

Basic Concepts on Database

Vu Tuyet Trinh

trinhvt@soict.hut.edu.vn

Department of Information Systems
School of Information Technology and Communication
Hanoi University of Technology

Example: University Environment

- What we are interested in ?
 - STUDENTs
 - COURSEs
 - SECTIONs (of COURSEs)
 - (academic) DEPARTMENTs
 - INSTRUCTORs
 - What we can do with these information ?
 - Who passed the examine of Database course?
 - Which courses are given by “John”?
 - ...
- *Put information into Database!*



Database

*A shared collection of related data
designed to meet the information needs
of an organisation*

- Logically coherent
- Internally consistent
- Specific purpose
- Representation of the real world

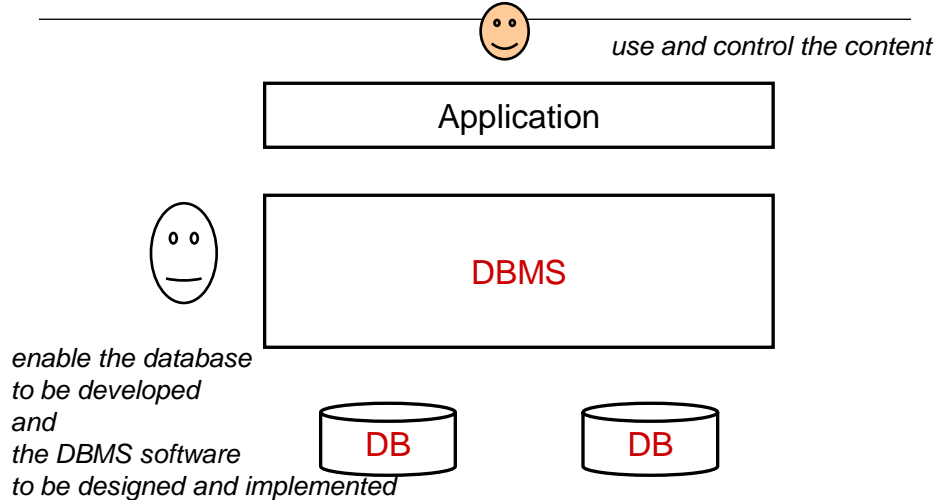


Database Management System

*A software to facilitate the creation and
maintenance of a database*

- Defining ~ specifying *types* of data
- Constructing ~ storing & populating
- Manipulating ~ querying, updating, reporting

Database Environment



Database Users

- ❑ **Database administrators**
 - authorize access to the database
 - co-ordinate and monitoring its use
 - acquire software, and hardware resources, controlling its use and monitoring efficiency of operations.
- ❑ **Database Designers**
 - define the content, the structure, the constraints, and functions or transactions against the database.
 - communicate with the end-users and understand their needs.
- ❑ **End-users**
 - use the data for queries, reports and some of them actually update the database content.
 - Casual end users
 - Naive users
 - Sophisticated end users



Characteristics of Database Approach

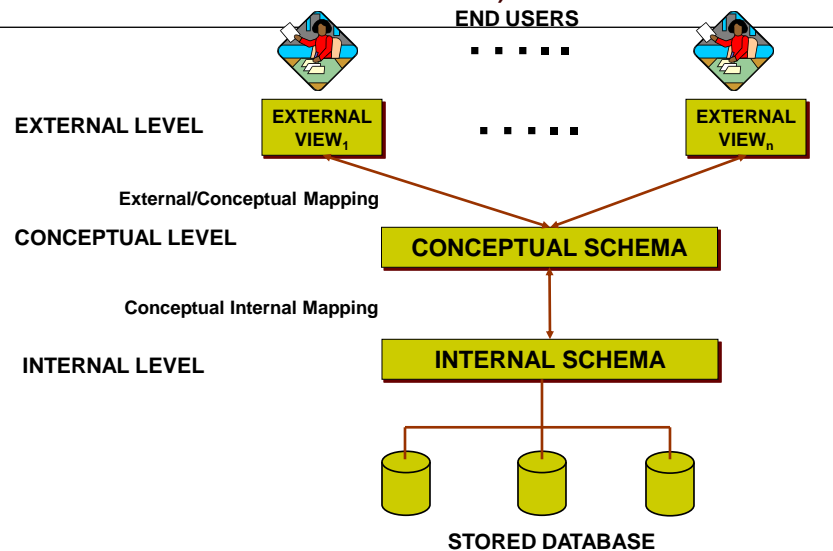
- Self-describing
 - **catalog** (or **meta-data**) stores the *description* of the database
 - allow the DBMS software to work with different DBs
- Data Abstraction:
 - **data model** used to hide storage details
 - present the users with a *conceptual view* of the DB
- Sharing of data
 - support multiple view of a DB
 - allow concurrent access on a DB



Characteristics ... (2)

- Persistence
 - store data on secondary storage
- Retrieval
 - a declarative query language
 - a procedural database programming language
- Performance
 - retrieve and store data quickly
 - deal with large volume of data

3-tier Schema Model (ANSI-SPARC Architecture)



Model vs. Schema vs. Instance

- **Data Model**
 - set of concepts used to describe the structure of a database: data types, relationships, constraints, semantics, ...
 - tool for data abstraction
- **Schema**
 - data structure fulfilled all features of the parts of the real world which is of interest to the users
- **Instance**
 - Data itself

Example

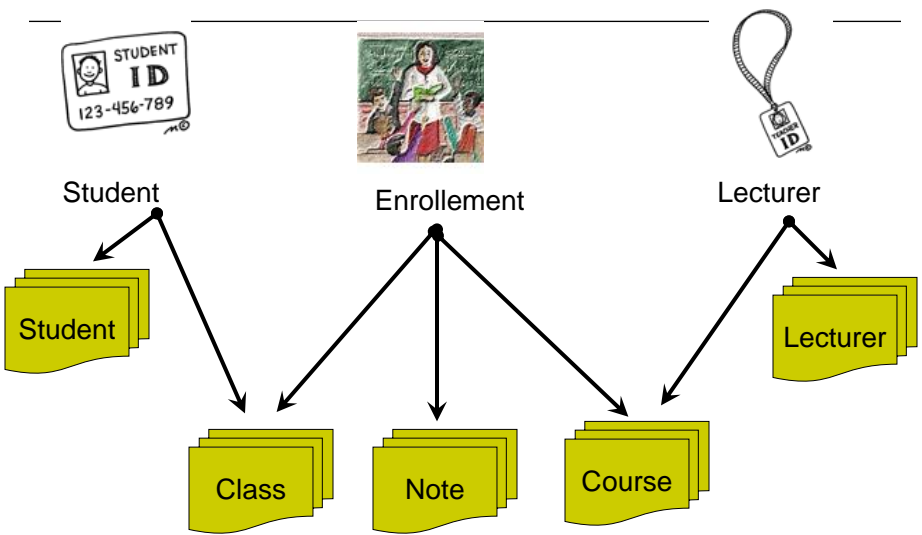
- Data Model

```
type <type_name> = record
  <field_name> : <data_type>;
  <field_name> : <data_type>;
  ...
end;
```
- Schema

```
type student = record
  ID : string;
  fullName: string;
  Birthday: date;
  Address: string ;
  Class: string;
end;
```
- Instance

(« Stud001 », « Nguyen », 1/4/1983, «l Dai Co Viet », « 1F VN K50 »)

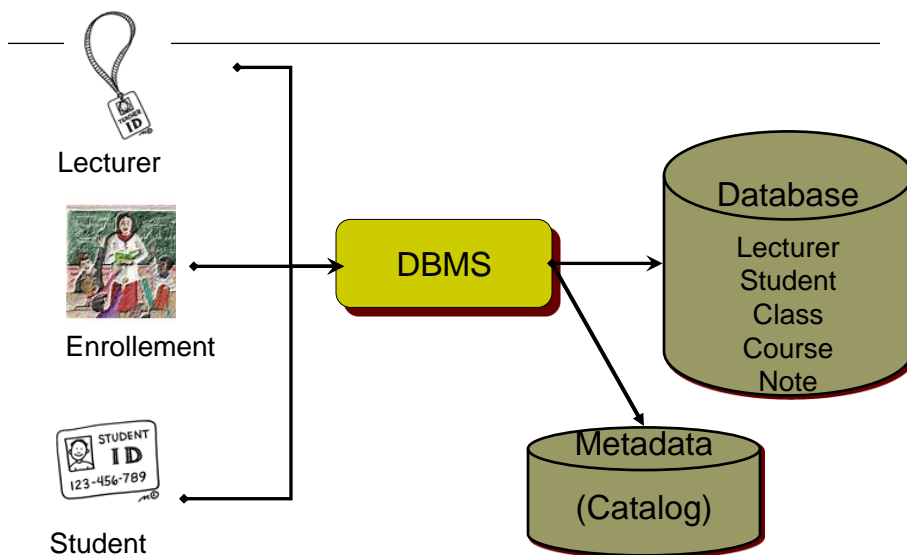
File System approach



Limitations

- Uncontrolled redundancy
- Inconsistent data
- Inflexibility
- Limited data sharing
- Poor enforcement of standards
- Low programmer productivity
- Excessive program maintenance
- Excessive data maintenance

Database approach





Advantages

- Controlled redundancy
 - consistency of data & integrity constraints
- Integration of data
 - self-contained & represents semantics of application
- Data and operation sharing
 - multiple interfaces
- Flexibility
 - data independence
 - data accessibility
 - reduced program maintenance
- Services & Controls
 - security & privacy controls
 - backup & recovery
 - enforcement of standards
- Ease of application development



Remarks

DBMS is

- more expensive
- more complex
- general

When to use DBMS ?

"More than 80 % of real world computer applications are associated with databases"*

* Korth & Silberschatz. Database System Concepts.