COMP20070:
DATABASE AND INFORMATION SYSTEMS I

Lecturer: Dr. Michela Bertolotto (module coordinator)
Room# B2.21, Email: michela.bertolotto@ucd.ie

Outline of the course

Theory:

- **Introduction to DB and Information Systems**
- **DB:**
  - **Relational Model**
  - **SQL**
  - **Data base design (E-R model)**
  - **Normalization**

Practicals: **Use of MySQL**

Lecture and lab material available on moodle:
https://csmoodle.ucd.ie/moodle

Click on **COMP20070 Databases and Information Systems 2019-20**
Use your Student ID as login/username
Key/password: **20070-1920**

Suitable Books

“Database Systems – Concepts, Languages and Architectures”
Atzeni, Ceri, Paraboschi, Torlon - McGraw-Hill
Available free online at: http://dbbook.dia.uniroma3.it/dbbook.pdf

“Database Systems”
Connolly, Begg, Strachan - Addison Wesley

“Database Management Systems”
Ramakrishnan, Gehrke - McGraw-Hill

“Database Systems Concepts”
Silberschatz, Korth, Sudarshan - McGraw-Hill

Examples of Information Systems

- *Payroll system*: handles payments of employees in an organization
- *Sales order processing system*: handles supply/orders of products
- *Project planning and control system*: handles time/resource requirements of activities related to a project
- *Geographic information system*: handles geo-spatial information (maps, satellite images, statistical data, etc. associated with a given geographic area)

Types of Information Systems

- *Transaction processing systems*: process individual operations in a business (e.g., marketing, financial operations, etc.)
- *Decision-support systems*: aid decision making of management (e.g., where to build a factory, which products to sell etc.)
- *Expert (knowledge-based) systems*: attempt to simulate the role of the human “expert”; use knowledge base of a given domain to provide solutions (e.g., diagnose reasons for failure of a given business)
- *Office automation systems*: handle office applications such as email, meeting management, fax transmission, etc.
- Etc.

Information System

- Component of an organization that manages (gets, processes, stores, communicates) the information of interest

  - each organization has an information system, possibly not made explicit in its structure
  - usually, the information system operates in support to other components of the organization

- The very notion of information system is partly independent of its computerization; however, we are mainly interested in information systems that are, to a large extent, computerized

Management of Information

Information is handled and recorded according to various techniques:

- informal ideas
- natural language (written or spoken)
- drawings, diagrams,
- numbers
- codes

Structured information

- As activities become systematized, appropriate forms of organization and codification for information have been devised
- Look at information about people
  - in most countries a structure for the name has been introduced in the last few centuries
  - later, it was realized that it could be useful to keep track of birthdate and birthplace (and use them in order to identify people, together with the name)
  - more recently, social security numbers (or tax codes) have been introduced in order to obtain unique identification

Information and data

• In most computer- based systems (as well as in many other places) information is represented by means of data

– data raw facts, to be interpreted and correlated in order to provide information

• An example:

– “John Smith” and 25755 are a name (or, better, a string) and a number: two pieces of data

– if they are provided as a reply to a request: “Who is the dept head, and which is his/ her extension,” then we get information out of them

Components of Computer-Based Information Systems (IS)

- Database (DB)
- Database software
- Application software
- Computer hardware (e.g., storage media)
- Personnel using and developing the system

NOTE: the DB is a fundamental component

DB and DBMS

- DB: collection of logically related data of interest to the IS
- DB is managed by the Database Management System (DBMS) = software component that interacts with the DB and the user application programs

DataBase Management System (DBMS)

Software system able to

1. manage collections of data that are:

   – large (bigger, often much bigger, than the main memory available)

   – shared (used by various applications and users)

   – persistent (with a lifespan that is not limited to single executions of the programs that use them)

2. ensure their reliability (so preserving the database in case of hardware or software failure) and privacy (controlling accesses and authorizations). Like any software product, a dbms must be efficient (using the appropriate amount of resources, such as time and space) and effective (supporting the productivity of its users).

Sharing

• Most organizations have a structure (departments, divisions, …) and each component is interested in a portion of the information system

• The data of interest of the various components often overlap

• A database is an integrated resource, shared by various components

• Integration and sharing allow a reduction of redundancy and the consequent possibility of inconsistency

• Since sharing is never complete, DBMS provide support for privacy of data and access authorizations

• Sharing also requires that multiple accesses to data are suitably organized: concurrency control techniques are used

Example: Online Bookseller

Data about **books** (title, authors, categories, etc.), **customers, orders, order histories, trends and preferences**, etc.

Very large amounts of data (too big for memory)
- 100s of gigabytes, more if we keep all order histories over time, clickstream logs (for user preferences), images of book covers and sample pages , etc.

Persistent data
- Data outlives programs that operate on it

Safe data
- From hardware failures, software failures, power outages, ...
- From malicious users

Multi-user access
- Many people/programs access same database/data simultaneously

File-Based System

- Collection of application programs that perform services for end-users (e.g., production reports)
- Each program defines and manages its own data (stored in a file)
- Used as first attempt to computerize the traditional manual filing system within an organization (e.g., office, etc.)

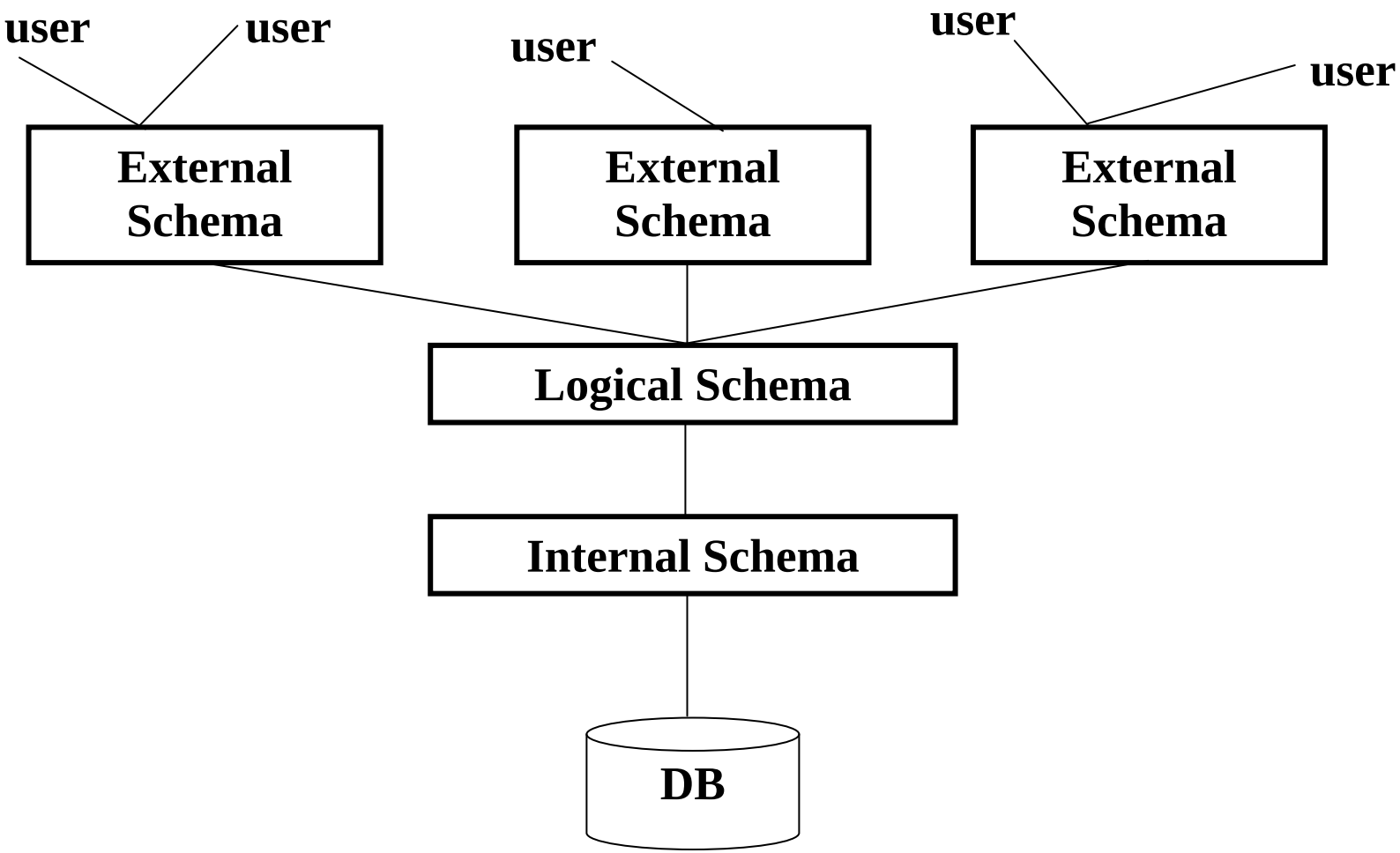
File Systems

- Allow to manage large and persistent sets of data
- Do not provide much support for data sharing

  ⇨ Replaced by DB systems

NOTE: DB systems use files to store the data but they provide users with abstract view of data, trying to hide storage and manipulation details

Standard (ANSI/SPARC) three-level architecture for a DBMS



ANSI/SPARC architecture: schemas

Logical schema: description of the whole database by means of the logical model adopted by the dbms

External schema: description of a portion of the database in a logical model (“ views,” possibly in different models)

Physical/internal schema: description of the implementation of the logical schema by means of physical storage structures

Data independence

Guaranteed by the multilevel architecture (which allows access only via the external level; could coincide with the logical one)

Two forms of *independence*

Physical : the logical and external level are independent of the the physical one; a relation is referred to always in the same way, regardless of its physical implementation (which could even vary over time)

Logical : the external level is independent of the logical one
- – addition of (or changes to) views do not require changes to the logical schema
- – changes to the logical schema need not affect the external schemas (provided that the definition of mappings are adjusted)

Database languages

Various forms:

1. Interactive textual languages, such as SQL

2. Interactive commands embedded in a host language (C, Java, etc.)

3. Interactive commands embedded in a ad-hoc development language, usually with additional features (for the production of forms, menus, reports, ...)

4. By means of non- textual user-friendly interfaces

A useful distinction

- Data definition language (DDL):
  Used to define the logical, external and physical schemas and access authorizations

- Data manipulation language (DML):
  Used for querying and updating database instances

DBMS: advantages and disadvantages

Pros

- data can be handled as a common resource
- centralized management and economy of scale
- availability of integrated services
- reduction of redundancies and inconsistencies
- data independence (favours the development and maintenance of applications)

Cons

- cost of the product (and associated tools) and of the migration
- difficulty in separating features and services (with possible lack of efficiency)