INTRODUCTION

Introduction to Database Systems

Mohammad Tanhaei
Ilam University

IN THIS LECTURE

- Course Information
- Databases and Database Systems
- Some History
- The Relational Model
- For more information
  - Connolly and Begg – Chapters 1 and 2
  - Ullman and Widom (2ed.) – Chapter 1

COURSE INFORMATION

- Contact details
  - Mohammad Tanhaei
  - m.tanhaei@ilam.ac.ir

TEXTBOOK

- Recommended textbooks:
  - 'Database Systems: A practical approach to design, implementation and management' by Connolly and Begg
  - `A first course in database systems' by Ullman and Widom.
- Other textbooks:
  - There are lots of database texts
  - Most of them would be fine also
- For example:
  - 'Database Systems' by CJ Date

COURSE OVERVIEW

- Several main topics
  - Database systems
  - Data models
  - Database design
  - SQL
  - Transactions
  - Concurrency
  - Administration
- Practical sessions
  - SQL
    - creating a database
    - querying a database

WHY STUDY DATABASES?

- Databases are useful
  - Many computing applications deal with large amounts of information
  - Database systems give a set of tools for storing, searching and managing this information
- Databases in CS
  - Databases are a ‘core topic’ in computer science
  - Basic concepts and skills with database systems are part of the skill set you will be assumed to have as a CS graduate

WHAT IS A DATABASE?

- "A set of information held in a computer"

  Oxford English Dictionary

- "One or more large structured sets of persistent data, usually associated with software to update and query the data"

  Free On-Line Dictionary of Computing

- "A collection of data arranged for ease and speed of search and retrieval"

  Dictionary.com

DATABASES

- Web indexes
- Library catalogues
- Medical records
- Bank accounts
- Stock control
- Personnel systems
- Product catalogues
- Telephone directories
- Train timetables
- Airline bookings
- Credit card details
- Student records
- Customer histories
- Stock market prices
- Discussion boards
- and so on…

DATABASE SYSTEMS

- A database system consists of
  - Data (the database)
  - Software
  - Hardware
  - Users
- We focus mainly on the software

- Database systems allow users to
  - Store
  - Update
  - Retrieve
  - Organize
  - Protect
- their data.

DATABASE USERS

- **End users**
  - Use the database system to achieve some goal
- **Application developers**
  - Write software to allow end users to interface with the database system
- **Database Administrator (DBA)**
  - Designs & manages the database system
- **Database systems programmer**
  - Writes the database software itself

DATABASE MANAGEMENT SYSTEMS

- A database is a collection of information

- A database management system (DBMS) is the software than controls that information

- Examples:
  - Oracle
  - DB2 (IBM)
  - MS SQL Server
  - MS Access
  - Ingres
  - PostgreSQL
  - MySQL

WHAT THE DBMS DOES

- Provides users with
  - Data definition language (**DDL**)
  - Data manipulation language (**DML**)
  - Data control language (**DCL**)
- Often these are all the same language

- DBMS provides
  - Persistence
  - Concurrency
  - Integrity
  - Security
  - Data independence
- Data Dictionary
  - Describes the database itself

DATA DICTIONARY - METADATA

- The dictionary or catalog stores information about the database itself
- This is **data about data** or ‘metadata’
- Almost **every aspect** of the DBMS uses the dictionary
- The dictionary holds
  - **Descriptions** of database objects (tables, users, rules, views, indexes,…)
  - **Information** about who is using which data (locks)
  - **Schemas** and **mappings**

FILE BASED SYSTEMS

- File based systems
  - Data is stored in **files**
  - Each file has a **specific format**
  - Programs that use these files depend on **knowledge** about that format
- Problems:
  - No standards
  - Data duplication
  - Data dependence
  - No way to generate ad hoc queries
  - No provision for security, recovery, concurrency, etc.

RELATIONAL SYSTEMS

- Problems with early databases
  - **Navigating** the records requires complex programs
  - There is **minimal data independence**
  - **No theoretical** foundations
- Then, in 1970, E. F. Codd wrote “A Relational Model of Data for Large Shared Databanks” and introduced the relational model

RELATIONAL SYSTEMS

- Information is stored as **tuples** or **records** in **relations** or **tables**
- There is a **sound** **mathematical theory** of relations
- Most modern DBMS are based on the relational model
- The relational model covers 3 areas:
  - **Data structure**
  - **Data integrity**
  - **Data manipulation**
- More details in the next lecture…

ANSI/SPARC ARCHITECTURE

- ANSI - American National Standards Institute
- SPARC - Standards Planning and Requirements Committee
- 1975 - Proposed a framework for DBs
- A **three-level** architecture
  - **Internal level:** For systems designers
  - **Conceptual level:** For database designers and administrators
  - **External level:** For database users

INTERNAL LEVEL

- Deals with physical storage of data
  - **Structure** of records on disk - files, pages, blocks
  - **Indexes** and **ordering** of records
  - Used by database system **programmers**
- Internal Schema

```
RECORD EMP                LENGTH=44
HEADER: BYTE(5)           OFFSET=0
NAME: BYTE(25)            OFFSET=5
SALARY: FULLWORD          OFFSET=30
DEPT: BYTE(10)            OFFSET=34
```

CONCEPTUAL LEVEL

- Deals with the organization of the data as a whole
  - **Abstractions** are used to remove unnecessary details of the internal level
- Used by **DBAs** and application programmers

- Conceptual Schema

```
CREATE TABLE
Employee (
 Name        VARCHAR(25),
 Salary      REAL,
 Dept_Name   VARCHAR(10)
)
```

EXTERNAL LEVEL

- Provides a view of the database tailored to a user
  - Parts of the data may be **hidden**
  - Data is presented in a **useful** form
- Used by **end users** and application programmers

- External Schemas

```
Payroll:
  String Name
  double Salary

Personnel:
  char *Name
  char *Department
```

MAPPINGS

- Mappings **translate** information from one level to the next
  - External/Conceptual
  - Conceptual/Internal
- These mappings provide data **independence**
- Physical data independence
  - Changes to internal level shouldn't affect conceptual level
- Logical data independence
  - Conceptual level changes shouldn't affect external levels

ANSI/SPARC ARCHITECTURE

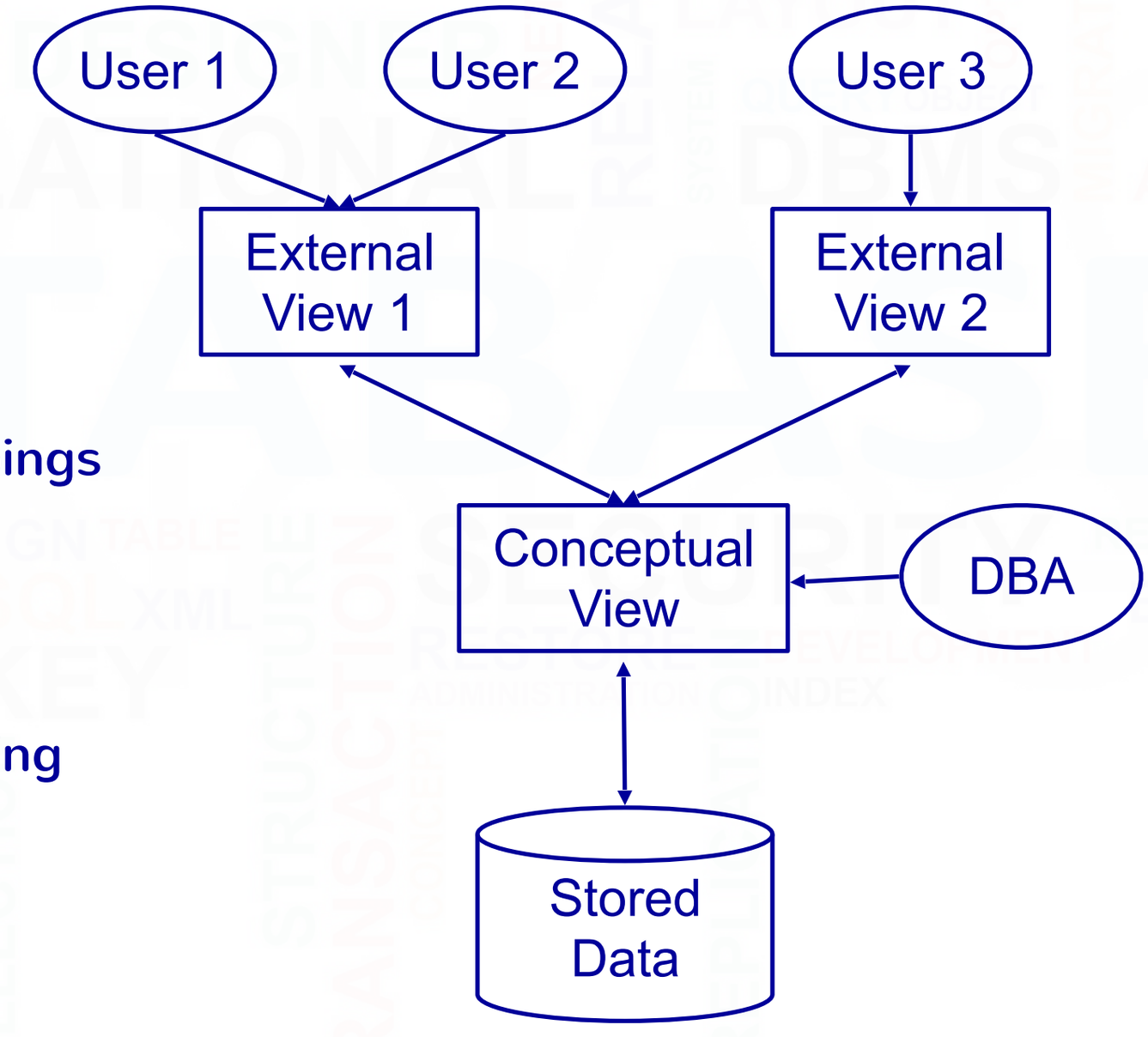
External Schemas

External/Conceptual Mappings

Conceptual Schema

Conceptual/Internal Mapping

Internal Schema



THIS LECTURE IN EXAMS

- Describe the three levels of the ANSI/SPARC model. You should include information about what each level is for, which users might be interested in which levels, and how the levels relate to one another.

END

But take a look at next slide!

NEXT LECTURE

The Relational Model

- Relational data structure
- Relational data integrity
- Relational data manipulation

For more information

- Connolly and Begg chapters 3 and 4
- Ullman and Widom (2 ed.) Chapter 3.1, 5.1
- E.F. Codd's paper