Information Management (IM)

CS1F - Computing Fundamentals
IM Lecture 1

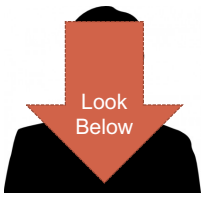
Dr. Craig Macdonald

2019-2020

CS1F Overview

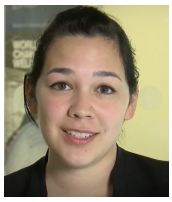
- *From now… Information Management*

Dr. Craig Macdonald
Lecturer
Room S171, Lilybank Gardens
Email: Craig.Macdonald@glasgow.ac.uk



- *After Oct. 29th – Professional Issues & Human Computer Interaction:*

Dr. Julie Williamson
Lecturer
Rm 407, Sir Alwyn Williams building
Email: Julie.Williamson@glasgow.ac.uk



CS1F: Where & When

3

- Lectures:
  - Tuesdays & Thursdays 12-1
    - Venue: Main Building Kelvin Gallery
- Tutorials & Labs
  - Start in Week 2, from 2nd October
  - You have been allocated a lab/tutorial group
  - Check your timetable for your tutorial group & location

CS1F & Routes

4

- You take CS1F if:
  - *Normal* Route: If you are studying 1st year Computing Science and taking CS1P
  - OR
  - *Alternate* Route: If you are in 2nd year and took CS1CT in first year (and not CS1F)
  - OR
  - *Exceptional* Route: You might *exceptionally* be taking CS1CT and CS1F
  - OR
  - *Faster* Route: If you are in the first year of a “fast track degree”

CS1F Overview: When?

5

Week 1 - Week 6: **Information Management**
Week 6 - week 12: **HCI & Professional Issues (PI)**
Week 13: Exam
Week 14, 15, 16: Christmas Holiday

CS1F Assessment

6

Exam	December	80%
Practical Work	Ongoing	20%
Resit Exam	*August*	*80%*

For CS1F, this equates to

- **Information Management Practical** is worth **8%** **=~ 8 hours**
- **Professional Issues Contribution** is worth **4%**
- **Human Computer Interaction Practical** is worth **8%** **=~ 8 hours**

Moodle

7

- Your gateway to all course information
  - http://moodle2.gla.ac.uk
- Specific page for CS1F
  - http://moodle2.gla.ac.uk/course/view.php?id=2890
  - Lecture notes
  - Assessed coursework
  - Additional reading material
  - News forum

Familiarisation Labs

8

Turn up to ONE of the following familiarisation labs **this week** (week 1)

They will be held in the Computer Labs @ Room 715 Boyd Orr Building:

Tuesday 24th (2-4)
Wednesday 25th (10-12)
Wednesday 25th (2-4)

You should bring your Student ID card

You will be issued with: (1) important lab documentation and (2) a lab worksheet for this week

What you will do in the lab:
(1) Log in and find your way around Moodle
(2) A short exercise (CS1CT or CS1P or FasterRoute) - upto 1 hour

It is your job to turn upto an appropriate lab session

THE INFORMATION MANAGEMENT MODULE

9

The Information Management Module

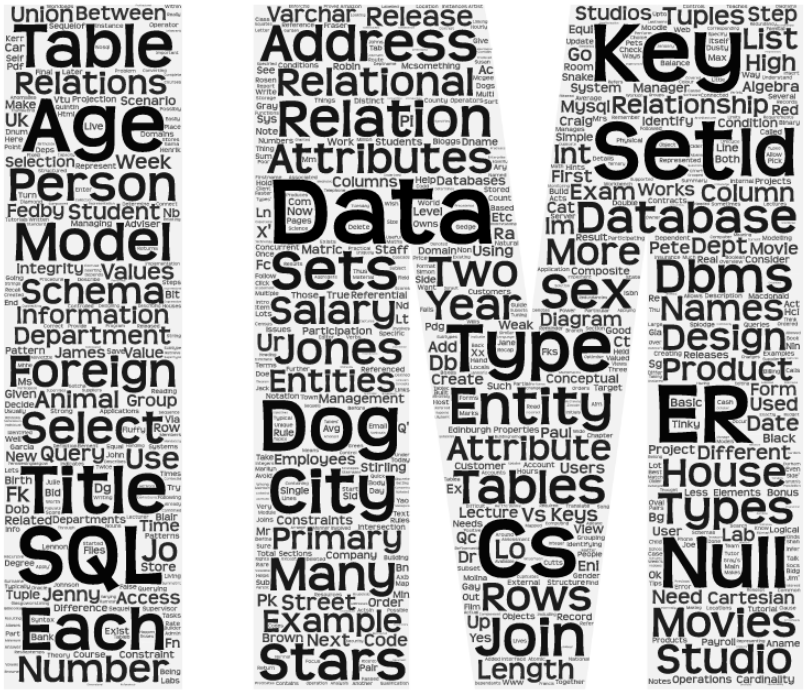
10

- **Aim**: to understand the ways in which databases contribute to the management of large amounts of data.
- **Objectives**:
  - understand the ***nature of applications*** built using programs clustered around databases and other large collections of data.
  - understand the overall ***architecture*** of a database management system.
  - be able to carry out all the ***operational tasks*** of setting up and using a relational database.

The Information Management Module

11

- Another aim: to understand the mathematics necessary behind databases and how to manage and query them
- Objectives
  - understand sets & relations
  - understand relational algebra, SQL



12

Lectures

13

- IM Lectures from week 1 through to week 6
- Topics
  - Issues in Data & Information Management
  - Data Modelling and ER Diagrams
  - Design & Implementation of a (Relational) Database Application
  - Querying a Database
  - Sets, Relations, and Relational Algebra

Tutorials & Labs

14

- Tutorials & Labs will be closely linked to the practical assessment and the exam

Week	Lecture (Tue)	Lecture (Thu)	Tutorial	Lab	Hand in
1	Introduction	DB modelling		Familiarisation	
2	ER Diagrams	ER diagram con't	ER diagram		**ER Diagram**
3	ER to Tables	Relational Model & Integrity		Tables	
4	Sets	Relational Algebra	Sets		
5	Basic SQL & Patterns	More SQL		Populate and Query	
6	*SQL Wrapup & Exam tech*	*HCI*		**Assess DB**	**DB report**
7	*HCI*	*HCI*	Feedback		

- Practical Assessment
  - worth 8% of 1F module mark
  - involves designing, building, testing and reporting on a simple database system (using a MySQL database server)

See also 1F overview document on Moodle

Why are Databases Important?

15

- Databases are a key technology
  - Used in a wide variety of applications to manage data
  - Growing in importance – amount of data we store is increasing
- Consider the management of all of Facebook's user accounts and data……..
  - This is a lot of data and a lot of data management involved!

Understanding Databases

16

- On this course you will design, build, populate and query your own database using **MySQL**
- It is important you understand the **theory,** as well as how to use **MySQL,** so that you will easily be able to design, build and use other database applications in the future

ISSUES IN DATA MANAGEMENT

What is data?

18

What is data?

19

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

Data vs. Information

20

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

structured representation (encoding)

data + meaning

true belief

Issues in managing data

21

- Consider the example of handling the billing & monitoring of all UK household telephone calls

Issues in managing data

22

- Consider the example of handling the billing & monitoring of all UK household telephone calls
  - What kinds of data would you have to store?

Issues in managing data

23

- Consider the example of handling the billing & monitoring of all UK household telephone calls
  - What kinds of data would you have to store?
    - People's names
    - Addresses
    - Phone numbers
    - Post codes
    - Account Codes
    - Bank details
    - Money owed
    - ……

Issues in managing data

24

- Consider the example of handling the billing & monitoring of all UK household telephone calls
  - Can you think of any issues associated with managing all this data and the tasks associated with it?
    - On paper?
    - In filing cabinets?
    - In spreadsheets or text documents?

Issues in managing data

25

- What about managing all of the data stored on Amazon?
  - 20 million products available
  - Need to hold terabytes of data
  - *310 million* users, with thousands of users active at the same time
  - Processing distributed around the world
  - Need access to data for monitoring and looking for significant patterns
  - Reliability and security important
    - E.g. required by EU's General Data Protection Regulation (GDPR)

Managing Data

26

- If we have lots of data to store, we need a really good way to store that data
- We also need good ways to be able to access that data quickly and easily
- And we don't want lots of different versions of the data all over the place - we want to avoid REDUNDANCY

Managing Data

27

- Our data storage tool must provide these features:
  - Data definition (data structuring)
  - Data entry (to add new data)
  - Data editing (to change existing data)
  - Querying (a means of extracting data by a description)
  - Persistence (data existing beyond a single operation or program invocation)

Strategies for Data Management 1

28

- A **program** where all the data is held in the program's memory
- But no data persistence between invocations of a program
- The data is reconstructed (or re-entered) at each invocation of the program

Strategies for Data Management 2

29

- **Files** of data on disk that can be accessed by different applications
  - Each application is responsible for its own representations of the data
  - Difficult to coordinate between applications
  - Might be different versions of the data

Strategies for Data Management 3

30

- Combine together all the functions of data storage and access for a related set of tasks, e.g.
  - handling student records
  - stock control in a warehouse
  - account management in a bank
- This is known as a

  Database Management System (DBMS)

What is a database?

What is a database?

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

Types of database

33

- **Hierarchic databases** (older)
- **Network databases** (older)
- **Relational databases** (very common)
- **Object Oriented Databases** (1990s)
- **NoSQL Databases** (2000s)

MySQL, Postgres, Oracle, MS SQLServer

Our focus

Informix, Greenplum

MongoDB, HBase

A Relational Database…

34

- A *relational* database can be thought of as a series of *tables* about *related* information
- For example, Amazon's database might have a table called **Products**:

Product ID	Product Title	Description	ISBN	Supplier	#
194729187	Database Systems: The Complete Book	For Database Systems and Database Design and Application courses ...	129202447X	2847	5
…					

- As well as *related* tables called **Customers**, **Suppliers**, **Orders**, **Reviews**, etc:

Customer ID	Email Address	…
…		

Supplier ID	Address	…
2847	14 Cider St	

Operations on a Database

35

- What operations can be done upon a database?
  - Views: read existing data
  - Manipulation: Amend existing data, or add new data
- All this may occur **concurrently,** by many **different users** with varying **permissions**

Product ID	Product Title	Description	ISBN	Supplier	#
194729187	Database Systems: The Complete Book	For Database Systems and Database Design and Application courses ...	129202447X	2847	5
…					

Customer ID	Email Address	…
…		

Supplier ID	Address	…
2847	14 Cider St	

Databases avoid Redundancy

36

- Ambiguity
  - Same thing with different name in different files
- Inconsistency
  - If data changes in one place it should also change in the other files is exists in

An example database

37

- Example table from a geographical database

town	county	population	County town?	Cathedral?
Welwyn Garden	Hertfordshire	40,570	no	no
St. Albans	Hertfordshire	123,800	no	yes
Hertford	Hertfordshire	2,023	yes	no
Durham	Durham	29,490	yes	yes

- Boss says: *"I want the county population details too, including Essex!"*

Extended database

Redundancy: Information about Hertfordshire is duplicated: difficult to update without *inconsistency*

- Add county details, including Es

town	county	population	County town?	Cathedral?	County population	County size
Welwyn Garden	Hertfordshire	40,570	no	no	937,300	631
St. Albans	Hertfordshire	123,800	no	yes	937,300	631
Hertford	Hertfordshire	2,023	yes	no	937,300	631
Durham	Durham	29,490	yes	yes	132,681	295
	Essex				1,464,200	1,528

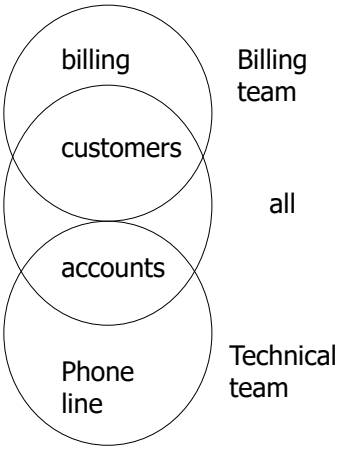
We have information about the population of Essex as a whole but none about any individual town.

- Redundant informatio
  - Consider update "popul up to 1.04M"

Databases avoid Redundancy (cont.)

39

- Ambiguity
  - Same thing with different name in different files
- Inconsistency
  - If data changes in one place it should also change in the other files is exists in
- Wasted effort
  - Data should be shared where possible to save time and effort



Databases avoid redundancy (2)

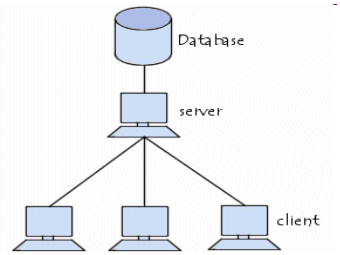
40

- Having data repeated in different places indicates poor relational database design
- The data relationships, inherent in a relational database, should allow you to:
  - maintain a single data field, at one location.
  - Such that the database's relational model is then responsible for porting any changes, to that data field, across the database.
- E.g. a *single* table with Town & County-level details is a poor design

Controlled Concurrent Access

41

- Databases can have **many users** *reading* and *writing* at the same time



- We need to make sure that each view of the data is correct or *consistent* for each user
  - So that concurrent access does not cause incorrect updates
  - Lets see an example…

Bank Account

42

- Imagine a bank's database of accounts

Customer	Balance (£)
Mr & Mrs Bloggs	100
…	…

- And the operation to withdraw £10 from a cash machine:

```
X = Get_balance();
Set_balance(X-10);
```

- Now what happens if Mr & Mrs Bloggs both withdraw £10 *concurrently,* i.e. at exactly the same time?

Bank Account

43

Customer	Balance (£)
Mr & Mrs Bloggs	90
…	…

Mr Bloggs 

```
X = Get_balance();
```

X=100

```
Set_balance(X-10);
```

Mrs Bloggs 

```
X = Get_balance();
```

X=100

```
Set_balance(X-10);
```

- Here, concurrent access resulted in an incorrect account balance being recorded

Controlled Concurrent Access

44

- Databases can have **many users** *reading* and *writing* at the same time
  - We need to make sure that each view of the data is correct or *consistent* for each user
  - So that concurrent access does not cause incorrect updates
- DBMS have concurrent control software to ensure that several users updating the same data do so in a controlled manner
- This happens through *transactions,* which make concurrent database interactions appear to happen *independently* & *sequentially*

Database Management Systems

45

- A DBMS is the software that can provide features needed by many databases:

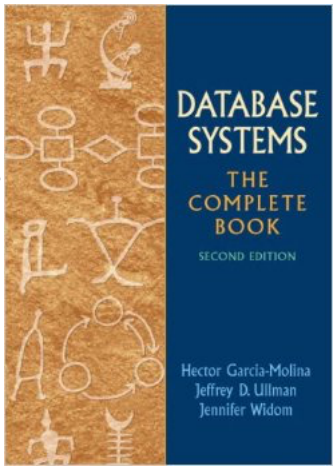
1. Sharing and integration of data
2. Multiple views of the same data
3. Controlled concurrent access
4. Management of security and integrity

- More on DBMS attributes on Thursday

Its good to read…

46

- A *recommended* textbook
- Database systems : the complete book. Garcia-Molina, Ullman, Widom. Pearson Education, 2013
  - **Free** eBook via University Library
    - Limited (9) copies online
  - 3 copies in the library
- I will always provide alternative reading material



Recommended Reading…

47

- Related to lecture 1
  - Garcia-Molina, Chapter 1
  - Chapter 1 is available for free online at http://infolab.stanford.edu/~ullman/fcdb/ch1.pdf
  - Or: http://philip.greenspun.com/sql/introduction.html
- Related to lecture 2
  - Garcia-Molina, Chapter 4
    - Sections 1 -1.5
  - OR: from Mamčenko's notes http://gama.vtu.lt/biblioteka/Information_Resources/i_part_of_information_resources.pdf pages 8 & 9