第1章データベースの概要

Overview of Database

KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

fit@hcmus4.0

コンテンツ

Introduction

Real business & application

Marketing, production, banking, education, entertainment, medical, … → **complex, richness**

develop

Automate & support tasks

Computers

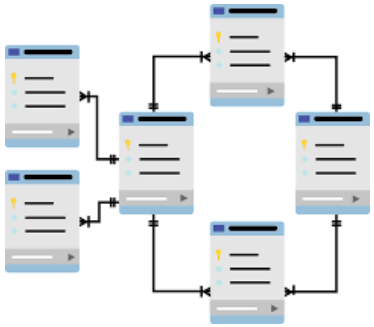
How can computers understand the real world domain to digitize & support automation?

導入

□ どうやって？

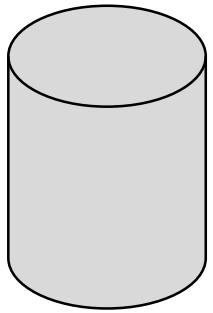
Business environment

Analyze & design



Data

Implementation



Database

Processes, business rules, interface

Application

導入

□ 組織の情報の例

- 銀行と金融
  - 顧客情報、アカウント、ローン、銀行取引
    - 顧客情報：名前、住所、電子メール、身元番号、…
  - 販売と購入の情報
- 教育
  - 学生情報、コース登録、成績
- 航空会社
  - 予約、フライトの情報、チケット価格
- 人事
  - 従業員、給与、給与、ピット（税金）、報酬、才能に関する情報

Introduction

What is data ?

- Facts that can be recorded and have meaning
- Pieces of data are individual pieces of information
- Data is a collection of facts, such as numbers, words, measurements, observations or even just descriptions of things represented in a form suitable for processing by computer

Example

- “Nguyễn Văn A” is a student’s name
- “11.12.008 ” is a student’s ID
- “19-02-2015” is a Tet day of 2015

Introduction

- **Database (DB)**
  - **A collection of related data**
  - **Contains information relevant to a business**

- **Example:**
  - Sale, purchase
  - Payable and receivable accounts
  - Employees
  - Printing of employee's weekly paychecks

Introduction

- **Database (DB)**
  - Definition:
    - A logically coherent collection of data with some inherent meaning
      - Random assortment of data cannot correctly be a database
  - is designed, built, and populated with data for a specific purpose, for intended group of users or applications
  - Example:
    - A list of students → data structure of group of students
    - A list of classes → data structure of classes
  - DB is stored using a structure → structured database
  - Other types of database: unstructured databases, document databases, graph databases

Example 1 – Course Management

MÔNHỌC	TênMH	MãMH	SốTC	Khoa
	Khoa học máy tính	CS1310	4	CNTT
	Cấu trúc dữ liệu	CS3320	4	CNTT
	Toán rời rạc	MATH2410	3	TOÁN
	Cơ sở dữ liệu	CS3380	3	CNTT

SINHVIÊN	Tên	MSSV	Lớp	Khoa
	Trang	17	1	CNTT
	Ngọc	8	2	CNTT

HỌCPHẦN	MãHP	MãMH	HọcKỳ	Năm	GiáoViên
	85	MATH2410	1	2008	Anh
	92	CS1310	1	2007	Tiến
	112	MATH2410	2	2008	Anh
	119	CS1310	2	2007	Tiến

KẾTQUẢ	MSSV	MãHP	Điểm
	17	112	10
	17	119	7
	8	85	6
	8	92	9

ĐIỀUKIỆN	MãMH	MãMH_Trước
	CS3380	CS3320
	CS3380	MATH2410
	CS3320	CS1310

cdio 4.0

fit@hcmus

Example 2 – PROJECT MANAGEMENT

EMPLOYEE	LName	MName	FName	SSN	BirthDate	SuperSSN	DNo
	Tran	Hong	Quang	987987987	03/09/1969	987654321	4
	Nguyen	Thanh	Tung	333445555	12/08/1955	888665555	5
	Nguyen	Manh	Hung	666884444	09/15/1962	333445555	5
	Tran	Thanh	Tam	453453453	07/31/1972	333445555	5

PROJECT	PName	PNumber	PLocation	DNum
	San pham X	1	VUNG TAU	5
	San pham Y	2	NHA TRANG	5
	San pham Z	3	TP HCM	5
	Tin hoc hoa	10	HA NOI	4

WORKS_ON	SSN	PNo	Hours
	123456789	1	32.5
	123456789	2	7.5
	666884444	3	40.0
	453453453	1	20.0

Example 2 – Project management

NHANVIEN	HONV	TENLOT	TENNV	MANV	NGSINH	MA_NQL	PHG
	Tran	Hong	Quang	987987987	03/09/1969	987654321	4
	Nguyen	Thanh	Tung	333445555	12/08/1955	888665555	5
	Nguyen	Manh	Hung	666884444	09/15/1962	333445555	5
	Tran	Thanh	Tam	453453453	07/31/1972	333445555	5

DEAN	TENDA	MADA	DDIEM_DA	PHONG
	San pham X	1	VUNG TAU	5
	San pham Y	2	NHA TRANG	5
	San pham Z	3	TP HCM	5
	Tin hoc hoa	10	HA NOI	4

PHANCONG	MA_NVIEN	SODA	THOIGIAN
	123456789	1	32.5
	123456789	2	7.5
	666884444	3	40.0
	453453453	1	20.0

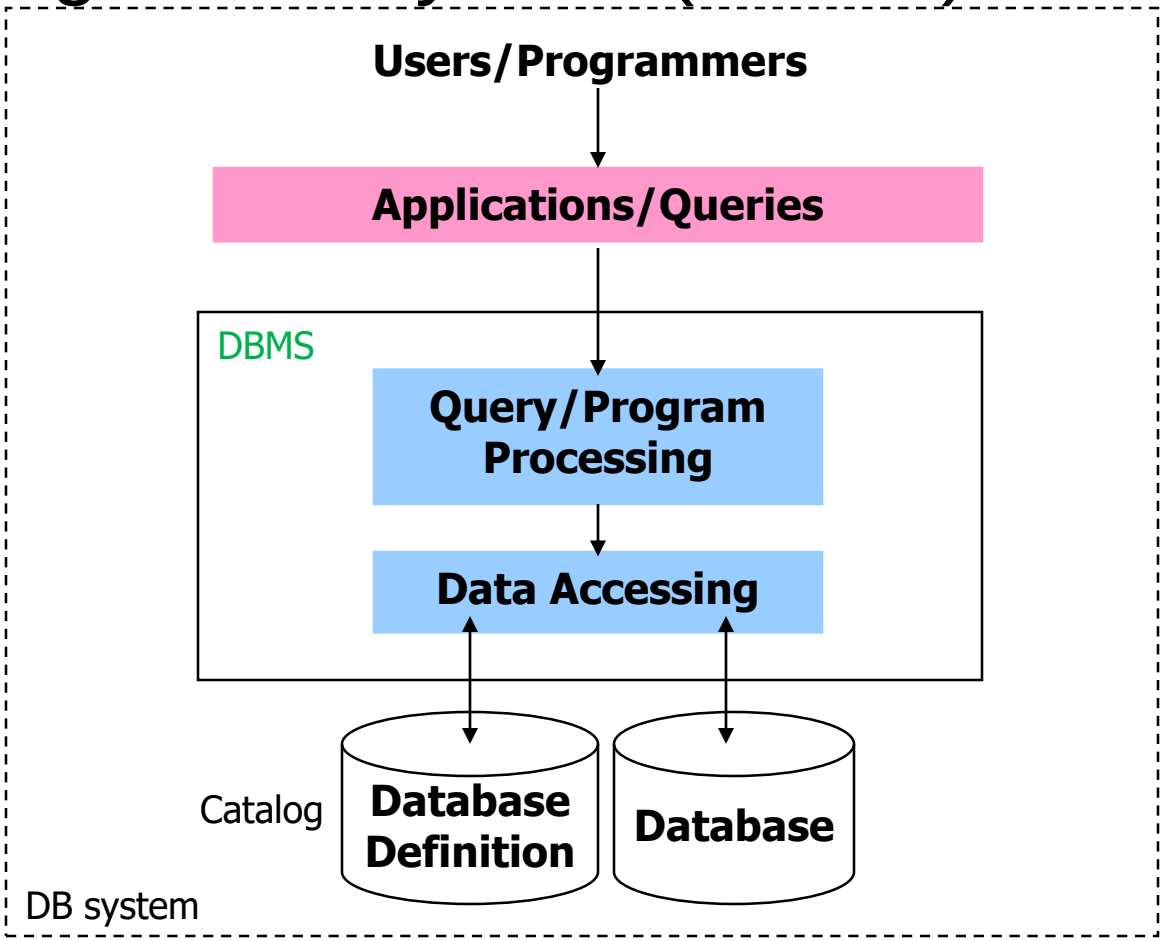
Introduction

- Database Management System (DBMS)
  - A **collection of programs** that enables users to build, operate and maintain a database = **software**
  - A *general-purpose software* system that facilitates
    - ***Definition*** – specifying the data types, structures, and constraints for the data
    - ***Construction*** – storing the data itself on some storage medium
    - ***Manipulation*** – querying the database to retrieve data, updating the database to reflect changes, generating reports from the data
    - ***Sharing*** – allowing multiple users/programs to access the database concurrently

Introduction

Database Management System (DBMS)

Set of programs that enable users to create, operate, and maintain databases



Example #3

- Company database - project management
  - Definition
    - Specify the structure of records, including data elements, data types
  - Construction
    - Store data to represent an employee, project, department… as a record
  - Manipulation
    - Querying: “Select the employees whose department is 5”
    - Updating: “Move the employee Nguyen Thanh Tung to department 1”

Quiz #1

- What below statements are TRUE?
  - A. DB is a collection of related data
  - B. DB is collection of facts
  - C. DB is stored in a structure called structured database
  - D. “22-01-2023” is a data

- DBMS is?
  - A. A data set with the same structure
  - B. Discrete data
  - C. Tools supporting data or application programming
  - D. A set of programs to build, operate, and maintain databases

cdio 4.0

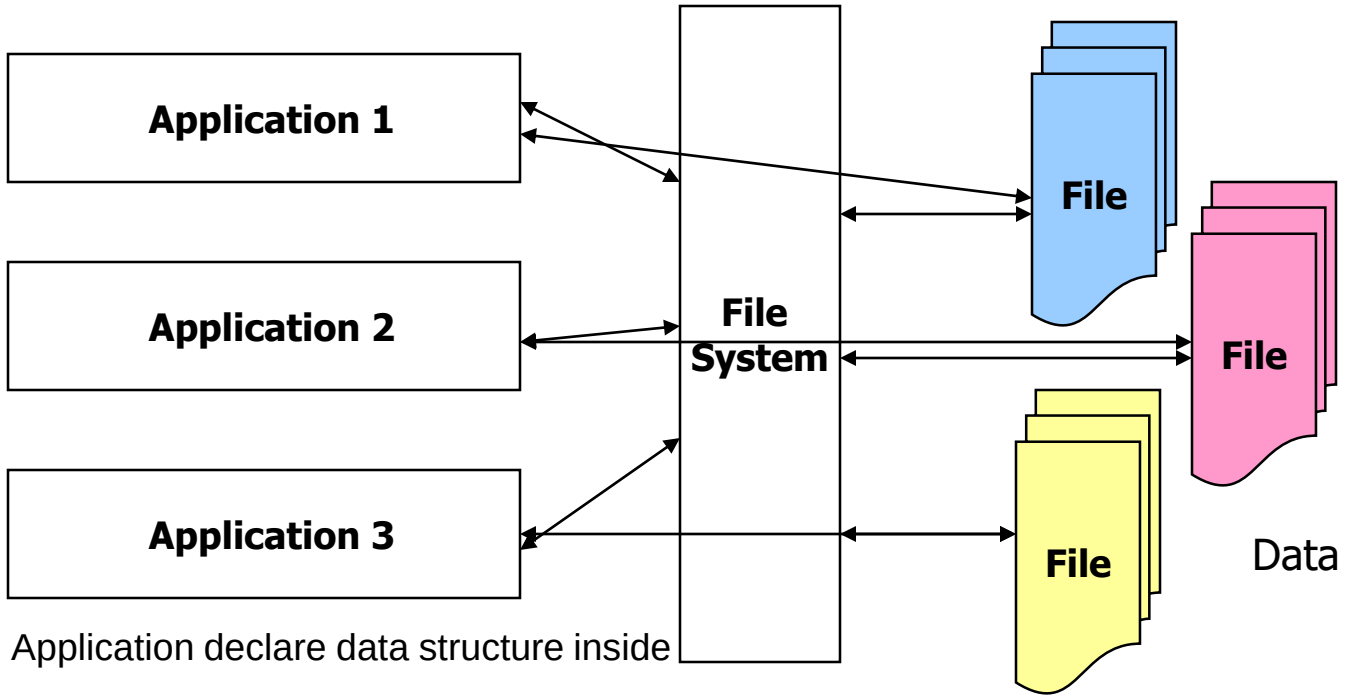
fit@hcmus

Content

- Introduction
- **The evolution of database systems**
- Characteristics of the database approach
- Database users
- Architecture of a DBMS
- Properties of DBMS
- Data models
- Database languages

Evolution

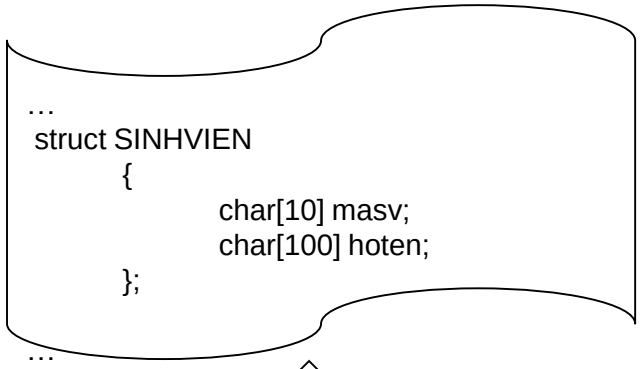
- File



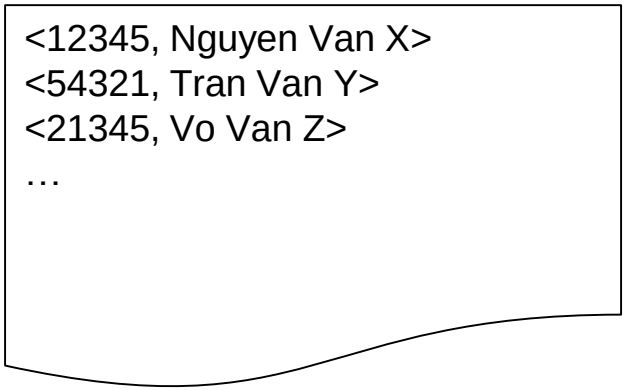
An application program has its own data

Evolution - Example

Program



Data file



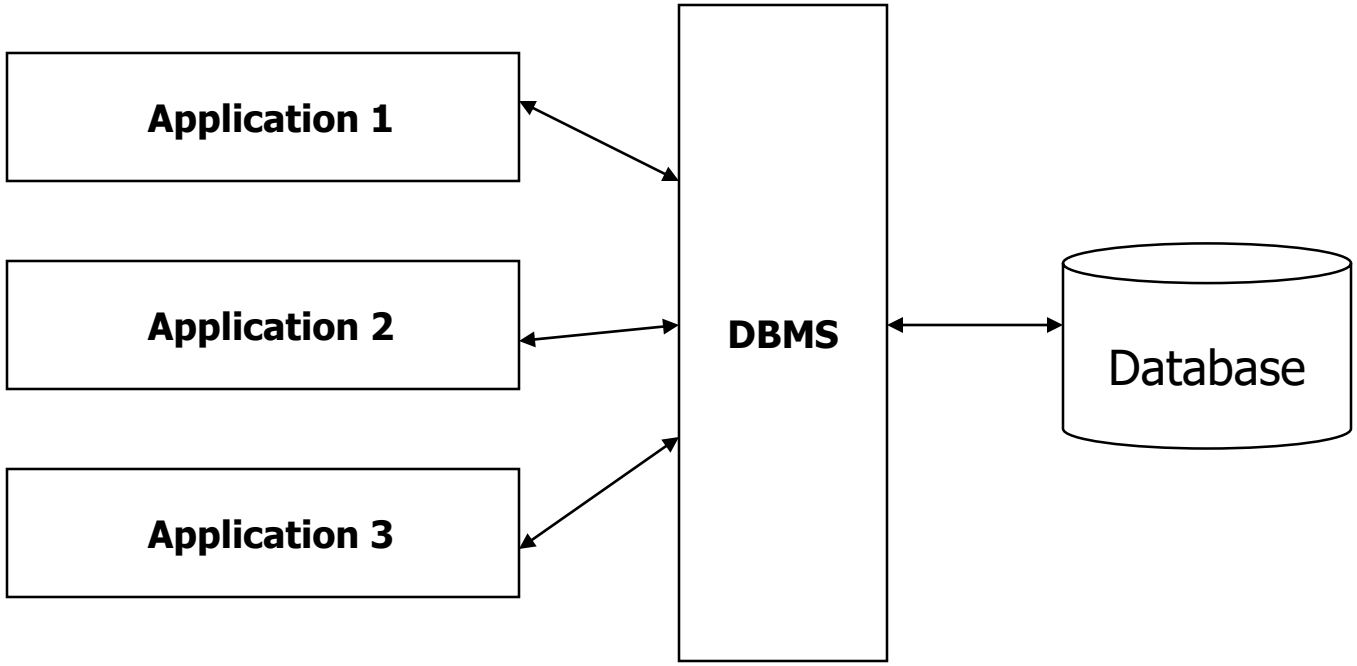
Evolution

- Limitations
  - Data redundancy
    - Wasted storage space
    - Opportunities of the inconsistency
  - Data sharing is limited
  - Difficult recovery
  - Low security

- But, still be used in some applications
  - Small size
    - Storing and accessing data only, not including other processing operations
  - Fee costs less
    - Operation or maintenance

Evolution

- Database



cdio 4.0

fit@hcmus

Content

- Introduction
- The evolution of database systems
- **Characteristics of the database approach**
  - Self-describing
  - Insulation between programs and data
  - Data abstraction
  - Views of data
  - Sharing of data
- Database users
- Architecture of a DBMS
- Properties of DBMS
- Data models
- Database languages

Self-Describing

- The DB system contains not only the DB itself, but also a complete *definition/description* of the DB structure
- The *definitions* are stored in ***catalog*** called "**metadata**"
  - Contains information such as the structure of data, type and storage format of data items, and constraints on the data
- Many applications can access to the DB
  - Refer to ***catalog***, knowing the structure of files in specific DB (type and format of data)

Self-Describing

- An example of a database **catalog/metadata**

RELATION

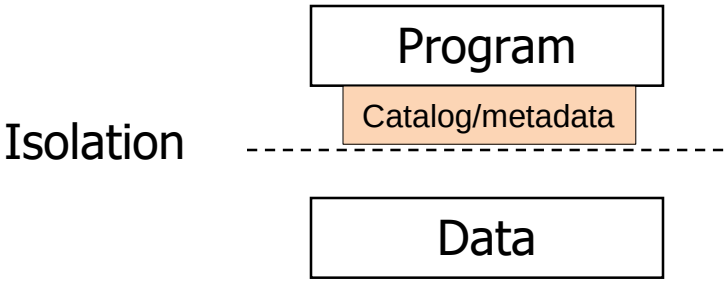
Relation_name	No_of_columns
EMPLOYEE	7
PROJECT	4
WORKS_ON	3

COLUMN

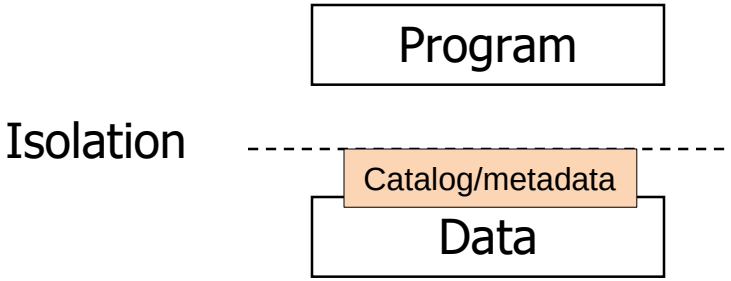
Column_name	Data_type	Belongs_to_relation
LName	Character(10)	EMPLOYEE
FName	Character(10)	EMPLOYEE
…	…	…

Isolation

- The structure of data is stored in *catalog* separately from the access programs
  - Program-Data independence



Program contains catalog → data structure depends on the program



DBMS contains catalog → data is independent of program

- A little change in the data structure
  - Application programs are rarely revised

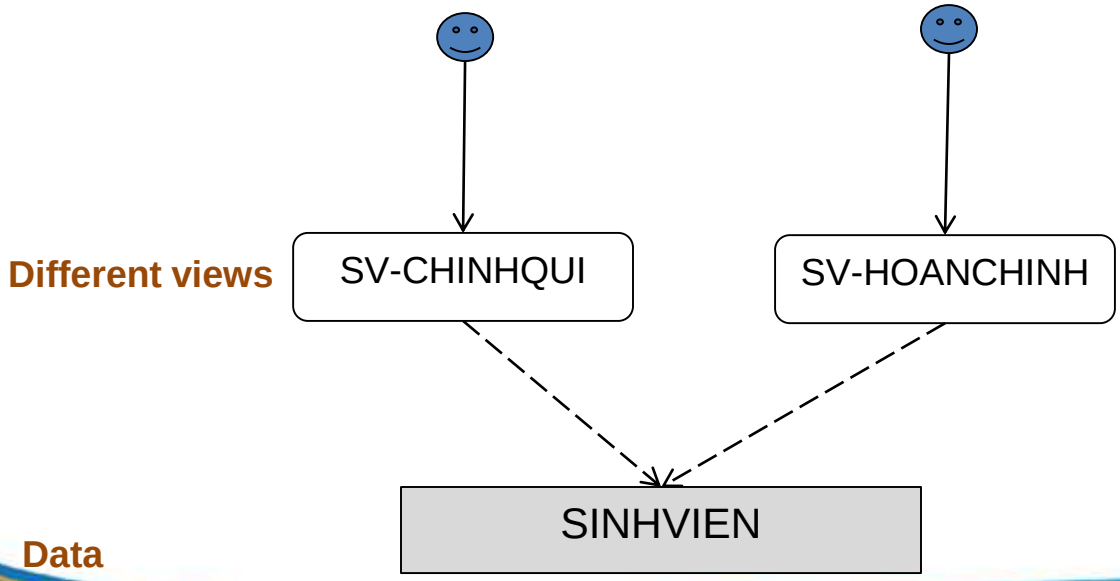
Data abstraction

- The DB system provides a ***conceptual representation*** of the data to hide certain details of how the data are stored and maintained

- Example
  - *Data model* is a type of data abstraction
    - Objects
    - Properties
    - Relationships
  - These logical concepts are easier for user to understand than computer storage concepts

Views of data

- A DB has many users
- Each user may require a different ***perspective or view*** of the database
- A view may be
  - A subset of the database
  - Aggregate data that are derived from the database



Sharing of data

- A multiuser DBMS
  - **Allows multiple users** accessing to DB at the same time
  - Data for many applications are integrated and maintained in *a single DB*
- Using concurrency control mechanisms to access the data reasonably
  - Avoid data contention (tranh chấp)
  - Ensure the data will always be valid when they are accessed

Quiz #2

1. Self-Describing is
   A. To aggregate data that are derived from the database
   B. To enable a DB to define in catalog called “metadata”
   C. To enable only one application can access to DB
   D. To define different views of DB
2. What below statements are TRUE?
   A. Sharing data: allows multiple users to access to DB at the same time
   B. A view can mainly store data of DB
   C. Isolation: combining program and catalog/metadata together
   D. Data abstraction provides a conceptual representation of the data to hide certain details of how the data are stored and maintained

Content

- Introduction
- The evolution of database systems
- Characteristics of the database approach
- **Database users:** Actors on the scene
  - Database administrator (DBA)
  - Database designer
  - End user
- Architecture of a DBMS
- Properties of DBMS
- Data models
- Database languages

Database administrator

- Many people use the same resources
  - Need a chief administrator to oversee and manage
- Responsibility
  - Administering the DB
  - Authorizing access to the DB
  - Coordinating and monitoring the use of DB
  - Acquiring software and hardware resources as needed

Database designer

- Responsibility
  - Identifying the data to be stored in the DB
  - Choosing appropriate structures to represent and store the DB
  - Communicating with all DB users to understand their requirements, to come up with a design that meet the requirements
- Can be
  - Staff of the DBA
  - Other staffs taking responsibilities after the DB designed is completed

End user

- People whose jobs require to access to the DB
  - Querying, updating, generating reports
- Categories
  - Casual end user
  - Naïve or parametric end user
  - Sophisticated end user

End user

- People whose jobs require to access to the DB
  - Querying, updating, generating reports

- Categories
  - Casual end user
    - Occasionally access the DB
    - Need different information each time
    - Use sophisticated DB query language to specify requests
    - Middle or high level manager
  - Naïve or parametric end user
  - Sophisticated end user

End user

- People whose jobs require to access to the DB
  - Querying, updating, generating reports
- Categories
  - Casual end user
  - Naïve or parametric end user
    - Constantly query and update the DB
    - Use standard types of queries and updates that have been programmed and tested
    - Employee
  - Sophisticated end user

End user

- People whose jobs require to access to the DB
  - Querying, updating, generating reports
- Categories
  - Casual end user
  - Naïve or parametric end user
  - Sophisticated end user
    - Be familiar with the facilities of the DBMS
    - Implement the applications to meet the complex requirements
    - Engineers, scientists, business analysts

cdio 4.0

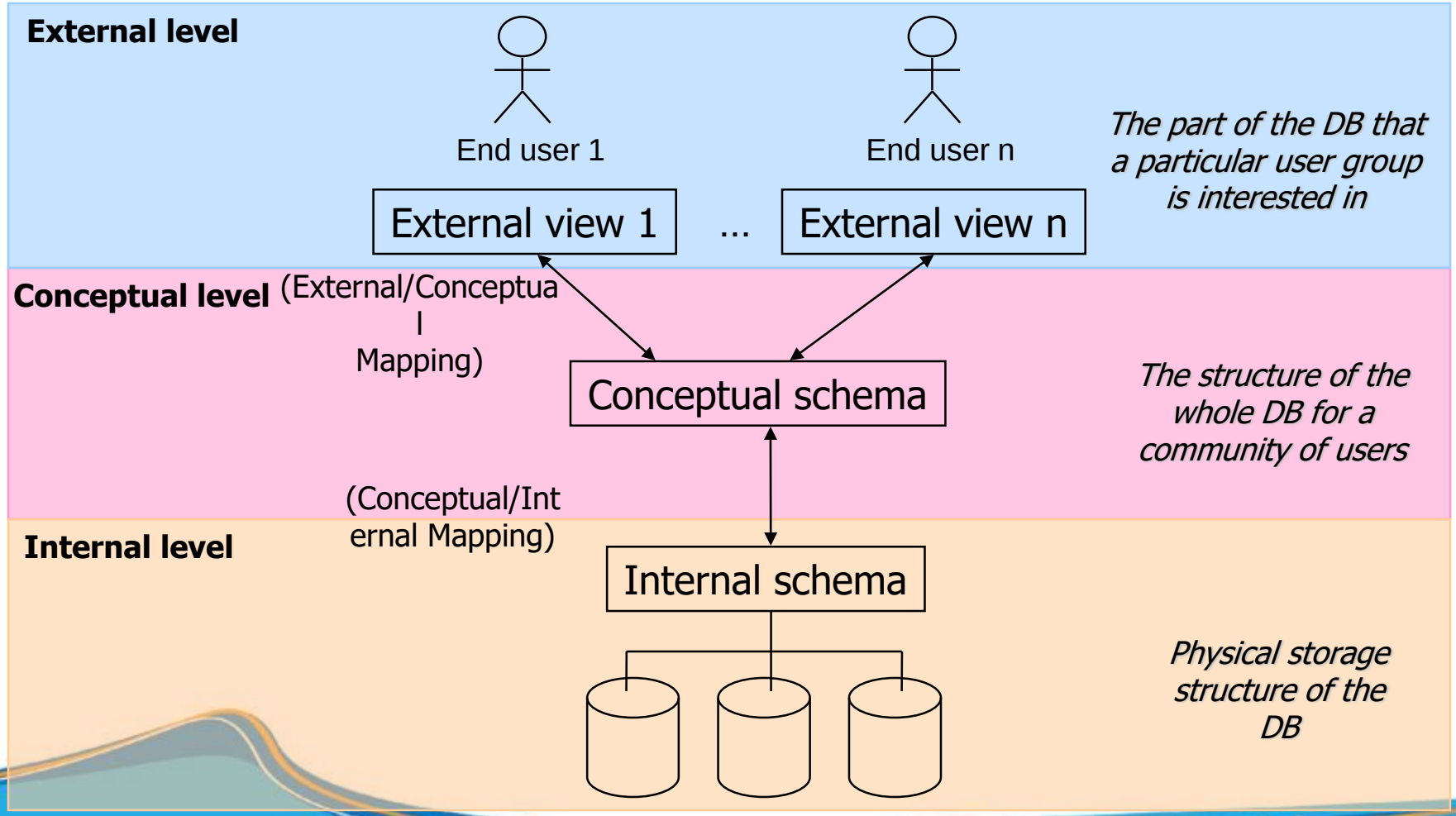
fit@hcmus

Content

- Introduction
- The evolution of database systems
- Characteristics of the database approach
- Database users
- **Architecture of a DBMS**
- Properties of DBMS
- Data models
- Database languages

Architecture

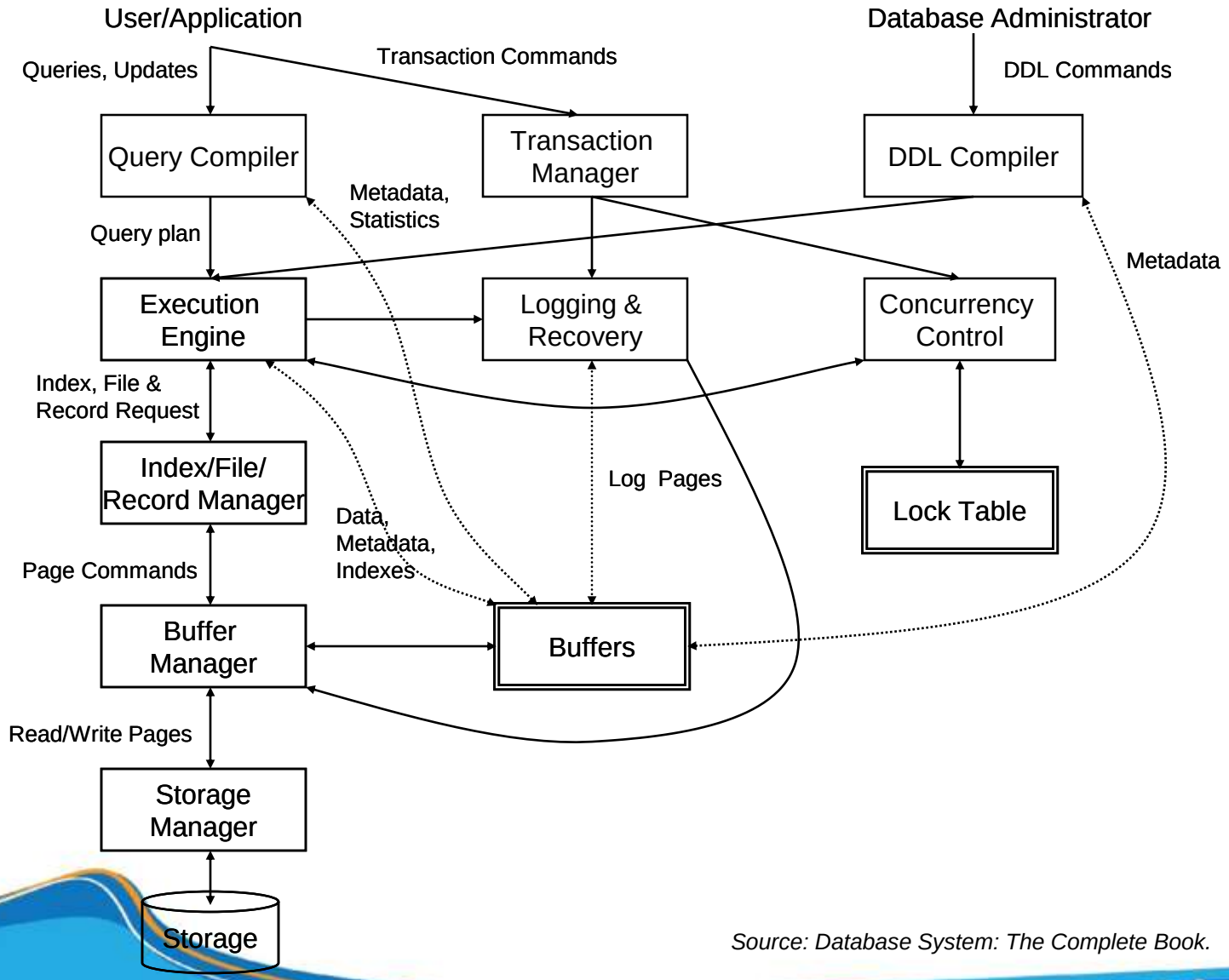
Three-schema architecture:



Architecture

- Data independence
  - Logic data independence
    - The capacity to change the conceptual schema without change to external schemas or application programs
    - Example
      - Adding/removing a record type or data item (expand/reduce DB)
      - Changing constrains
  - Physical data independence
    - The capacity to change the internal schema without change to the conceptual schema
    - Example
      - Physical files had to be reorganized to improve the performance of retrieval or update

Architecture



Source: Database System: The Complete Book.

cdio 4.0

fit@hcmus

Content

- Introduction
- The evolution of database systems
- Characteristics of the database approach
- Database users
- Architecture of a DBMS
- **Properties of DBMS**
- Data models
- Database languages

Properties of DBMS

- Controlling redundancy
  - By placing all the data together, we do not have to search multiple files to collect this data
- Data sharing
  - In multiple user environment, concurrency data access is allowed
- Restricting unauthorized access
  - Users or user groups are given account numbers protected by passwords to gain access to the DB
- Providing multiple user interfaces
  - Provide query languages for casual users, programming language interfaces for programmers, forms and command codes for parametric users

Properties of DBMS

- Enforcing integrity constraints
  - Integrity constraints
    - Rules/conditions are derived from the meaning/semantics of the data or the miniworld it represents
  - Some constraints
    - Can be specified to the DBMS and automatically enforced
    - May have to be checked by update programs
- Providing backup and recovery
  - Provide facilities for recovering from hardware and software failures
  - Make sure the DB is restored to the state it was before

Properties of DBMS

- Others
  - Potential for enforcing standards
    - Permit DBA to define and enforce standards among database users in a large organization
  - Flexibility
    - It may be necessary to change the structure of a DB as requirements change without affecting the stored data and the existing application programs
  - Reduced application development time
  - Availability of up-to-date information
    - As soon as one user's update is applied to the DB, all other users can immediately see this update

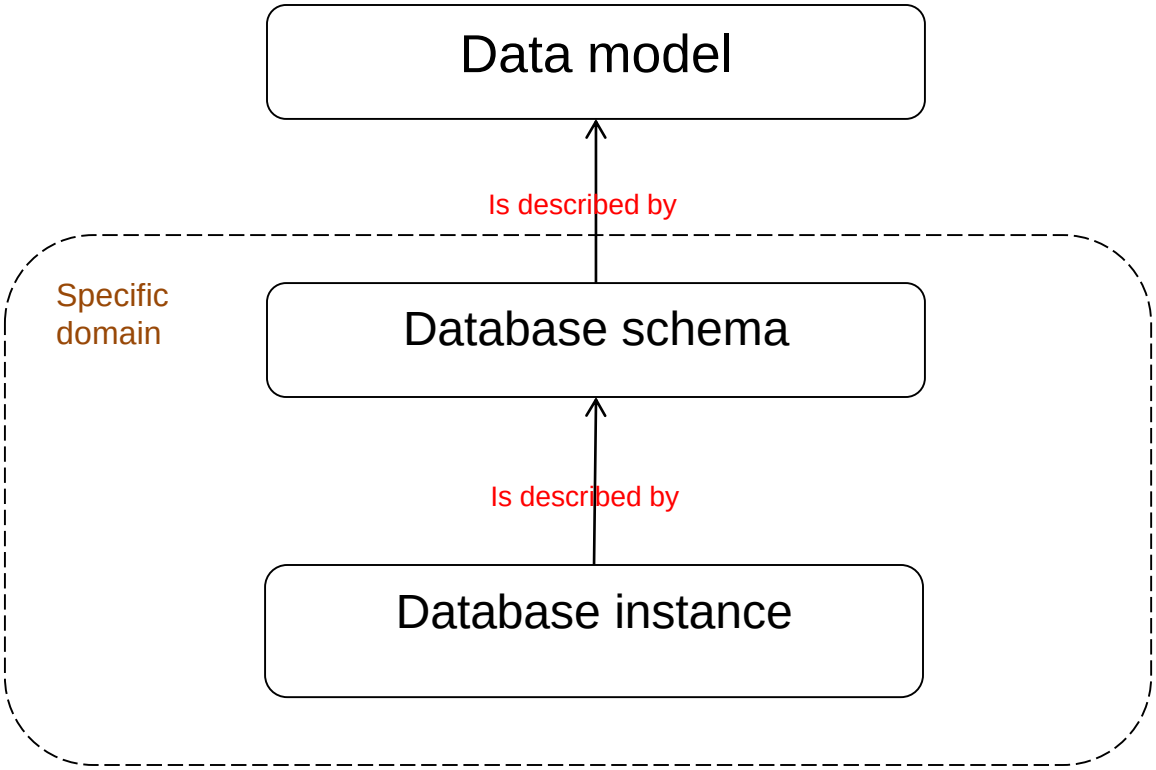
cdio 4.0

fit@hcmus

Content

- Introduction
- The evolution of database systems
- Characteristics of the database approach
- Database users
- Architecture of a DBMS
- Properties of DBMS
- **Data models**
- Database languages

Data models


Data model
Is described by
Specific domain
Database schema
Is described by
Database instance

Data models

- Definition
  - A collection of ***concepts*** that can be used to describe the ***structure of a DB***
    - Data types, relationships, and constraints
  - Including a set of basic ***operations*** for specifying retrievals and updates on the DB
- Categories
  - High level or conceptual data models
  - Representational or implementation data models
  - Low level or physical data models

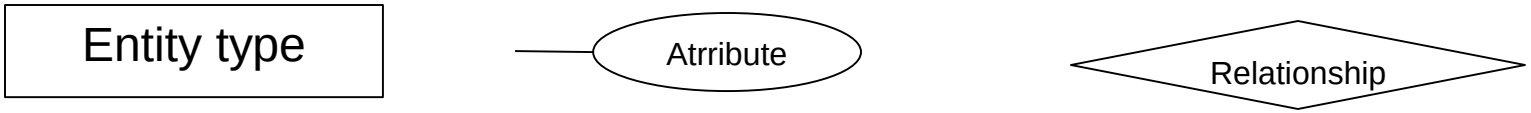
Data models

- High level data model
  - Provide concepts that are close to the way users perceive data
  - E.g.: entity relationship model, object-oriented model…
- Implementation data model
  - Provide concepts that may be understood by end users, but that are not too far from the way data is organized within the computer
  - E.g.: relational model, network and hierarchical models…
- Low level data model
  - Provide concepts the describe the details of how data is stored in the computer

Example of data model

Entity Relationship Diagram

Concepts



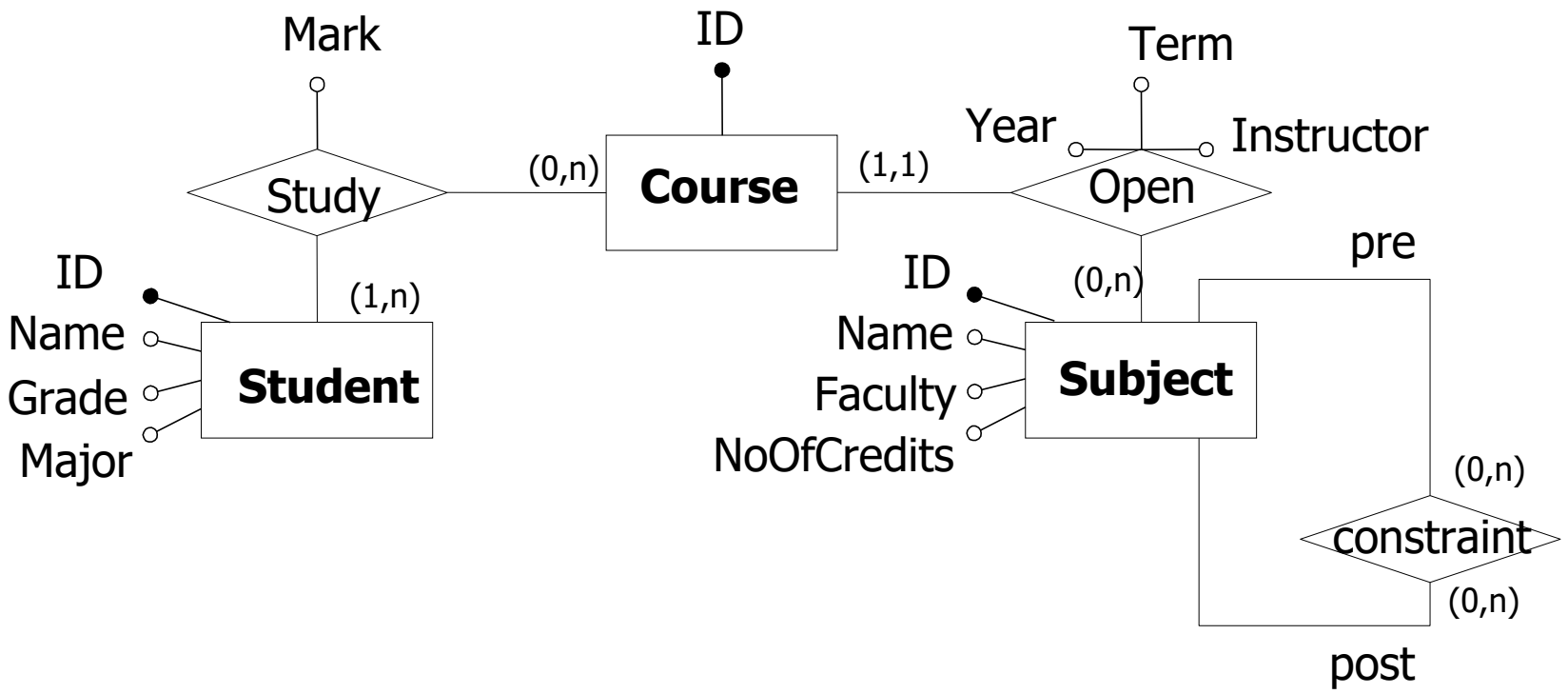
Network data model

Concepts

cdio 4.0

fit@hcmus

Database schema - Example of ER Model



Database schema

Definition

Description of the structure and constraints on the database about a particular domain (banking, education, marketing, etc.)

Example:

SINH VIÊN	TÊNSV	MÃSV	LỚP	NGÀNH

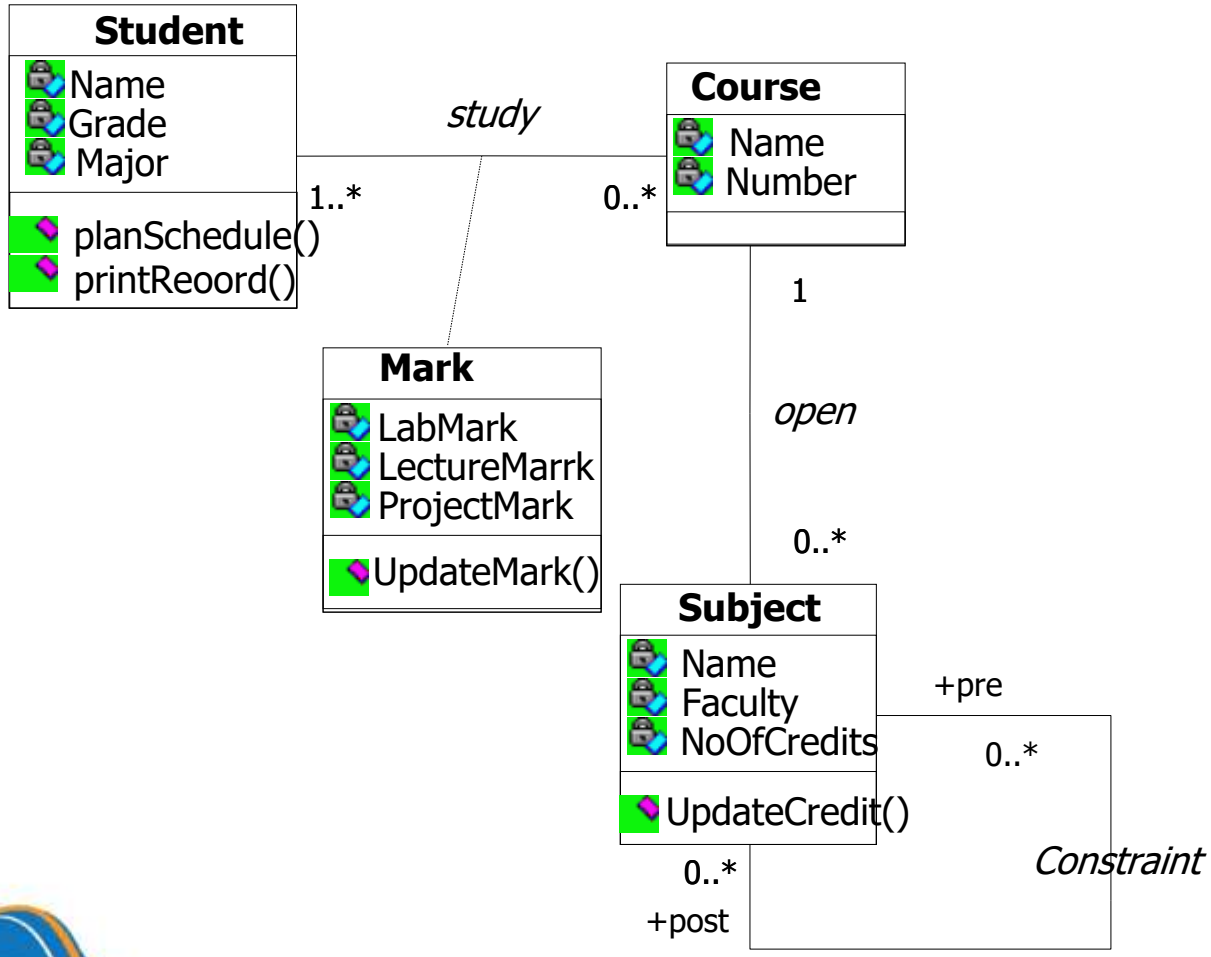
MÔN HỌC	TÊNMH	MÃMH	KHOA	TÍNCHỈ

ĐIỀU KIỆN	MÃMH_TRƯỚC	MÃMH

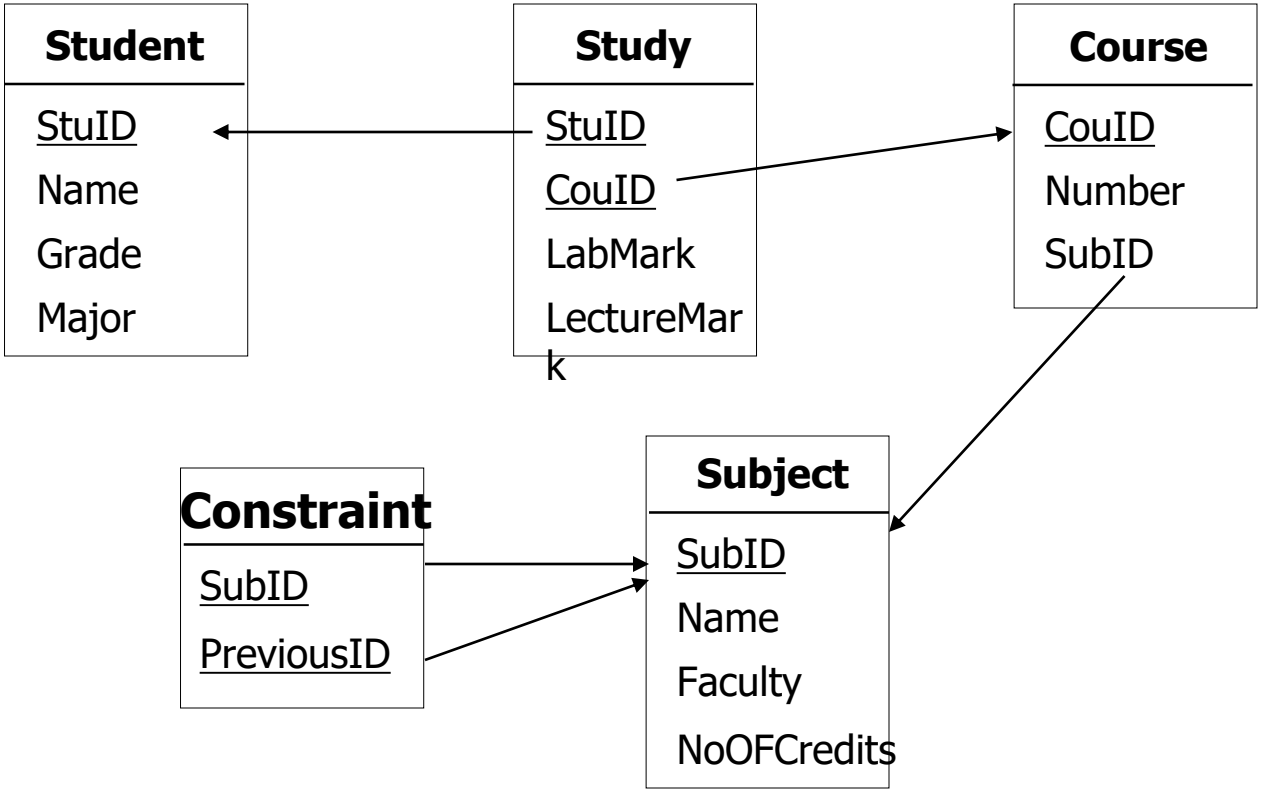
HỌC PHẦN	MÃHP	GIÁOVIÊN	HỌCKỲ	NĂM

KQ_HỌC	MÃSV	MÃHP	ĐIỂM

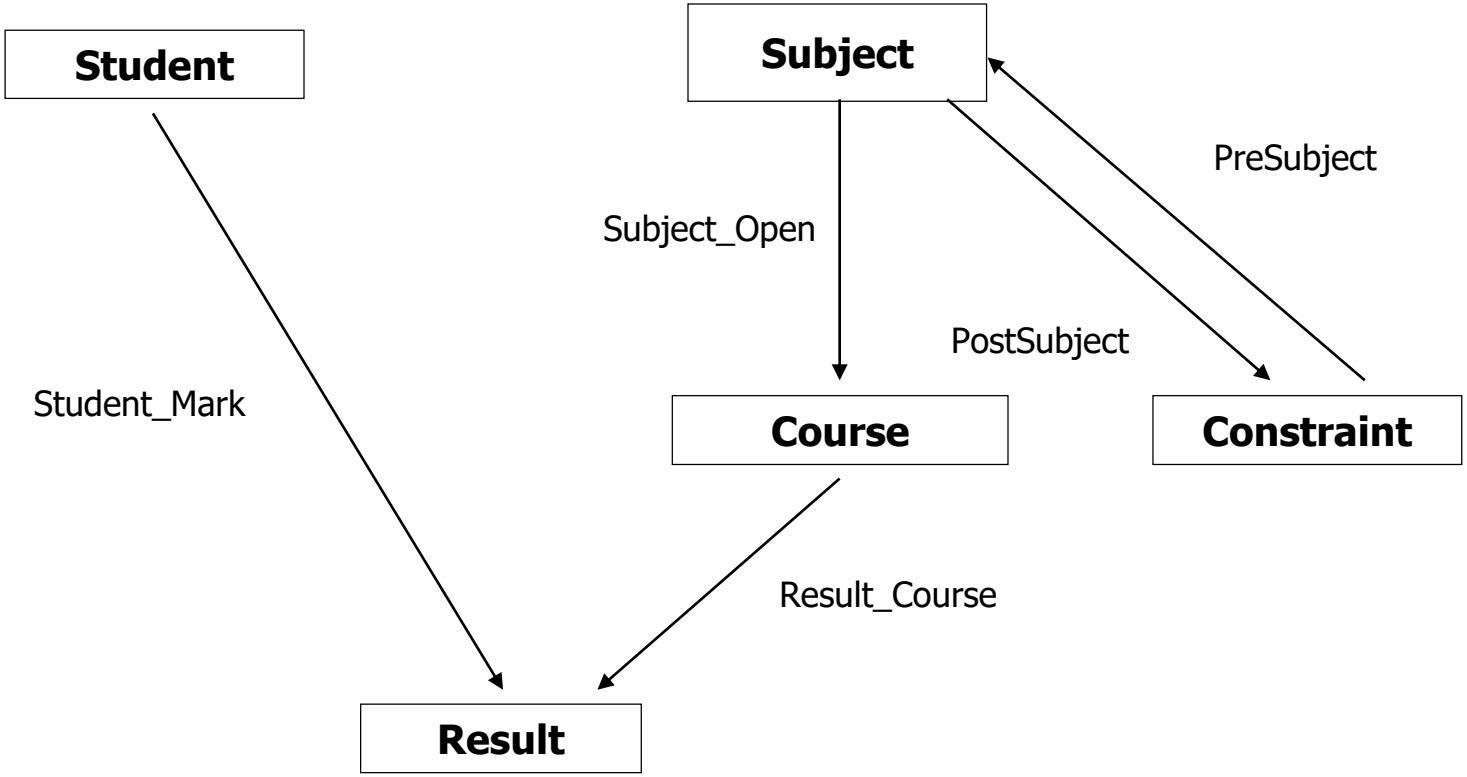
Database schema - Example of Object-Oriented Model



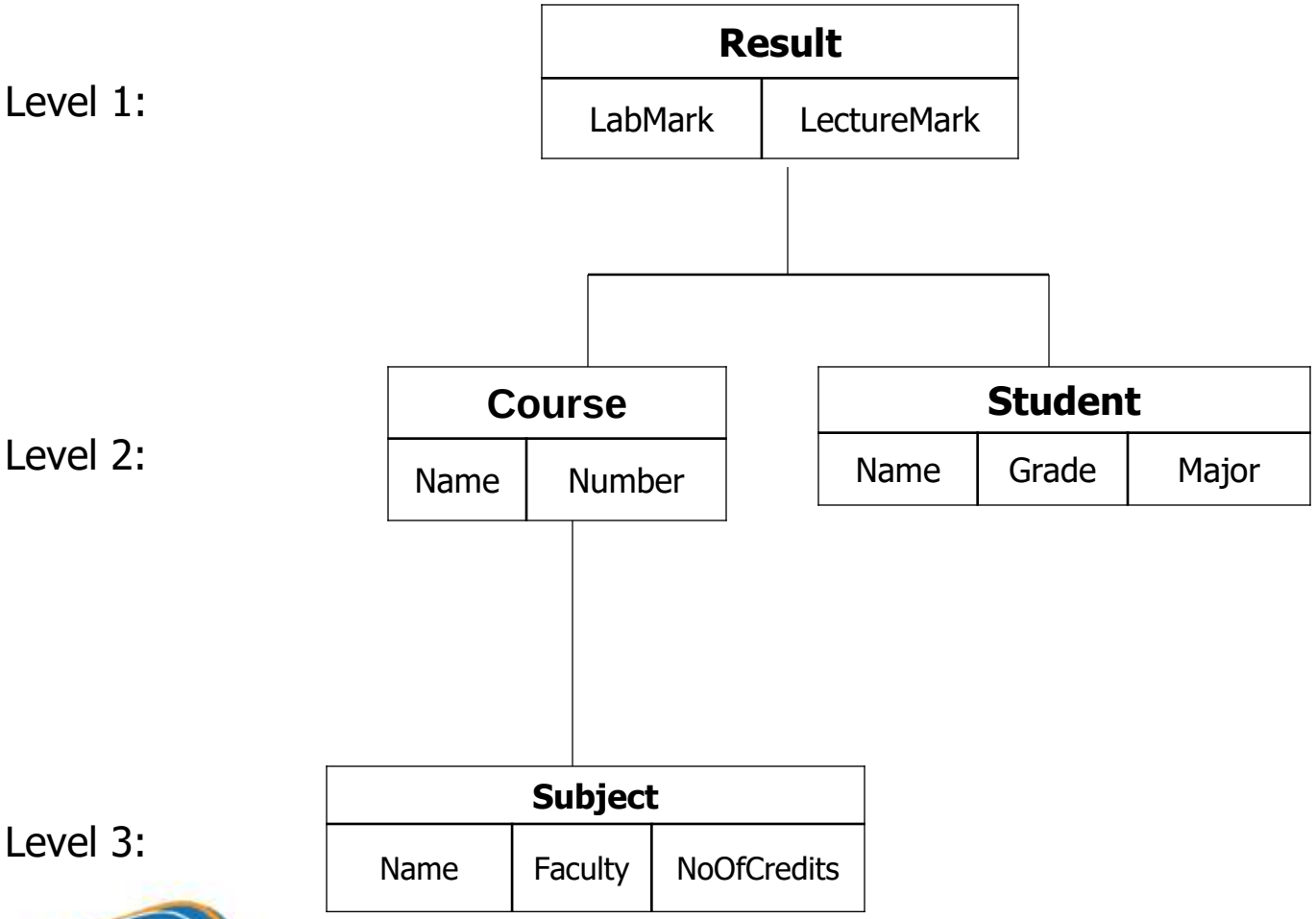
Database schema - Example of relational model



Database schema - Example of network data model



Database schema - Example of hierarchical data model



Database instance or status

Definition

The data stored in database at a particular moment of time is called instance of database.

MÔNHỌC	TênMH	MãMH	SốTC	Khoa
	Khoa học máy tính	CS1310	4	CNTT
	Cấu trúc dữ liệu	CS3320	4	CNTT
	Toán rời rạc	MATH2410	3	TOÁN
	Cơ sở dữ liệu	CS3380	3	CNTT

SINHVIÊN	Tên	MSSV	Lớp	Khoa
	Trang	17	1	CNTT
	Ngọc	8	2	CNTT

HỌCPHẦN	MãHP	MãMH	HọcKỳ	Năm	GiáoViên
	85	MATH2410	1	2008	Anh
	92	CS1310	1	2007	Tiến
	112	MATH2410	2	2008	Anh
	119	CS1310	2	2007	Tiến

KẾTQUẢ	MSSV	MãHP	Điểm
	17	112	10
	17	119	7
	8	85	6
	8	92	9

ĐIỀUKIỆN	MãMH	MãMH_Trước
	CS3380	CS3320
	CS3380	MATH2410
	CS3320	CS1310

cdio 4.0

fit@hcmus

Content

- Introduction
- The evolution of database systems
- Characteristics of the database approach
- Database users
- Architecture of a DBMS
- Properties of DBMS
- Data models
- **Database languages**

Database language

- DDL – Data Definition Language
  - Identify descriptions of the schema constructs
  - Store the schema description in the DBMS catalog
- SDL – Storage Definition Language
  - Specify the internal schema and the mappings between two schemas
- VDL – View Definition Language
  - Specify user views and their mapping to the conceptual schema

Database language

- DML – Data Manipulation Language
  - Provide a set of operations including retrieval, insertion, deletion and modification of the data
  - Two types
    - High level (nonprocedural)
      - Entered interactively from a display monitor/terminal, or
      - Embedded in a general-purpose programming language
    - Low level (procedural)
      - Must be embedded in a general-purpose programming language

A query in a high-level DML often specifies *which* data to retrieve rather than *how* to retrieve it; therefore, such languages

Discussion

- When will we use or not use the DB approach?