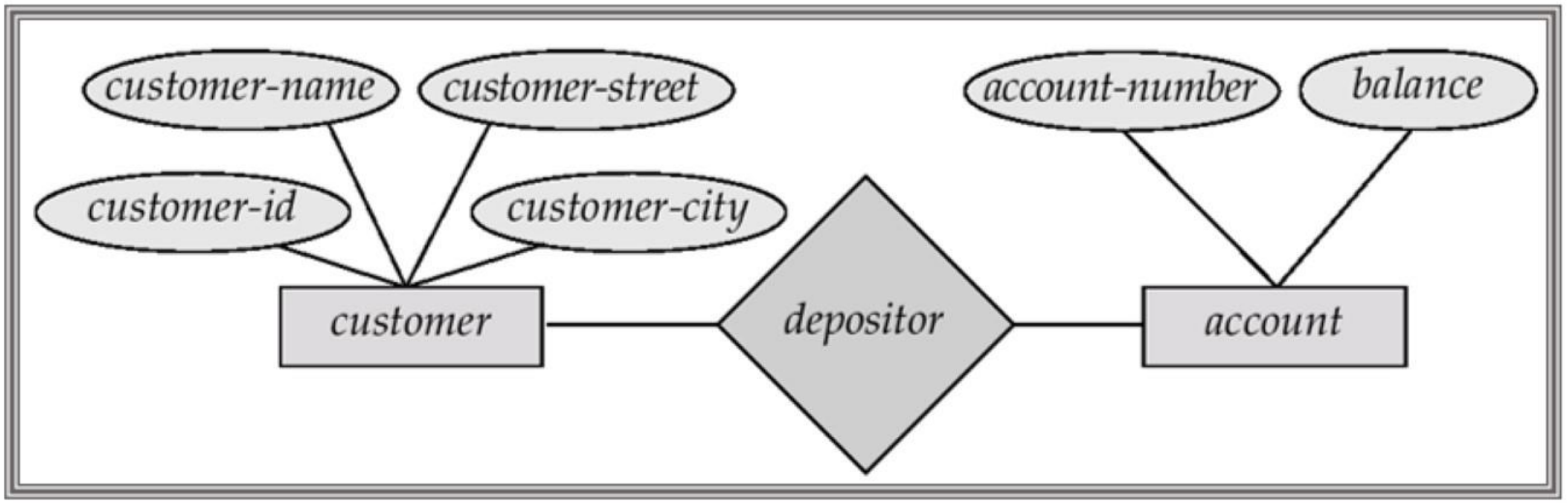
Courses Taken by Students				
Student ID	Course Code	Year	Semester	Grade

Data Models

- Most commonly used data models are:
  - Entity-Relationship model
  - Relational model
  - Object-oriented model
  - Semi-structured data models
  - Network model
  - Hierarchical model

Entity-Relationship Model

Example of an entity-relationship model



Defining Data Models

- To define and create data models special languages are used.
- These languages are referred to as Data Definition Languages (DDL)
- Data definition language depends on the model type that we want to use.

Data Definition Language (DDL)

- Specifies the notation used for defining the data model
  - E.g.

```
create table account (
        account-number    char(10),
        balance           integer)
```

- DDL compiler generates a set of tables stored in a *data dictionary*

- Data dictionary contains metadata (i.e., data about data)

Data Manipulation Language (DML)

- Language for accessing and manipulating the data organized by the appropriate data model
  - DML also known as query language
- Two classes of languages
  - Procedural – user specifies what data is required and how to get those data
  - Nonprocedural – user specifies what data is required without specifying how to get those data
- SQL is the most widely used query language

Structures Query Language (SQL)

- The SQL language was first developed in the 1970s by IBM researchers Raymond Boyce and Donald Chamberlin.
- The language was based on the paper by Edgar Frank Todd "A Relational Model of Data for Large Shared Data Banks," in 1970.
- SQL can be used by
  - Language extensions to allow embedded SQL
  - Application program interface (e.g. ODBC/JDBC) which allow SQL queries to be sent to a database

Structures Query Language (SQL)

- SQL: widely used non-procedural language
  - E.g. find the name of the customer with customer-id 192-83-7465

    select *customer.customer-name*
    from *customer*
    where *customer.customer-id* = '192-83-7465'
  - E.g. find the balances of all accounts held by the customer with customer-id 192-83-7465

    select *account.balance*
    from *depositor*, *account*
    where *depositor.customer-id* = '192-83-7465' **and**
    depositor.account-number = *account.account-number*

Summary

- Working with large bulk of data requires considerations to avoid problems such as redundancy, inconsistency, and so on.
- Besides, we want to have efficiency in accessing and using data
- Database management systems are used to avoid problems in storing and accessing data, and to hide unnecessary details

Information about the course

What is included?

- We will discuss file management systems.
- We are interested in file management methods for improving data access.
- The methods we will discuss are:
  - Sorting and searching in files
  - Indexing
  - Hashing
- In addition in this course we will learn how to manage and model data

Course Outline

- Introduction
  - Files, optimizing file access, indexing, hashing
  - Data, Database, Database Management Systems
  - Data Models
  - Database Languages
- Relational Databases
  - Relations and Tables
  - Attributes
  - Integrity Constraints
- Relational Algebra
- SQL
  - Queries
  - Data Manipulations
- Database Design
  - Entity Relationship Model (ER Model)
  - Normalization
- Transactions

Reference Books

- Database Systems, Concepts, Languages and Architectures, By: Paolo Atzeni, Stefano Ceri, Stefano Paraboschi, and Riccardo Torlone, Published by McGraw-Hill, ISBN:0077095006
- Fundamentals of Database Systems, Fourth Edition, Elmasri and Navathe, Addison Wesley Press, ISBN:0-312-20448-4
- Database system concepts / Abraham Silberschatz. — 6th ed. ISBN 978-0-07-352332-3
- Note: The course webpage will include updated slides

Evaluation

- Homework
  - Assignments, you may use any DBMS 20%
- Term Project 30%
  - ✓ Includes designing a database, creating tables, inserting data, writing queries.
  - ✓ Report
  - ✓ Presentation + Demo
- Midterm 20%
- Final 30%