CSCU9B3:
Database Principles and Applications

Introduction to Databases

Course Content and Aims

- To introduce the principles of relational database systems and how they can be used via web applications.
  - Introduction to the Database Approach
  - The Relational model
  - Relational languages: Relational Algebra
  - ER modelling
  - Normalisation
  - The ANSI-SPARC model
  - Backup and Concurrency
  - Data protection – the legal aspects
  - mySQL
  - PHP

Course Organisation

- See the Schedule on Canvas
  - 2 lectures per week
  - 1 practical (most weeks)
  - 1 tutorial (two weeks)

- Assessment
  - Lab checkpoints
  - Coursework
  - Examination

- How to get the most of the class
  - Prepare for all of the sessions
  - Ask questions!
  - Re-read material after the session

Traditional File-Based Systems

- The first precursor of databases was the manual file-based system (the kind that people store in a filing cabinet).
- For example, organisations may keep data related to projects, products, tasks, clients, or employees within separate files held within secure filing cabinets.
- Many people keep a similar system at home to store information such as bank statements, receipts, guarantees, and invoices.
- Using such a filing system, we look up information by searching through the files until we find the one that we want.
- The better the filing system, the more efficient information retrieval can be.

Problems with Manual Filing Systems

- Manual filing systems work fine when the number of files is small, or if an efficient indexing system is employed.
- However, manual systems break down when we need to cross-reference the stored information or process it in some way.
- For example, consider an estate agent who keeps separate files for each property, buyer, and member of staff.
- Answering questions like these can be very time consuming:
  - What flats do you have for rent within 3 miles of the city centre?
  - What is the average rent for a 2-bedroom flat?
  - What are the phone numbers of the owners of property in Alloa?
  - How does last month's turnover compare with the projected figure for this month?

Why Do These Questions Create Problems?

- The kind of questions we asked on the previous slide can be very difficult to answer with a manual filing system.
  - This is because we generally need to search through every file to pick out and combine the information required to answer the question.
- It is not possible to design a filing system that will be efficient for answering all questions that we might pose.
- To cope with the ever-increasing demand for information, organisations started to store information in electronic files to improve the efficiency of retrieval.
- However, departments within an organisation tended to manage their own information, simply replacing their own paper files by electronic files. This leads to practical problems, as we shall see.

Decentralised Information Management

- For example, consider two departments within an estate agency.
- The Sales department is responsible for selling and renting properties to clients.
- Two kinds of clients are dealt with:
  - Those that wish to sell properties.
  - Those that wish to rent or buy a property.
- Information relating to the dealings with each kind of client may be maintained electronically using a special purpose program
- Three files containing property, owner, and renter details are required.
- Examples of these files are shown on the next slide.

Sales Department Information

Properties

Pno	*Street*	*Area*	*City*	*Type*	*Rooms*	*Rent*	*Ono*
PA14	16 Holhead	Dee	Aberdeen	House	6	650	CO46
PL94	6 Argyll St	Kilburn	London	Flat	4	400	CO87
PG4	6 Lawrence St	Partick	Glasgow	Flat	3	350	CO40

file

Owners

Ono	*Name*	*Address*	*Phone*
CO46	Joe Keogh	2 Fergus Dr	01224-861212
CO87	Carol Farrel	6 Achray St	0141-357-7419
CO40	Tina Murphy	63 Well St	0141-943-1728

record

field

Renters

Rno	*Name*	*Address*	*Phone*	*Pref_Type*	*Max_Rent*
CR76	John Kay	56 High St	0171-774-5632	Flat	425
CR56	Aline Stewart	64 Fern Dr	0141-848-1825	Flat	350
CR74	Mike Ritchie	18 Tain St	01475-392178	House	750

The Contracts Department

- The Contracts department is responsible for handling the lease agreements associated with properties for rent.
- When a client agrees to rent a property, the Sales department provides leasing details to the Contracts department.
- The Contracts department complete each lease agreement by allocating a reference number and determining the payment and rental period.
- They maintain three files containing lease, property, and renter details.
- Some of this information is similar to that maintained by the Sales department.
- Examples appear on the next slide.

Contracts Department Information

Lease

Lno	*Pno*	*Rno*	*Rent*	*Payment*	*Deposit*	*Paid*	*Start*	*Finish*	*Duration*
10024	PA14	CR62	650	VISA	1300	Y	01-Jun-95	31-May-96	12
10075	PL94	CR76	400	Cash	800	N	01-Aug-95	31-Jan-96	6
10012	PG21	CR74	600	Cheque	1200	Y	01-Jul-95	30-Jun-96	12

Property

Pno	*Street*	*Area*	*City*	*Rent*
PA14	16 Holhead	Dee	Aberdeen	650
PL94	6 Argyll St	Kilburn	London	400
PG21	18 Dale Rd	Hyndland	Glasgow	600

Renter

Rno	*Name*	*Address*	*Phone*
CR76	John Kay	56 High St	0171-774-5632
CR74	Mike Ritchie	18 Tain St	01475-392178
CR62	Mary Tregear	5 Tarbot Rd	01224-196720

Observations

- Each department has its own set of special-purpose programs for entering and manipulating the information in its files.
- The programs automate the storage and retrieval of information and improve the overall efficiency compared to the manual system.
- One important point to note about the programs is that they specify the details of physical structure and storage of the data files and records. The program(s) will work only if the data is organised in their particular way.
- Also, due to the relative autonomy of each department, there is a significant amount of duplication of information.
- From these observations, a number of problems can be identified.

Problems with Decentralised Files (1)

- Separation and isolation of data
  - Isolating data in separate files makes it difficult to answer the style of cross-referencing questions we looked at before.
  - For example, to produce a list of houses matching the requirements of potential renters, we must first extract all renters who are interested in houses. Using this list, we must then scan the property file to find houses for which the rent is less than that listed for each renter.
  - This is a fairly simple query, yet it is quite complex to write a program that can do it. Imagine trying to handle more complex queries!
- Duplication of data
  - Uncontrolled duplication of data can occur.
  - This is wasteful of both time and space.
  - Great care must be taken to maintain the consistency of duplicated data.

Problems with Decentralised Files (2)

- Data dependence
  - Since the structure and organisation of files and records are defined in the program code, making changes to the structure is difficult.
  - A potentially large number of programs would need to be updated.
  - A one-off program must be created to convert the existing data to the new format.
- Incompatible file formats
  - Since each department maintains its own information, it is likely that the structure of files will vary from one department to another.
  - Furthermore, departments might write their programs in different languages, each of which stores information in a different format.
  - This would make it difficult to share information and would exacerbate the duplication problem.

Problems with Decentralised Files (3)

- Range of operations available.
  - From the user's point of view, electronic file-based systems were a major improvement over manual systems.
    - **This led to an increase in the number and types of operations that users wished to perform on their data.**
  - However, electronic file-based systems require appropriate programs.
  - Support for a particular operation requires a dedicated program to process the data files and insert or extract the information as required.
  - Thus, either the range of available operations is limited to that supported by a basic set of programs, or a plethora of programs must exist to satisfy the information requirements of the organisation.
- As a result of these sorts of problems, organisations turned to a different approach for information handling.

The Database Approach

- Textbook (Ritchie): Chapter 1
- The limitations of the file-based approach can be attributed to two factors:
  - The definition of the data is embedded in the programs which are used rather than being stored separately and independently.
  - There is no control over the access and manipulation of data beyond that imposed by the programs.
- A more effective approach requires two new concepts:
  - the Database
  - the Database Management System
- A database is a shared collection of logically related data (and a description of this data), designed to meet the information needs of an organisation.

The Database

- A database is a single, large repository of data that is defined once and used simultaneously by many users (e.g. in different departments.)
- Rather than having disconnected files with redundant data, all data is integrated, with minimum duplication.
- No data is owned by a single department; all data is now a shared corporate resource.
- A database holds not only an organisation's operational data, but also a description (more likely a number of separate descriptions) of that data.
  - The description is known as the database schema or meta-data (the data about data).
  - This provides a degree of independence between programs and the data since the structure of the data is separated from the programs and stored in the database.

Logically Related Data?

- When we analyse the information needs of an organisation we try to identify entities, attributes, and relationships.
- An entity is a distinct object (a person, place or thing, etc.) in the organisation that must be represented in the database.
  - e.g. a Renter, a Property, a Lease Agreement, etc.
- An attribute is a property that describes some aspect of the object that we wish to record.
  - e.g. Rent, Number of rooms, Name, Address, etc.
- A relationship is an association among two or more entities.
  - e.g. Properties have Owners, Renters are associated with a Lease Agreement and a Property, etc.
- A database combines these aspects: it holds data that is logically related in ways that correspond with the relationships.

The Database Management System

- The Database Management System (DBMS) is the software that enables users to define, create, and maintain the database and which provides controlled access to this database.
- The DBMS comes between the users and the database.
- First, the DBMS allows users to define the database, usually through a Data Definition Language (DDL).
  - The DDL allows users to specify the data types and structures, and the constraints on the data to be stored in the database.
- Second, the DBMS allows users to insert, update, delete, and retrieve data from the database, usually through a Data Manipulation Language (DML).

The Database Management System

- The fact that the database provides a central storage repository for data allows the DML to provide a general enquiry facility for this data, called a query language.
  - Query languages help prevent the problems of either a fixed set of operations or a proliferation of programs that occur with file-based systems.
- The DBMS provides controlled access to the database.
  - Prevents access by unauthorised users.
  - Protects integrity of data by maintaining its consistency.
  - Controls access to allow sharing.
  - Facilitates recovery following hardware or software failures.

The Database Environment

- A database system aims to provide users with an abstract view of data by hiding certain details of how data is stored and manipulated.
- Therefore, the starting point for the design of a database must be an abstract description of the information requirements of the organisation. This will be in terms of entities, attributes and relationships, as we saw before.
- And it will not be concerned at all with the mechanisms which might be used to store or retrieve the data. (It is abstract.)

An Abstract View

- For example, in the estate agent example we may build an abstract view which contains the following:
  - Entities: Staff, Property, Owner, and Renter (maybe others, too).
  - Attributes describing properties or qualities of each entity (e.g. Staff have names, addresses, and salaries).
  - Relationships between these entities (e.g. Owners own Properties).
- Since a database is a shared resource, we may also be concerned to provide different users with different views of the data held in the database.

End of Lecture

Would you like to ask anything?

Don’t forget to read the notes again, and Ritchie Chapter 1.