# **Database Foundations**

### **Option Presentation - M1 MOSIG**

Silviu Maniu, LIG, Univ. Grenoble Alpes

## Databases

**Database**: *organized collection* of data; *regular structure,* to capture a part of the *real world* 

- Started with tabular data: records (employee, census, banking)
- Nowadays more varied data: medical databases, Web search logs, sensor logs, financial data (time-series)

				A To	ur Paï ordre	d'une d	efpece	de fo	rmula	ire fa	it en	
Chastellenie de												
	Parroisse de S. N. Ruë Fr.											
		Maifons	Noms & qualitez.	Hommes	Femmes	Grands Garçons	Giandes Filles	Petits Garçons	Petites Filles	Vallets	Servantes	Nomb. des Familles
and a strength of		I	Mr le Conte de Seigneur du lieu, y refidant actuellement.	I	I	2	0	0	D	6	2	12





### **Database Management Systems**

**DBMS -** dedicated *software* to manage databases: *data storage*, *data transactions*, *querying data* 

#### Applications:

- Enterprise management: accounting, supply chain, staff, order
- Web applications: online retailers (Amazon), streaming providers, search engines
- Web backends: CMS, emails, certificates
- Scientific data: medicine, biology, etc.



#### **Genealogy of Relational Database Management Systems**





#### Key to lines and symbols

Acquisition

DBMS name (Company)

v9.2006 Versions – Discontinued 🕅 Branch (intellectual and/or code)

e) Crossing lines have no special semantics

Felix Naumann, Jana Bauckmann, Claudia Exeler, Jan-Peer Rudolph, Fabian Tschirschnitz Contact - Hasso Plattner Institut, University of Potsdam, felix.naumann@hpi.de Design - Alexander Sandt Grafik-Design, Hamburg Version 6.0 - October 2018 https://hpi.de/naumann/projects/rdbms-genealogy.html

### **Database Foundations** Objectives

- 1. To learn the **fundamental concepts** behind database models: relations, constraints, dependencies
- 2. To learn how to **design databases** for your application
- 3. To learn how to query and program databases
- 4. To learn how classical database models apply to current data models

### **Database Foundations** Contents

- Applications of databases
- Data models: relational model, constraints, functional dependencies
- **Database design**: entity/relationship model, normalisation
- Database programming: relational algebra+SQL, procedural languages, triggers, views
- (Bonus) "Modern" data models: JSON (document stores), column stores, vector databases

"Cours-TD": lecture + practical lab (exercises, implementation)