Lecture 2

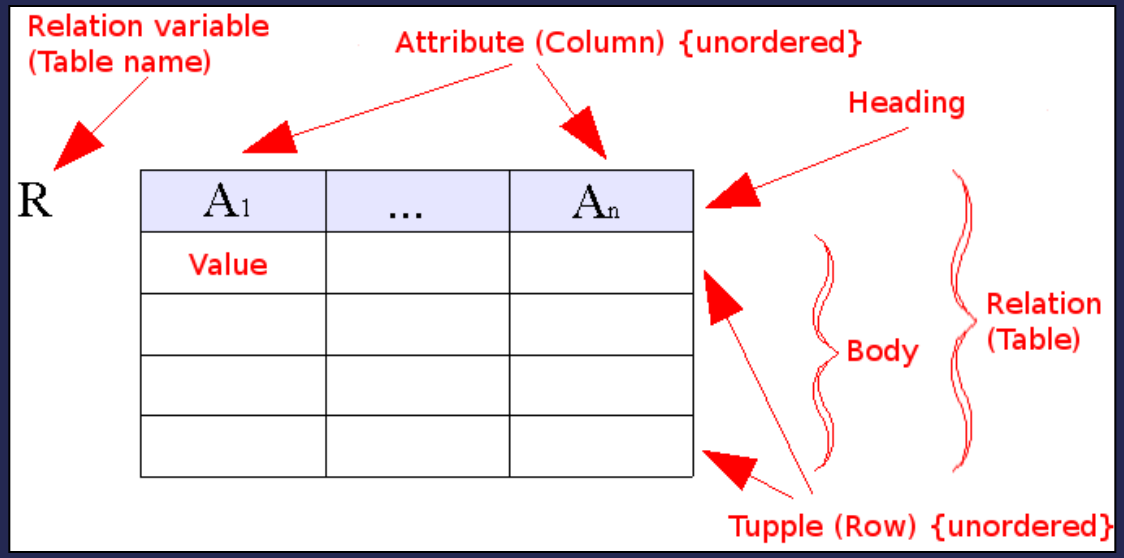
DATABASE MANAGEMENT SYSTEM

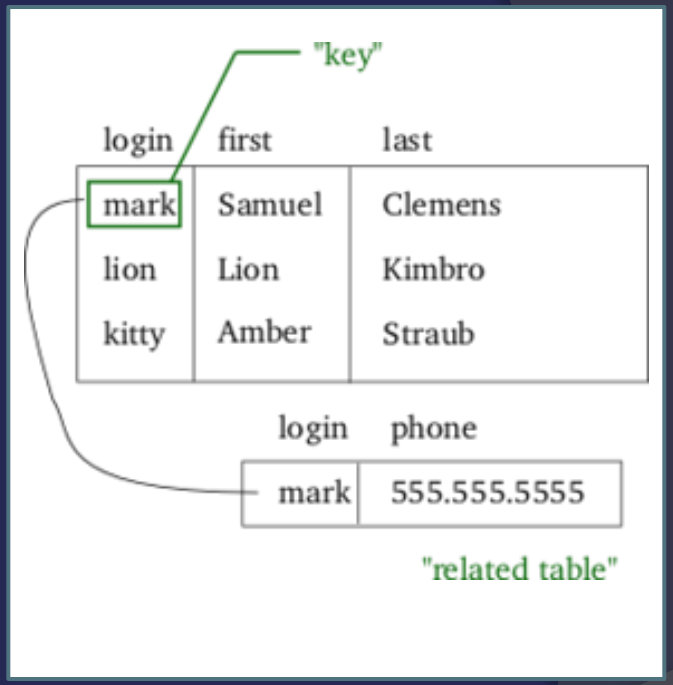
Data Models

- A **collection of concepts** that can be used to describe the structure of a database.
- Provides the **means to achieve data abstraction** by hiding details of data storage.
- Include a set of **basic operations** for specifying retrievals and updates on the database.

- Examples: Hierarchical model, Network model, R**elational model**, Object-oriented model

Relational model (1/2)





Relational model (2/2)

- A table is a collection of records and each record in a table contains the same fields.

- **Properties** of Relational Tables:
  - Values Are Atomic
  - Each Row is Unique
  - Column Values Are of the Same Kind
  - The Sequence of Columns is Insignificant
  - The Sequence of Rows is Insignificant
  - Each Column Has a Unique Name

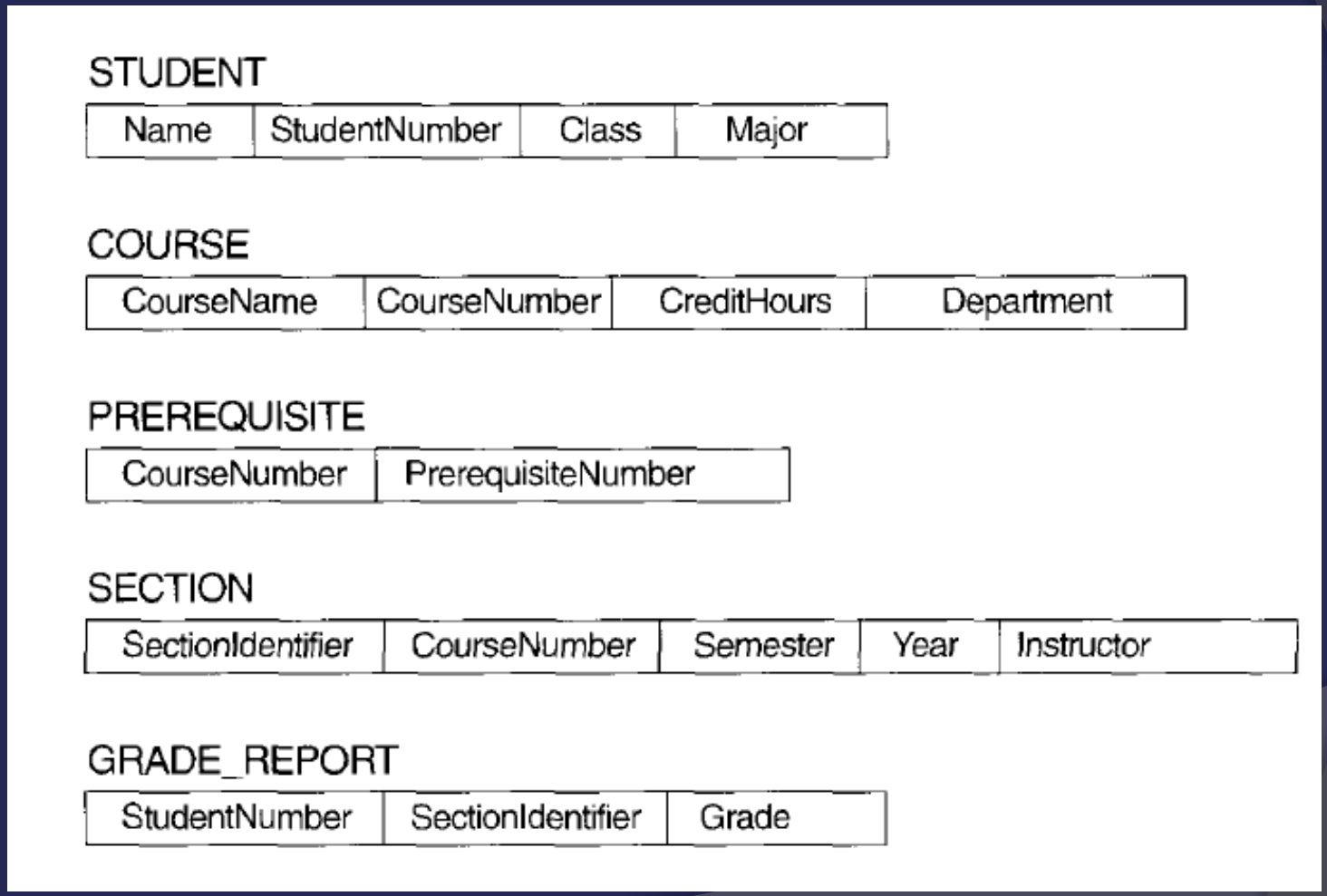
Levels of Data Models

- **Conceptual** data models: Provide concepts that are close to the way many users perceive data.
- **Physical** data models: Provide concepts that describe details of how data is stored in the computer.
- **Implementation** (**representational**) data models: Provide concepts that fall between the above two, balancing user views with some computer storage details.

Terms to know

- **Database Schema**:
  - The **description** of a database.
  - Includes descriptions of the database structure and the **constraints** that should hold on the database.
- **Schema Diagram:**
  - A **diagrammatic** display of (some aspects of) a database schema**.**
- **Database Instance / Database State**
  - The **actual data stored** in a database at a particular moment in time.

Schema diagram



Distinction

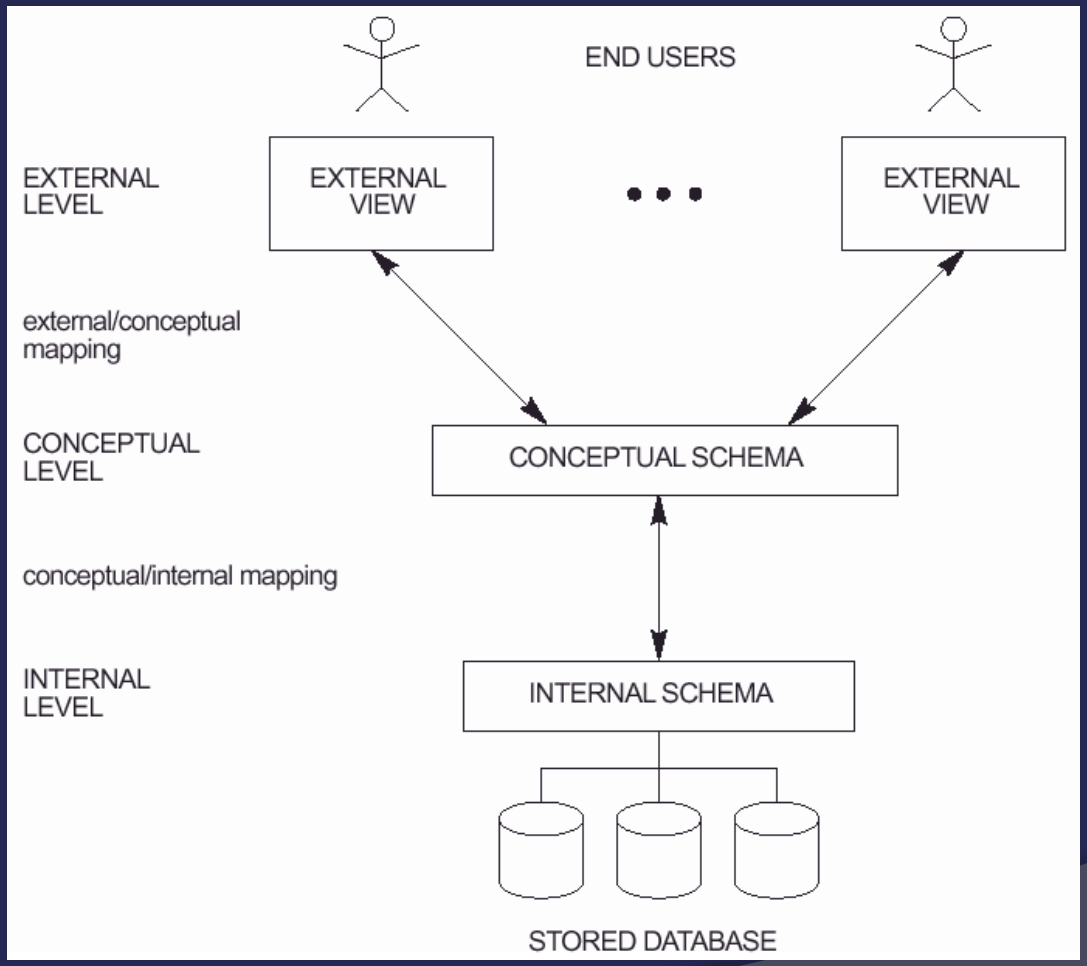
Database schema

- Changes *very* ***infrequently***.
- **Schema** is also called **intension**

Database state

- Changes ***every time the database is updated***.
- State is also called **extension**

Three-Schema Architecture (1/4)



Three-Schema Architecture (2/4)

- Defines DBMS schemas at *three levels*:

  - **Internal schema**
    - **Internal level**
    - Describe **physical storage structures and access paths.**
    - Uses a *physical* data model.

Three-Schema Architecture (3/4)

- **Conceptual schema**
  - **Conceptual level**
  - Describe the **structure and constraints** for the ***whole*** **database** for a community of users.
  - Uses a ***conceptual*** or an ***implementation*** **data model.**

- **External schemas**
  - **External level**
  - Describe the **various user views**.
  - Uses the same data model as the conceptual level.

Three-Schema Architecture (4/4)

- **Mappings**
  - Needed to transform requests and data.
  - Programs refer to an external schema, and are mapped by the DBMS to the internal schema for execution.

Data Independence (1/5)

- **Logical Data Independence**:
  - The capacity to **change the conceptual schema without having to change the external schemas** and their application programs.
  - Logical data independence refers to the **immunity** of application programs to the logical structure of database.

Data Independence (2/5)

- Example : If we add more attributes to a relation in database, the data access methods used by application should not be affected
- Addition or removal of new entities, attributes, or relationships to the conceptual schema should be possible without having to change existing external schemas or having to rewrite existing application programs

Data Independence (3/5)

- **Physical Data Independence**:
  - The capacity to **change the internal schema without having to change the conceptual schema.**
  - Application programs should not be concerned with the physical data organization.
  - Therefore the application programs should not need to be modified when physical data organization is changed

Data Independence (4/5)

- Example: Change to the internal schema, such as using different file organization or storage structures, storage devices, or indexing strategy, should be possible without having to change the conceptual or external schemas.

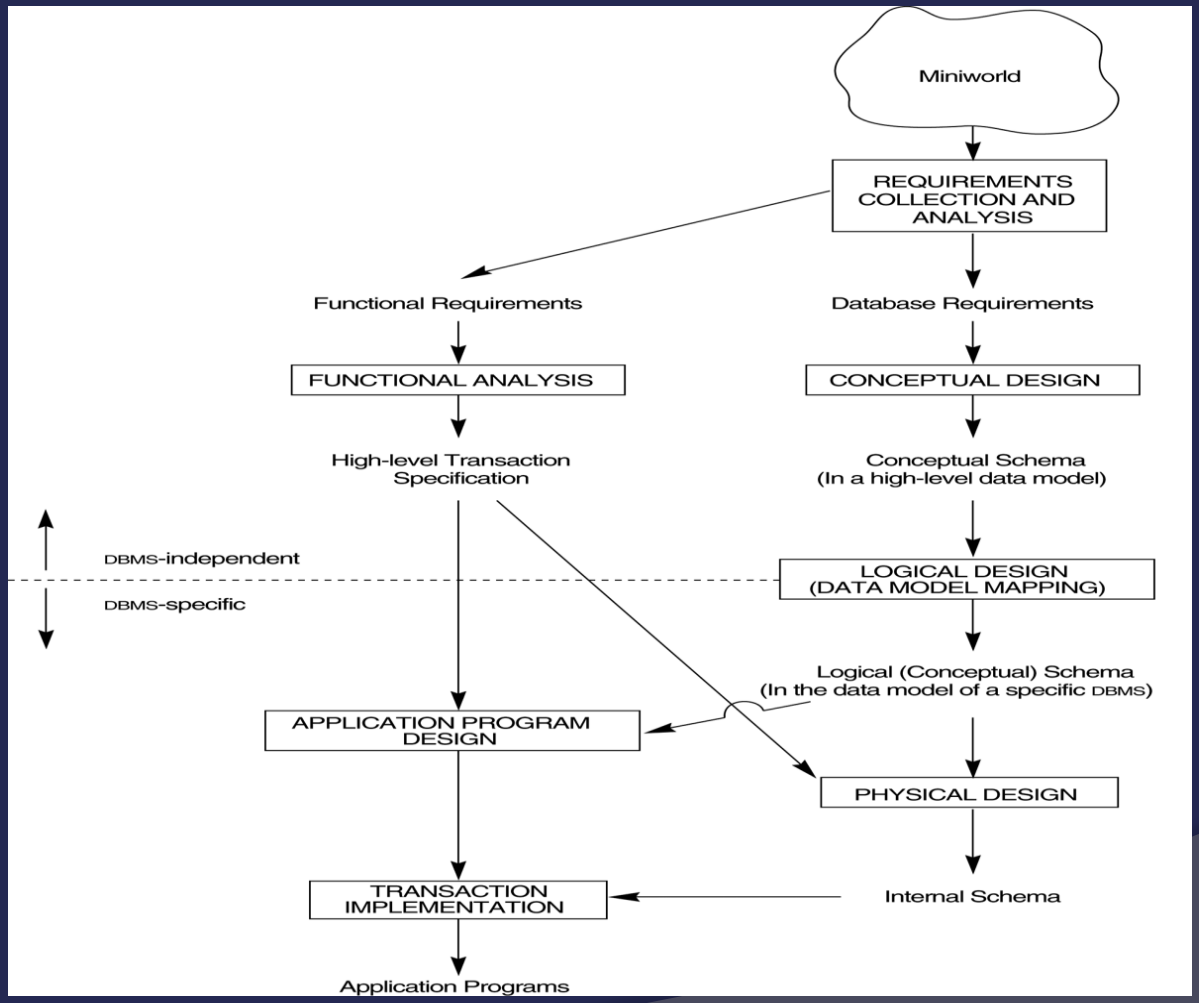
Data Independence (5/5)

- When a schema at a lower level is changed, only the **mappings** between this schema and higher-level schemas need to be changed in a DBMS that fully supports data independence.
- The higher-level schemas themselves are ***unchanged***.
- Hence, the application programs need not be changed since they refer to the external schemas.

Database Design Process (1/8)

- Two main activities:
  - **Database design :** To **design the conceptual schema** for a database application
  - **Applications design :** Focuses on the **programs and interfaces that access the database**

Database Design Process (2/8)



Database Design Process (3/8)

- **Requirements collection and analysis**
  - Database designers interview prospective database users to understand and document their data requirements.
  - Result of this step is a concisely written set of users' requirements.
  - Requirements should be specified in as detailed and complete a form as possible.

Database Design Process (4/8)

- **Functional requirements**
  - Consist of the user-defined operations that will be applied to the database, including both retrievals and updates.
  - In software design, it is common to use *data flow diagrams, sequence diagrams, scenarios,* and other techniques for specifying functional requirements

Database Design Process (5/8)

- **Conceptual schema**
  - Use a high-level conceptual data model.
  - This step is called conceptual design.
  - The conceptual schema is a concise description of the data requirements of the users and includes detailed descriptions of the entity types, relationships, and constraints
  - These are expressed using the concepts provided by the high-level data model.

Database Design Process (6/8)

- During or after the conceptual schema design, the basic data model operations can be used to specify the high-level user operations identified during functional analysis.

- Modifications to the conceptual schema can be introduced if some functional requirements cannot be specified using the initial schema

Database Design Process (7/8)

- Most current commercial DBMS use an implementation data model such as the relational model so the conceptual schema is transformed from the high-level data model into the implementation data model

- This step is called logical design or data model mapping, and its result is a database schema in the implementation data model of the DBMS

Database Design Process (8/8)

- **Physical design phase**
  - The internal storage structures, indexes, access paths, and file organizations for the database files are specified.

- In parallel with these activities, application programs are designed and implemented as database transactions corresponding to the high-level transaction specifications