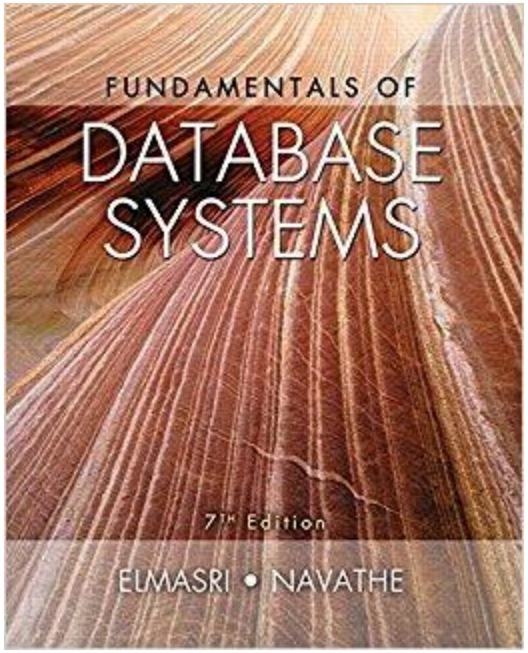
Database Systems

Zaobo He
Department of Computer Science
Georgia State University



Chapter 1: Introduction

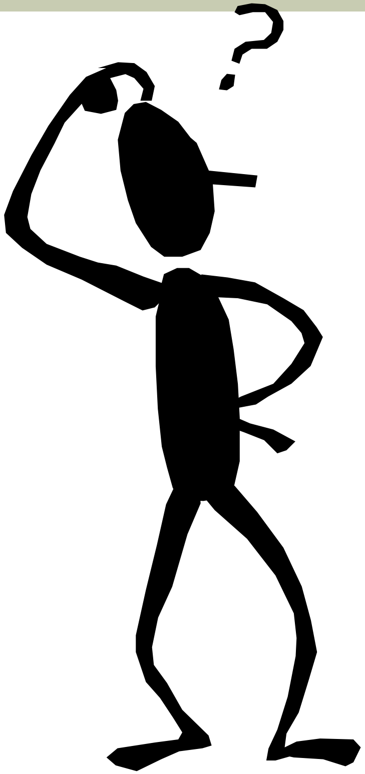
This material is a modified version of the slides provided by Ramez Elmasri and Shamkant Navathe for their book "Fundamentals of Database Systems", 5th edition.

Outline

- Database Definition
- Types of Databases and Database Applications
- Database Management System (DBMS)
- Example of a Database (UNIVERSITY)
- Main Characteristics of the Database Approach
- When Not to Use Databases

Data and Database

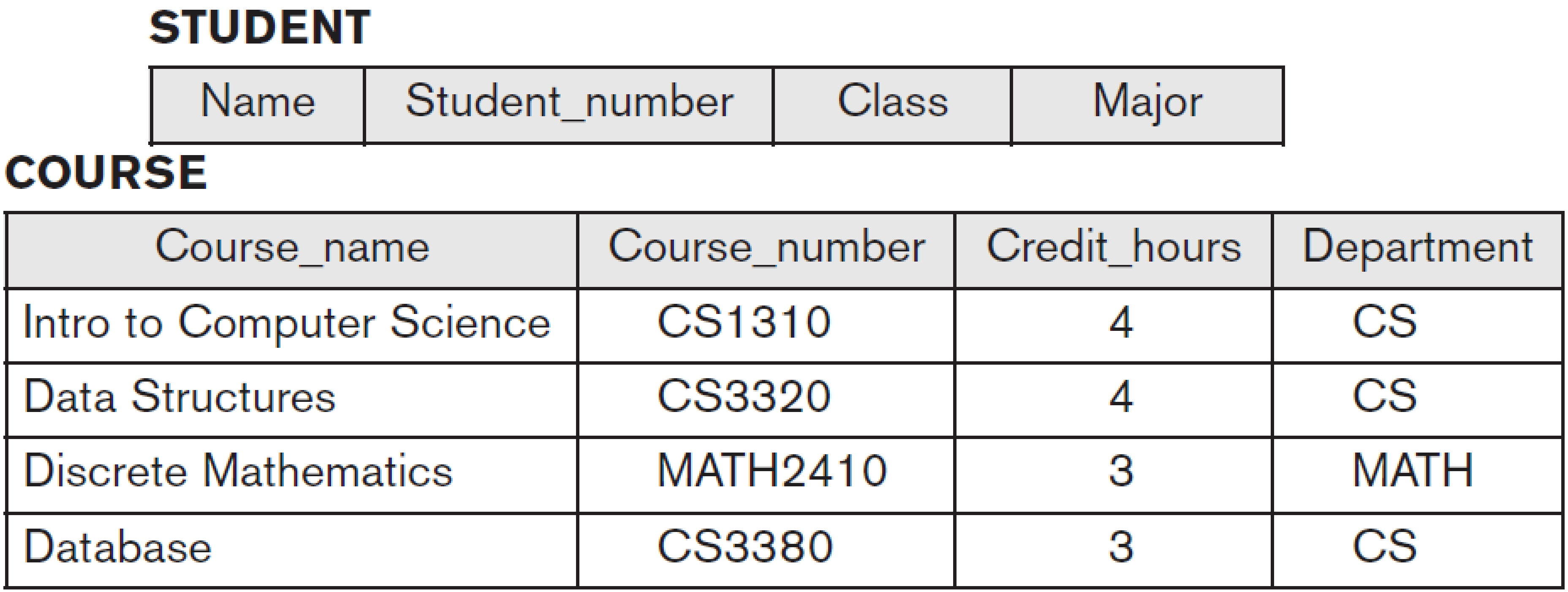
- What is a database?
  - Collection of related data.
- What is data?
  - Known facts that can be recorded and have an implicit meaning.



How about there is no database?

Data and Database (cont.)

- In a database, group only related data together and store them under one group called **table**.
  - Manipulate data efficiently



STUDENT

Name	Student_number	Class	Major

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

Quite general! Need a more restricted definition

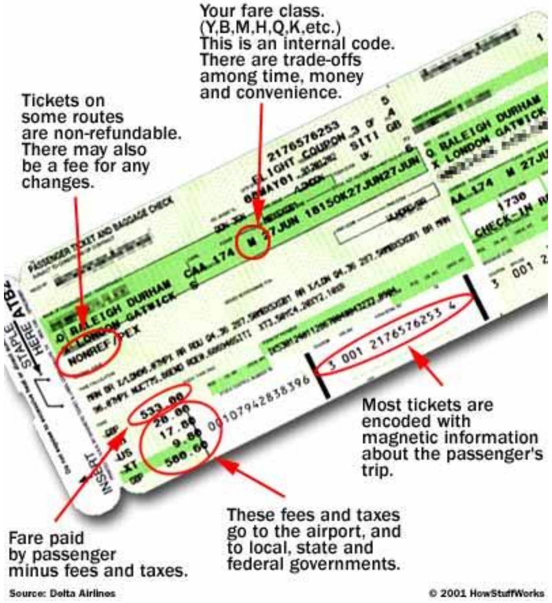
Formal Database Definition

- “A database has some source from which data are derived, some degree of interaction with events in the real world, and an audience that is actively interested in the contents of the database”
  - Source — logically coherent collection of data with inherent meaning
  - Interaction — changes to the miniworld are reflected in the database
  - Audience — has an intended users and applications in which these users are interested

Databases Everywhere

amazon.com

ebay



Your fare class. (Y,B,M,H,Q,K,etc.) This is an internal code. There are trade-offs among time, money and convenience.

Tickets on some routes are non-refundable. There may also be a fee for any changes.

Most tickets are encoded with magnetic information about the passenger's trip.

Fare paid by passenger minus fees and taxes.

These fees and taxes go to the airport, and to local, state and federal governments.

Source: Delta Airlines

© 2001 HowStuffWorks

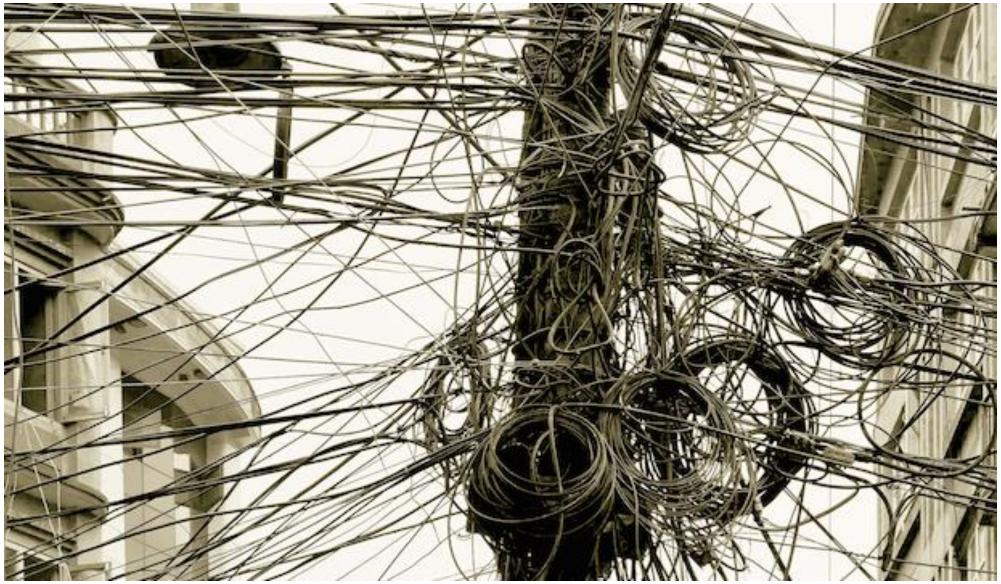
A database can be of any size and complexity

Types of Databases and Database Applications

- Traditional Applications:
  - Numeric and Textual Databases
- More Recent Applications:
  - Multimedia Databases
  - Geographic Information Systems (GIS)
  - Data Warehouses
  - Real-time and Active Databases
  - Many other applications

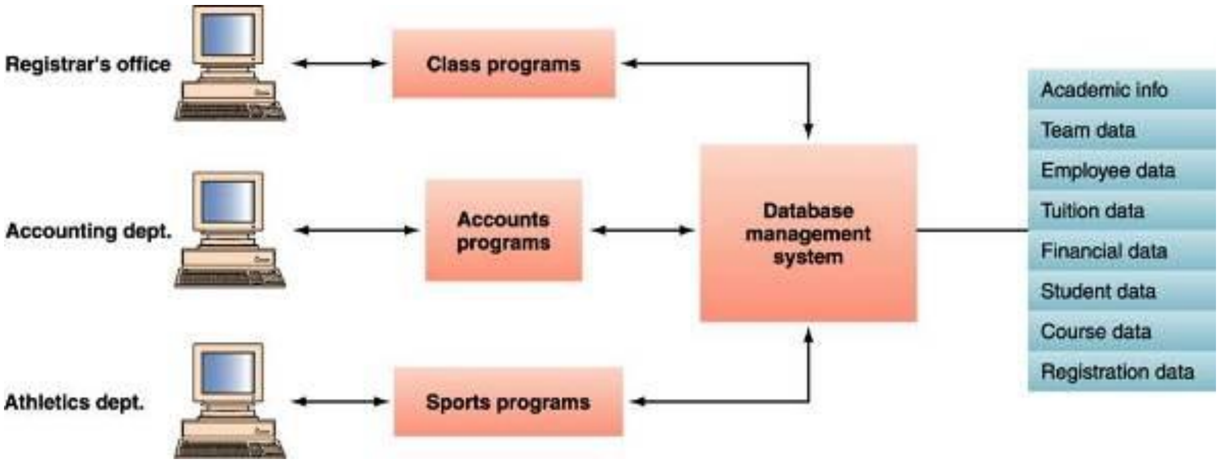
Database Management System (DBMS)

- Do we store all these information under one table?
- Will database be quick in getting the data or updating?
- Certainly Not! Chaos!



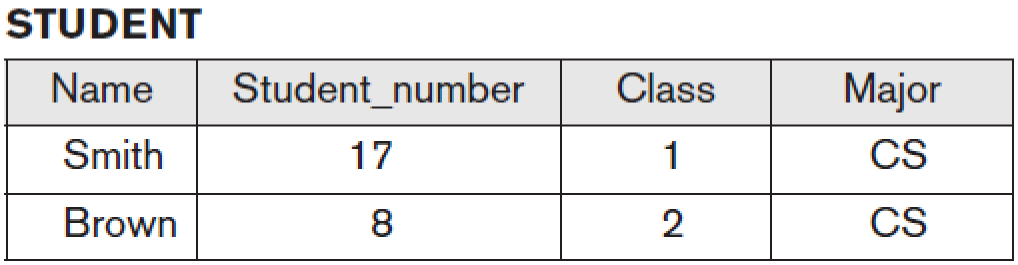
Database Management System (DBMS) (cont.)

- DBMS
  - It tells us how to divide related information into different tables and inter-relate them so that we can manipulate data easily and efficiently
  - A software system that facilitates the processes of *defining*, *constructing*, *manipulating* and *sharing* databases



Database Implementation (cont.)

- Defining a database
  - Data types
    - e.g., char, text, int, float, double, etc.
  - Structures
    - Logical data structure, physical data structure
  - Constraints
    - Specify some restrictions on valid data

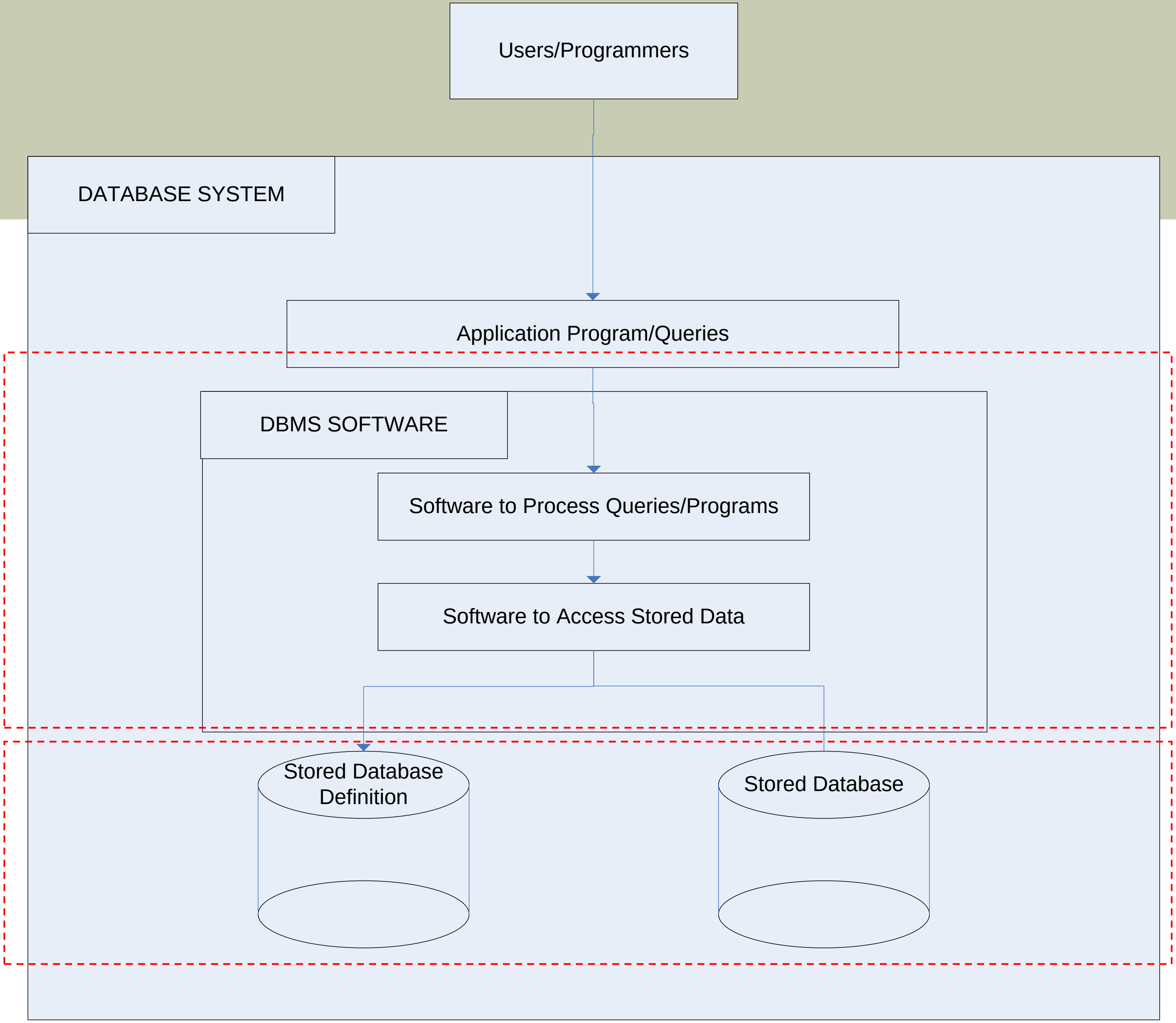


STUDENT

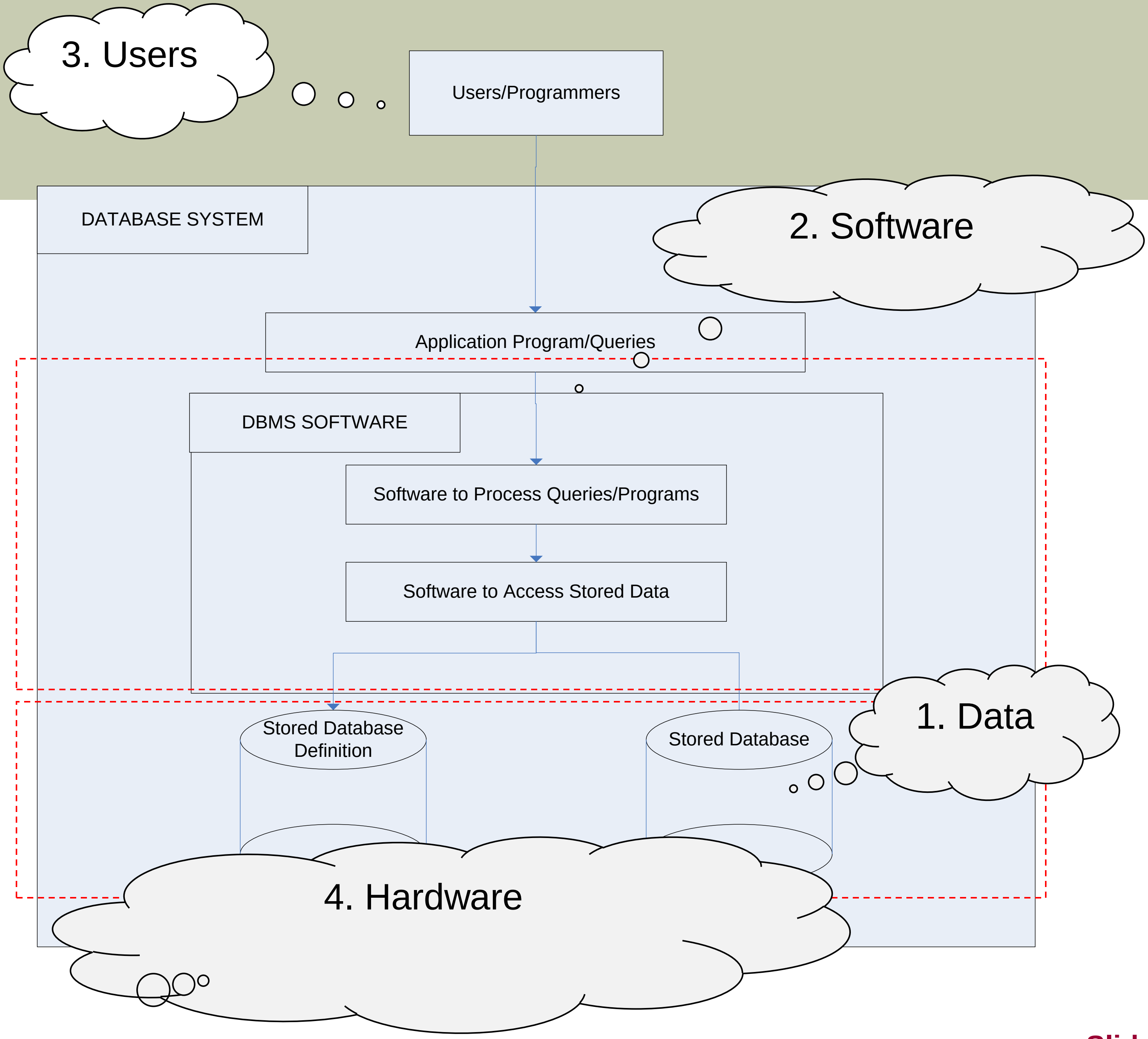
Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

Database Implementation (cont.)

- Constructing a database
  - Storing the data itself on a storage medium
- Manipulating a database
  - Querying
  - Updating
  - Generating reports
- Sharing a database
- Protecting a database
  - System protection
  - Security protection
- Maintaining a database
  - Allowing the system to evolve as requirements change over time


Users/Programmers
DATABASE SYSTEM
Application Program/Queries
DBMS SOFTWARE
Software to Process Queries/Programs
Software to Access Stored Data
Stored Database Definition
Stored Database

Database + DBMS == Database System



3. Users

Users/Programmers

DATABASE SYSTEM

2. Software

Application Program/Queries

DBMS SOFTWARE

Software to Process Queries/Programs

Software to Access Stored Data

Stored Database Definition

Stored Database

1. Data

4. Hardware

Summary of Basic Definitions

- Database:
  - A collection of related data.
- Data:
  - Known facts that can be recorded and have an implicit meaning.
- Mini-world:
  - Some part of the real world about which data is stored in a database. For example, student grades and transcripts at a university.
- Database Management System (DBMS):
  - A software package/ system to facilitate the creation and maintenance of a computerized database.
- Database System:
  - The DBMS software together with the data itself. Sometimes, the applications are also included.

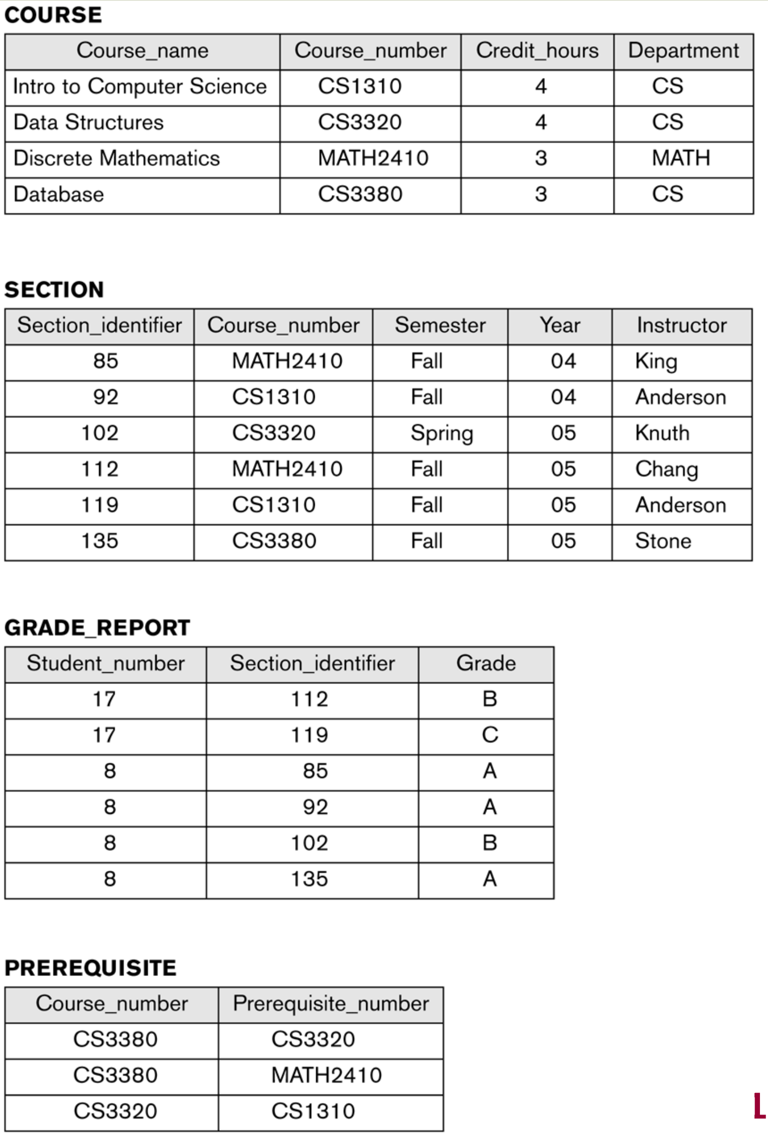
Example of a Database

- Mini-world for the example:
  - Part of a UNIVERSITY environment.
- Some mini-world *entities*:
  - STUDENTs
  - COURSEs
  - SECTIONs (of COURSEs)
  - (academic) DEPARTMENTs
  - INSTRUCTORs

Database Structure and Sample Data Record

STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS



COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
92	CS1310	Fall	04	Anderson
102	CS3320	Spring	05	Knuth
112	MATH2410	Fall	05	Chang
119	CS1310	Fall	05	Anderson
135	CS3380	Fall	05	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

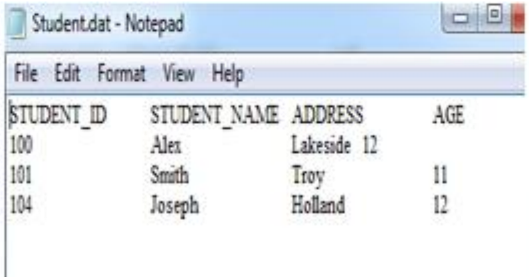
Data Relationship

- Records in the various files may be related.

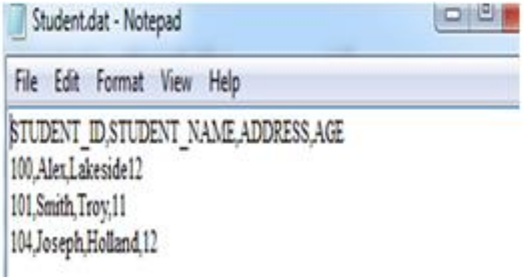
- Some mini-world *relationships*:
  - SECTIONs *are of specific* COURSEs
  - STUDENTs *take* SECTIONs
  - COURSEs *have prerequisite* COURSEs
  - INSTRUCTORs *teach* SECTIONs
  - COURSEs *are offered by* DEPARTMENTs
  - STUDENTs *major in* DEPARTMENTs

Database V.S. File

Search GSU students. Never ending task

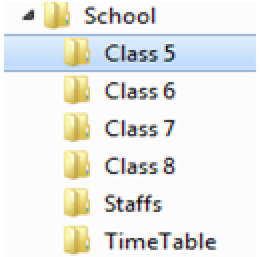


```
STUDENT_ID     STUDENT_NAME  ADDRESS        AGE
100            Alex          Lakeside  12
101            Smith         Troy           11
104            Joseph        Holland        12
```

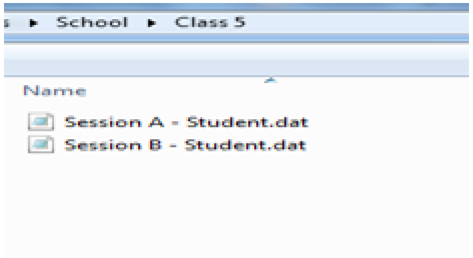


```
STUDENT_ID,STUDENT_NAME,ADDRESS,AGE
100,Alex,Lakeside12
101,Smith,Troy,11
104,Joseph,Holland,12
```

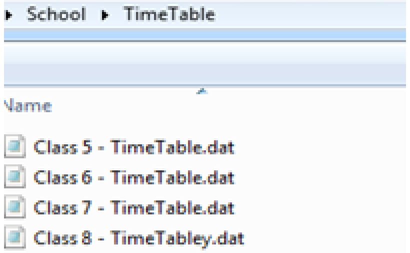
query/update data with C, C++, etc.



- School
  - Class 5
  - Class 6
  - Class 7
  - Class 8
  - Staffs
  - TimeTable



Session A - Student.dat
Session B - Student.dat



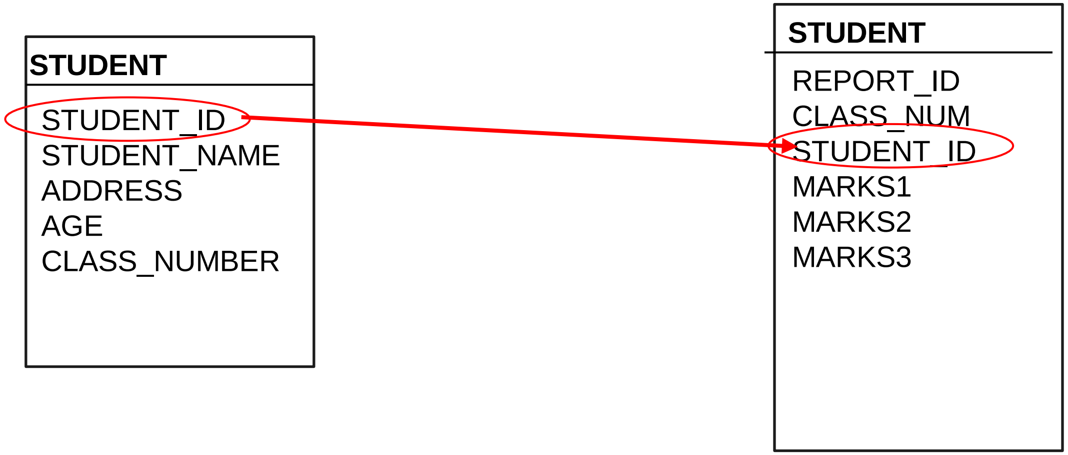
Class 5 - TimeTable.dat
Class 6 - TimeTable.dat
Class 7 - TimeTable.dat
Class 8 - TimeTabley.dat

All the files were grouped based on their categories

Database V.S. File (cont.)

Search GSU students. Never ending task

- As data grow, handling them with file becomes difficult
  - Data Mapping and Access.
    - No mapping between any two files, i.e.; any two dependent files are not linked



STUDENT

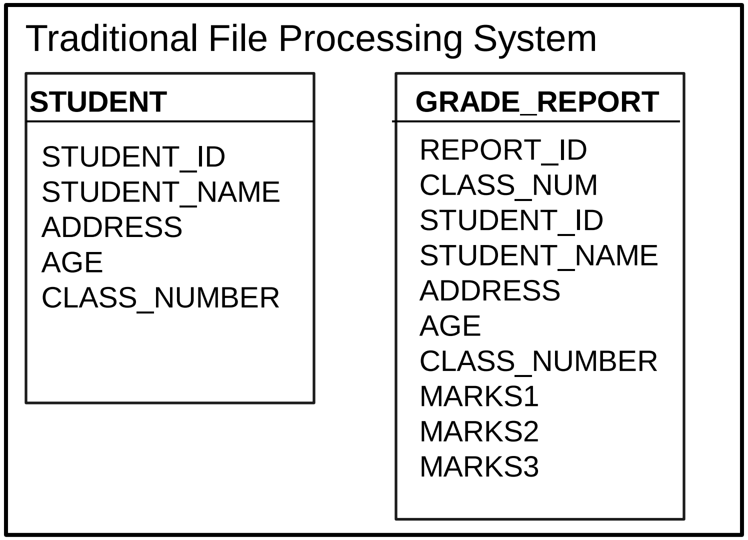
STUDENT_ID
STUDENT_NAME
ADDRESS
AGE
CLASS_NUMBER

STUDENT

REPORT_ID
CLASS_NUM
STUDENT_ID
MARKS1
MARKS2
MARKS3

Database V.S. File (cont.)

- As data grow, handling them with file becomes difficult
  - Data Redundancy
    - No methods to validate the insertion of duplicate data in file
    - Wastes storage space and effort used to maintain the common data up-to-date.



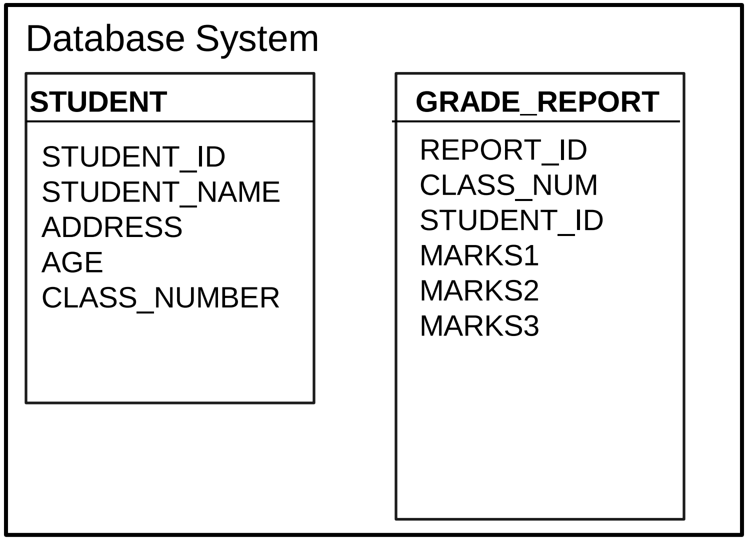
Traditional File Processing System

STUDENT

STUDENT_ID
STUDENT_NAME
ADDRESS
AGE
CLASS_NUMBER

GRADE_REPORT

REPORT_ID
CLASS_NUM
STUDENT_ID
STUDENT_NAME
ADDRESS
AGE
CLASS_NUMBER
MARKS1
MARKS2
MARKS3



Database System

STUDENT

STUDENT_ID
STUDENT_NAME
ADDRESS
AGE
CLASS_NUMBER

GRADE_REPORT

REPORT_ID
CLASS_NUM
STUDENT_ID
MARKS1
MARKS2
MARKS3

Database V.S. File (cont.)

Search GSU students. Never ending task

- As data grow, handling them with file becomes difficult
  - Data Dependency
    - Any change in data affect all the places where this file is being used

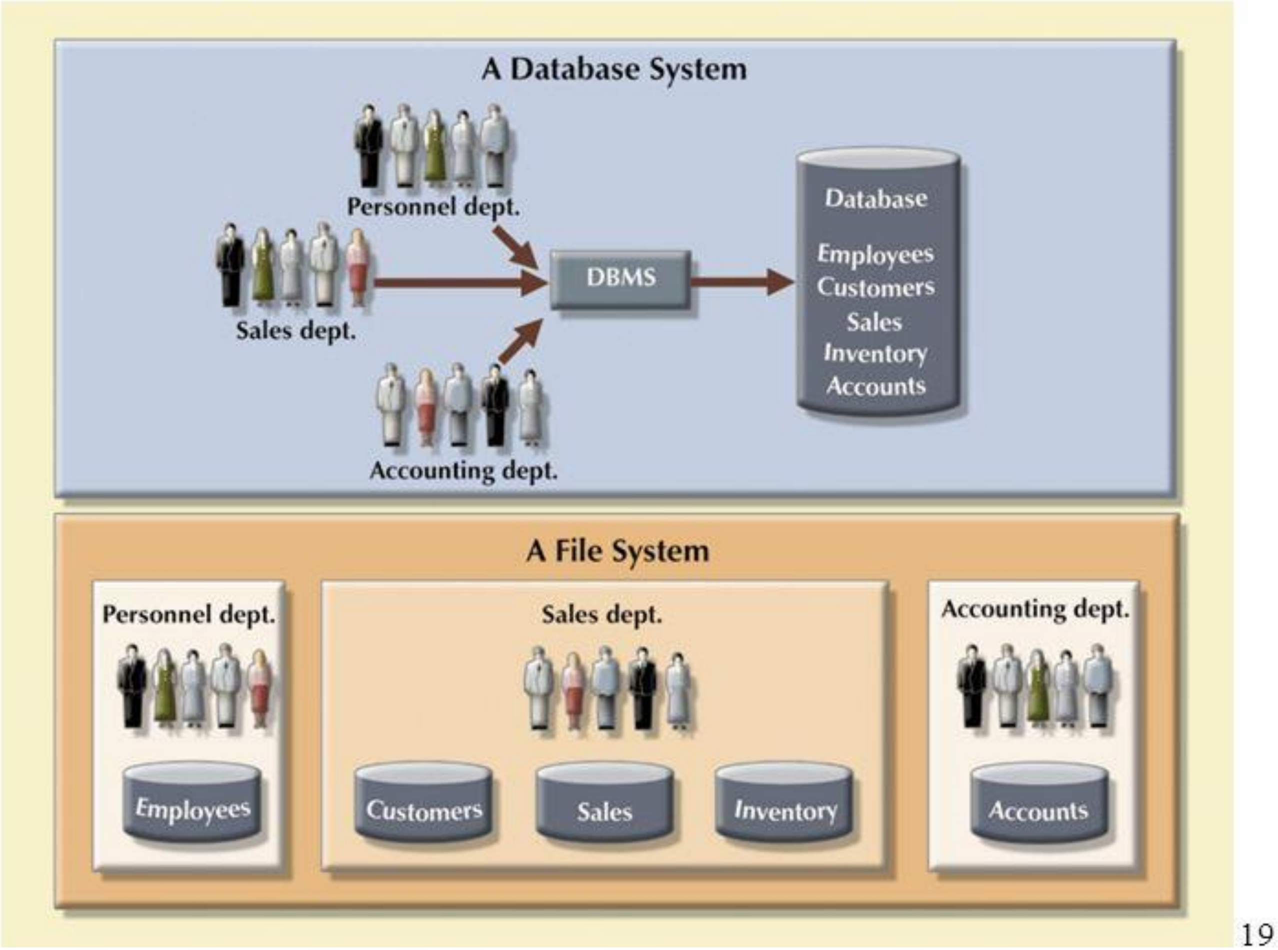
Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

Program: what's the Major of Smith?

Name	Student_number	Class	Grade
Smith	17	1	A
Brown	8	2	A

Program: what's the Major of Smith?

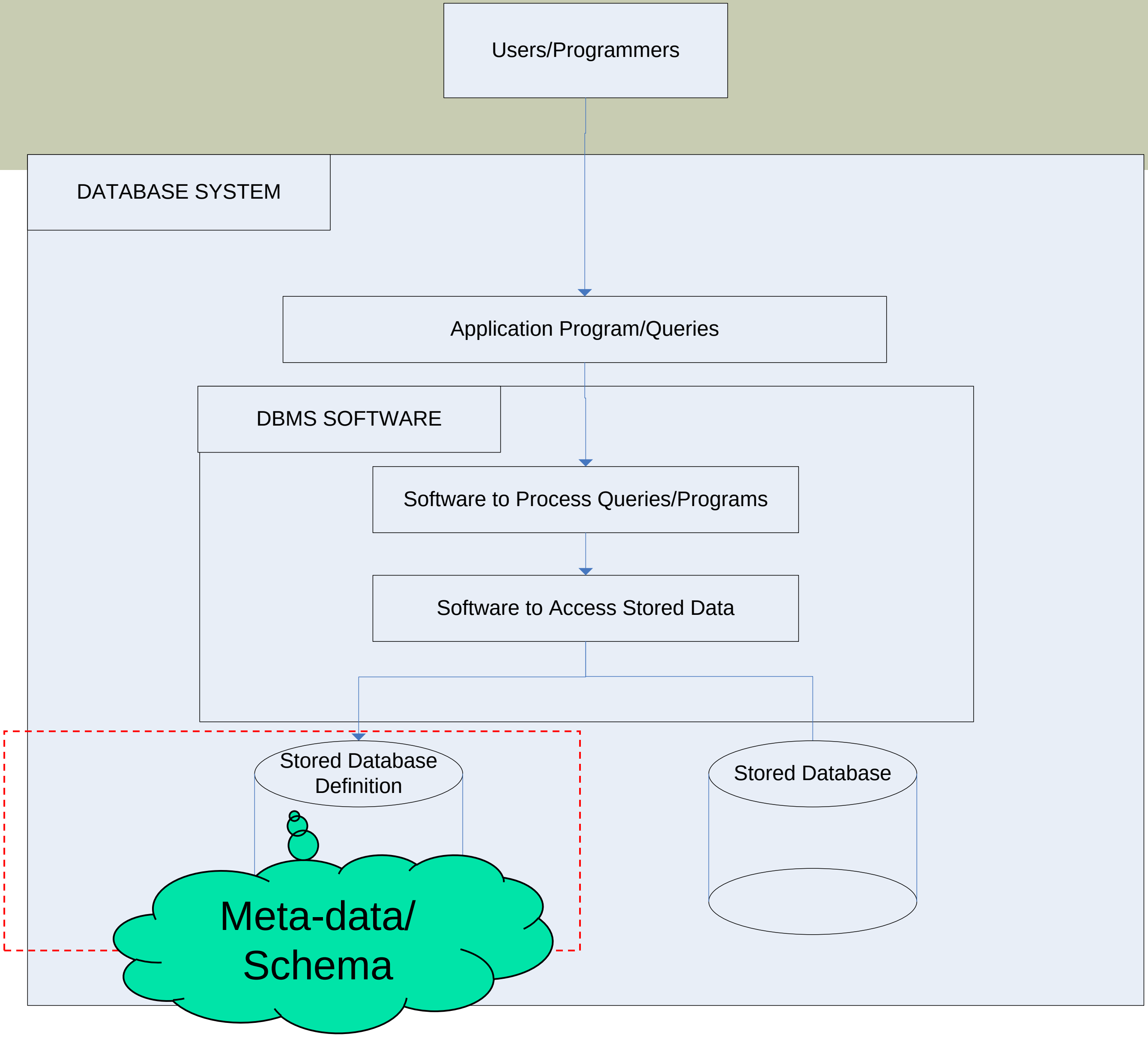
Database vs. File Systems



In the database approach, a single repository maintains data that is defined once and then is accessed by various users repeatedly through queries and application programs

Main Characteristics of the Database Approach

- 1) Self-describing nature of a database system:
  - A DBMS **catalog** stores the description of a particular database (e.g. data structures, types, storage format and constraints)
  - The description is called **meta-data**.
  - This allows the DBMS software to work with different database applications.


Users/Programmers
DATABASE SYSTEM
Application Program/Queries
DBMS SOFTWARE
Software to Process Queries/Programs
Software to Access Stored Data
Stored Database Definition
Stored Database
Meta-data/ Schema

Self-Describing Nature

Example of file approach

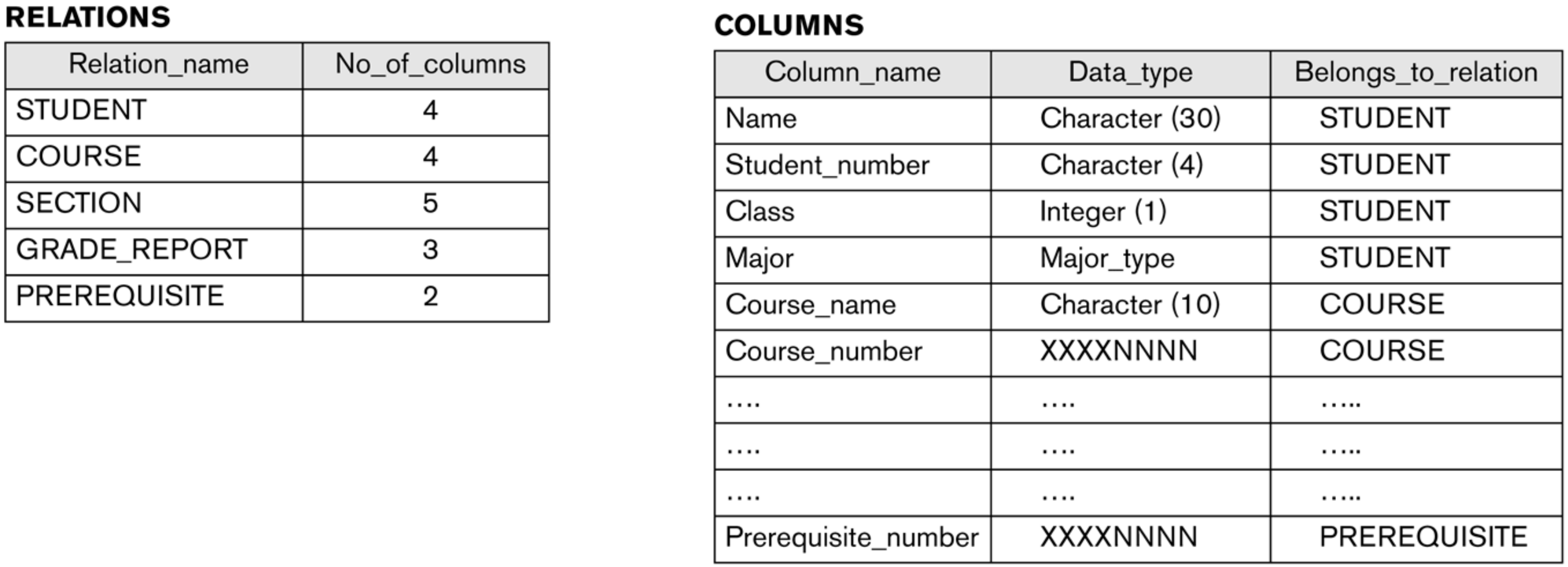
```
class Rectangle {
    int width, height;
  public:
    void set_values (int,int);
    int area() {return width*height;}
};

void Rectangle::set_values (int x, int y) {
  width = x;
  height = y;
}

int main () {
  Rectangle rect;
  rect.set_values (3,4);
  cout << "area: " << rect.area();
  return 0;
}
```

- Data definition is typically part of the application programs themselves
- Programs are constrained to work with *one specific database*

Example of a simplified database catalog

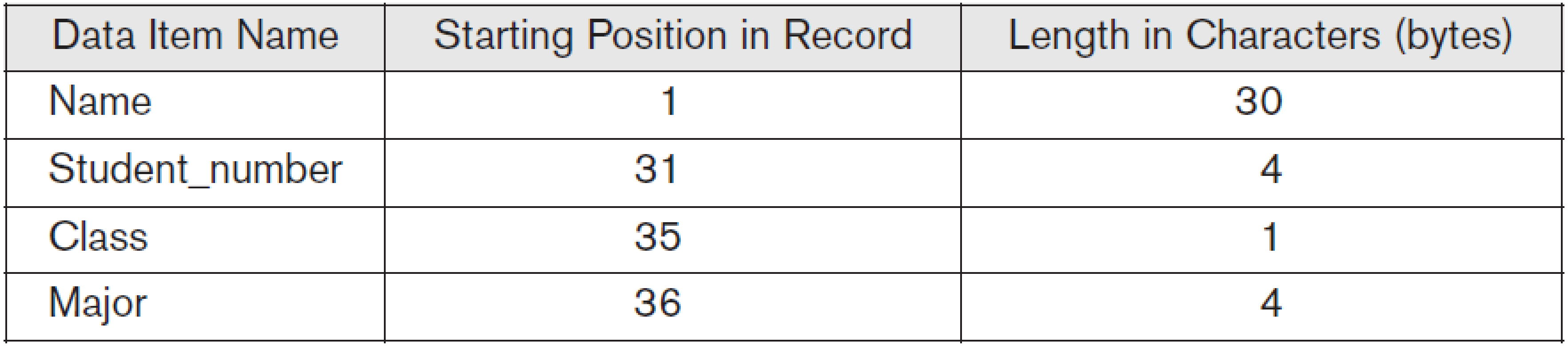


RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

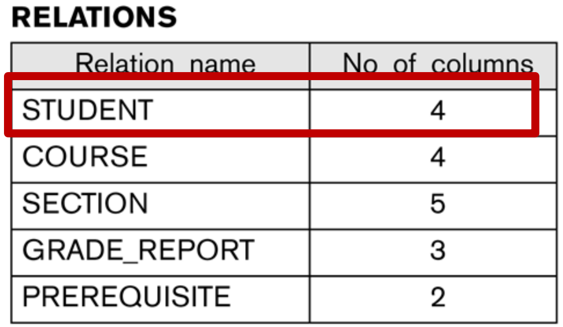
COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
….	….	…..
….	….	…..
….	….	…..
Prerequisite_number	XXXXNNNN	PREREQUISITE



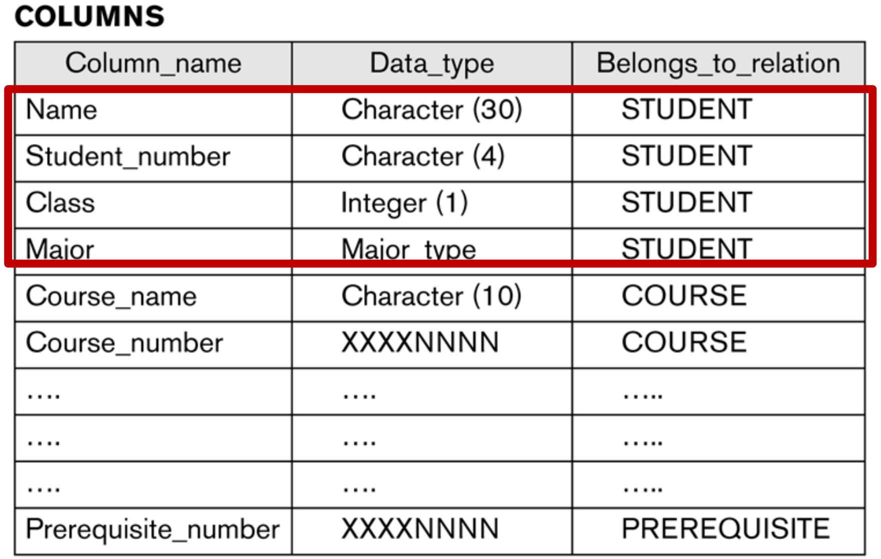
Data Item Name	Starting Position in Record	Length in Characters (bytes)
Name	1	30
Student_number	31	4
Class	35	1
Major	36	4

Example of a simplified database catalog (cont.)



RELATIONS

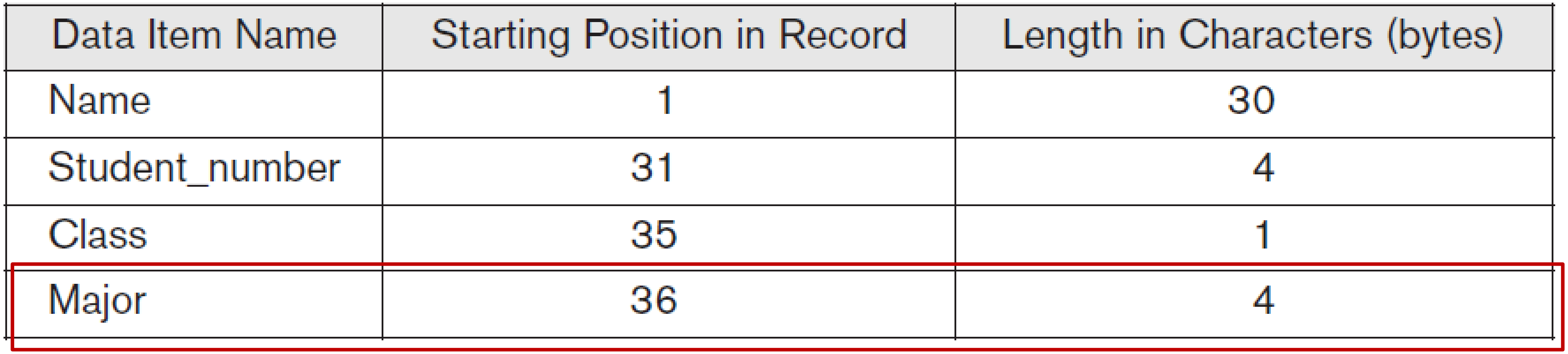
Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2



COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
….	….	…..
….	….	…..
….	….	…..
Prerequisite_number	XXXXNNNN	PREREQUISITE

Query: what is the Major of ‘Smith’

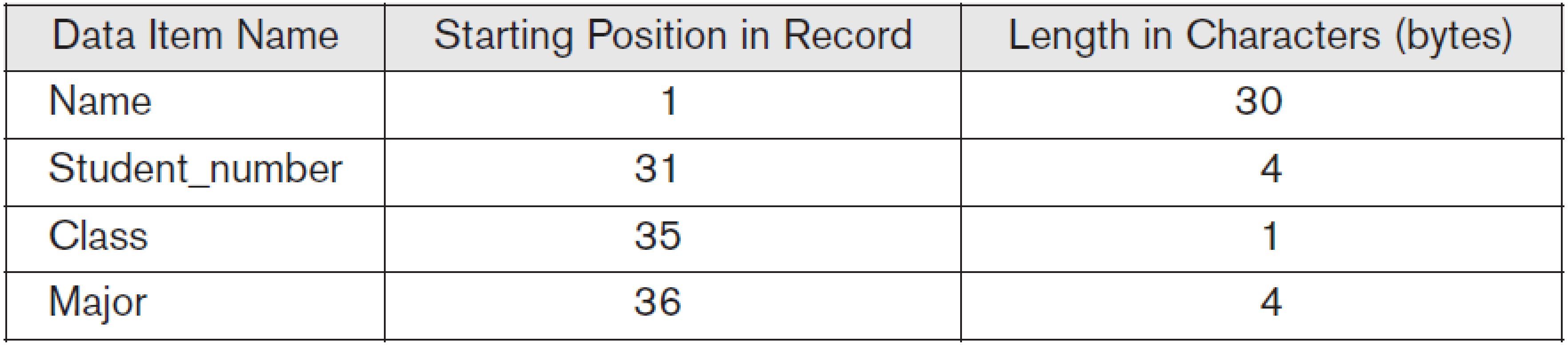


Data Item Name	Starting Position in Record	Length in Characters (bytes)
Name	1	30
Student_number	31	4
Class	35	1
Major	36	4

Main Characteristics of the Database Approach (cont.)

- 2) Insulation between programs and data:
  - Structure of data is stored in catalog separately from the access programs
  - Called **program-data independence**.
  - Allows changing data structures and storage organization without having to change the DBMS access programs.

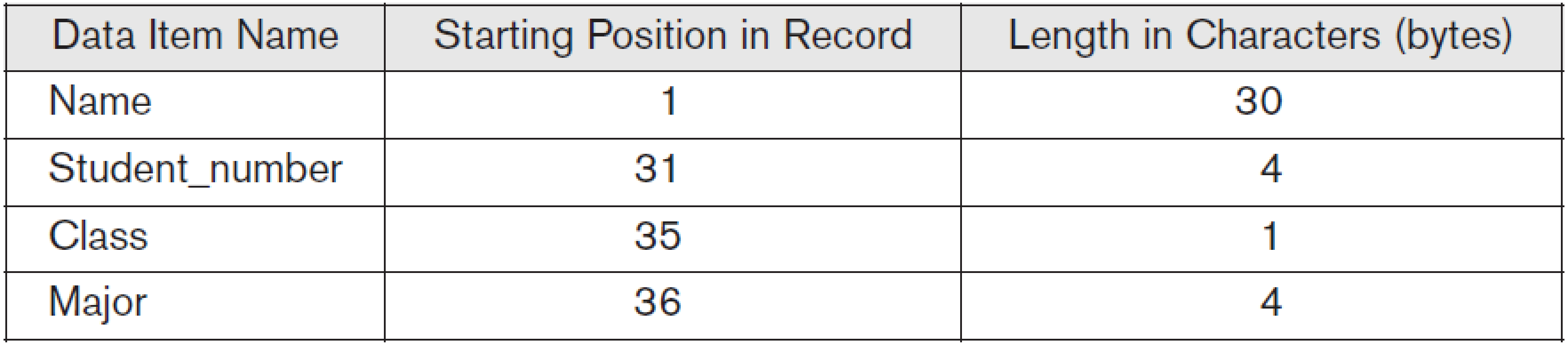
- An example



Data Item Name	Starting Position in Record	Length in Characters (bytes)
Name	1	30
Student_number	31	4
Class	35	1
Major	36	4

Main Characteristics of the Database Approach (cont.)

- 3) Data Abstraction:
  - A **conceptual representation** of data that does not include many of the details
    - How the data is stored
    - How the operations are implemented
  - A typical users is not concerned with the details, such as

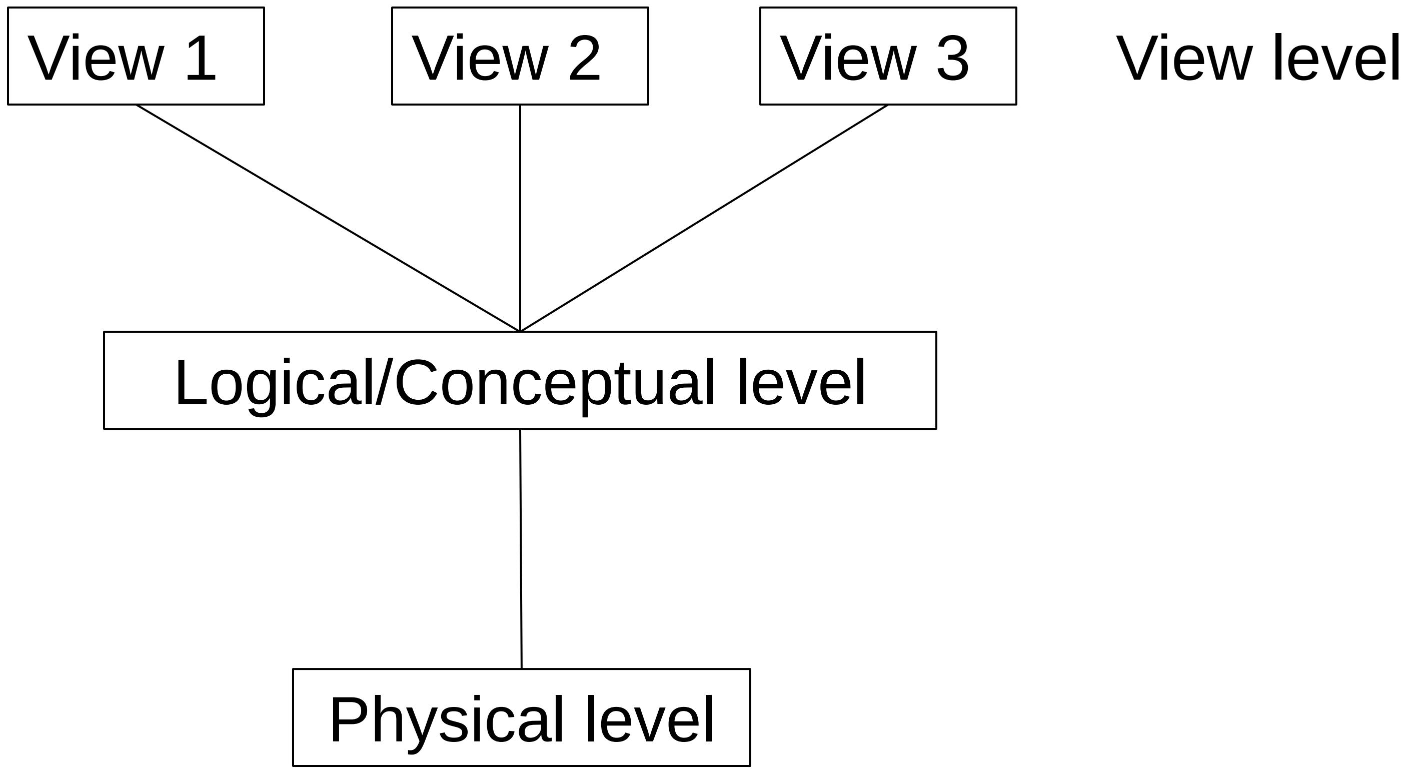


Data Item Name	Starting Position in Record	Length in Characters (bytes)
Name	1	30
Student_number	31	4
Class	35	1
Major	36	4

A conceptual representation of the STUDENT records is shown in Figure 1.2

Main Characteristics of the Database Approach (cont.)

- 3) Data Abstraction (cont.):



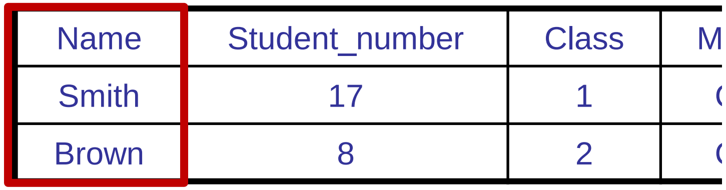
Three levels of data abstraction

Main Characteristics of the Database Approach (cont.)

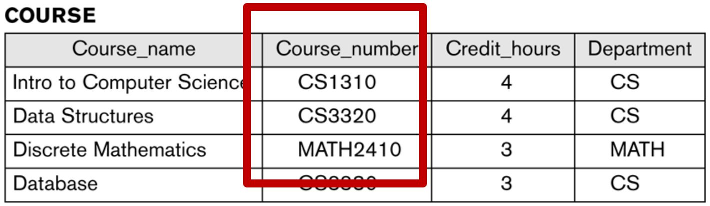
- 4) Support of multiple views of the data:
  - Each user may see a different view of the database, which describes **only** the data of interest to that user.

- Example:
  - Interested only in accessing and printing the transcript of each student **(See next figure)**
  - Interested only in checking that students have taken all the prerequisites of registered course **(See next figure)**

Main Characteristics of the Database Approach (cont.)

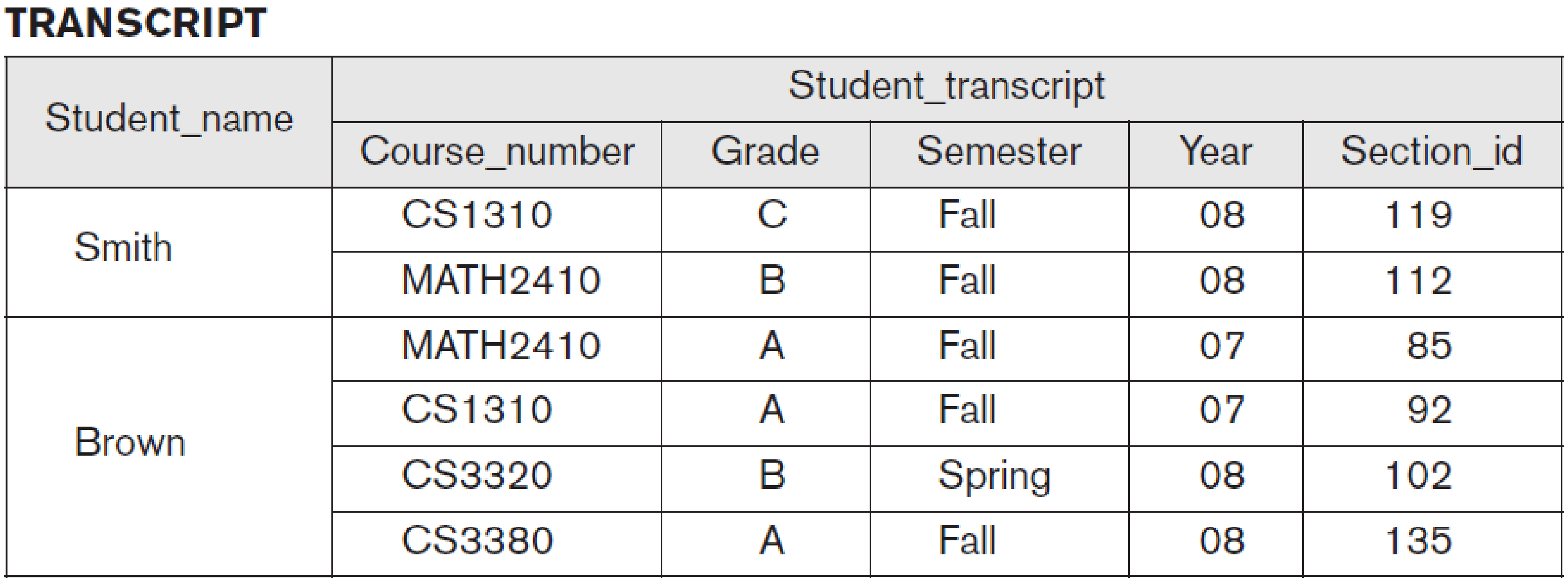


Name	Student_number	Class	M
Smith	17	1	(
Brown	8	2	(



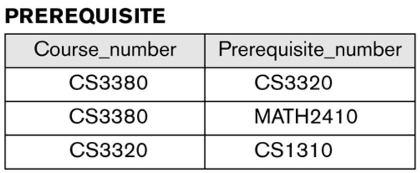
COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS



TRANSCRIPT

Student_name	Student_transcript				
	Course_number	Grade	Semester	Year	Section_id
Smith	CS1310	C	Fall	08	119
	MATH2410	B	Fall	08	112
Brown	MATH2410	A	Fall	07	85
	CS1310	A	Fall	07	92
	CS3320	B	Spring	08	102
	CS3380	A	Fall	08	135

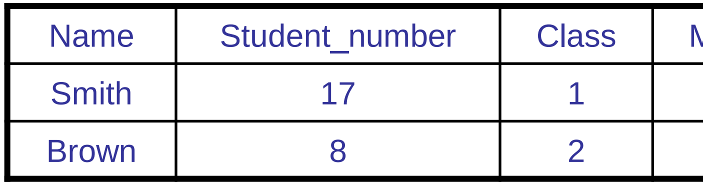


PREREQUISITE

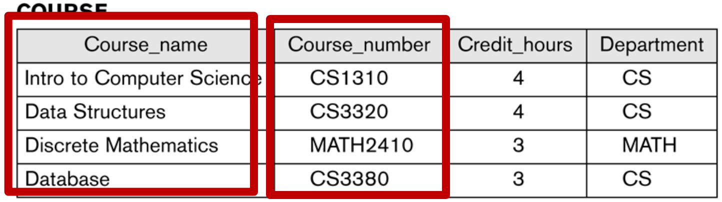
Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Figure 1.2
A database that stores student and course information.

Main Characteristics of the Database Approach (cont.)

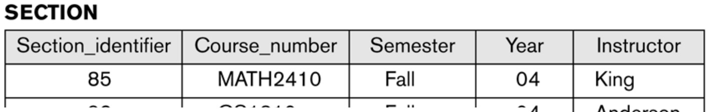


Name	Student_number	Class	M
Smith	17	1	
Brown	8	2	



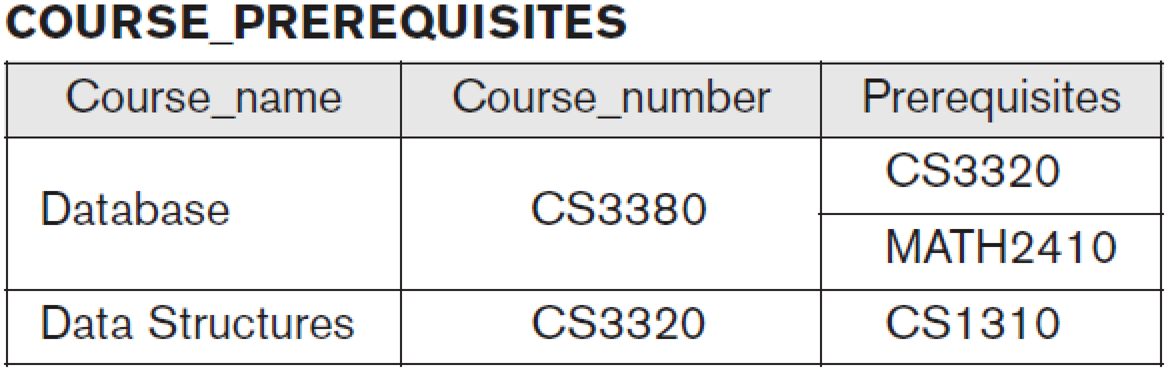
COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS



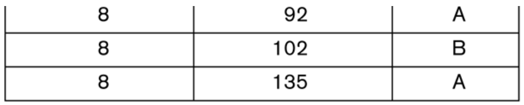
SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
			4	Anderson
			5	Knuth
			5	Chang
			5	Anderson
			5	Stone



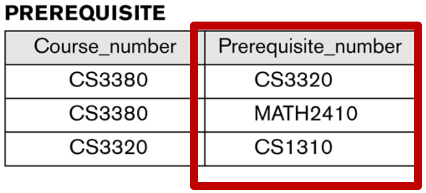
COURSE_PREREQUISITES

Course_name	Course_number	Prerequisites
Database	CS3380	CS3320 MATH2410
Data Structures	CS3320	CS1310



8	92	A
8	102	B
8	135	A

The COURSE_PREREQUISITES view



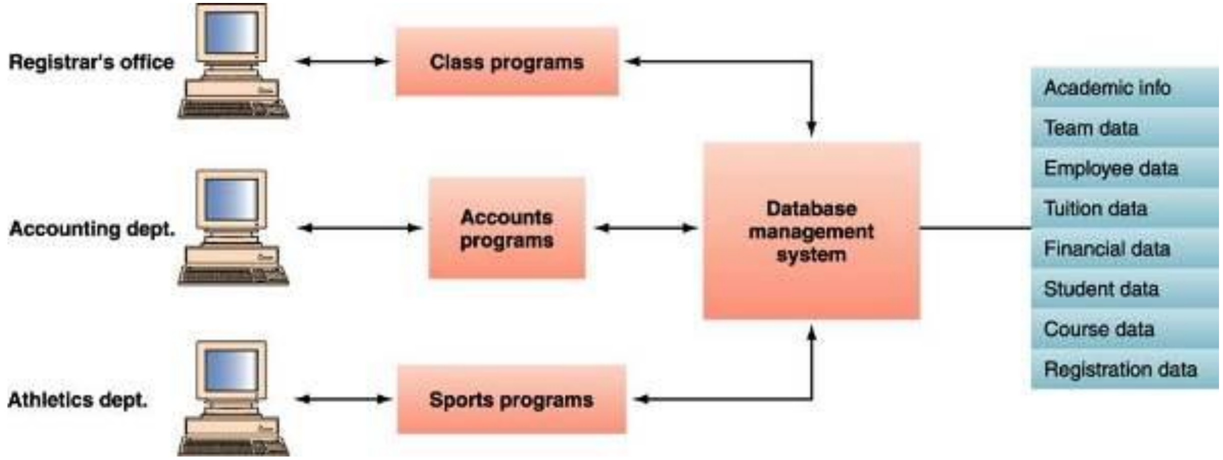
PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Figure 1.2
A database that stores student and course information.

Main Characteristics of the Database Approach (cont.)

- 5) Sharing of data and multi-user transaction processing:
  - Allowing a set of **concurrent users** to retrieve from and to update the database.
  - The DBMS must include **concurrency control** software



Advantages of Using the Database Approach

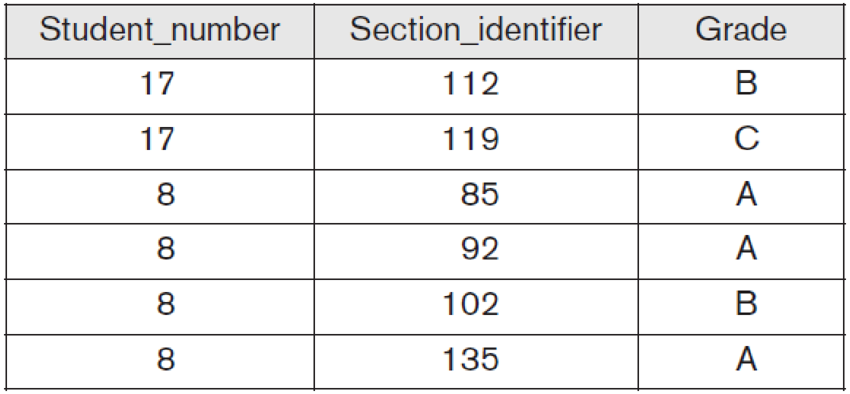
- Controlling redundancy in data storage and in development and maintenance efforts.
  - Sharing of data among multiple users.

- Redundancy leads to several problems:
  - Duplication of effort
  - Storage space is wasted
  - Data inconsistent
  - Updates are applied

Advantages of Using the Database Approach (cont.)

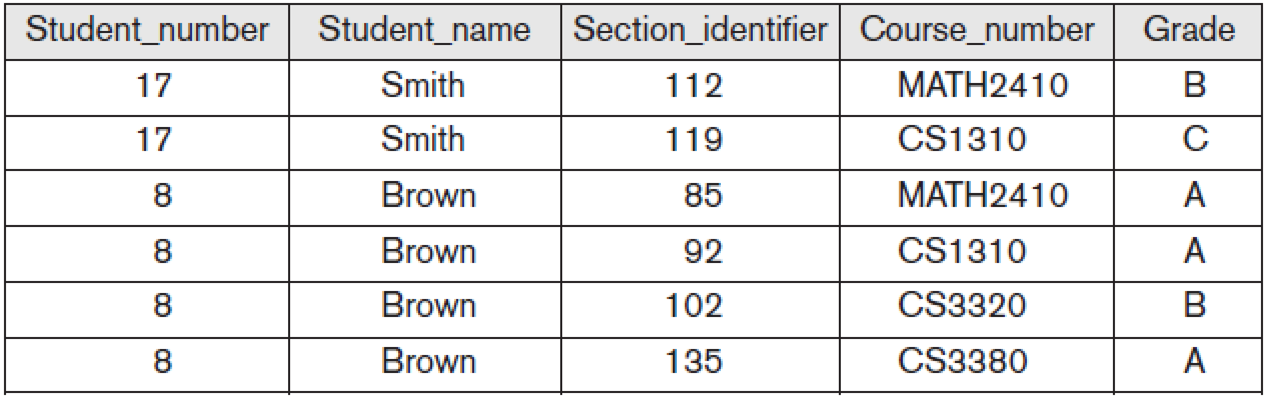
- Controlled redundancy (cont.)
  - To improve the performance of queries.
  - Automatically checking to avoid uncontrolled redundancy

GRADE_REPORT



Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

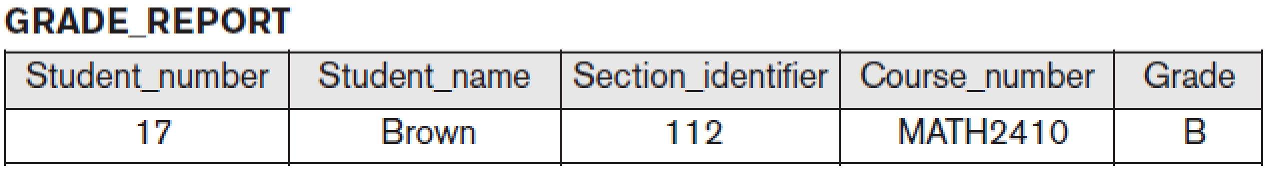
GRADE_REPORT



Student_number	Student_name	Section_identifier	Course_number	Grade
17	Smith	112	MATH2410	B
17	Smith	119	CS1310	C
8	Brown	85	MATH2410	A
8	Brown	92	CS1310	A
8	Brown	102	CS3320	B
8	Brown	135	CS3380	A

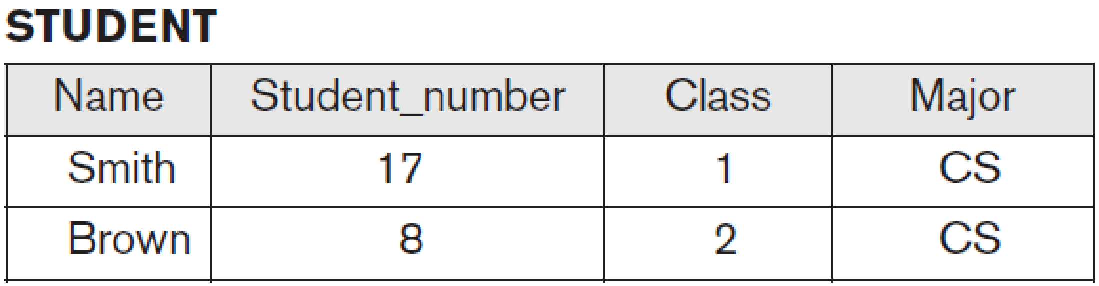
Advantages of Using the Database Approach (cont.)

- Controlled redundancy (cont.)
  - To improve the performance of queries.
  - Automatically checking



GRADE_REPORT

Student_number	Student_name	Section_identifier	Course_number	Grade
17	Brown	112	MATH2410	B



STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

Which part is inconsistent?

Advantages of Using the Database Approach (cont.)

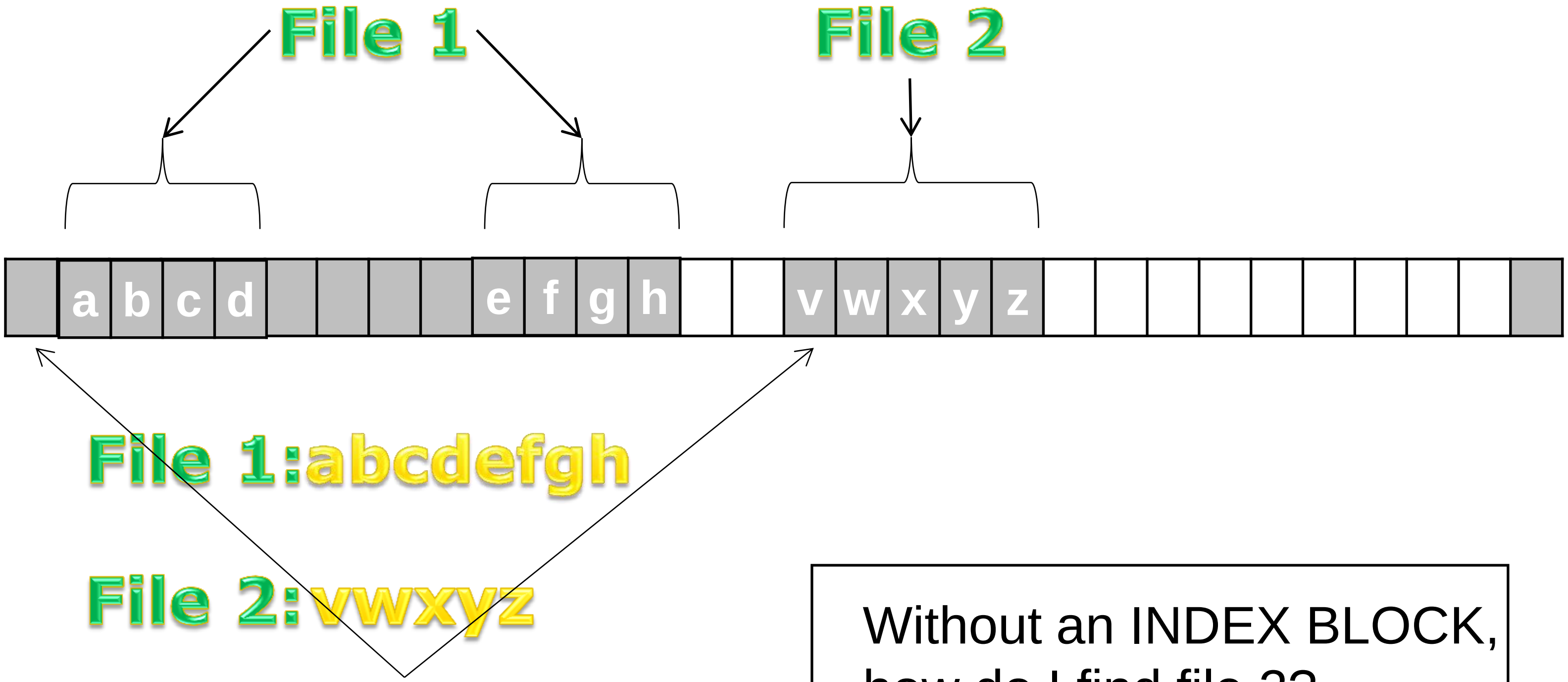
- Restricting unauthorized access to data
  - Security and authorization subsystem
  - For example, privileged software

Advantages of Using the Database Approach (cont.)

- Providing Storage Structures (e.g. indexes) and Search Techniques for efficient Query Processing
  - Efficiently executing queries and updates
    - Indexes based on tree data structures or hash data structures
    - DBMS often has a buffering or caching module
    - Query processing and optimization

- Example:
  - Indexed Storage means that as well as the records in the file, an index block is created, with pointers to each individual file.

Indexed Storage



File 1:abcdefgh

File 2:vwxyz

Is this file 2?

YES

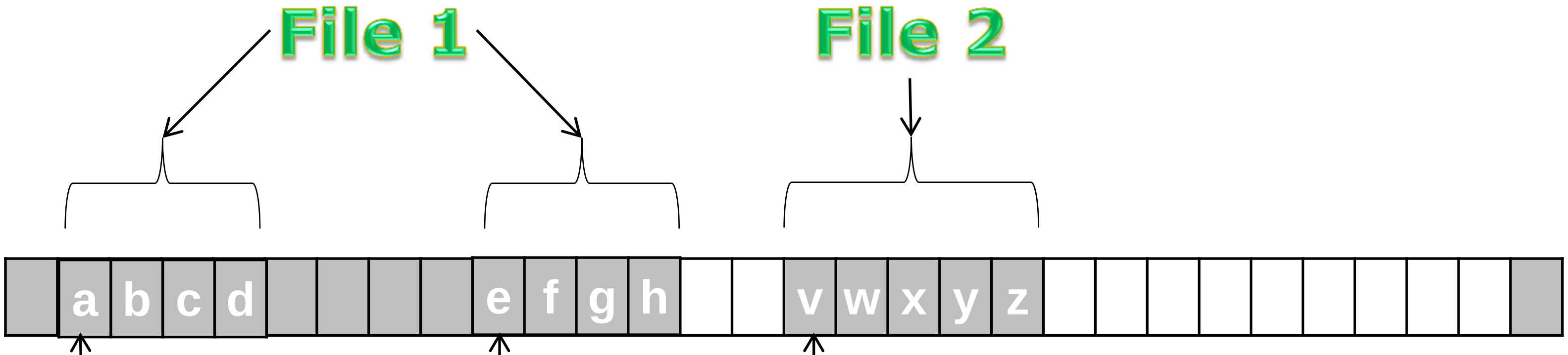
NO

Without an INDEX BLOCK, how do I find file 2?

We do a sequential search.

So how does it work with an INDEX BLOCK?

Indexed Storage



File 1:abcdefgh

File 2:vwxyz

INDEX BLOCK:

File	Address	Size	Next
File 1	1	4	9
File 1	9	4	-
File 2	15	5	-

When not to use a DBMS

- It may be more desirable to develop customized database applications when:
  - Simple, well-defined database applications that are not expected to change at all
  - Stringent, real-time requirements for some application programs that may not be met because of DBMS overhead
  - Embedded systems with limited storage capacity, where a general-purpose DBMS would not fit
  - No multiple-user access to data

Summary

- Types of Databases and Database Applications
- Basic Definitions
- Typical DBMS Functionality
- Example of a Database (UNIVERSITY)
- Main Characteristics of the Database Approach
- Advantages of Using the Database Approach
- When Not to Use Databases