CS105.3
Database Management Systems

By: Chalani Oruthotaarachchi

Delivery Schedule

Course Duration

15 weeks

12 Lectures + 2 Revision + 1 Mid Exam

Practical

Mandatory 1 × 2 hour practical per week

Tutorials

1 × 1 hour tutorial per week (we'll notify you)

Answers will be discussed ONLY during the Tutorial class.

Evaluation Criteria

Formal Exam	60%	(3 hour paper)
In course Assignments	40%	(Quizzes, Assignments, Tutorials)

Recommended Texts

Fundamentals of Database Systems' by Elmasri/Navathe

'Database Systems: A practical approach to design, implementation and management' by Connolly and Begg

Road Map

- DB Concepts
- Introduction to DBMS
- Database Architecture
- Data Models
- Database Design Methodology
  - Conceptual Model
  - Logical Model
  - Physical Model

Road Map

- Entity Relationship Diagrams
  - Entities
  - Relations
  - Attributes
  - Mapping Conceptual model into Relational Schema
- Data Normalization
  - INF
  - 2NF
  - 3NF
- **SQL**

Data vs. Information…

Data is generated manually and automatically

Data is used (consumed) by end-users and different applications

Audio

Text

Files

Numbers

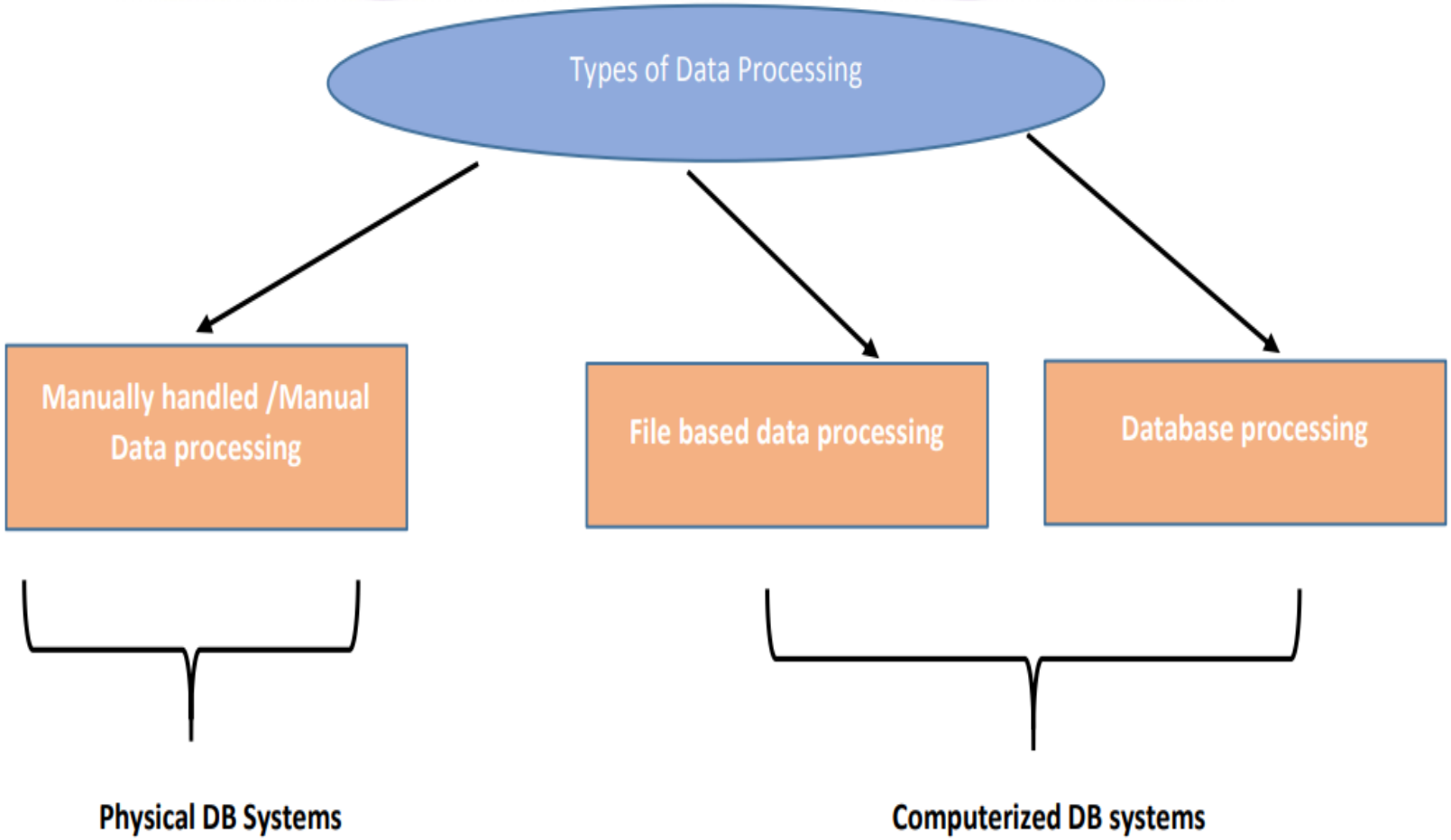
Video

Amounts

Images

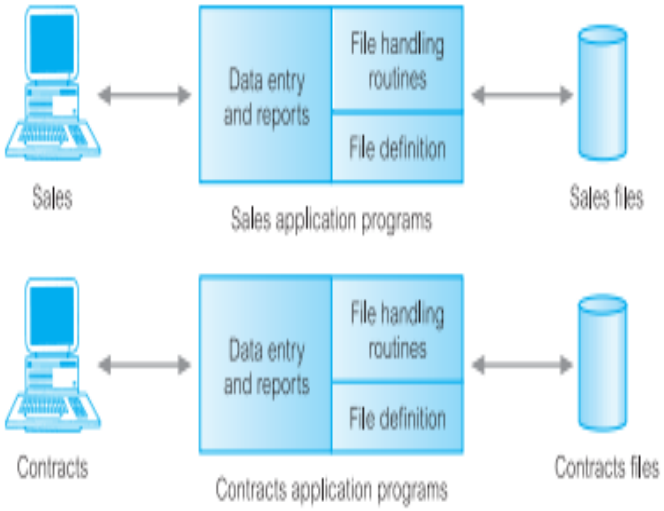
Dates

If You Have Data, Do You Need a Database to store them ???

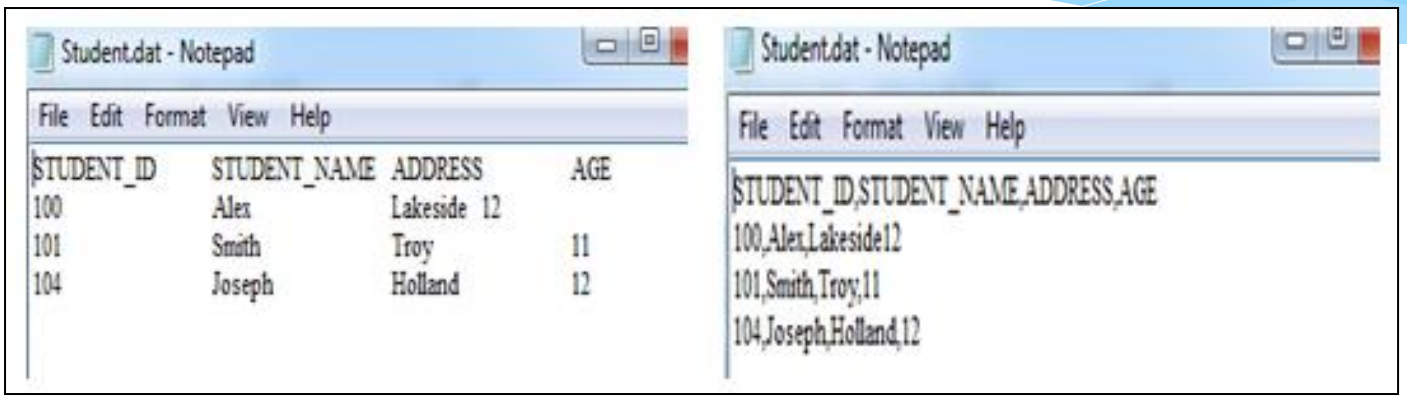

Types of Data Processing
Manually handled /Manual Data processing
File based data processing
Database processing
Physical DB Systems
Computerized DB systems

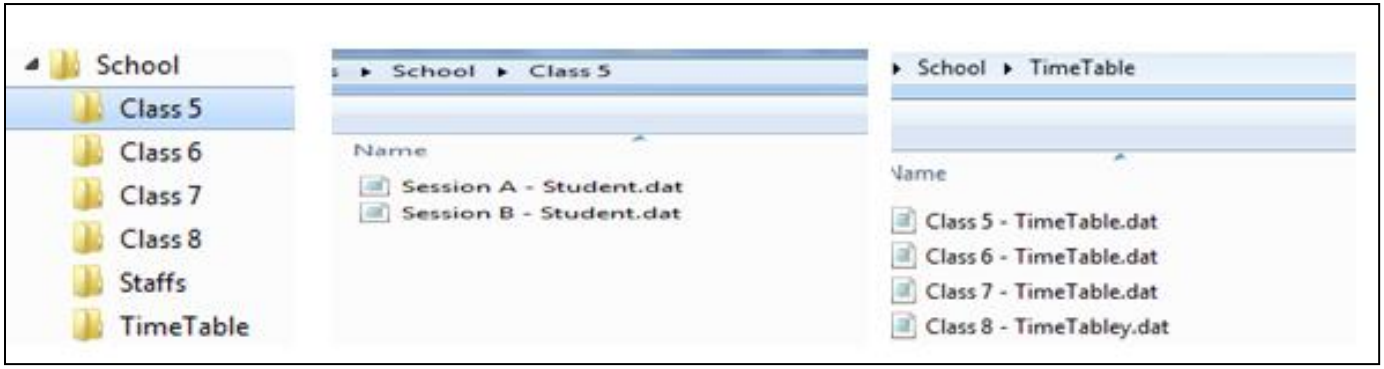
File based systems:

- Data is stored in files
- Each file has a specific format
- Programs that use these files depend on knowledge about that format



File Processing Systems





What are the problems that you may find in the file-processing approach?

Disadvantages of file processing

1. Limited data Mapping and Access
2. Data Redundancy
3. Data Dependence
4. Data inconsistency
5. Data Isolation
6. Security is less
7. Integrity is limited
8. Concurrent Access is limited

Database Concepts

* The need of database systems arose in the early 1960s in response to the traditional file processing system.

* In the database approach, the data is stored in the form of files, and a number of application programs are written by programmers to **add, modify, delete, and retrieve** data to and from appropriate files.

Things to be remembered…

* A database..
  * Is a miniworld.
  * Is a logically coherent collection of data with some inherent meaning.
  * Is designed, built, and populated with data for a specific purpose.

Example of a simple database

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

Figure 1.2
A database that stores student and course information.

Examples of Applications of Database concept

* Purchases from the supermarket
* Purchases using your credit card
* Booking a holiday at the travel agents
* Using the local library
* Taking out insurance
* Using the Internet
* Studying at university

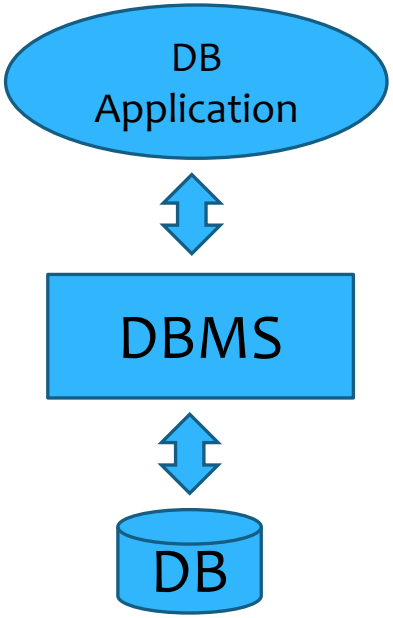
Types of Databases

* **Traditional applications:**
  * Numeric and textual databases
* **More recent applications:**
  * Multimedia databases
  * Geographic Information Systems (GIS)
  * Biological and genome databases
  * Data warehouses
  * Mobile databases
  * Real-time and active databases

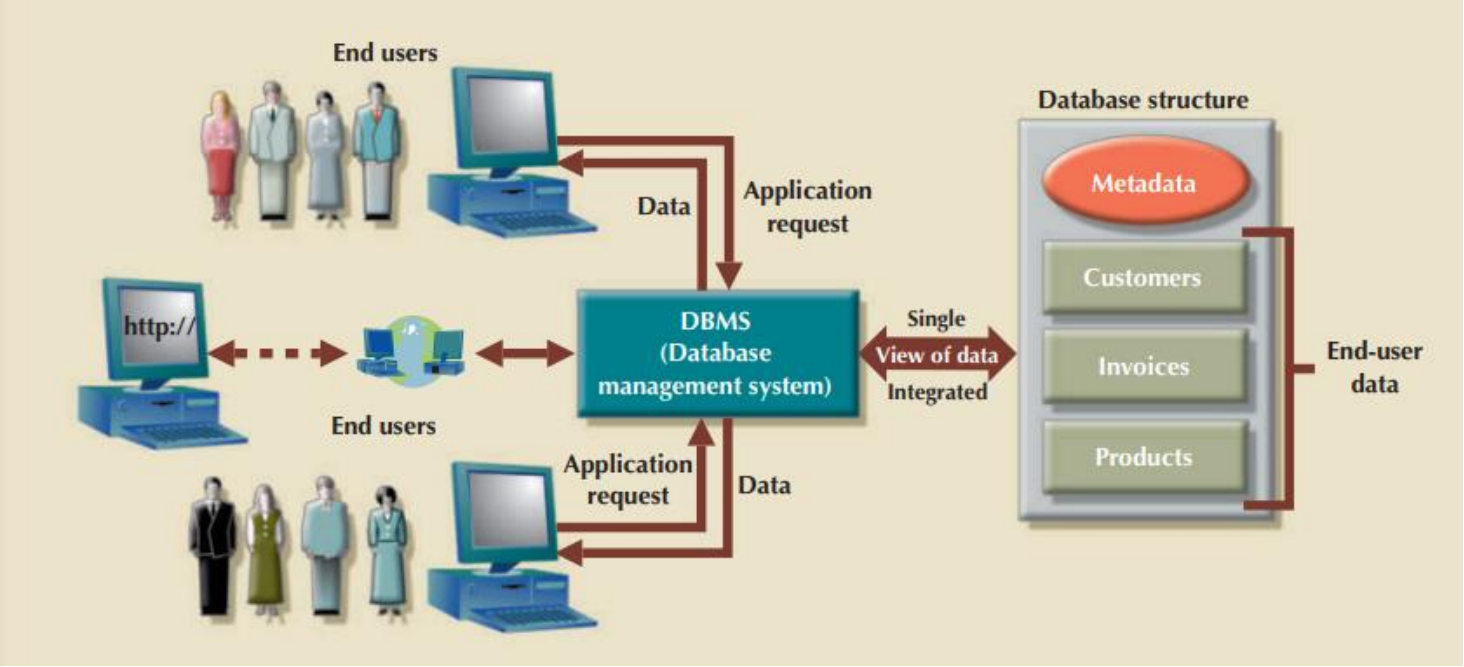
Database vs DBMS

* What is a database?
  * Collection of data related in some way.
  * Student database/University/School database

* What is a Database Management System?
  * A Program to manage large Databases providing reliable and efficient access to data
  * General-purpose software system that facilitates the processes of *defining*, *constructing*, *manipulating*, and *sharing* databases among various users and applications.
  * Not required always



* The DBMS manages the interaction between the end user and the database



Typical DBMS Functionality

* ***Define*** a particular database in terms of its data types, structures, and constraints
* ***Construct*** or Load the initial database contents on a secondary storage medium
* ***Manipulating*** the database:
  * Retrieval: Querying, generating reports
  * Modification: Insertions, deletions and updates to its content
  * Accessing the database through Web applications
* ***Processing*** **and** ***Sharing*** by a set of concurrent users and application programs

Main Characteristics of the DBMS Approach

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

* **Insulation between programs and data:**
  * Called **program-data independence**.
  * Allows changing data structures and storage organization without having to change the DBMS access programs.

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

Note: Major_type is defined as an enumerared type with all known majors. XXXXNNNN is used to define a type with four alpha characters followed by four digits

Figure 1.3
An example of a database catalog for the database in Figure 1.2.

Main Characteristics of the DBMS Approach (continued)

* **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.

* **Sharing of data and multi-user transaction processing:**
  * Allowing a set of **concurrent users** to retrieve from and to update the database.
  * *Recovery* subsystem ensures each completed transaction has its effect permanently recorded in the database
  * **OLTP** (Online Transaction Processing) is a major part of database applications. This allows hundreds of concurrent transactions to execute per second.

What are the advantages of using the DBMS approach?

Advantages of Using the DBMS Approach

* Controlling redundancy in data storage and in development and maintenance efforts.
  * Sharing of data among multiple users.
* Restricting unauthorized access to data.
* Providing backup and recovery services.
* Providing multiple interfaces to different classes of users.
* Representing complex relationships among data.
* Enforcing integrity constraints on the database.
* Drawing inferences and actions from the stored data using deductive and active rules

When Not to Use DBMS

- Main inhibitors (costs) of using a DBMS:
  - High initial investment and the possible need for additional hardware
  - Overhead for providing generality, security, concurrency control, recovery, and integrity functions
- When a DBMS may be unnecessary:
  - If the database and applications are simple, well-defined, and not expected to change
  - If access to data by multiple users is not required
- When a DBMS may be infeasible:
  - In embedded systems where a general-purpose DBMS may not fit in available storage

Recent Developments (1)

* **Social Networks** started capturing a lot of information about people and about communications among people-posts, tweets, photos, videos in systems such as:
  - Facebook
  - Twitter
  - Linked-In

* All of the above constitutes data

* **Search Engines**, Google, Bing, Yahoo: collect their own repository of web pages for searching purposes

Recent Developments (2)

* New technologies are emerging from the so-called non-SQL, non-database software vendors to manage vast amounts of data generated on the web:
  * **Big data** storage systems involving large clusters of distributed computers (Chapter 25)
  * **NOSQL** (Non-SQL, Not Only SQL) systems (Chapter 24)

* A large amount of data now resides on the "cloud" which means it is in huge data centers using thousands of machines.

Thank You