BASIC CONCEPTS

Chapter 2:
Database Systems
Architecture

Outline of Chapter 2

1 Data Models

2 Schemas and Instances

3 The Three Level Architecture

4 Data Independence

5 DBMS Languages

6 DBMS Interfaces

7 DB System Utilities

8 DBMS Classification

9 Overview of the data models

1. Data Models

- **A data model**: A set of concepts to describe the *structure* of a database.

- The **structure** of a Database refers to the *data types*, *relationships*, and *constraints* that the database should obey.

- A Data Model can also contain **user-defined operations** that specify database retrievals and updates by referring to the concepts of the data model.

1. Data Models (Cont.)

Categories of data models:

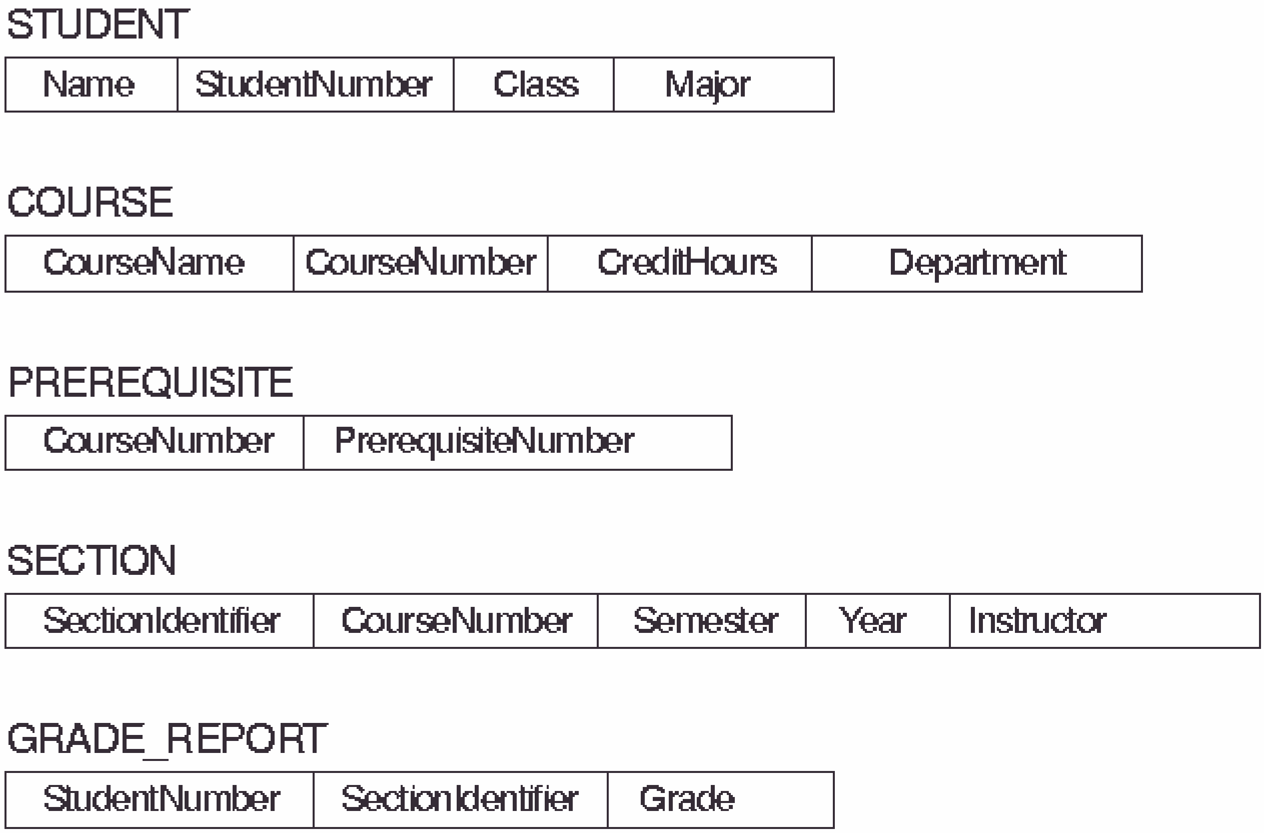
- **Conceptual (high-level, semantic)** data models
  (Also called **entity-based** or **object-based** data models).
- **Physical (low-level, internal)** data models.
- **Implementation (record-oriented)** data model.

Object data models can be seen as *higher-level implementation data models* that are closer to conceptual data models

2. Schemas and Instances

- **Database Schema**
- **Schema Diagram**

Figure 2.1 Schema diagram for the database of Figure 1.2.



STUDENT

Name	StudentNumber	Class	Major

COURSE

CourseName	CourseNumber	CreditHours	Department

PREREQUISITE

CourseNumber	PrerequisiteNumber

SECTION

SectionIdentifier	CourseNumber	Semester	Year	Instructor

GRADE_REPORT

StudentNumber	SectionIdentifier	Grade

2. Schemas and Instances (Cont.)

- **Database Instance**: The actual data stored in a database at a *particular moment in time* . Also called **database state** (or **occurrence**).

Schema is also called **intension**, whereas **state** is called **extension**.

3. The three level Architecture

The three level architecture is proposed to support the DBMS characteristics of:

- **Program-data independence**.
- **Multiple views** of the data.

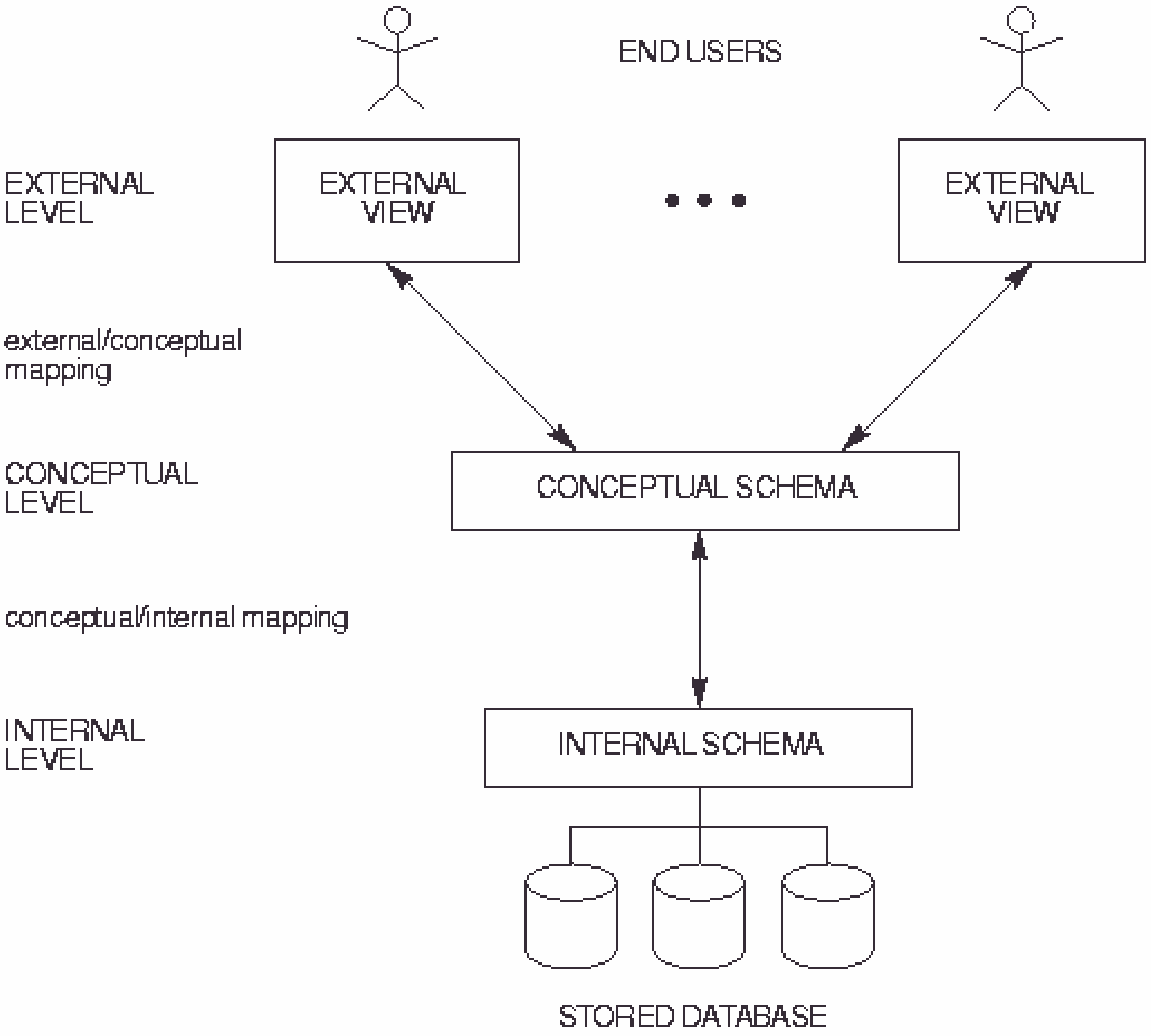
It defines DBMS schemas at *three levels* :

- **Internal schema** (Typically uses a *physical data model*.)
- **Conceptual schema** (Uses a *conceptual* or *an implementation data model*.)
- **External schemas** (Usually uses *the same data model as the conceptual level.*)

3.The three level Architecture (Cont)

Mappings among schema levels are also needed. Programs refer to an external schema, and are mapped by the DBMS to the internal schema for execution.

Figure 2.2 Illustrating the three-schema architecture.



4. Data Independence

Logical Data Independence

Physical Data Independence

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.

5. DBMS Languages

Data Definition Language (DDL)

-Storage definition language (SDL)

-View definition language (VDL)

Data Manipulation Language (DML)

In current DBMS a *comprehensive integrated language* is used that includes constructs for *conceptual schema definition*, *view definition* and *schema and data manipulation.*

5. DBMS Languages (Cont.)

DML commands can be *embed-ded* in a general-purpose programming language (**host language**).

Alternatively, *stand-alone* DML commands can be applied directly (**query language**).

A DML that can be used on its own to specify complex DB operations in a concise manner is called **high-level DML** (as opposed to a **low-level DML**.)

A high-level DML can specify and retrieve many records in a single DML statement (**set-at-a-time DML**). A low-level DML retrieves and process each time one record from a set of records (**record-at-a-time DML**).

A query in a high-level DML often specifies which data to retrieve (**declarative language**) rather than how to retrieve it (**procedural language**).

6. DBMS Interfaces

-Stand-alone query language interfaces.

-Programmer interfaces for embedding DML in programming languages:

- Pre-compiler Approach
- Procedure (Subroutine) Call Approach

-User-friendly interfaces:

- Menu-based
- Graphics-based (Point and Click, Drag and Drop etc.)
- Forms-based
- Natural language
- Combinations of the above
- Speech as Input (?) and Output
- Web Browser as an interface

-Parametric interfaces using function keys.

-Report generation languages.

-Interfaces for the DBA:

- Creating accounts, granting authorizations
- Setting system parameters
- Changing schemas or access paths

7. DBMS Utilities

Most DBMS have utilities that help the DBA to perform certain functions such as:

-*Loading* data stored in files into a database.

-*Backing up* the database periodically on tape.

-*Reorganizing* database file structures.

-*Generating reports.*

-*Monitoring performance.*

-Other functions, such as *sorting*, *user monitoring*, *data compression* , etc.

8. DBMS Classification

DBMS can be classified according to various criteria.

- **Based on the data model used**:
  -*Traditional*: Relational, Network, Hierarchical.
  -*Emerging*: Object-oriented, Object-relational.

- **Based on the number of users**:
  -*Single-user* (typically used with micro-computers)
  -*multi-user* (most DBMSs).

- **Based on the cost**:
  DBMSs usually range from free to a few thousand dollars.

- **Based on the generality**:
  -*Special purpose DBMS*
  -*General purpose DBMS*.

8. DBMS Classification (Cont.)

- **Based on the number of sites**:
  - *Centralized*
  - *Distributed*

Distributed Database Systems have now come to be known as **client server based database systems**.

The DBMS of the different DBs in a Distributed DBMS can be the same (**Homogeneous DBMSs**) or different (**Heterogeneous DBMSs**).

Further Reading

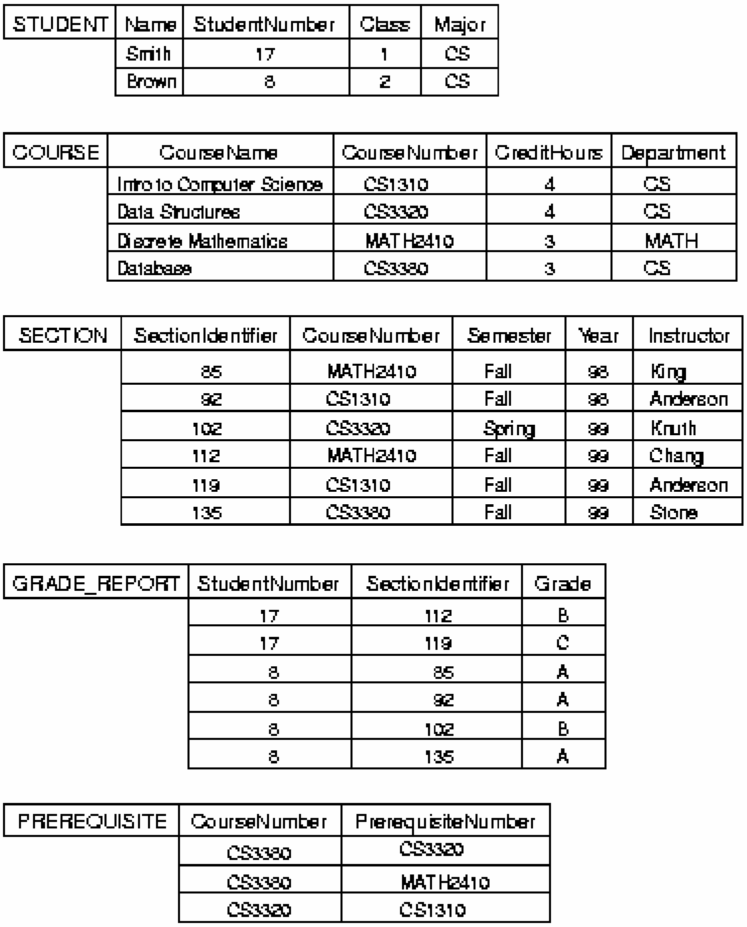
9. Overview of the data models (1)

The **Relational data model** is based on the notion of Relation.

It represents a database as a collection of tables.

Most relational databases use a high-level query language called SQL and support user views.

Example:



STUDENT	Name	StudentNumber	Class	Major
	Smith	17	1	CS
	Brown	8	2	CS

COURSE	CourseName	CourseNumber	CreditHours	Department
	Intro to Computer Science	CS1310	4	CS
	Data Structures	CS3320	4	CS
	Discrete Mathematics	MATH2410	3	MATH
	Database	CS3380	3	CS

SECTION	SectionIdentifier	CourseNumber	Semester	Year	Instructor
	85	MATH2410	Fall	98	King
	92	CS1310	Fall	98	Anderson
	102	CS3320	Spring	99	Knuth
	112	MATH2410	Fall	99	Chang
	119	CS1310	Fall	99	Anderson
	135	CS3380	Fall	99	Stone

GRADE_REPORT	StudentNumber	SectionIdentifier	Grade
	17	112	B
	17	119	C
	8	85	A
	8	92	A
	8	102	B
	8	135	A

PREREQUISITE	CourseNumber	PrerequisiteNumber
	CS3380	CS3320
	CS3380	MATH2410
	CS3320	CS1310

9. Overview of the data models (2)

The **Object data model** defines a database in terms of objects, their properties, and their operations.

Objects belong to classes; classes are organized into hierarchies; the operations of each class are specified in terms of predefined procedures called methods.

Many Object DBMS use a high-level query language called OQL.

The model of relational DBMS has been extended to incorporate object database concepts and other capabilities (Object-Relational systems).

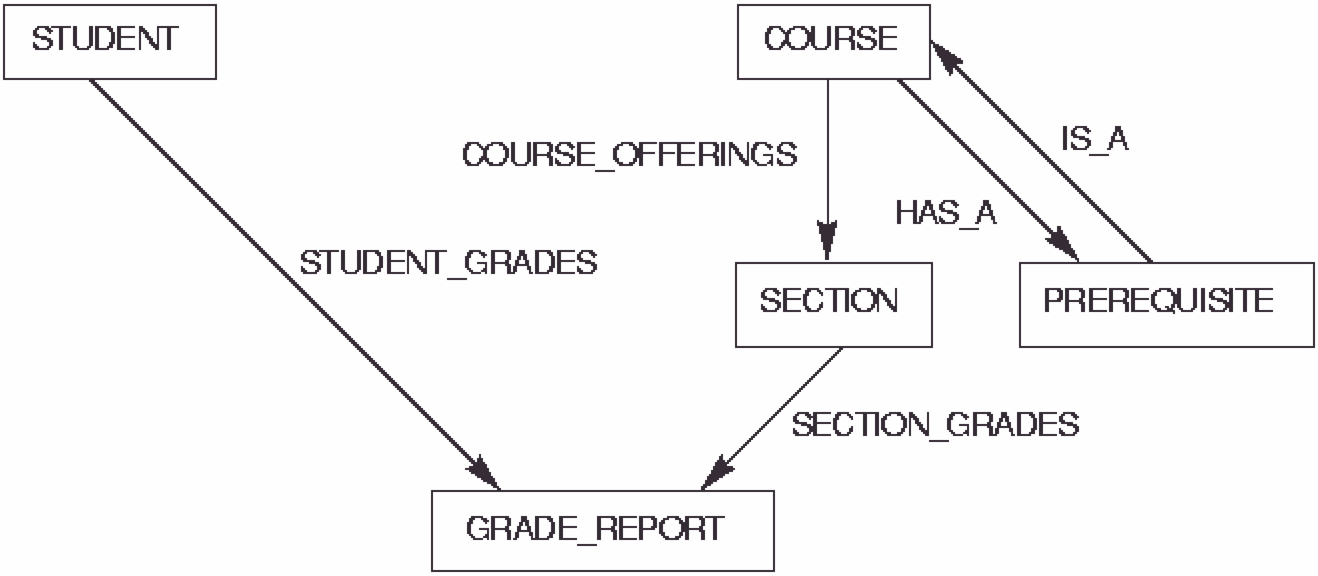
9. Overview of the data models (3)

The **Network data model** represents data as record types and 1:N relationships (called set types).

It has an associated record-at-a-time language that must be embedded in a host programming language.

Example:

Figure 2.4 The schema of Figure 2.1 in the notation of the network data model.



© Addison Wesley Longman, Inc. 2000, Elmasri/Navathe, Fundamentals of Database Systems, Third Edition

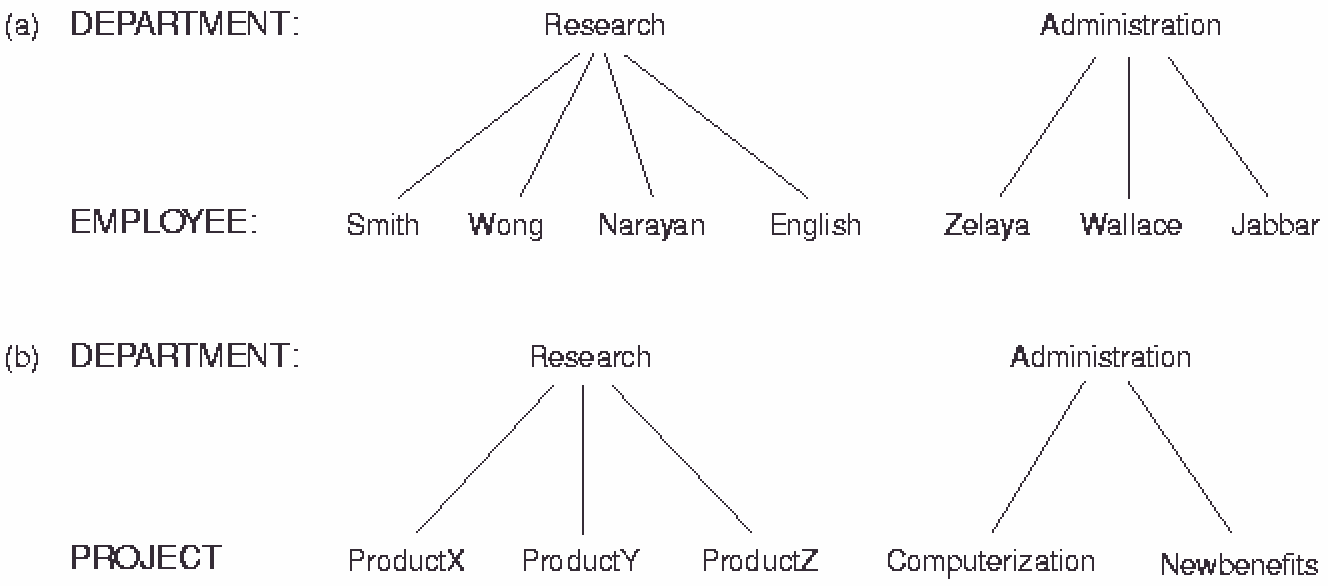
9. Overview of the data models (4)

The **Hierarchical data model** represents data as hierarchical tree structures (parent-child relationship types)

There is no standard language. Most hierarchical DBMS use a record-at-a-time language.

Example:

Figure D.2 Occurrences of Parent-Child Relationships.
(a) Two occurrences of the PCR type (DEPARTMENT, EMPLOYEE).
(b) Two occurrences of the PCR type (DEPARTMENT, PROJECT).


(a) DEPARTMENT:
Research
Administration
EMPLOYEE:
Smith
Wong
Narayan
English
Zelaya
Wallace
Jabbar
(b) DEPARTMENT:
Research
Administration
PROJECT
ProductX
ProductY
ProductZ
Computerization
Newbenefits