University of Glasgow

Data Storage and Retrieval

Lecture 1

Dr. Graham McDonald

Graham.McDonald@glasgow.ac.uk

Who am I?

Dr. Graham McDonald

Senior Lecturer

Email: graham.mcdonald@glasgow.ac.uk

- About Me:
  - PhD in Information Retrieval
  - Research interests: Responsible Information Retrieval Sensitive information identification and fair retrieval strategies

Updates to Course

Each year we make incremental updates to the course

- Some changes are operational, but student feedback is an important input to this process
- Please complete the EvaSys feedback at the end of the semester!

This year's main updates to the course:

- Additional NoSQL examples
- New section on vector databases for inexact queries

You will have plenty of opportunities for feedback and questions

- E.g. in lectures, labs, on coursework etc.
- If you have a question, ask!

University of Glasgow

Topics Covered

- Issues in data & information management
- Data modelling and ER diagrams
- Design & implementation of a (relational) database application
- Sets, relations, and relational algebra
- Querying a database
- Transactions and views
- Beyond relational databases: NoSQL
- Beyond exact queries: vector databases

This course will mostly focus on issues relating to the design, implementation and querying of relational databases. However, we will also introduce concepts that go beyond the relational model.

University of Glasgow

Aims & Objectives

- **Aim**: To understand the ways in which data storage and retrieval (e.g. 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.
  - Understand the mathematics that are necessary to manage and query relational databases.
  - Be able to carry out all the ***operational tasks*** of setting up and using a relational database.
  - Understand methods of data storage and retrieval that go beyond the relational model.

Course Structure

- Class Structure:
  - 2nd hour will be delivered as live lectures.
  - 1st hour will be interactive sessions covering previous day's lecture.
- Assessment:
  - Written examination 60%,
  - Mid semester class tests 10%,
  - In-class quiz 5% and
  - Course work 25%

Day	Date	Schedule
Mon	28/10	
Tue	29/10	
Wed	30/10	
Thu	31/10	Coursework 1 Handout & Lab
Fri	01/11	
Mon	04/11	
Tue	05/11	**Coursework 1 Due**
Wed	06/11	Coursework 2 Handout + Lab
Thu	07/11	Workplace Days
Fri	08/11	
Mon	11/11	
Tue	12/11	
Wed	13/11	
Thu	14/11	Lab
Fri	15/11	**Course Work 2 Due**
Mon	18/11	
Tue	19/11	
Wed	20/11	**Class Test**
Thu	21/11	Workplace Days
Fri	22/11	

Moodle

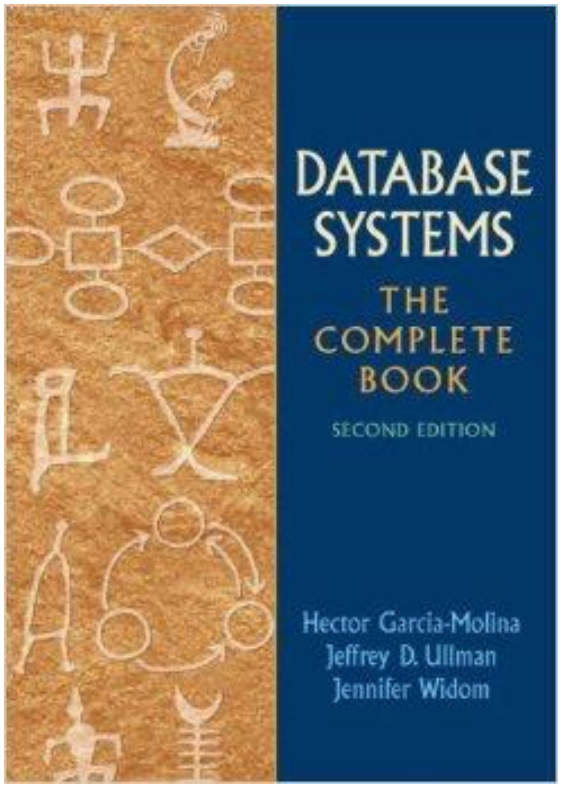
- Data Storage and Retrieval course page
  - https://moodle.gla.ac.uk/course/view.php?id=44988
  - Lecture notes
  - Assessed coursework
  - Additional reading material
  - News forum

- Moodle GA General Information
  - https://moodle.gla.ac.uk/course/view.php?id=45978

A Good Read ..

- A *recommended* textbook

- Database systems : the complete book. Garcia-Molina, Ullman, Widom. Pearson Education, 2013
  - **Free** eBook via University Library
  - 3 copies in the library
  - Available to buy online also (this is not a requirement)



Advert!

Terrier Team

Automated Financial Advisory Experiment

We are conducting an online experiment and are looking for participants who meet the following criteria:

- Fluent English speaker
- Over 18 years old
- Interested in Finance and Investment

The experiment involves interacting with a chatbot and answering questions. The session will last approximately 60 minutes.

QR code for participation!

Compensation:

Participants will receive a **£10 Amazon gift card for completing the experiment**.

How to Participate:

To express your interest, please choose one of the following options:

- Email us and include interest in taking part in the experiment, or
- Scan the QR code and fill in your email address on the provided form

Once we have received your information, we will provide invitation details.

Email Contact:

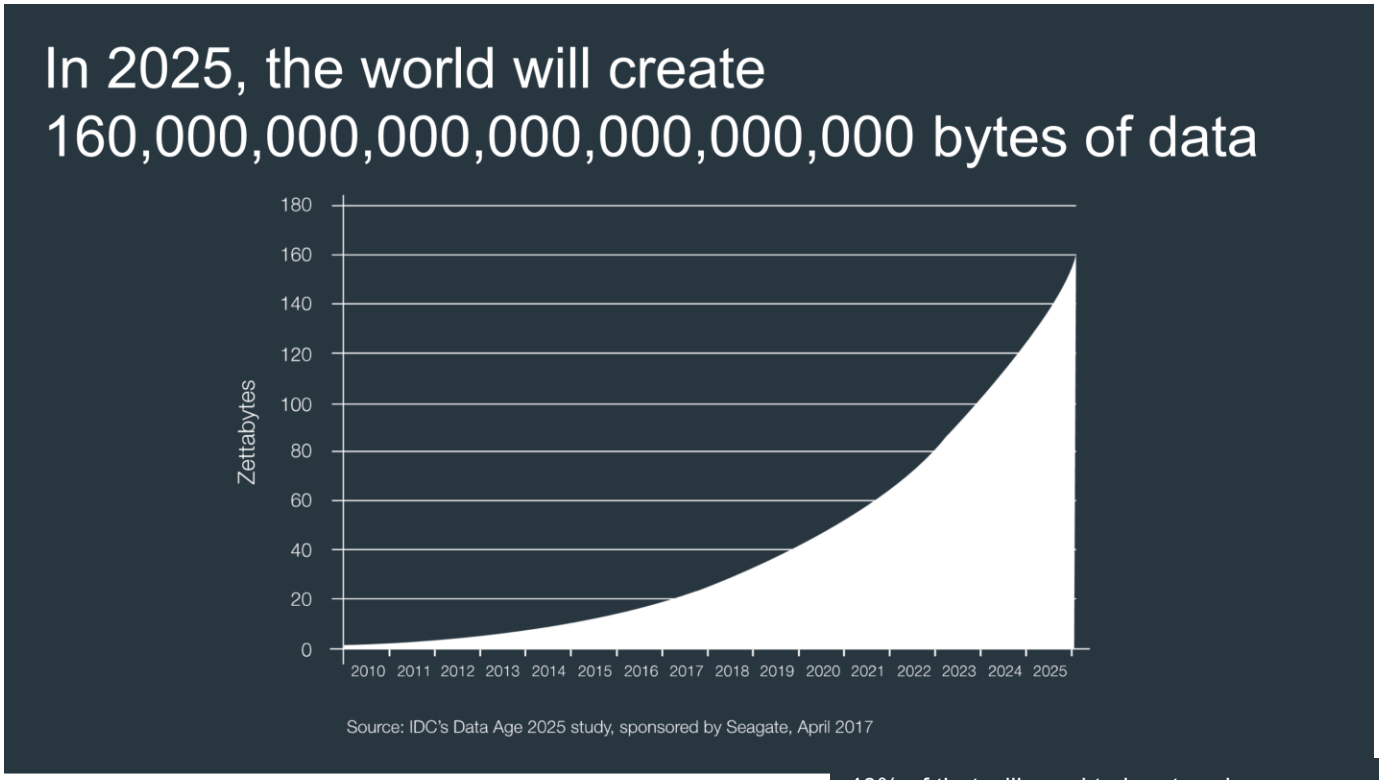
Contact: Takehiro Takayanagi

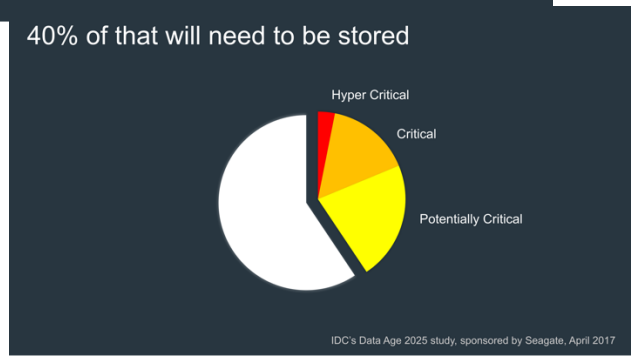
Email: 3057508T@student.gla.ac.uk, takayanagi-takehiro590@g.ecc.u-tokyo.ac.

University of Glasgow

Issues in Data Management

Why are Databases Important?





https://www.digitaltrends.com/cool-tech/dna-data-catalog-startup/

University of Glasgow

Why are Databases Important?

- 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!

University of Glasgow

Understanding Databases

- 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

Data, Information and Knowledge

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

Data, Information and Knowledge

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

structured representation (encoding)

data + meaning

true belief

University of Glasgow

Issues in Managing Data

- 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

- 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

- 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?

University of Glasgow

Issues in Managing Data

- What about managing all of Amazon's data?
  - 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

- 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

University of Glasgow

Data Storage

- 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)

University of Glasgow

Strategies for Data Management 1

- 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

- **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

- 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?

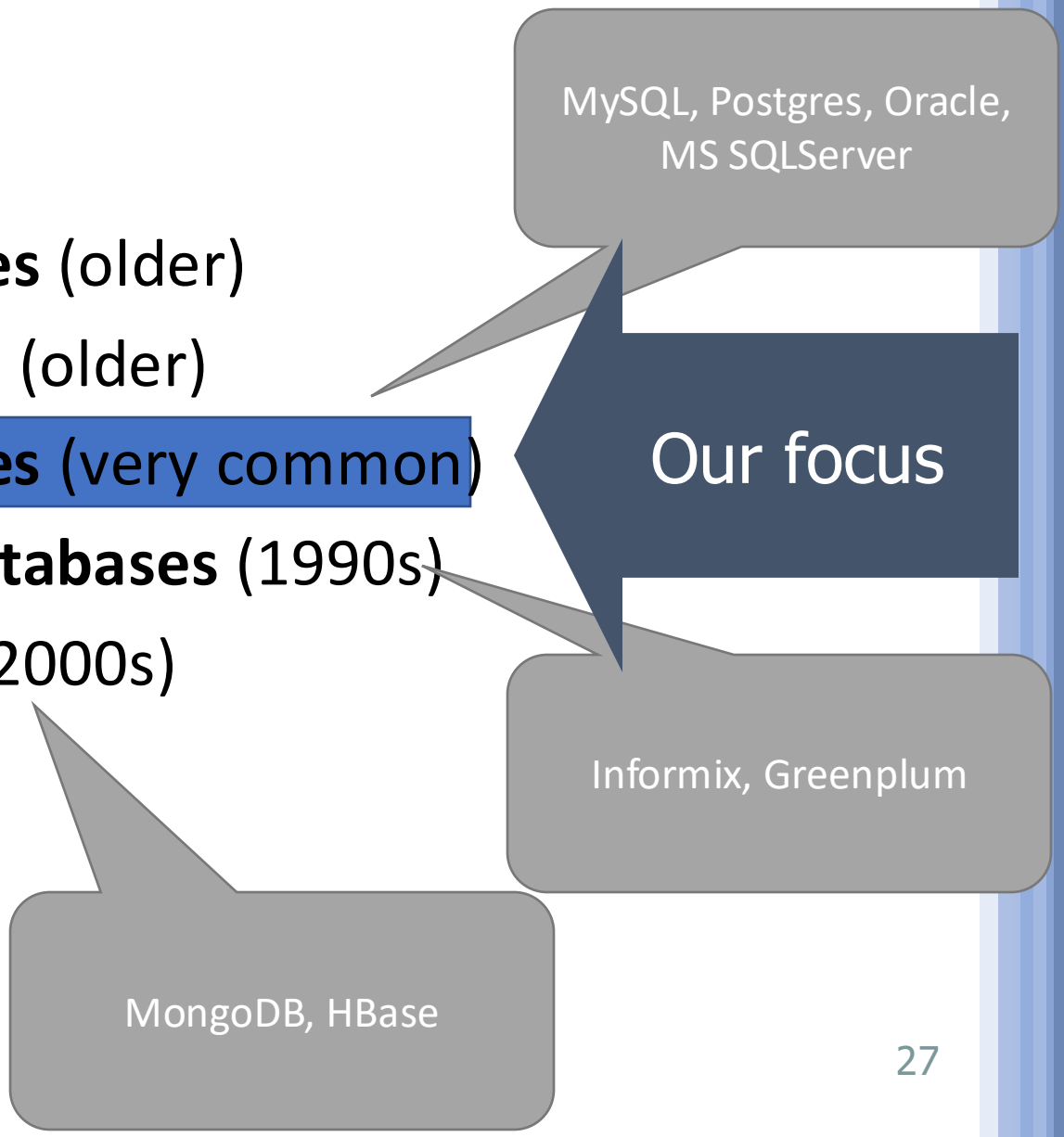
University of Glasgow

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

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



A Relational Database…

- 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

- 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	

University of Glasgow

Databases Avoid Redundancy

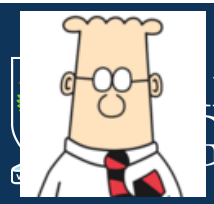
- Ambiguity
  - Same thing with different name in different files
- Inconsistency
  - If data changes in one place it should also change in the other files is exists in

An Example Database

- 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!”



iversity
Glasgow

Extended Da

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

- Add county details, includin

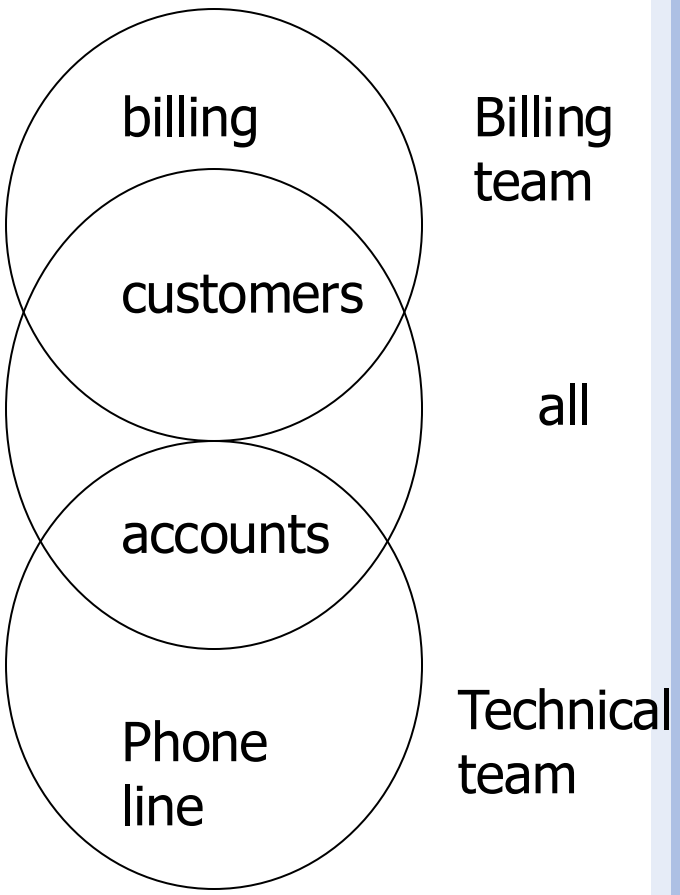
town	county	population	County town?	Cathedral?	County population	County size
Welwyn Garden	Hertfordshire	40,570	no	no	937,300	631
St. Albans	Hertfordshire	123,800	no	yes	937,300	631
Hertford	Hertfordshire	2,023	yes	no	937,300	631
Durham	Durham	29,490	yes	yes	132,681	295
	Essex				1,464,200	1,528

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

- Redundant information
  - Consider update "popula o 1.04M"

Databases Avoid Redundancy

- 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



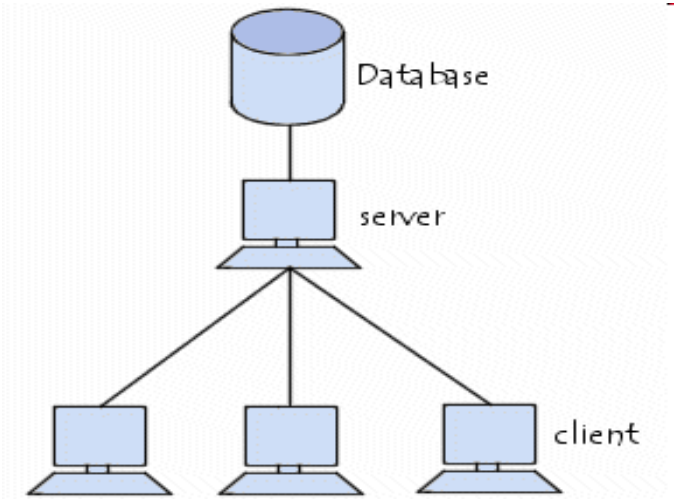
Databases Avoid Redundancy (2)

- 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

- 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

- 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

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

University of Glasgow

Controlled Concurrent Access

- 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

University of Glasgow

Database Management Systems

- 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 in next lecture!