

*SMI6U05L : Bases de données*  
*Une introduction aux BDs et aux SGBDs*

Luigi Santocanale  
LIS, Aix-Marseille Université

Contenu basé sur le cours de Rémi Eyraud

# Plan

Informations administratives

Introduction

Historique

Modélisation des données

# Plan

Informations administratives

Introduction

Historique

Modélisation des données

# Organisation du cours

- Page web du cours :

<https://pageperso.lis-lab.fr/~luigi.santocanale/teaching/BD/>

- Calcul de la note finale :  $NF = (E+CC)/2$
- CC : deux TP notés
- Attention : pas de rattrapage pour le CC

# Objectifs du cours

- Apprendre le langage SQL (SQLite)
- Savoir concevoir et administrer une BD
- Apprécier les fondements mathématiques de la théorie des bases de données

# *Structure du cours*

- Introduction
- Le modèle relationnel
- Le langage SQL
- Conception de BD

## Repères bibliographiques

- La “bible” francophone : *Bases de données*, Georges Gardarin, éditions Eyrolles, 2003. Disponible depuis le site web de l’auteur (avec plein d’autres documents) :

[http://georges.gardarin.free.fr/Cours\\_Total/IndexTotal.html](http://georges.gardarin.free.fr/Cours_Total/IndexTotal.html)

- *Foundations of Databases*, par Serge Abiteboul, Richard Hull, Victor Vianu, 1995.
- *Using SQLite*, par Jay A. Kreibich, 2010.

# Plan

Informations administratives

**Introduction**

Historique

Modélisation des données

# Où trouve-t-on des bases de données ?

- Gestion des entreprises
  - ▶ commerce et marketing
  - ▶ gestion des stocks
  - ▶ administrations
- Sciences :
  - ▶ médecine
  - ▶ biologie
  - ▶ astronomie
  - ▶ mathématiques, e.g.
    - ▶ MathsciNet, ZentralBlatt,
    - ▶ arXiv, Hal
    - ▶ OEIS
- Audiovisuel
- Internet
- Gestion des données personnelles :
  - ▶ agenda
  - ▶ répertoire
  - ▶ smartphone
  - ▶ bibliothèque
- Loisirs
- Art

# Qu'est-ce qu'une base de données ?

Une BD est<sup>1</sup> :

- un ensemble **structuré**
- de données **persistantes**
- représentant une **réalité extérieure au système**
  
- **partagé** par plusieurs applications
- d'une **même entreprise**

## Qu'est-ce qu'une base de données (suite) ?

Une **base de donnée** regroupe,

- un ensemble de données,
- au sein d'un **stockage informatique**,
- de façon **suffisamment organisée**,  
pour que ces données puissent être **facilement** :
  - ▶ retrouvées
  - ▶ interrogées
  - ▶ enrichies
  - ▶ mises à jour
  - ▶ supprimées
- sans perturber la **cohésion** de l'ensemble.

## *Exemples et contre-exemples*

1. Toute source de donnée
2. Un fichier texte
3. Un ensemble de fichiers textes
4. Un site internet quelconque
5. Le web dans son ensemble
6. Un répertoire de téléphone portable
7. L'ensemble des informations utilisées par la SNCF pour l'achat d'un billet de train.

## *Exemples et contre-exemples*

1. Toute source de donnée
2. Un fichier texte
3. Un ensemble de fichiers textes
4. Un site internet quelconque
5. Le web dans son ensemble
6. Un répertoire de téléphone portable
7. L'ensemble des informations utilisées par la SNCF pour l'achat d'un billet de train.

## *Exemples et contre-exemples*

1. Toute source de donnée
2. Un fichier texte
3. Un ensemble de fichiers textes
4. Un site internet quelconque
5. Le web dans son ensemble
6. Un répertoire de téléphone portable
7. L'ensemble des informations utilisées par la SNCF pour l'achat d'un billet de train.

## *Exemples et contre-exemples*

1. Toute source de donnée
2. Un fichier texte
3. Un ensemble de fichiers textes
4. Un site internet quelconque
5. Le web dans son ensemble
6. Un répertoire de téléphone portable
7. L'ensemble des informations utilisées par la SNCF pour l'achat d'un billet de train.

# Usagers des BD

3 catégories d'usagers des bases de données :

- Les usagers interactifs
- Les programmeurs d'applications
- Les administrateurs

## *Les usagers interactifs*

- Une personne quelconque (ne sachant même pas qu'il utilise une BD).
  - ▶ Ex. : quelqu'un consultant des horaires de trains sur le site de la SNCF
- Une personne visualisant le contenu d'une base à l'aide d'une interface graphique conviviale (Exemple : web).

# Les programmeurs d'applications

- Construisent les interfaces destinées aux usagers interactifs.
- Sont spécialistes d'un *langage de définition et d'interrogation de données* (par exemple, le langage SQL).
- Exemple : un développeur web.

# Les administrateurs

- Définissent (parfois) et maintiennent la base de données.
- Possèdent plus de droits sur la base.
- 2 types :
  - ▶ concepteur
  - ▶ administrateur
- Exemple : vous, à la fin de ce cours 😊

# Types de Base de Données

- 4 types :
  - ▶ BD personnelles : 10 à 100 Ko
  - ▶ BD professionnelles : 100 Ko à 1 To (téraoctet = 1000 Go)
  - ▶ Très grande BD : de 1 To à 1 Po : (petaoctet = 1 milliard Go)
  - ▶ Très très très grande : plus de 1 Po
  
- Exemples :
  - ▶ Répertoire téléphonique
  - ▶ Données clients d'une PME
  - ▶ SNCF (10 To en 2016)
  - ▶ Facebook ( > 100 Po, + 500 To par jour)

## Système de Gestion de Bases de Données

En anglais : DBMS, DataBase Management System

Ensemble de logiciels dont le(s)

- But : simplifier la tâche des usagers (tout les types).
- Rôle : aide au développement et à l'exploitation des BD
- Caractéristiques :
  - ▶ définition et programmation de BD
  - ▶ optimisation des accès à la base
  - ▶ contrôle et sécurité de BD
- Fourni : le(s) langage(s) nécessaire(s) (déclaratifs ou procéduraux) à la manipulation des données (lecture & écriture).

## SGBD (suite)

- Gère les couches **logiques ET physiques**
- Certains SGBD intègrent des outils connexes :
  - ▶ générateurs (de rapport, d'applications, d'interface, etc.)
  - ▶ utilitaires (calcul de statistiques, ...)
  - ▶ interface web
  - ▶ outils de fouille de données
- Un administrateur doit parfaitement connaître le SGBD de sa base.

# Plan

Informations administratives

Introduction

**Historique**

Modélisation des données

# Bref historique

- 3 grandes phases :
  - ▶ La préhistoire (1950-1970)
  - ▶ La révolution (1970)
  - ▶ L'ère post-révolutionnaire

## Historique : la pré-préhistoire

- Basée sur les fichiers
- Chaque application définit et gère ses propres fichiers, reliés par des pointeurs
- **Positif** : premières idées
- **Négatif** :
  - ▶ structure du stockage visible dans le code (struct en C)
  - ▶ dépendance données/programme (portabilité nulle)
  - ▶ redondance des données (taille, maintenance, ...)
  - ▶ pas de vrai SGBD

## Historique : la préhistoire (1965-1970)

- Approche *navigationale*
- Systèmes hiérarchiques et réseaux
- A base de fichiers gérés par arborescence de pointeurs
- **Positif** : fin de la redondance, premier vrai SGBD
- **Négatif** : mélange logique/physique, accès procéduraux (programmation lourde), “*one-record-at-a-time*”
- Toujours utilisés (SGBD : IDS2, IMS2, PHOLAS)

# Historique : la révolution

- Approche **relationnelle**
- Premier SGBD relationnel en 1980
- Idée majeure :
  - ▶ séparation application/implémentation,
  - ▶ distinction QUOI/COMMENT
  - ▶ **indépendance des données**
- **Positif** :
  - ▶ simplicité des concepts de base
  - ▶ existence d'un **cadre théorique** simple et bien établi
  - ▶ modèle de données (respect de l'intégrité)
  - ▶ approche déclarative de haut niveau (SQL, Structured Query Language, 1974)
  - ▶ SGBD performants (accès concurrents, optimisation, pannes)
  - ▶ portabilité des bases et des applications

## Historique : la révolution (suite)

- **Négatif :**
  - ▶ types de données prédéfinis (inadaptés aux données complexes)
  - ▶ modélisation conceptuelle pauvre
  - ▶ trop grande séparation données/opération (les procédures ne font pas parties du modèle)
  - ▶ théorie relationnelle pas entièrement exploitée (ex : contraintes d'intégrité)
  - ▶ grand écart entre SQL et langages de programmation
- **La plupart des bases actuelles sont relationnelles**  
(SGBDR : Oracle, PostgreSQL, SQLite, DB2, Informix, ...)

## Historique : la nouvelle génération

- Approche post-relationnelle (Not Only SQL - NoSQL)
- Pour des applications de + en + complexes
- Plein de catégories :
  - ▶ approche **orientée objet**
  - ▶ approche par **documents semi-structurés** (XML)
  - ▶ données hétérogènes complexes (vidéos, images, sons, etc.)
- Permet une intégration des idées de l'intelligence artificielle (e.g. base de donnée déductives)
- Pas autant utilisée que les SGBDR, mais en progression constante ?

# Plan

Informations administratives

Introduction

Historique

Modélisation des données

# Modélisation des données

- Principe premier :  
séparer description des données et manipulation par des programmes.
- **Description** : spécification de la structure des données
- **Manipulation** : opérations
  - ▶ d'interrogation,
  - ▶ d'insertion,
  - ▶ mise à jour,
  - ▶ suppression.

## *Décrire les données : définitions*

- Données = **ensembles** reliés par des associations (= **ensembles de liens**).

Exemple : base de données cinématographique

- ▶ Ensemble des films
  - ▶ Ensemble des acteurs
  - ▶ Association entre les films et les acteurs  
(ensemble de liens film ↔ acteurs)
- **Décrire les données c'est définir ces ensembles via leurs propriétés/caractéristiques**

## Décrire les données : exemples

- Le type *Entier*, muni des opérations standards de l'arithmétique, est un **type de donnée élémentaire**.  
12 est une instance de ce type.
- Le type *Film* avec ses **propriétés** :  
*titre, réalisateur, langues, acteurs, durée, année, ...*  
est un **type composé**.
- Le film  
("Usual suspects", "Bryan Singer", [anglais, français],  
["Byrne", "Spacey", "Del Toro"], 108, 1994, ...)  
est une **instance** du type *Film*.

# Décrire les données : définitions

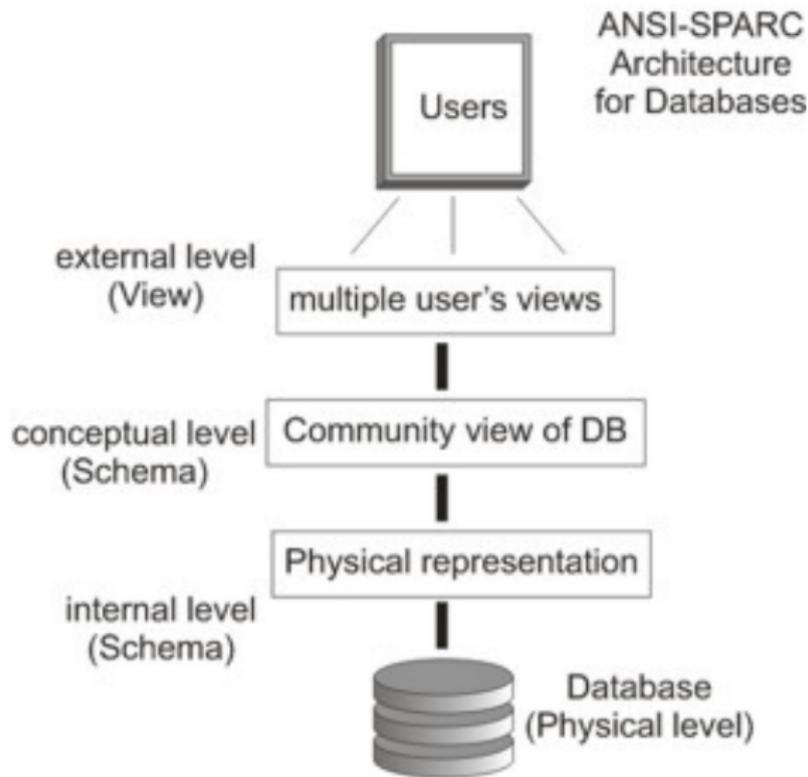
- 4 grandes notions :
  - ▶ **Type de donnée** : construction informatique spécifiant un ensemble de données via leurs caractéristiques et les opérations applicables.
  - ▶ **Instance** : une donnée particulière appartenant à un type spécifié + identifiant.  
Créée par *insertion* dans la base
  - ▶ **Modèle de description** : spécification d'un ensemble de types, parfois graphique.
  - ▶ **Schéma** : mise en oeuvre du modèle de description à l'aide du **Langage de Description de Données** (LDD).

## Les niveaux d'abstraction

### ANSI/X3/SPARC :

- 3 niveaux spécifient clairement le passage de la vision de *chaque utilisateur à la représentation en machine.*
- Facilite la construction, la maintenance, l'évolution des BD
- Important de comprendre ces 3 niveaux lors de la phase de conception.
- Les SGBD actuels ont tendance à ne mettre en oeuvre que 2 niveaux

# ANSI-SPARC<sup>2</sup> (1975, Bachmann 1974)



2. American National Standards Institute, Standards Planning And Requirements Committee

## *Le niveau conceptuel*

- Niveau le plus important, détaillé avant même la réalisation de la base.
- Il conditionne :
  - ▶ la qualité de la base
  - ▶ les développements au sein du SGBD
  - ▶ la perception que les utilisateurs auront de la BD
- il correspond à l'explication et la description sémantique des concepts et des relations du domaine modélisé.
- Indépendant d'un SGBD, il reflète une réalité cohérente du domaine ET des acteurs du domaine.

## Le niveau conceptuel (suite)

Le schéma conceptuel d'une BD couvre (mini) :

- La référence aux **concepts élémentaires**.  
Ex. : la durée d'un film est un entier positif correspondant au nombre de minutes.
- La **définition** des types composés : regroupement des propriétés nommées d'un ensemble d'objets.  
Cet ensemble est soit un concept du monde modélisé, soit une association entre concepts. Ex. : à suivre
- Les **règles** que doivent respectées les données au cours de leur vie.  
Ex. : un Film doit obligatoirement avoir un titre, un acteur doit avoir joué dans au moins un Film, etc.

## Le niveau conceptuel, exemple

Ex. de définition de types composés et d'associations

- Types d'objets :

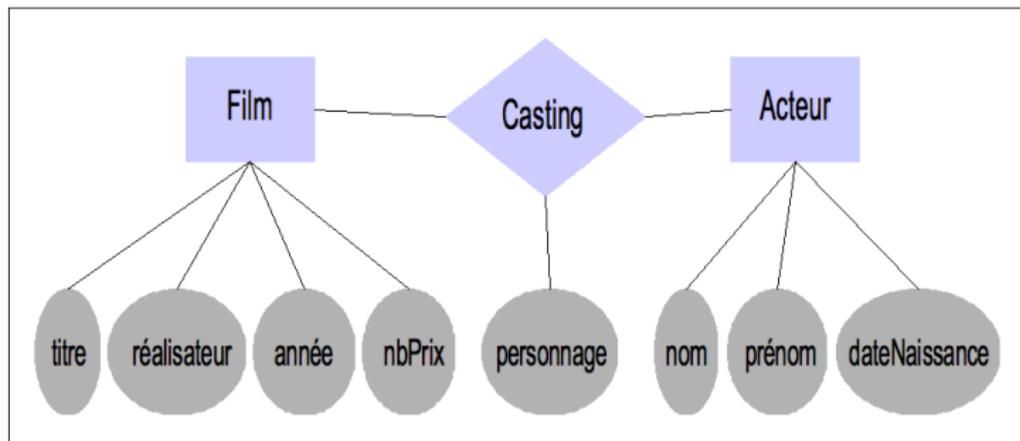
```
FILM(titre, année, durée, nbPrix, réalisateur)  
ACTEUR(nom, prénom, dateNaissance, photo)
```

- Type d'association :

```
CASTING(FILM, ACTEUR, personnage)
```

Note : toutes les relations ne sont pas explicités.  
Par exemple : deux acteurs peuvent être en relation parce qu'ils ont joué dans le même film.

## Langages graphiques pour le MCC, exemple



## *Le niveau interne (Cours de Master 1 Info)*

- Niveau le plus proche de l'implémentation
- Concerne la façon dont les données sont stockées.
- La définition du schéma interne est fortement lié au choix du SGBD.
- Contenu :
  - ▶ les fichiers contenant les données
  - ▶ les *articles* de ces fichiers  
(leur longueur, mode de placement, etc.)
  - ▶ le mode d'accès à ces articles  
(pointeur/index, etc.)
- *Dérive du schéma conceptuel.*

## *Le niveau externe (fin du cours, peut-être)*

- Le niveau externe spécifie des **vues** différentes associées aux différents groupes d'utilisateurs.
- Eloigné du niveau interne, dépendant du niveau conceptuel.
- On définit donc **1 ou plusieurs** schémas externes.
- Chaque schéma définit la manière dont un utilisateur perçoit la base dans ses applications.
- Il spécifie les droits d'accès.

## Le niveau externe

Exemples de schémas externes possibles pour le schéma conceptuel de filmographie :



Schéma externe 1

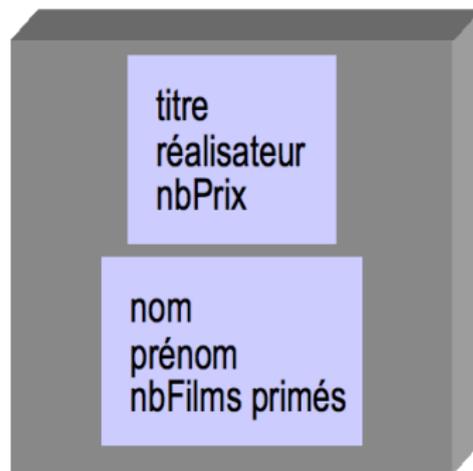
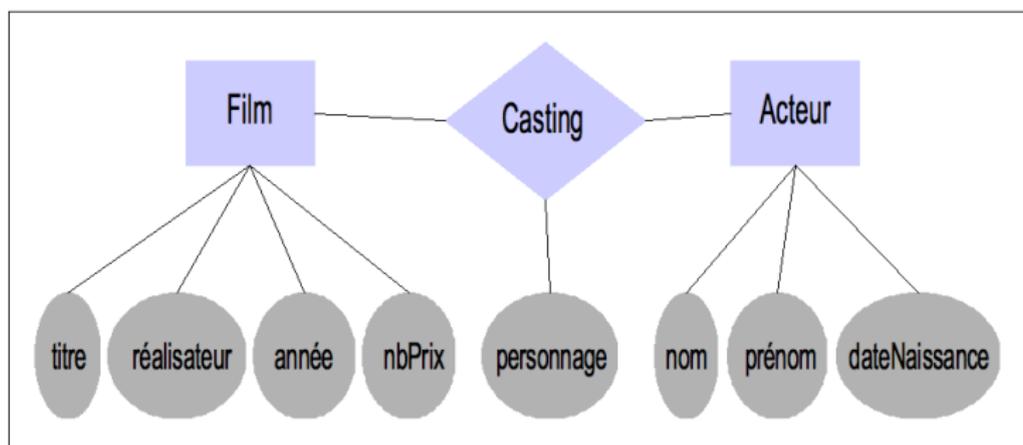


Schéma externe 2

## *Le modèle entité-association* *(Brown 1975, Chen 1976)*

- But : réaliser un **schéma conceptuel**
- Permet : description visuelle des **entités** (=types d'objets) et de liens entre elles (= **relations**)
- Notion principale : **agrégation** (= la construction d'un objet complexe/composé à l'aide d'objets plus élémentaires).
- monde = distinction entre agrégations de données élémentaires (entités) et des liaisons entre ces entités (=relations)
- Entité = agrégation d'**attributs** (nommés et typés)

## Modèle entité-association : exemple



Spécifier un schéma conceptuel revient (presque) à fournir un modèle entité-relation et à le préciser par diverses informations.