EBU5503 Database Systems

Welcome to Database Systems

Content

- Week 1
  - Introduction
  - Relational model
  - Relational Algebra
  - Entity-relationship(ER) modelling
- Week 2
  - EER modelling
  - ER to Relational Model mapping
  - SQL
  - Database design

Content

- Week 3
  - Normalization
  - Advanced Normalization
  - Transaction management
- Week 4
  - Distributed DBMS
  - Database security and Ethics
  - XML
  - NoSQL

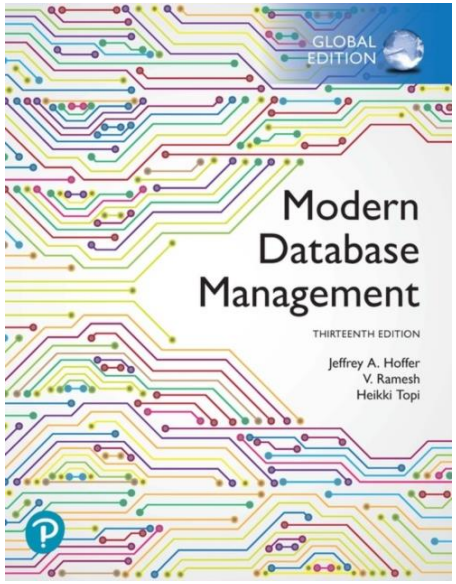
Organisation of the module

- Lectures
- Lab: Five 2-hour lab sessions (starting in teaching week 2)
- Assessment
  - Two tests: QMplus timed quizzes, mid-term test – total 10%
  - Coursework (practical group project): 10%
  - Exam: 80%
  - QMplus Quizzes: beneficial to check your own learning, not assessed
- Modules Reps
  - One module representative for each teaching group
  - Anyone interested please contact the teaching group lecturer

- Textbook:

  Modern Database Management, Thirteenth edition / Jeffrey A. Hoffer, V. Ramesh, Heikki Topi.

  Available from QMUL library as e-book
  https://www.qmul.ac.uk/library/



- Recommended reading:

  Database systems: a practical approach to design, implementation, and management, 6th edition, Thomas M. Connolly, Carolyn E. Begg.

Tips on doing well for this module

- Attend the lectures
- Ask when you don't understand anything
  - No questions too "silly"!
  - Ask in live lectures
  - Use Student forum on QMPlus
  - Use Mentimeter during live lectures
  - Ask in Office hour
- Check the "Learning Outcomes"
- Do the exercises and attend labs
  - QMplus quizzes, tutorial, in-lecture quizzes/exercises
  - Turn up to scheduled labs! There are TAs there to help.

Database

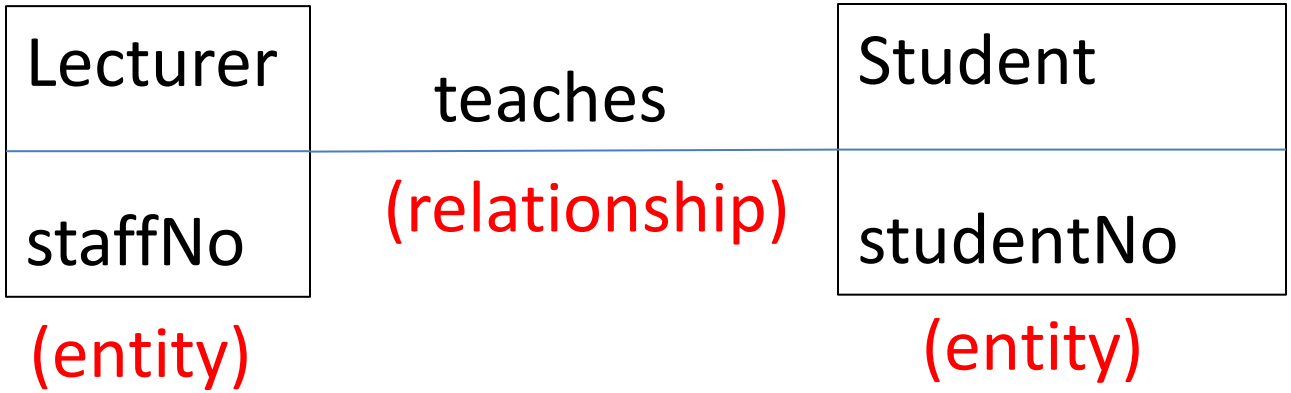
- **Data**: “facts and statistics collected together for reference or analysis” (Oxford dictionary)

- **Database**: a shared collection of **logically related data** (and a **description of this data**), designed to ***meet the information needs of an organization***.

- **System catalogue (metadata)**
  - description of data to enable program–data independence.

“Original database” by shinichi, flickr

- Logically related data comprises entities, attributes, and relationships of an organization's information.
- Entity: a distinct object in the organization that is to be represented in the database.
- E.g. person, place, thing, etc
- Attribute: a property that describes some aspect of the object that we wish to record.
- Relationship: an association between entities.



Database Management Systems (DBMS)

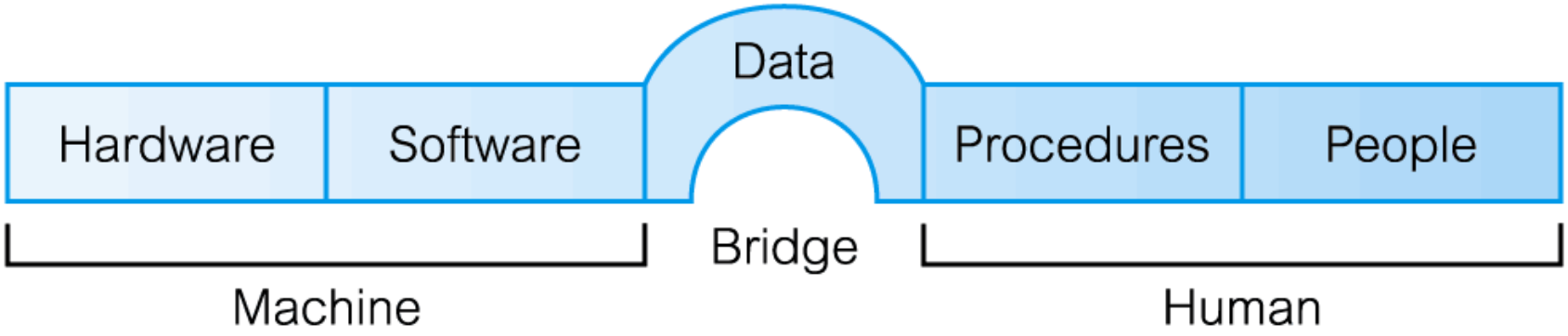
- DBMS: A software system
  - enables users to define, create, maintain, and control access to the database.

- (Database) application program: a computer program that interacts with database
  - issues appropriate requests (eg SQL statements) to the DBMS.

Key concepts

- Data model
  - A model is a representation of ‘real world’ objects and events, and their associations.
- Schema vs Data
  - The description of the database is the **database schema**.
  - **Data** is the actual information stored in the database.

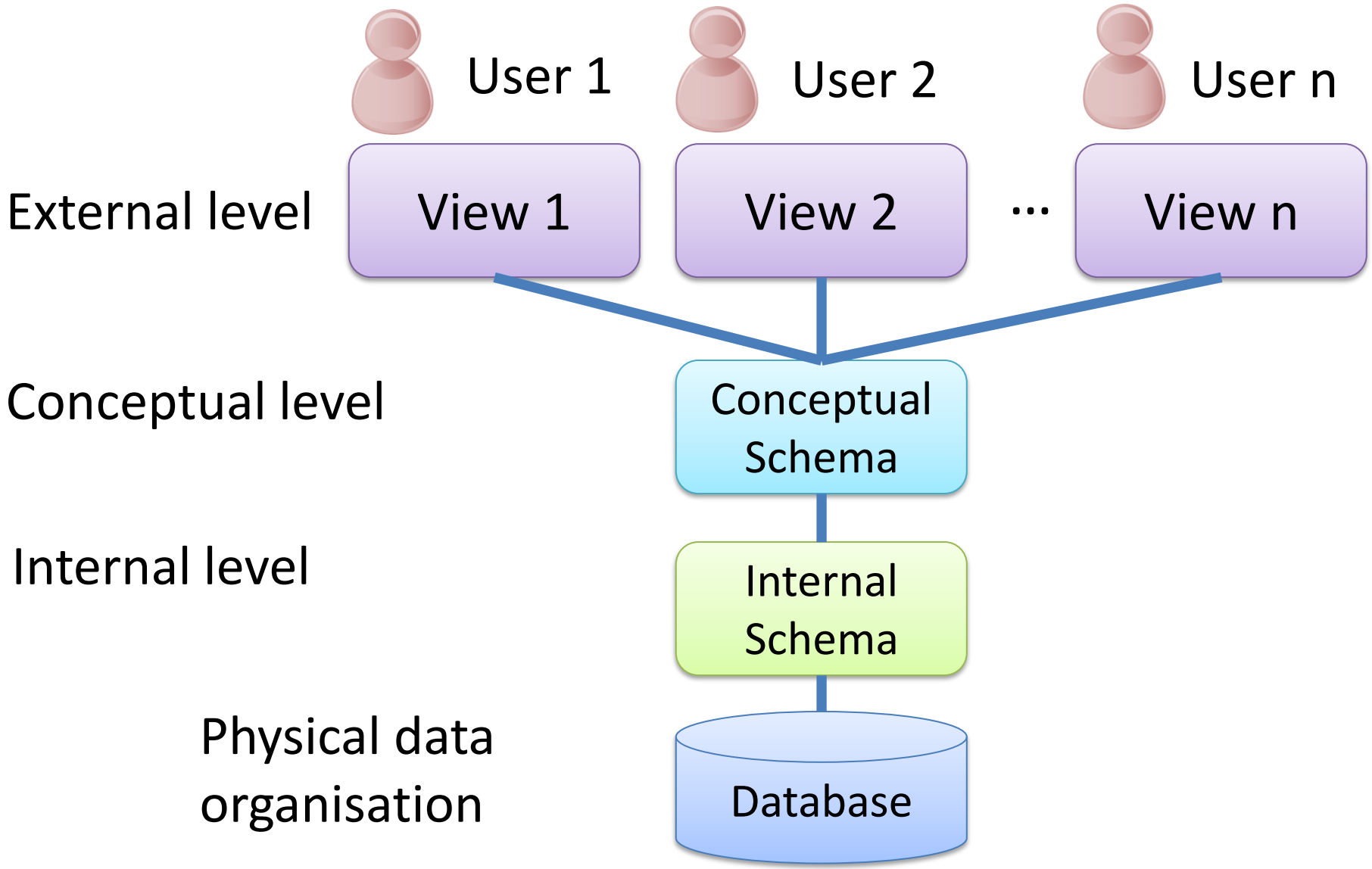
Components of DBMS Environment



Roles in the Database Environment

- Data Administrator (DA)
- Database Administrator (DBA)
- Database Designers (Logical and Physical)
- Application Developers
- End Users (naive and sophisticated)

Three level ANSI-SPARC architecture



The three-level ANSI-SPARC architecture

Objective of three level ANSI-SPARC architecture: Separate each user's view of the database from the way the database is physically represented.

- Users should be able to:
  - Access data but with customised view.
  - Change user's view without affecting other users.
- Users should not have to deal with physical database storage details.
  - Interaction with database independent of storage considerations.

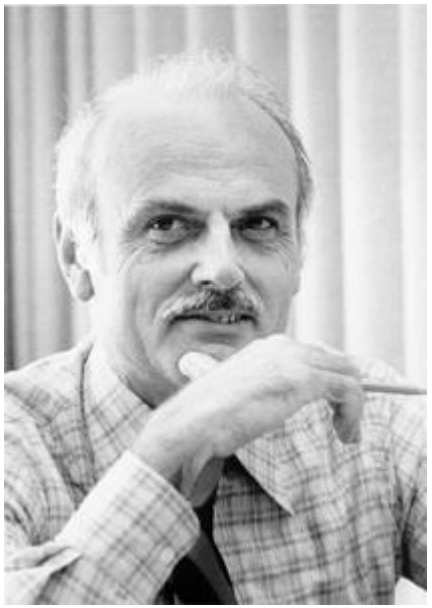
The three-level ANSI-SPARC architecture

Objective of three level ANSI-SPARC architecture: Separate each user's view of the database from the way the database is physically represented.

- Users' views **should be unaffected by** changes to the **physical aspects of storage**.
- Internal structure of the database **should be unaffected by** changes to the **physical aspects of storage**.
- The DBA should be able to **change the conceptual structure of the database** with **minimum affect to users' views**.

History of DBMS (1)

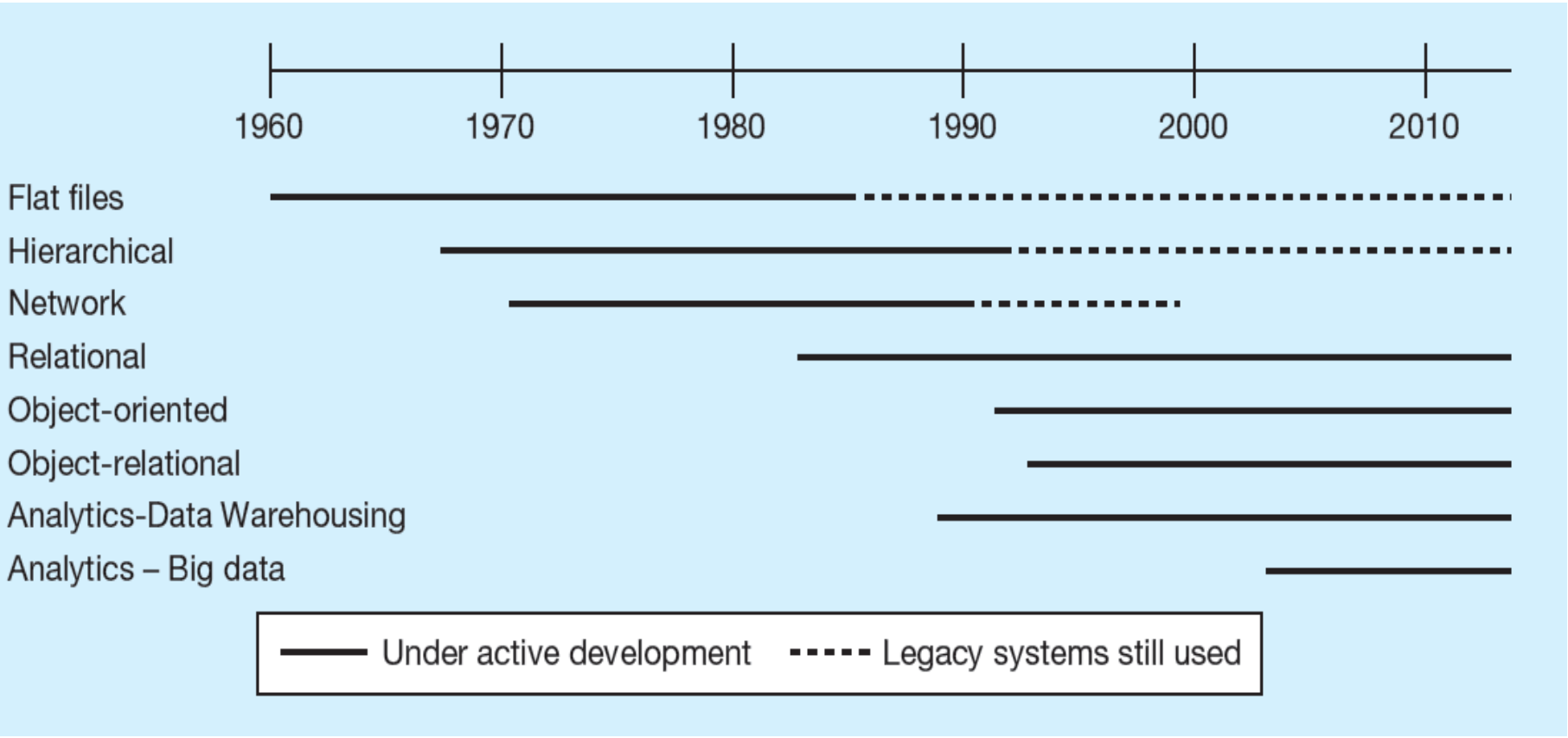
- Early 60s
  - Charles Bachman (Turing Award 1973)
  - Integrated Data Store
  - Network Data Model
  - CODASYL
- Late 60s
  - IBM
  - Information Management System (IMS)
  - Hierarchichal Data Model
- 70s
  - Edgar Codd, IBM (Turing Award 1981)
  - Relational Data Model



Edgar F. Codd - Wikipedia

History of DBMS (2)

- 80s
  - Relational Data Model
  - Standard Query Language (SQL)
  - Transaction Management (James Gray, Turing Award 1999)
- Now
  - Object-oriented Data Model
  - Data warehouse and data mining
  - Accessing databases through the web/internet
  - Multimedia data
  - Text data (information retrieval)
  - Structure of the data (XML,JSON)


1960
1970
1980
1990
2000
2010
Flat files
Hierarchical
Network
Relational
Object-oriented
Object-relational
Analytics-Data Warehousing
Analytics – Big data
Under active development
Legacy systems still used

Learning Outcomes

- Be able to explain the key terms:
  - Data, Database, DBMS, Database application programme, Data model, schema,
- Be able to explain the components and roles in database environment
- Be able to explain the three level ANSI-SPARC architecture and the purpose of it:
  - External level, Conceptual level, Internal level
- Understand the history of database development