

# An introduction to database management system

Fabrice Rossi

CEREMADE  
Université Paris Dauphine

2020

## Informal definitions

1. Collection of measured and recorded values
2. Set of values of entities (subjects) with respect to variables (attributes)

## Informal definitions

1. Collection of measured and recorded values
2. Set of values of entities (subjects) with respect to variables (attributes)

## Actors

id	first_name	last_name	gender	film_count
933	Lewis	Abernathy	M	1
2547	Andrew	Adamson	M	1
2700	William	Addy	M	1
2898	Seth (I)	Adkins	M	1
2925	Charles (I)	Adler	M	1
3226	Casey	Affleck	M	1
4306	Shigekazu	Aida	M	1
4856	Julliet	Akinyi	M	1

## Actors

id	first_name	last_name	gender	film_count
529149	Pamela	Abdy	F	1
530401	Kelly	Adkins	F	1
533380	Karen (I)	Allen	F	1
533504	Sara	Allen	F	1
533710	Jennifer	Allswang	F	1
534915	Suzy	Amis	F	1
535586	Jo (I)	Anderson	F	1
535754	Sharon	Anderson	F	1

## Entities and attributes

- ▶ entities: actors
- ▶ attributes: first name, last name, gender, etc.

# Database?

## A possible definition

A persistent and structured collection of related data.

# Database?

## A possible definition

A persistent and structured collection of related data.

## Important aspects

- ▶ persistent: stored on a permanent support (e.g. hard drive)
- ▶ structured: not a data dump
- ▶ related: unrelated data should be in different databases

# Example

id	first_name	last_name	gender	film_count
933	Lewis	Abernathy	M	1
2547	Andrew	Adamson	M	1
2700	William	Addy	M	1
2898	Seth (I)	Adkins	M	1
2925	Charles (I)	Adler	M	1

id	name	year	rank
192017	Little Mermaid, The	1989	7.30
300229	Shrek	2001	8.10
306032	Snatch.	2000	7.90
333856	Titanic	1997	6.90

actor_id	movie_id	role
933	333856	Lewis Bodine
2547	300229	Duloc Mascot
2700	306032	Tyrone
2898	333856	Slovakian three-year-old boy
2925	192017	Additional Voices

# Example

## Remarks

- ▶ multiple entities
  - ▶ actors
  - ▶ movies
  - ▶ roles
- ▶ attributes can be specific to some entities (first name, title, etc.)
- ▶ relation between entities
  - ▶ ids with naming conventions
  - ▶ is role an entity or a relationship?

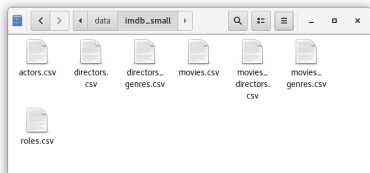
id	first_name	last_name	gender	film_count
933	Lewis	Abernathy	M	1
2547	Andrew	Adamson	M	1
2700	William	Addy	M	1
2898	Seth (I)	Adkins	M	1
2925	Charles (I)	Adler	M	1

id	name	year	rank
192017	Little Mermaid, The	1989	7.30
300229	Shrek	2001	8.10
306032	Snatch.	2000	7.90
333856	Titanic	1997	6.90

actor_id	movie_id	role
933	333856	Lewis Bodine
2547	300229	Duloc Mascot
2700	306032	Tyrone
2898	333856	Slovakian three-year-old boy
2925	192017	Additional Voices



# A database in CSV files



	A	B	C	D	E
1	id	first_name	last_name	gender	film_count
2	933	Lewis	Abernathy	M	1
3	2547	Andrew	Adamson	M	1
4	2700	William	Addy	M	1
5	2898	Seth (I)	Adkins	M	1
6	2925	Charles (I)	Adler	M	1
7	3226	Casey	Affleck	M	1
8	4306	Shigekazu	Aida	M	1
9	4856	Julliet	Akinyi	M	1
10	6005	Henri	Alciatore	M	1
11	6990	Dean	Alexandrou	M	1

```
"id","first_name","last_name","gender","film_count"  
933,"Lewis","Abernathy","M",1  
2547,"Andrew","Adamson","M",1  
2700,"William","Addy","M",1  
2898,"Seth (I)","Adkins","M",1  
2925,"Charles (I)","Adler","M",1  
3226,"Casey","Affleck","M",1  
4306,"Shigekazu","Aida","M",1  
4856,"Julliet","Akinyi","M",1  
6005,"Henri","Alciatore","M",1
```

# Some limitations of file storage

## Size

- ▶ almost all programs must load the data in the RAM
- ▶ strong limit on data size

## Shared data

- ▶ shared physical storage
- ▶ crude access control

## Consistency/Integrity

- ▶ no support for rules in storage
- ▶ data can become inconsistent

## Atomicity

- ▶ no support for atomic operation
- ▶ transient inconsistent states

## Integrity

List integrity constraints that must be fulfilled by the movie database.

## Atomic operations

Describe a natural modification of the movie database that needs atomicity.

## Definition

A DBMS is a software system used to manage databases providing

- ▶ access to the data while maintaining integrity, security and efficiency
- ▶ a high level api to build applications on the databases

## Architecture

- ▶ DBMS follow the client-server model
- ▶ the DBMS itself the server
- ▶ clients connect concurrently to the server to access the databases

## Abstraction

- ▶ DBMS provide data abstraction
- ▶ different levels
  - ▶ physical level (how): the way data are stored on the hard drives
  - ▶ logical level (what): what data are stored and what relationship they have, *independently from the physical level*
  - ▶ view level: what is shown to some specific users

## Schemas

- ▶ DBMS store metadata: data (schemas) about the data (instances)
- ▶ schemas describe the structure of a given database
- ▶ levels are specified by schemas (physical schema, logical schema, external schema)

# Example

id	first_name	last_name	gender	film_count
933	Lewis	Abernathy	M	1
2547	Andrew	Adamson	M	1
2700	William	Addy	M	1
2898	Seth (I)	Adkins	M	1
2925	Charles (I)	Adler	M	1

## Physical level

```
"id", "first_name", "last_name", "gender", "film_count"  
933, "Lewis", "Abernathy", "M", 1  
2547, "Andrew", "Adamson", "M", 1  
2700, "William", "Addy", "M", 1  
2898, "Seth (I)", "Adkins", "M", 1  
2925, "Charles (I)", "Adler", "M", 1  
3226, "Casey", "Affleck", "M", 1  
4306, "Shigekazu", "Aida", "M", 1  
4856, "Julliet", "Akinyi", "M", 1  
6005, "Henri", "Alciatore", "M", 1
```

## Illustration only!

This is generally not the way data are stored in DBMS

# Example

id	first_name	last_name	gender	film_count
933	Lewis	Abernathy	M	1
2547	Andrew	Adamson	M	1
2700	William	Addy	M	1
2898	Seth (I)	Adkins	M	1
2925	Charles (I)	Adler	M	1

## Physical level

```
id;first_name;last_name;gender;film_count
933;Lewis;Abernathy;M;1
2547;Andrew;Adamson;M;1
2700;William;Addy;M;1
2898;Seth (I);Adkins;M;1
2925;Charles (I);Adler;M;1
3226;Casey;Affleck;M;1
4306;Shigekazu;Aida;M;1
4856;Julliet;Akinyi;M;1
6005;Henri;Alciatore;M;1
```

## Illustration only!

This is generally not the way data are stored in DBMS

# Example

## Logical level

- ▶ an actor is described by
  - ▶ a unique id (integer)
  - ▶ a first name and a last name (strings)
  - ▶ a gender (M or F)
  - ▶ a film count (integer)
- ▶ a movie is described by
  - ▶ a unique id (integer)
  - ▶ a title (name, string)
  - ▶ a release year (valid year)
  - ▶ a grade (rank, decimal number)
- ▶ actors play roles in movies

id	first_name	last_name	gender	film_count
933	Lewis	Abernathy	M	1
2547	Andrew	Adamson	M	1
2700	William	Addy	M	1
2898	Seth (I)	Adkins	M	1
2925	Charles (I)	Adler	M	1

id	name	year	rank
192017	Little Mermaid, The	1989	7.30
300229	Shrek	2001	8.10
306032	Snatch.	2000	7.90
333856	Titanic	1997	6.90

actor_id	movie_id	role
933	333856	Lewis Bodine
2547	300229	Duloc Mascot
2700	306032	Tyrone
2898	333856	Slovakian three-year-old boy
2925	192017	Additional Voices



# Example

## View level

id	first_name	last_name	role	name
529149	Pamela	Abdy	Casting Assistant	Garden State
530401	Kelly	Adkins	Charmaine, Lifted Security Gu	Ocean's Eleven
533380	Karen (I)	Allen	Katy	Animal House
533504	Sara	Allen	Spatula Neighbor	UHF
533710	Jennifer	Allswang	Earring customer	Planes, Trains & Automobiles
534915	Suzy	Amis	Lizzy Calvert	Titanic
535586	Jo (I)	Anderson	Julia Ann Mercer	JFK
535754	Sharon	Anderson	Morning Show Hostess	Fargo
536750	Jennifer	Aniston	Joanna	Office Space
537264	Tracey	Antosiweicz	Hooker	Garden State
539150	Rosanna	Arquette	Jody	Pulp Fiction
540103	Jennifer	Aspen	Nina	Vanilla Sky
540321	Nao	Asuka	Premium Fantasy Woman	Lost in Translation
541008	Agnes	Aurelio	District Attorney staff membe	JFK
541111	Julie (I)	Austin	Mrs. Morrison	Braveheart

## Two languages

- ▶ data definition language (DDL)
  - ▶ used to specify the schemas
  - ▶ data type and relationships
  - ▶ additional constraints such as positivity of an attribute
  - ▶ database access authorizations
  - ▶ data storage
- ▶ data manipulation language (DML)
  - ▶ used to query a database
  - ▶ specifies computation on a database, including modifications of the data

## DBMS architecture

- ▶ storage manager
  - ▶ handles saving data and meta data on the storage system
  - ▶ manages part of the concurrency aspects
- ▶ query processor
  - ▶ core part of the DBMS
  - ▶ handles computation on the database (e.g. search, insertion, deletion, modification, aggregation)
  - ▶ optimize queries for efficiency
- ▶ high level tools
  - ▶ meta data management
  - ▶ connection and security manager

## Data model categorization

- ▶ old approaches: hierarchical DBMS and network DBMS
- ▶ current standard: relational DBMS
  - ▶ based on a relational data model
  - ▶ typical based on SQL (Structured Query Language, sequel)
- ▶ niche approach: object-oriented DBMS
- ▶ large scale or unstructured data: NoSQL
  - ▶ not-only SQL (or more specifically not relational)
  - ▶ document stores
  - ▶ graph oriented databases
  - ▶ column oriented databases

## An introduction to Relational DBMS

## An introduction to Relational DBMS

### What's missing?

- ▶ A lot!
- ▶ DBMS are a very rich and complex subject
  - ▶ numerous theoretical aspects
  - ▶ getting under the hood is yet another subject
  - ▶ practical aspects are also very complex

- ▶ November 2020: initial version



This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

<http://creativecommons.org/licenses/by-sa/4.0/>



Last git commit: 2020-11-23

By: Fabrice Rossi (Fabrice.Rossi@apiacoa.org)

Git hash: 312a0636ceb585db2da88a95e73b59651b34a3fb