uc3m | Universidad Carlos III de Madrid

OpenCourseWare

Database

Tema 1. Introduction

Lourdes Moreno López
Paloma Martínez Fernández
José Luis Martínez Fernández
Rodrigo Alarcón García

HULAT
Human Language & Accessibility Technologies

Departamento de Informática

Content

- Introduction. Database Systems History
- Database. Database Management Systems (DBMS). Database Users
- Development Methodology. Data Modelling. Data Model

Fuente: https://pxhere.com/es/photo/1571969 (CC0 Dominio publico)

Learning objectives

- The student should be able to:
  - Know in a broad way, what is a database and its importance in nowadays
  - Know in a broad way, what is a Database Management System (DBMS)
  - Know in a broad way, the methodology for design a database

Introduction. Database Systems History

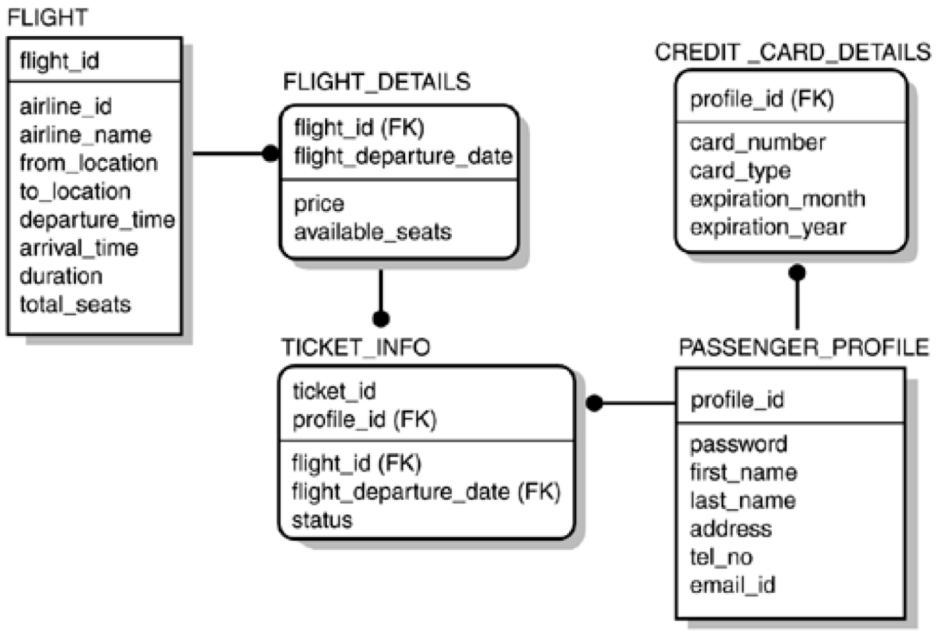
Introduction

- Database systems are an **essential component of our life**
- **Most of us make activities every day that involve a database, examples:**
  - Going to the bank to deposit or withdraw funds
  - Making a hotel or airline reservation
  - Accessing a computerized library catalog to search for a bibliographic item
  - Purchasing something online
- Because these activities involve a computer accessing a database.

Introduction

- Example: App for making an airline reservation (flight booking)

Fuente: https://www.pexels.com/es-es/foto/persona-que-usa-la-aplicacion-google-maps-a-traves-de-un-telefono-inteligente-android-negro-35969/



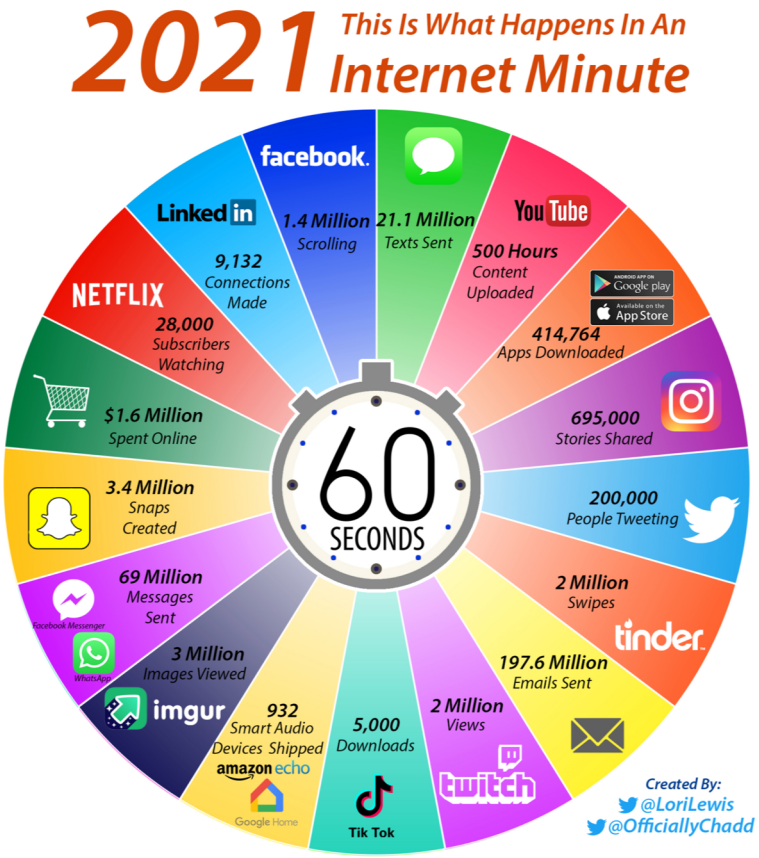
Introduction

- These applications are **traditional database applications**
- These database systems are called **SQL systems**
- The most of the information that is stored is either **textual or numeric**
- **Examples:**

  - Dates
  - Phone numbers
  - Social security numbers
  - Credit card numbers
  - Customer names
  - Addresses
  - Product names and numbers
  - Transaction information

Introduction

- The proliferation of social media Web sites, such as Facebook, Twitter, ... have led to **new applications of database systems**



Introduction

- Example: Personalized Recommendation System for e-Commerce

© Amazon

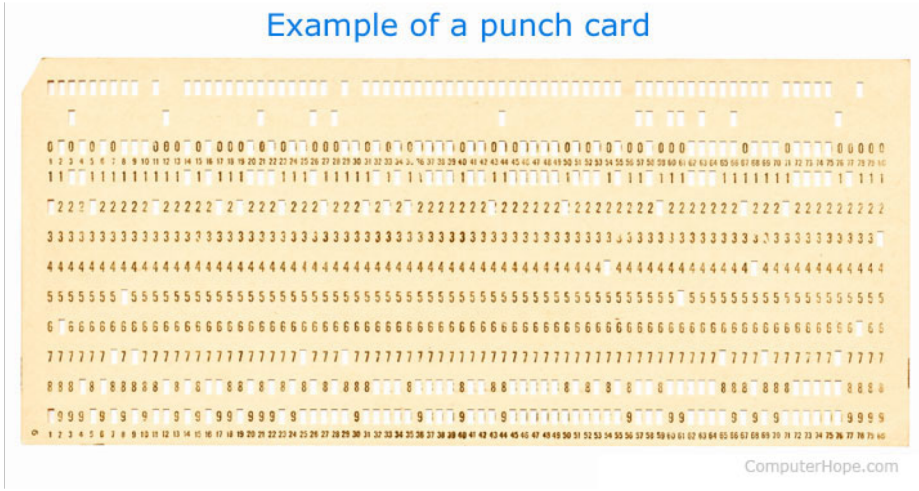
Introduction

- New types of database systems have been created referred to as **big data storage systems**, or **No-SQL systems**
- These database **systems store nontraditional data such as tweets, images, video, documents**
- Examples:

  - Text files
  - Reports
  - Email messages
  - Audio files
  - Video files
  - Images
  - Surveillance imagery

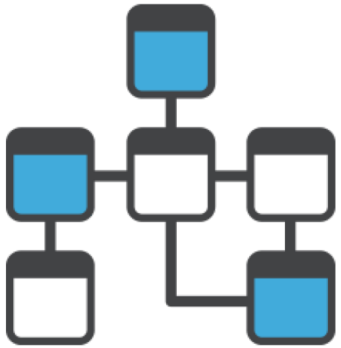
Database Systems History

- **1950s and early 1960s:**
  - Data processing using magnetic tapes for storage
    - Tapes provide only sequential access
  - Punched cards for input



Database Systems History

- **Late 1960s and 1970s:**
  - Hard disks allow direct access to data
  - Network and hierarchical data models in widespread use
  - **Ted Codd defines the relational data model**
    - Would win the ACM Turing Award for this work
    - IBM Research begins System R prototype
    - UC Berkeley begins Ingres prototype
  - High-performance (for the era) transaction processing



Database Systems History

- **1980s:**
  - **Research relational prototypes evolve into commercial systems**
    - SQL becomes industry standard
  - Parallel and distributed database systems
  - Object-oriented database systems

Database Systems History

- **1990s:**
  - **Large decision support and data-mining applications**
  - Large multi-terabyte data warehouses

=> Emergence of web commerce applications

Fuente: https://www.pexels.com/es-es/foto/telefono-inteligente-ordenador-portatil-macbook-tecnologia-6214479/

Database Systems History

- **2000s:**
  - XML and XQuery standards
  - Automated database administration
  - **Increasing use of highly parallel database systems => Web-scale distributed data storage systems**

Database Systems History

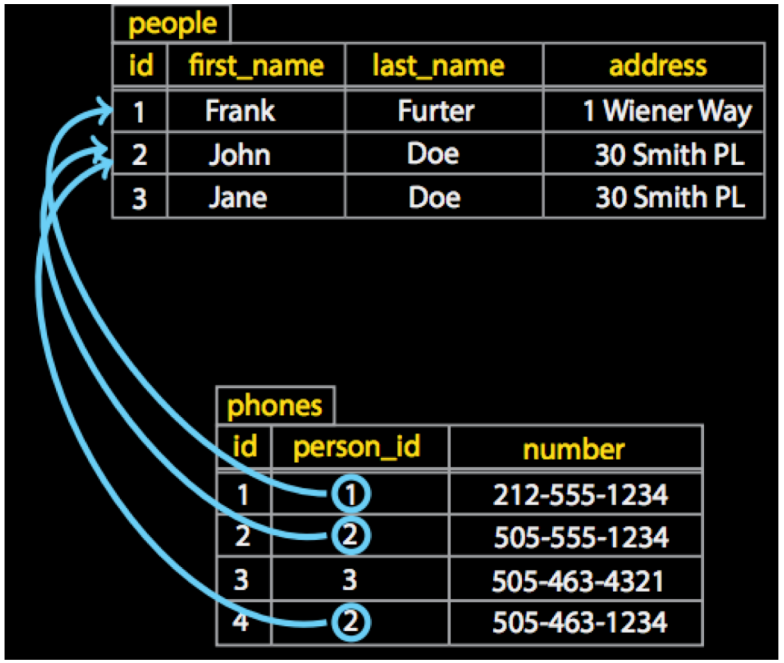
- **2010s:**
  - New types of database systems were necessary to manage these huge databases:
    - *systems that would provide **fast search** and **retrieval** as well as **reliable** and **safe storage of nontraditional types of data**, such as social media posts and tweets —*
  - **The proliferation of applications and platforms** such as social media Web sites, large e-commerce companies, Web search indexes, and cloud storage/backup led to a surge in the amount of data stored on large databases and massive servers

=> **Emergence of Big Data Storage Systems and NOSQL Databases**

Database. Database Management Systems (DBMS). Database Users

Database

- A database is a **collection of related data**
  - E. g.: names, telephone numbers, and addresses of the people



Database

- A database
  - **represents some aspect of the real world**, called the universe of discourse or miniworld
  - is a **logically coherent collection** of data with some **inherent meaning**
  - is designed and built with data for **a specific purpose**

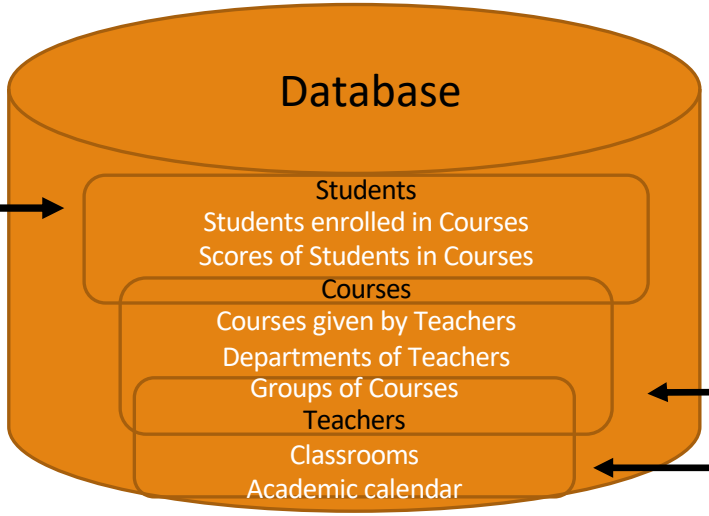
Database

E. g. Academic Management DB

Teaching Planning

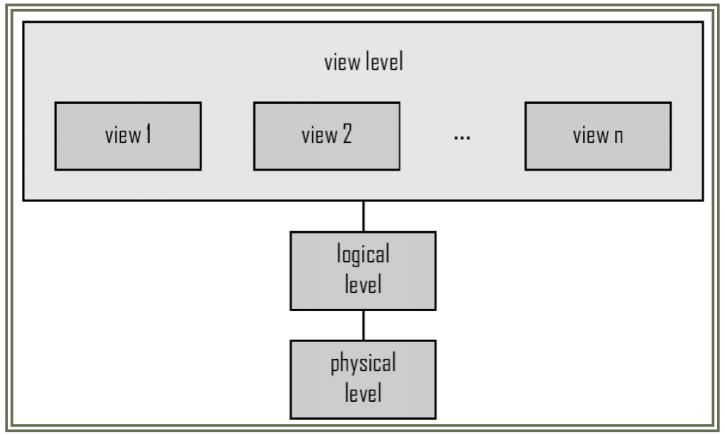
Scheduling

Enrollment Management



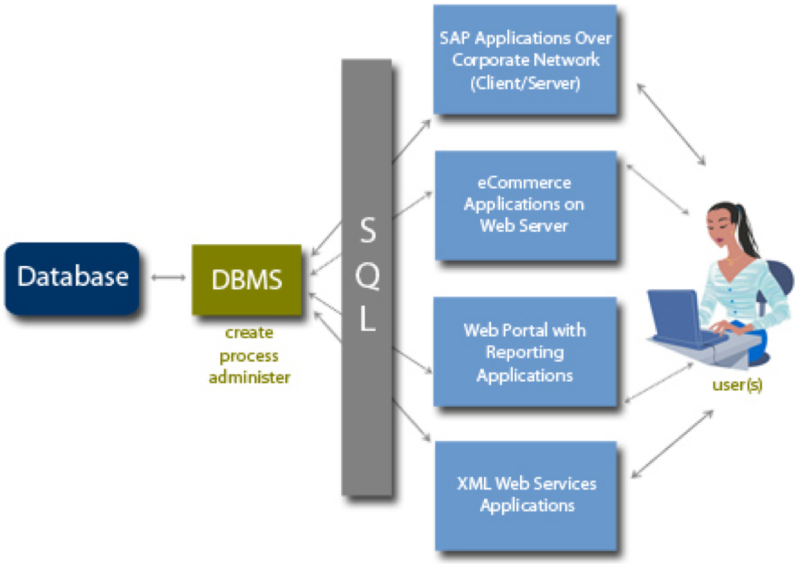
Database
Three-levels Architecture

- **View level:** includes a user views of application programs which hide details of data types.
- **Logical level**: describes data stored in database, and the relationships among the data.
- **Physical level**: describes how a record is stored.



DataBase Management Systems (DBMS)

- The **DataBase Management System (DBMS)** is a general-purpose software system that **facilitates the processes of defining, constructing, manipulating, and sharing databases among various users and applications.**



DataBase Management Systems (DBMS)

- Functions provided by the DBMS include:
  - manipulating a database includes functions such as:
    - **querying the database** to retrieve specific data
    - **updating the database** to reflect changes in the miniworld
    - **generating reports** from the data
  - protecting the database and **maintaining** it over a long period of time
  - **a security protection** against unauthorized or malicious access

Database Users

- Database Administrators
- Database Designers
- End Users
- System analysts and Application Programmers (Software Engineers)

Fuente: https://www.pexels.com/es-es/foto/foto-de-mujer-con-laptop-3194518/

Database Users
Database Administrators

- **Database Administrators** coordinate all the activities of the database system
  - Administrator has a good understanding of the enterprise's information resources and needs.

Fuente: https://www.pexels.com/es-es/foto/mujer-de-pie-mientras-lleva-portatil-1181354/

Database Users
Database Administrators

- **Database administrator's** duties include:
  - Defining storage structure and access method
  - Modifying schema and physical organization
  - Granting users authority to access the database
  - Backing up data
  - Monitoring performance and responding to changes

Database Users
Database Designers

- **Database designers** are responsible for:
  - Identifying the data to be stored in the database and for choosing appropriate structures to represent and store this data
  - These tasks are mostly undertaken before the database is actually implemented
  - Communicate with all database users in order to understand their requirements and to create a design that meets these requirements

Database Users
System analysts and Application Programmers (Software Engineers)

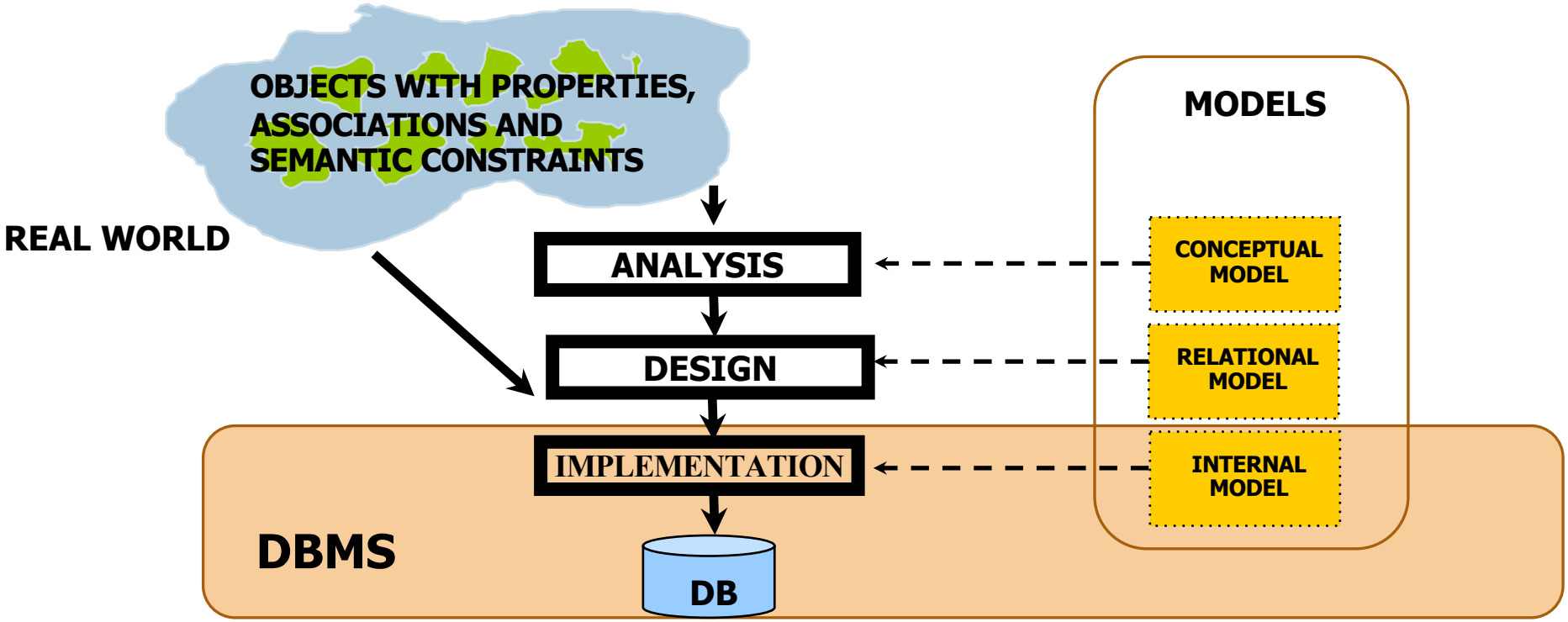
- **System analysts** determine the requirements of end-users and develop specifications for these requirements
- **Application programmers** implement these specifications as programs; then they test, debug, document, and maintain these canned transactions.
- Analysts and programmers should be familiar with the full range of capabilities provided by the DBMS to accomplish their task

Database Users
End Users

- **Users** are differentiated by the way they expect to interact with the system
  - Application programmers – interact with system through DML calls
  - Sophisticated users – form requests in a database query language
  - Specialized users – write specialized database applications that do not fit into the traditional data processing framework
  - Naïve users – invoke one of the permanent application programs that have been written previously

Development Methodology. Data Modelling. Data Model

Development Methodology



Data Model
Model, Schema, Instance

- A **Data model** provides a set of concepts, rules and conventions that allow us to describe the structure of a database and manipulate data stored in it
- A **Schema** is a logical structure of the DB that is specified during database design and is not expected to change frequently
- An **instance** is the actual content of the database at a particular point in time.

Data Model
Model, Schema, Instance

Database Schema

STUDENT

Name	Student_number	Class	Major

COURSE

Course_name	Course_number	Credit_hours	Department

PREREQUISITE

Course_number	Prerequisite_number

SECTION

Section_identifier	Course_number	Semester	Year	Instructor

GRADE_REPORT

Student_number	Section_identifier	Grade

Instance

STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Data Model
Structured Query Language

- To communicate with DBMS we need a language:
  1. To describe DB schemas → **Data Definition Language (DLL)**
  2. To access and manipulate the data organized by the appropriate data model → **Data Manipulation Language (DML)**

  In Relational Databases is **SQL (Structured Query Language)**

Data Model

- Now we know that to build a DB we need:

1. To **know the elements of a model to define** a database
2. To learn **how to apply the model to obtain** a diagram representing the information to be stored in the DB

=> (next class) **Topic 2.1 Relational Model in order to design a database**

Bibliography

- Connolly, Thomas M, Begg, Carolyn E. *Database systems: a practical approach to design, implementation, and management.* Addison Wesley. 2015
- Elmasri, Ramez, Navathe, Sham. *Fundamentals of database systems*. Pearson Addison Wesley. 2017