Information Management (IM)

CS1Q/CF2

IM Lecture 1 – **Part 2**

Dr. Craig Macdonald

Recap...

What is data?

3

data	52
information	J Smith's score on the final exam is 52%
knowledge	I've passed!

What is a database?

4

- A database (abbreviated *DB*) is an entity in which data can be stored in a **persistent** and **structured manner**, with as **little redundancy** as possible
- A database centralises users access to data, which they can view, enter, or update
  - within the limits of the access rights granted to them
- It is viewable by many users at the same time (**controlled concurrent access**)

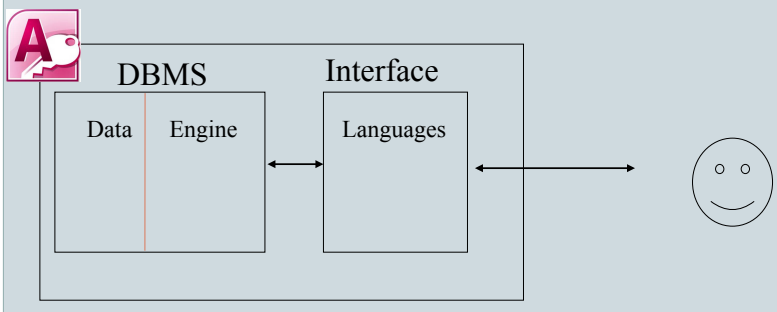
What is a Database Management System (DBMS)?

5

- The DBMS is a suite of services (software applications) for managing databases, which involves:
  - enabling simple access to data
  - allowing multiple users access to the information
  - manipulating the data found in the database (inserting, deleting, editing)
- It also controls the **security** and **integrity** of the database
  - The DBMS accepts requests for data from the application program and instructs the operating system to transfer the appropriate data

Models of data access – Local Database

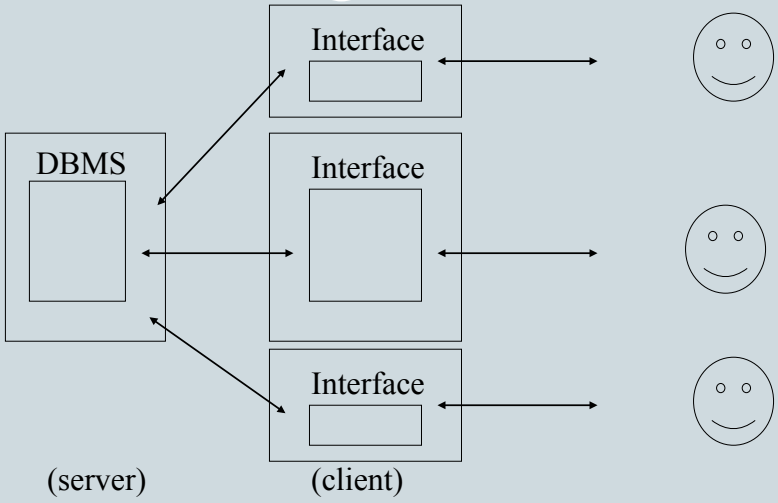
6



6

Models of data access - Client/Server

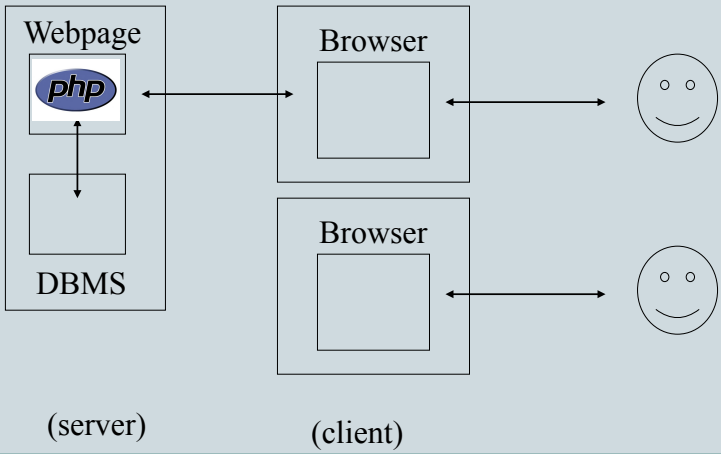
7



7

Models of data access – Web-based

8



8

9

- Databases are used by people…
- … to perform particular tasks
- Databases therefore need interfaces to allow access to the data
- Many people may need to access the same database
- Web pages are just one way of implementing an interface to a database

9

Database design lifecycle

10

- Requirements analysis
  - User needs; what must database do?
- Conceptual design
  - High-level description; often using E/R model
- Logical design
  - Translate E/R model into (typically) relational schema
- Schema refinement
  - Check schema for redundancies and anomalies
- Physical design/tuning
  - Consider typical workloads, and further optimise

Later years

Material in L2 and L3

11

- Data Modeling and Database Design
- Entities, relationships, and attributes
- ER diagrams

Database Design

12

- How do we go about designing a database from scratch?

Firstly some terminology

13

- A **data model**: a description of the objects that could be represented by a computer system together with their properties and relationships
  - these are typically "real world" objects such as products, suppliers, customers, and orders
- A **schema**: a description of how a database can be designed to represent a *data model*
  - *E.g. tables with columns definitions: Suppliers have names, addresses, etc*
- A **database**: an instance of a schema with corresponding data
  - E.g. Amazon's suppliers/customers/orders.

Database Design

14

- Creating a database involves:
  - **(1) Capturing user requirements**
  - **(2) Representing them in a MODEL**
  - **(3) Converting model into a SCHEMA**
  - **(4) Implementation on DBMS**
- Many different ways to implement a database
- Many different models and tools you can use
  - All require the stages above

People involved

15

- Users
  - access the data only (casual vs expert)
  - need an effective means of accessing the data
- Database designers:
  - specify schema and content
- (web) Application developers:
  - extend functionality; provide means of data access for a particular application
- Database administrators
  - Maintain accuracy and integrity
- Web-site designers

All need to think about the final users

15

1 - Identifying **User** Requirements

16

- Talk to client
  - E.g. CEO of the bank, the chief of BT…..
- Talk to customers
  - End users of the system
  - Those that might view the data
- Talk to different levels of users
  - Admin, programmers, technical staff….
  - People who might need to add/update/query data

Users

1 - Identifying **Data** Requirements

17

- Write down all the different 'THINGS' that you need to store data about
  - Customers, branches, accounts…..
- Take note of any **relationships** between the things talked about
  - All customers must belong to one branch only
  - All accounts must only have one account number

Organising into Data Objects

18

<u>Customer</u>
- Name
- address
- overdraft limit
- address
- ID

<u>Branch</u>
- name
- address
- manager
- ID

- This could start to get quite complicated if there are lots of things to store information about in the database

2 - Data Modelling

19

- We need a way to represent all the data we have captured relating to what we want to store in our database
  - Helps us during design and implementation
  - Can help communicate ideas to other members of the team

2 - Data Models

20

- There are models at each of three levels :
  - High-level **Conceptual Data Models** (or semantic data models) describe data in a manner close to their real world (**external**) meaning - as entity types, attributes and relationships
  - The conceptual data model can then converted into a **conceptual schema** describing how data is stored - as tables and records, for instance
    - These are **Implementation-level/logical Data Models**
  - **Low-level** or **Physical Data Models** describe how data is stored on the computer: files, storage structures, etc.
    - This is handled by the DBMS, with occasional help from the DBA

Election Example

23

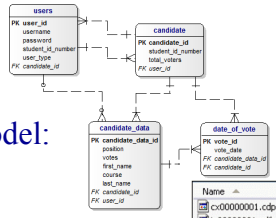
External

Rank any *two* candidates in your order of preference.

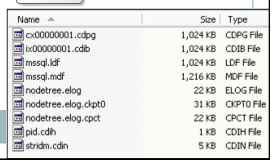
- Joe Smith
- 1 John Citizen
- Jane Doe
- Fred Rubble
- 2 Mary Hill

Conceptual Model

Implementation Model: Conceptual Schema



Physical Model



Conversion depends on Database Model used by DBMS

2 – Different Models of a Database

(different from a data model!)

24

StudentId	First Name	Last Name
482952s	Jane	Smith
195824b	Joe	Bloggs

- **Relational Model**
  - information is represented by a collection of two dimensional tables with rows and columns
    - More on this later (lecture 4)

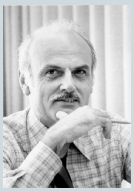
Background

- Network Model
  - information is represented by a directed graph of records and links between the records
- Hierarchical Model
  - the information is represented as a tree of records organised in sets connected by ownership links
- Object oriented Model
  - Collections of 'classes' with certain 'behaviours'/methods

Relational DBMS

27

- In older DBMS, the code for data management and application were all tangled together
  - Hard to modify, hard to generalise
  - Data manipulation code written at very low levels of abstraction
- Instead most modern DBMS follow the *relational* model (RDBMS)
  - Data is stored in relational *tables*
  - It links very well with **Entity/Relationships (E/R)** form of Conceptual Data Modelling
- E/R modelling and corresponding Relational DBMS will be the focus of the next lectures



E.F. Codd
1923-2003

Next Lecture

28

- How to construct an ER diagram
- More on relationships and attributes

Note

- you will need notes from lecture 2 (this one) and lecture 3 (Tuesday) for your first IM (1Q) tutorial next week!

Essential Reading

29

- After this lecture
  - Rolland, Chapter 2
    - 2.1 and 2.3.1
- Before next lecture
  - Rolland, Chapter 4
    - 4.1