Information Management II

CS3041 – CS4D2a – 4CSLL1

Introduction

Prof. Vincent Wade

Vincent.Wade@scss.tcd.ie

Course Overview

- 12 Week Course
  - Thursday 16:00-18:00
  - Wednesday 14:00-15:00

- Approx. 28 Lectures and 5 Tutorials

Course Overview

- Online Course
  - SQL
- Assessment
  - Exam
  - Continuous Assessment
    - Project Work
    - Online Course

Course Outline

- System Engineering:
  - State of the Art in Database Technology; Transaction Processing; Concurrency Control; Metadata Representation; Recovery; Database Security; Web Databases and Emerging Database Technologies.
- Information Design:
  - Relational Modelling; Functional Dependency Modelling; Normalisation; Implementation of Databases and Database Applications.

Course Layout (week 1 – 5)

1. Introduction to Databases
2. Database Architectures
3. Database Models
4. Relational Algebra for Data Manipulation
5. Designing Databases: Functional Dependency
6. Designing Databases: The Entity Relationship (ER) approach
7. Designing Databases: Mapping from ER to Relations

Recommended Reading

- Database Systems: Models, Languages, Design and Application Programming
  - Elmasri & Navathe
  - ISBN-10: 0132144980 | ISBN-13: 9780132144988 | Edition: 6th edition

Information Management II

1. Introduction to Databases

CS4D2a – 4CSLL1 – CS3041

Vincent Wade

Vincent.wade@scss.tcd.ie

What is Data?

- *Data* is any information that you want to store and refer to again. Data can be:
  - Text
  - Numbers
  - Dates
  - Images
  - Videos
  - Files
  - Any other types of information.
- For example, if you sell cakes, you can store the names, pictures, and recipes of your cakes, the prices and quantities of boxes and the dates of sales…..

What is a Database?

- An organised collection of Information, or Data…
  - "A database is a persistent collection of related data supporting several different applications within an organisation"
- Organised to:
  - model aspects of reality
  - in a way that supports processes that require this information
    - A collection of medical records in a Hospital
    - Finding records by a specific Doctor or Patient
  - mostly, to make the data more useful!

Metadata

- Metadata adds Context to Data

Metadata	***Data***
Student Number:	89041258
Name:	John Patrick Smith
Account Balance:	132.56

- Metadata can include:
  - data type, name of element, size, restrictions etc.
- Can be used at any level of aggregation

Database Management Systems

- Database Management System (DBMS)
- Goal of a DBMS is to simplify the storage of, and access to, data
- DBMS support:
  - Definition
  - Manipulation
  - Querying
- A DBMS can manage a single, or set of, DBs

DBMSs Provide…

- Efficient, reliable and secure management of large amounts of persistent data.
- Language(s) for defining the DB
  - *data definition language*
  - This data about data (e.g. student number is a seven digit number plus one check digit) is called *metadata*
- Languages for storing, retrieving and updating data in the DB
  - *data manipulation languages*

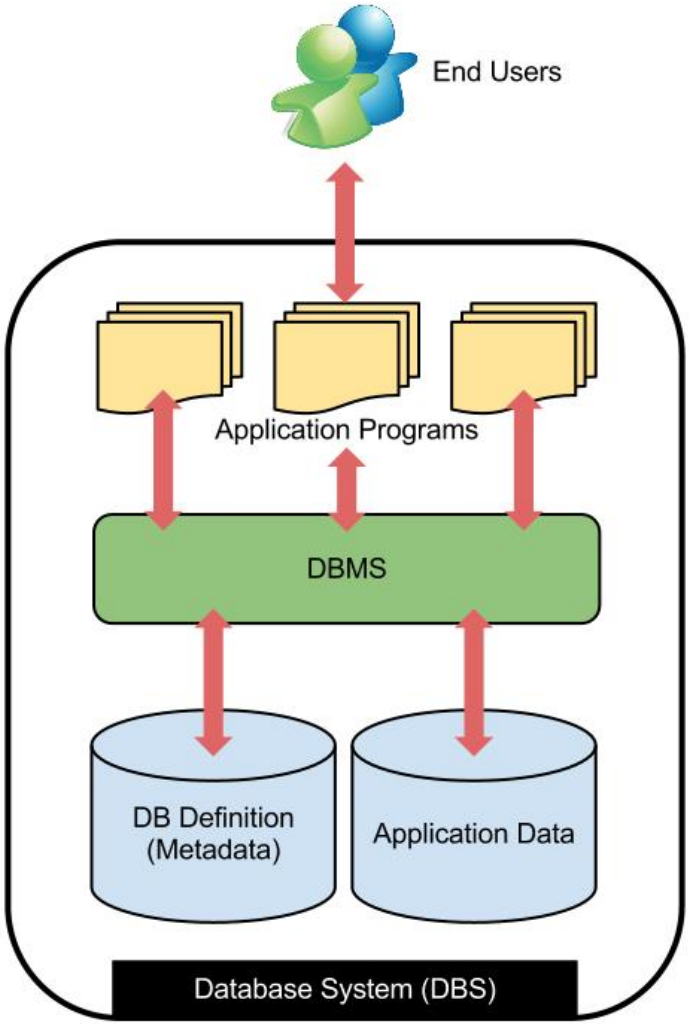
DBMS

- Well known DBMS:
  - Proprietary:

  - Open Source:

Database Systems

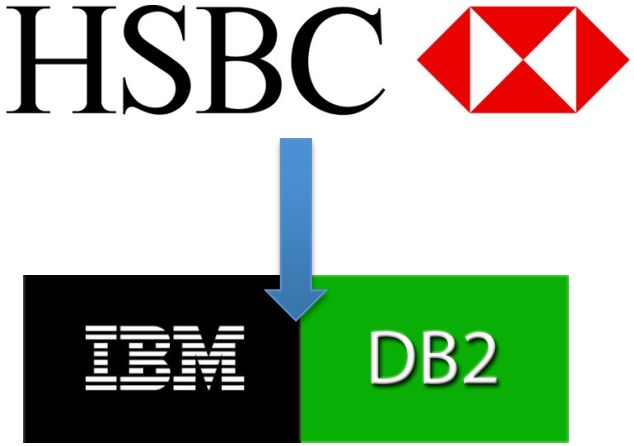
- Database System (DBS)
  - DBMS
  - DB
    - application data
    - associated metadata
  - Application programs
- Metadata and data are stored separately

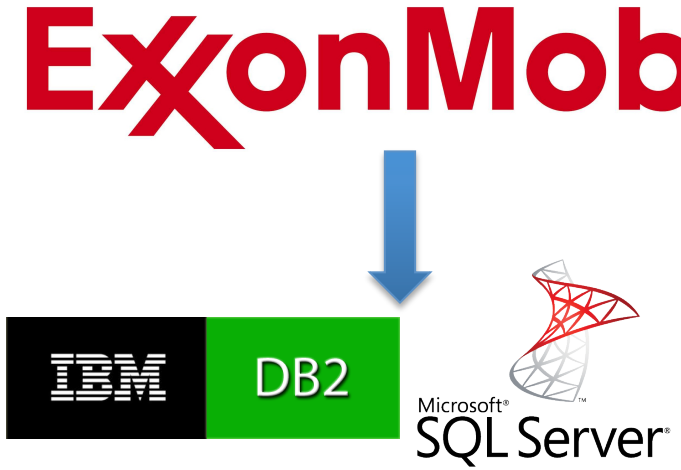


Why should I care?

- Ubiquity
- Software Market
  - roughly same size as OS market – approx. $20B annually.
- The majority of large corporations, web sites, scientific projects… all manage both day to day operations as well as business intelligence and data mining using databases

Why should I care?





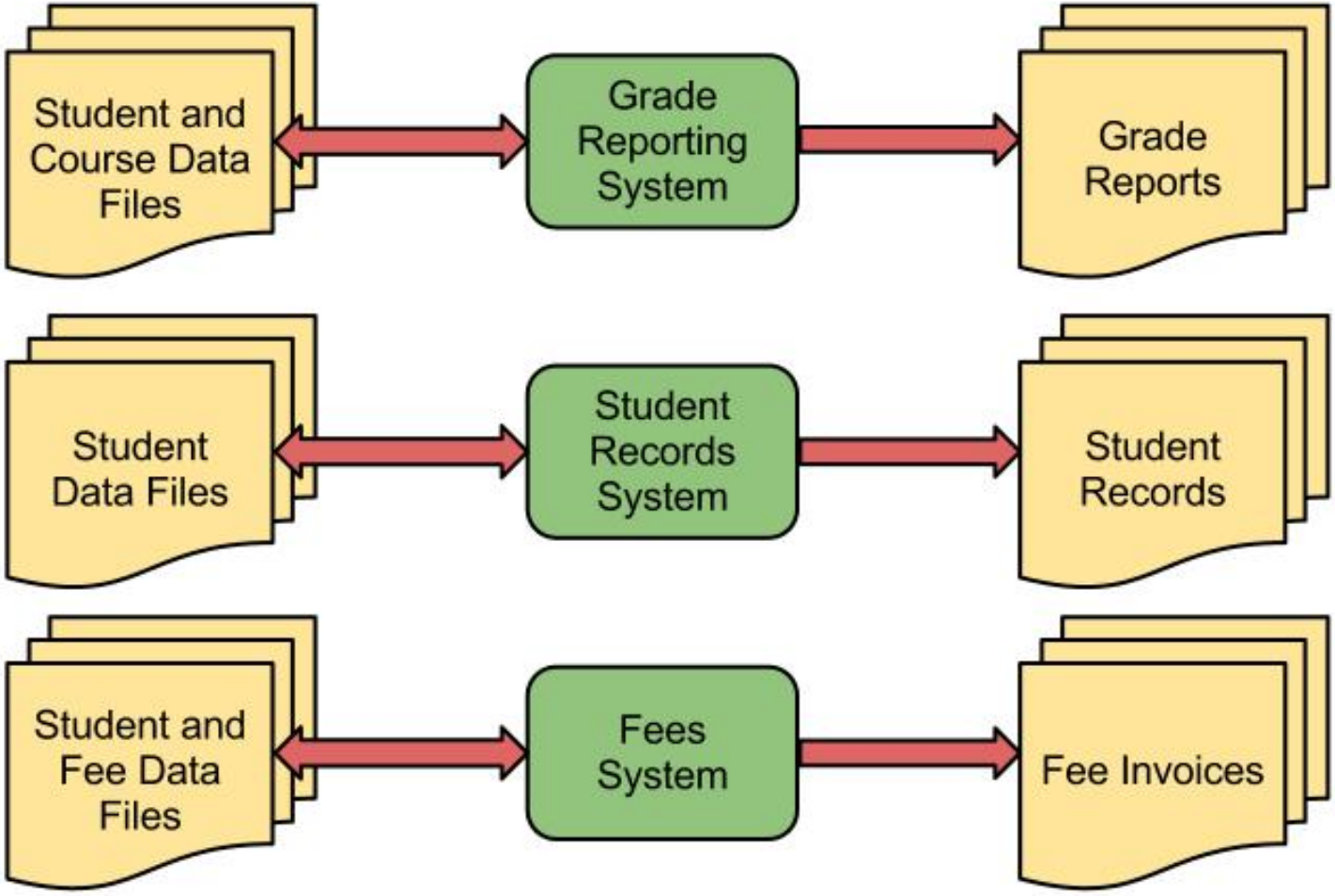
Why use a Database?

- Pre-DB era (up to end of 1960's) was characterised by file processing systems
- File systems offered
  - efficient, direct access to individual records
  - fast sequential processing
- Choice of file organisation technique was based on the needs of a particular application
- However, if multiple applications want to share data, this can give rise to wasteful duplication
  - Patient record application and Accounting application
  - Patient names, addresses, visit charges etc.

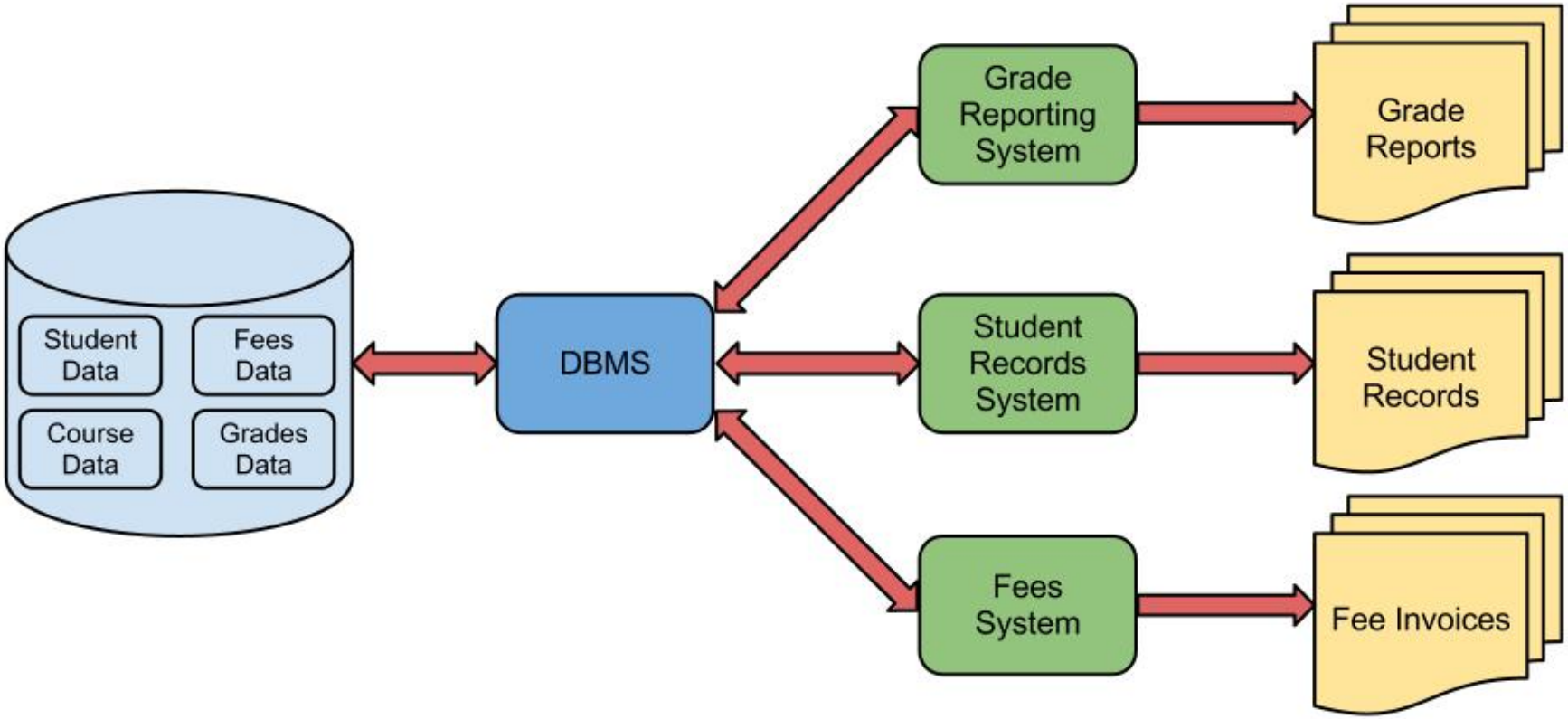
Why use a Database?

- Duplication of data
  - Wasteful of storage
  - Inefficient
  - Most importantly, leads to inconsistencies
- DB approach aims to eliminate such *redundancy* (data duplication)
- Data from all applications is integrated and stored once in the DB
- All applications access the same physical copy of the data

File-Based Approach



Database Approach



Disadvantages of File System

List five disadvantages of using a file system rather than a database ……

1.

2.

3.

4.

5.

How do Databases and DBMS address these problems?

Data Independence

- File-based systems are ***data dependent***
  - as the way in which data is viewed by an application and the way in which it is physically stored are built into the logic of the application program
- DBMS support ***logical data independence***
  - by allowing the view of the data to be changed and data added without affecting it's underlying organisation
- DBMS support ***physical data independence***
  - as they ***insulate*** the way in which data is viewed by the applications/users from the way in which it is physically stored

Data Integrity

- Data Integrity is concerned with the ***consistency*** and ***accuracy*** of the data in the Database
- Data Redundancy is a major threat to Data Integrity
- Support for Data Integrity is a key feature of any DBMS

Data Integrity

- Databases model parts of the real world in which many rules apply
  - "A student has only one address"
  - "A student must take 5 courses in the final year or 4 courses plus a project"
- DBMS express such rules by means of "integrity constraints"
- Validation of data values being entered into the DB is another aspect of Data Integrity
- Many users/applications simultaneously updating the Database can threaten Data Integrity
  - This requires "concurrency control"

Backup and Recovery

- The only facility available to file processing systems to restore data following failure is if a back-up was scheduled/manually taken
  - Time Machine on MacOSx
  - Backup and Recovery in Windows
- Insufficient in many on-line environments and organisations where data is a strategic resource
- DBMS provide very sophisticated recovery mechanisms

Query Language Support

- File systems are basically tools for physical storage of data
- They make data much less accessible to users than Database systems
  - If a GP wanted to examine all records for a single patient, this would be very difficult
  - Even if they were meticulous in where they stored them
  - Potentially would need an application to process and combine the data
- DBMS provide a variety of interfaces to suit the needs of a wide range of users

Metadata Management

- In applications which process data from a file system, metadata is often part of the application program
- This can lead to duplication of metadata across applications
  - leading to integrity problems
- Imagine a patient record

1234	Sheldon Cooper	2311 N. Los Roblos Ave., Pasadena	290846	Penicillin

Metadata Management

- To interpret the data in this record, we would need to look at an application program:

```
public class Patient {
        private int patient_ID;
        private String patient_name;
        private String patient_address;
        private int patient_phone;
        private String patient_allergy;
        ……
```

Metadata Management

- With the Database approach:
  - Metadata is stored centrally in the catalog
  - Database catalog entry for patient record
    - Patient_record contains basic details on patient

Patient_ID	int(4)	Unique
Patient_Name	varchar(255)	Firstname followed by Surname
Patient_Address	varchar(255)	Truncate if necessary
Patient_Phone	int(10)	Home phone
Patient_Allergies	varchar(255)	Drug name or None

Advantages of Databases

- Search and Retrieval Capabilites
  - Filtered according to specific needs
- Reduced Data Redundancy
  - Ease of Update
- Greater Data Integrity
- Independence from Applications, Concurrent Access
- Improved Data Security
- Reduced Costs for Data Entry, Storage and Retrieval

Disadvantages of the DB Approach

- Training required for management and querying
- Database systems are complex and time-consuming to design
- Cost
  - Software
  - Hardware
  - Training
- Loss of autonomy brought about by centralised control of the data
- Inflexibility due to complexity

Database Languages

- Programming languages which are used to:
  - Define a database
    - its entities and the relationships between them
  - Manipulate its content
    - insert new data and update or delete existing data
  - Conduct queries
    - request information based upon defined criteria
- The Structured Query Language (SQL) is the most commonly used language for Relational Databases
  - Supported by all relational DBMS and is a standard.

SQL

- SQL is split into four sets of commands which are divided based upon the tasks they are used for:
  - Data Definition Language
  - Data Modification Language
  - Data Query Language
  - Data Control Language


Data Definition Language

- SQL uses a collection of imperative verbs whose effect is to modify the schema of the database
- Can be used to add, change or delete definitions of tables or other objects.
- These statements can be freely mixed with other SQL statements
  - so the DDL is not truly a separate language.


Data Manipulation Language

- The data manipulation language comprises the SQL data change statements
  - Modifies stored data
  - Does NOT modify the schema or database objects
    - This is always the responsibility of the Data Definition Language
- Used for inserting, deleting and updating data in the tables of a database

Data Query Language

- The data query language allows users of a database to formulate requests and generate reports
- There is one primary command used in SQL to query the database - the SELECT Statement
  - This statement is used to query or retrieve data from a table in the database.
  - A query may retrieve information from specified columns or from all of the columns in the table
  - A query may have specified criteria that must be met in order for data to be returned

Transactions

- A way to group actions that must happen atomically
  - all or nothing
- Guarantees to move the DB content from one consistent state to another
- Isolates these actions from parallel execution of other actions/transactions
- Ensures the DB is recoverable in case of failure
  - e.g. the power goes out

Backup and Recovery

- Ensures that the DB can be returned to a stable state in case of errors, such as:
  - Transaction failure
  - System errors
  - System crash
  - Data Corruption
  - Disk failure

Users

- DBMS implementer
  - Builds the DBMS System
- Database designer
  - Designs the Database, Establishes the Schema
- Database application developer
  - Develops programs that operate upon the DB
- Database administrator
  - has overall responsibility for the DB including specifying access constraints, selection of appropriate backup and recovery measures, monitoring performance etc.

Emergent Databases

- XML Databases
  - Document-Oriented
- NoSQL Databases
  - Web Scale, Non-Relational, Open Source
- In Memory Databases
  - Stores data in main memory rather than on disk
- Others
  - Massively parallel processing (MPP) databases
  - Online analytical processing (OLAP) databases