Coding for Data Science and Data Management
Module of Data Management

Introduction to relational databases

Stefano Montanelli
Department of Computer Science
Università degli Studi di Milano
stefano.montanelli@unimi.it

Database

- *A collection of data used to represent information of interest to an information system* (generic definition)
- *A collection of data managed by a DBMS* (more technical definition)
- Examples of databases are bank databases, university databases, biological databases

DBMS

- A DBMS - Data Base Management System is a software system able to support the definition, construction, manipulation, and sharing of persistent and large databases
  - *Persistent*: their lifespan is not limited to single executions of application programs that use them
  - *(Very) large*: much more than the main memory available (e.g., gigabyte)

DBMS

- The goal of a DBMS:
- To support efficient data retrieval
- To enforce data protection, so that only authorized users can access data in the database

DBMS functionalities

- **DB definition:** specification of type, structure, and integrity constraints for the data to be stored in the database
- **DB construction:** population of the database and storage of data in a persistent way
- **DB manipulation:** operations for i) data retrieval (query) and modification (insert, update, delete), and ii) report generation
- **DB sharing:** multiple applications and users at the same time access the stored data

Data abstraction

- A basic feature of the database approach to data management is that it provides some level of data abstraction
- Data abstraction means to hide details about data organization/storage and to highlight core data features, to improve their understanding
- Users perceive data at different levels of abstraction, suitable to their skills and jobs
- Data abstraction is achieved through data models

The relational data model

- The relational data model is based on the relation construct
- Data are organized as sets of homogeneous records that can be represented as tables

A database example

- For our examples, we rely on a movie database inspired to the IMDb (Internet Movie Database) application
- https://www.imdb.com/
- The database is about movies and people involved in the movie crew (e.g., actors, producers, directors)
- Further database contents are about movie ratings and advertisement

Example of table (db relation)

- Movie

id	official_title	year	length
1375666	Inception	2010	148
0816692	Interstellar	2014	169
3460252	The Hateful Eight	2015	167

- Person

id	first_name	last_name	birth_date
0634240	Christopher Johnathan James	Nolan	30/07/1970
0362766	Edward Thomas	Hardy	15/09/1977
0004266	Anne Jacqueline	Hathaway	12/11/1982

Schema and instance of a database

- In a database we have:
  - the schema, rather stable over time, that describes the structure of the database (intensional component)
    - The table headings in the example
    - The first step in the development of a database is the definition of the database schema
  - the instance, varying very rapidly, that is the actual data stored in the database (extensional component)
    - The table contents in the example
    - The subsequent step in the development of a database is the population of the database with data

Example of schema

- Movie

id	official_title	year	length

- Person

id	first_name	last_name	birth_date

Example of instances

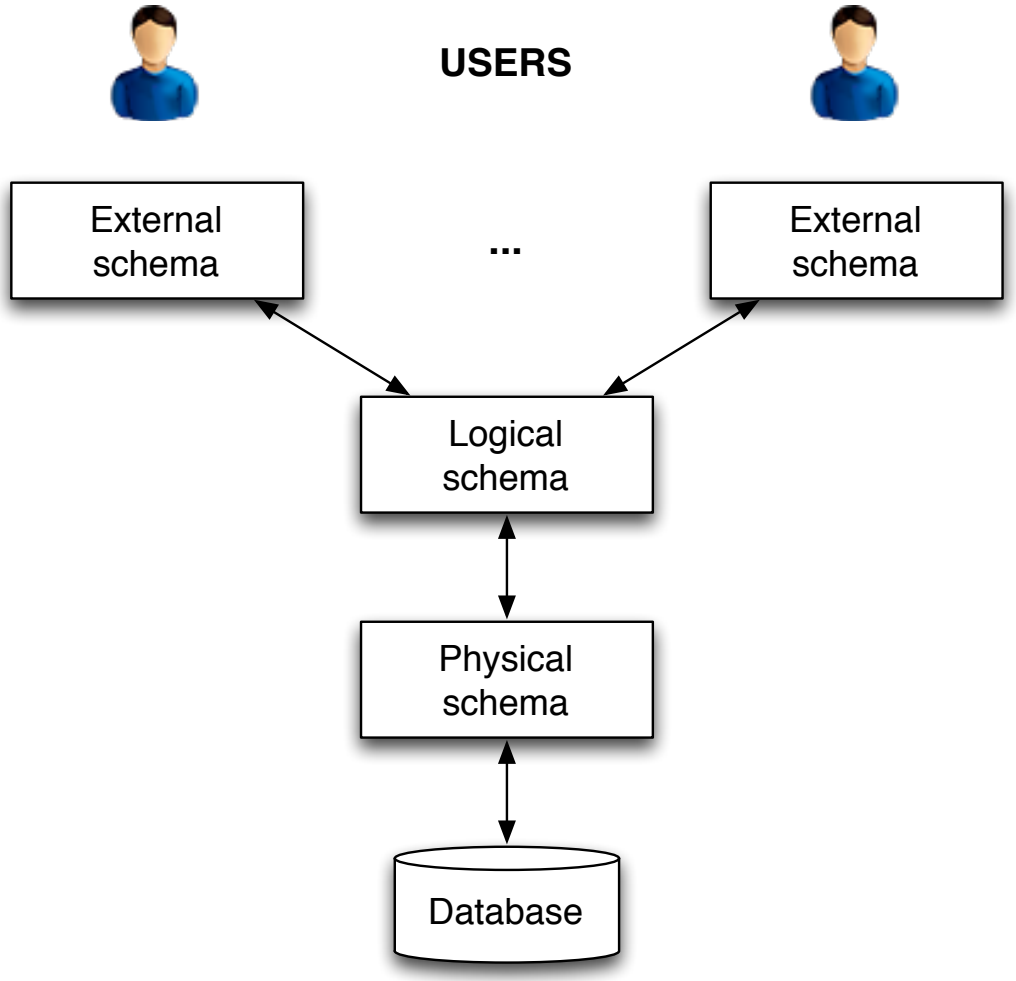
- Movie

1375666	Inception	2010	148
0816692	Interstellar	2014	169
3460252	The Hateful Eight	2015	167

- Person

0634240	Christopher Johnathan James	Nolan	30/07/1970
0362766	Edward Thomas	Hardy	15/09/1977
0004266	Anne Jacqueline	Hathaway	12/11/1982

Data abstraction layers in a db



Data abstraction layers in a db

- Logical schema: description of the whole database by means of the data model adopted by the DBMS (e.g., relational)
- External schema or View: description of a portion of the database for specific users
  - Multiple views on the same database are possible
- Physical schema: description of the implementation of the logical schema by means of physical storage structures

DBMS languages

- **DDL (Data Definition Language)**
  - Language for defining the database schema (logical, external, physical) and the access authorizations
- **DML (Data Manipulation Language)**
  - Language for querying and editing the database instances

Additional stuff

- Attached to the slide, you can find two spreadsheets (movie.xlsx, crew.xlsx) containing examples of data collections
- Using spreadsheets as databases is a solution that presents a lot of limitations
  - Data duplication
  - Data integrity violation
  - Possible errors during data entry
  - Missing checks over data types
  - Difficulties in data correlation across sheets
  - Difficulties in access control

Additional stuff

- Attached to the slide, you can find two spreadsheets (movie.xlsx, crew.xlsx) containing examples of data collections
- The use of a DBMS allows to overcome these limitations
- The relational data model supports the creation of a relational database to be managed by a DBMS

Additional stuff

- Thoughout the course, we are going to use the PostgreSQL relational DBMS
- PostgreSQL is available for free download:

  https://www.postgresql.org/download/