

# BASES DE DONNEES

Éléments de conception et d'exploitation d'une Base de Données

Equipe Bases de Données, F. De Marchi

LYON 1 - UFR Informatique - Laboratoire LIRIS

31 janvier 2021

- 1 Le modèle relationnel
  - Intuition
  - Structure et contraintes de clés
  - Traduction E/A vers Relationnel
  - Langages
  - Les transactions

- 1 Le modèle relationnel
  - Intuition
  - Structure et contraintes de clés
  - Traduction E/A vers Relationnel
  - Langages
  - Les transactions

# Qu'est ce qu'un modèle de données ?

Un formalisme commun pour représenter et interroger les données

- Une structure pour représenter les données
- Des contraintes pour garantir la sémantique du cahier des charges
- Des langages pour interroger et modifier les données

Les niveaux logiques et physiques sont dédiés à un modèle de données commun. Le niveau externe est plus libre, peu correspondre à des exportations vers d'autres modèles ou formats.

# Les principaux modèles de données

- *Modèle relationnel*

- *Structure* : ensembles de **relations**, qui sont des ensembles de *tuples*.
- *Contraintes* : principalement les **clés primaires** (identifiants de tuples, **dépendances fonctionnelles**) et les **clés étrangères** (liens entre les relations, **dépendances d'inclusion**)
- *Manipulation* : **algèbre relationnelle**, **calcul relationnel**, **SQL**, clauses de Horn sans récursion.

- *Modèle déductif*

- *Structure* : celle du modèle relationnel à laquelle on ajoute des règles de déduction.
- *Contraintes* : les mêmes que le modèle relationnel
- *Manipulation* : langages logiques comme *Datalog*. Contrairement aux langages du modèle relationnel, il admet la récursivité.

# Les principaux modèles de données

- *Modèle de graphe (e.g., RDF)*
  - *Structure* : graphe orienté et étiqueté, on enregistre les triplets  $(s; p; o)$
  - *Contraintes* : un identifiant pour chaque nœud, un mécanisme de référence entre des nœuds
  - *Manipulation* : parcours de graphes, SPARQL.
- *Modèle hiérarchique (e.g., XML)*
  - *Structure* : arborescente (forêt d'arbre)
  - *Contraintes* : un identifiant pour chaque nœud, un mécanisme de référence entre des nœuds
  - *Manipulation* : navigation hiérarchique, XPATH, XQUERY.

- *Modèle objet*

- *Structure* : logique objet, soit des classes, des objets, des attributs et des méthodes. Peut être vu comme un graphe orienté.
- *Contraintes* : identifiant pour les objets, référence entre objets.
- *Manipulation* : extensions de SQL comme OSQL ou OQL.

- *Modèle Entité/Association*

- *Structure* : Entités (avec des attributs) et associations entre des entités.
- *Contraintes* : identifiants, cardinalités sur les associations
- *Manipulation* : aucun (c'est un langage de modélisation, qui n'est pas implémenté tel que).

Les bases "de production" sont très très majoritairement relationnelles. Des représentations objet ou XML sont bien souvent intégrées comme des "surcouches" de SGBD relationnels.

# Notations

- $\{A; B; C|D\}$  sera noté  $ABC$
- Les notations  $ABC$  et  $BCA$  sont équivalentes puisque l'ordre n'a pas d'importance.
- L'ensemble  $AABC$  est le même que  $ABC$ , puisque l'élément  $A$  ne peut exister qu'en un seul exemplaire.
- Soit les ensembles  $X = ABC$  et  $Y = BD$ , alors leur union  $X \cup Y$  sera noté  $XY = ABCD$ .



# Le modèle relationnel : Intuition

<i>Étudiants</i>	<i>NUMETUD</i>	<i>NOMETUD</i>	<i>PRENOMETUD</i>	<i>AGE</i>	<i>FORMATION</i>
	28	Codd	Edgar	20	3
	32	Armstrong	William	20	4
	53	Fagin	Ronald	19	3
	107	Bunneman	Peter	18	3

<i>Enseignants</i>	<i>NUMENS</i>	<i>NOMENS</i>	<i>PRENOMENS</i>	<i>GRADE</i>
	5050	Tarjan	Robert	PR
	2123	Mannila	Heikki	MCF
	3434	Papadimitriou	Spiros	PR
	1470	Bagan	Guillaume	CR

<i>Encadre</i>	<i>NUMENS</i>	<i>NUMETUD</i>	<i>DATE</i>
	5050	53	2005
	3434	28	2020
	5050	28	2015
	2123	32	2019

TABLE – Exemple de bases de données

- 1 Le modèle relationnel
  - Intuition
  - Structure et contraintes de clés
  - Traduction E/A vers Relationnel
  - Langages
  - Les transactions

# Le modèle relationnel : structure

- Soit  $\mathcal{U}$ , un ensemble infini dénombrable de *noms d'attributs* ou simplement *attributs*, appelé **univers**.
- Soit  $\mathcal{D}$  un ensemble infini dénombrable de **constantes** (ou valeurs).
- Soit  $A \in \mathcal{U}$  un *attribut*, le **domaine** de  $A$  est un sous-ensemble de  $\mathcal{D}$ , noté  $DOM(A)$ .

## Schémas de relations et de bases de données

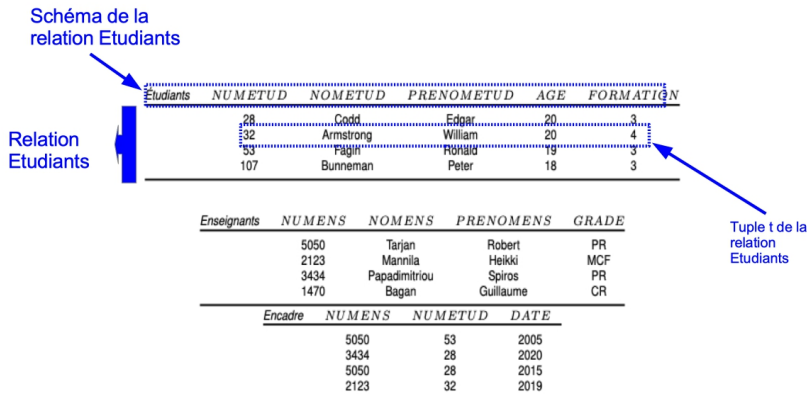
- Un **schéma de relation**  $R$  est un ensemble fini d'attributs (donc  $R \subseteq \mathcal{U}$ ).
- Un **schéma de base de données**  $\mathbf{R}$  est un ensemble fini de schémas de relation.

# Le modèle relationnel : structure

## Tuple, Relation et Base de Données

- Soit  $R = A_1 \dots A_n$  un schéma de relation. Un **tuple** sur  $R$  est un élément du produit cartésien  $DOM(A_1) \times \dots \times DOM(A_n)$ .
- Une **relation**  $r$  sur  $R$  (appelée aussi instance ou vulgairement table) est un ensemble *fini* de tuples.
- Une **base de données**  $\mathbf{d}$  sur un schéma de base de données  $\mathbf{R} = \{R_1, \dots, R_n\}$  est un ensemble de relations  $\{r_1, \dots, r_n\}$  définies sur les schéma de relation de  $\mathbf{R}$ .
  
- Si  $t$  est un tuple défini sur un schéma de relation  $R$ , et  $X$  un sous-ensemble de  $R$ , on peut restreindre  $t$  à  $X$  en utilisant la **projection**, notée  $t[X]$  qui est la restriction de  $r$  à  $X$ .

# synthèse



- Les deux schémas de relation forment le schéma de la BD
- Les deux relations forment la BD
- $t[NOMETUD, PRENOMETUD] = (\text{Armstrong}, \text{William})$

# Modèle relationnel : contraintes fondamentales

## Deux contraintes de base

- Une **clé** est un ensemble d'attributs où les doublons sont interdits. C|D'est donc un identifiant unique pour les tuples d'une relation.
  - Exemple : *NUMETUD* est une clé de la relation *Etudiants*.
  - Une relation a **obligatoirement une clé**. A défaut, c'est la concaténation de l'ensemble des attributs.
- Une **clé étrangère** est un ensemble d'attributs qui permet de faire référence à une clé d'une autre table, et donc à un tuple précis d'une autre table.
  - Exemple : *NUMETUD* est une clé étrangère de la relation *Encadre* faisant référence à l'attribut *NUMETUD* de *Etudiants*.

Ce mécanisme permet entre autre de reconstituer les informations d'une relation à l'autre.

- 1 Le modèle relationnel
  - Intuition
  - Structure et contraintes de clés
  - Taduction E/A vers Relationnel
  - Langages
  - Les transactions

# Traduction d'un schéma EA en relationnel

## Traduction des entités

Chaque entité devient une relation.

- Les identifiants de l'entité deviennent des clés
- Entités spécialisées et entités faibles
  - La clé de l'entité forte est ajoutée en clé étrangère
  - Pour les entités spécialisées, cette clé étrangère est la clé.
  - Pour les entités faibles, cette clé étrangère PLUS la clé locale constituent la clé.



# Traduction d'un schéma EA en relationnel

## Traduction des associations

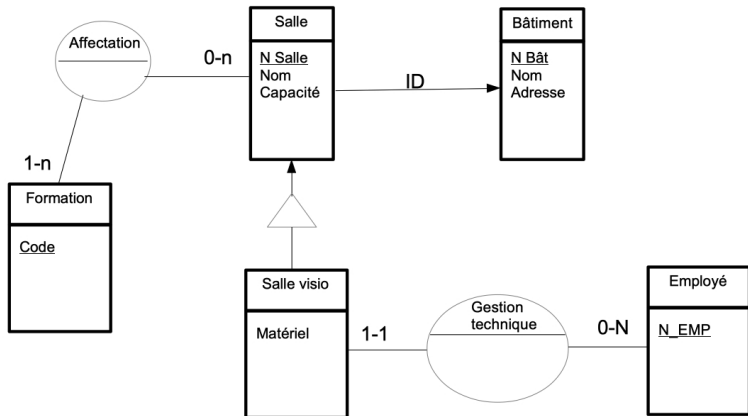
Chaque association devient une relation.

- Les attributs de l'association sont attributs de la relation
- On ajoute comme attributs une duplication de la clé de chaque entité participante, avec une contrainte de clé étrangère.
- La définition de la clé dépend de la connectivité
  - la clé étrangère correspondant à chaque entité qui possède une connectivité de 1 devient une clé.
  - Si toutes les connectivités sont N, alors la clé est composée de toutes les clés étrangères des entités participantes.

## Solution alternative

Lorsque la connectivité d'une entité à une association est 1, une autre solution consiste à placer une clé étrangère dans cette entité qui fait référence à la clé de l'autre entité.

# Exercice



# Traduction d'un schéma EA en relationnel

## Le travail n'est pas fini !

- On peut simplifier, abandonner des relations, ... En justifiant et en documentant.
- Intégrer les contraintes avancées (T,X,...) ou non schématisées en E/A
  - Les contraintes du relationnel sont plus expressives que E/A
  - Des contraintes complexes peuvent être programmées sur le serveur BD
  - En dernier recours, elles devront être programmées dans les applications.
- Une conception plus fine peut être poursuivie en relationnel
  - Normalisation et optimisation du schéma (compétences avancées)
  - Retour éventuel pour faire évoluer le schéma E/A

Maintenir un diagramme E/A "commenté" équivalent au modèle logique implémenté

- 1 Le modèle relationnel
  - Intuition
  - Structure et contraintes de clés
  - Traduction E/A vers Relationnel
  - Langages
  - Les transactions

## Modèle relationnel : Langages

- Langages théoriques d'interrogation des données :
  - Langage procédural : algèbre relationnelle
  - Langage déclaratif (logique) : calcul relationnel (tuple ou domaine), Datalog.
- Langage implémenté pour l'interrogation et la manipulation des données
  - Structured Query Language (SQL)
  - SQL est l'implémentation du calcul relationnel pour la partie interrogation.
  - Mais nombreuses extensions et possibilité d'ajouter des fonctions.

### On manipule des relations

Une requête prend une plusieurs relations en entrée, et retourne une relation. On peut donc interroger des sous-requêtes ; toutefois seul Datalog est récursif.

# Algèbre Relationnelle

Soient des relations  $r, r_1, r_2$  définies respectivement sur les schémas  $R, R_1, R_2$ .

- Sélection ( $\sigma_C(r)$ ) filtre les tuples de  $r$  selon la condition  $C|D$ .
  - On obtient un sous ensemble de  $r$
- Projection ( $\pi_X(r)$ ) ne conserve que les attributs de  $X$  ( $X \subseteq R$ )
  - On écarte les colonnes qui n'intéressent pas la requête
- Jointure ( $r_1 \bowtie r_2$ ) "combine" entre eux les tuples de  $r_1$  et  $r_2$ 
  - On combine les tuples qui sont égaux sur  $R_1 \cap R_2$ .
- Renommage ( $\rho_{[X/X']}(r)$ ) opération de renommage des attributs.

## Algèbre Relationnelle (suite)

Puisque les relations sont des ensembles de tuples, on bénéficie en plus de tous les opérateurs ensemblistes.

- A condition d'avoir  $R = S$  :
  - Différence ( $r_1 \setminus r_2$ ).
  - Intersection ( $r_1 \cap r_2$ ).
  - Union ( $r_1 \cup r_2$ ).
- A condition d'avoir  $R \cap S = \emptyset$
- • Produit cartésien ( $r_1 \times r_2$ ). La relation obtenue est sur le schéma  $R_1 \cup R_2$ .

Pour les conditions sur le schéma, on peut les "forcer" par le renommage préalable

# Exemples

Répondre aux questions suivants par une requête algébrique.

- Quel est le prenom et le nom de tous les étudiants
- Quel est le prenom et le nom des enseignants qui sont PR
- Quel est le nom des enseignants qui encadrent l'étudiant 53 ?
- Quels est le num des étudiants qui n'ont pas d'encadrant.
- Lister le prenom et nom de tous les étudiants et enseignants



# Exercices

Répondre aux questions suivants par une requête algébrique.

- Quel est le prenom et le nom de tous les étudiants

$$\bullet \pi_{PRENOMETUD, NOMETUD}(Etudiants)$$

- Quel est le prenom et le nom des enseignants qui sont PR

$$\bullet \pi_{PRENOMENS, NOMETUD}((\sigma_{GRADE='PR'}(Enseignants)))$$

- Quel est le nom des enseignants qui encadrent l'étudiant 53 ?

$$\bullet \pi_{NOMENS}(Enseignants \bowtie \sigma_{NUMETUD=107}(Encadre))$$

- Quels est le num des étudiants qui n'ont pas d'encadrant.

$$\bullet \pi_{NUMETUD}(Etudiants) \setminus \pi_{NUMETUD}(Encadre)$$

- Lister le nom de tous les étudiants et enseignants

$$\bullet \rho_{NOMETUD/NOM}(\pi_{NOMETUD}(Etudiants)) \cup$$

$$\rho_{NOMENS/NOM}(\pi_{NOMENS}(Enseignants))$$

Chaque requête est un arbre parenthésé, et plusieurs requêtes peuvent répondre à une même question.

# Calcul Relationnel à Variable Tuples

- Syntaxe :

$$\{x^{(n)} \mid F(x)\}$$

où  $x^{(n)}$  est un  $n$ -uplet (c'est à dire un tuple à  $n$  champs) et  $F$  est une formule logique du premier ordre ;  $F(x)$  exprime donc de façon déclarative les conditions que chaque tuple  $x$  doit vérifier pour appartenir au résultat.

- $x$  est une variable libre de  $F(x)$ .
- On introduit si besoin des variables liées par des quantificateurs  $\exists$  ou  $\forall$ . Ces variables permettent par exemple de parcourir les relations, pour être comparées à  $x$ .

# Exemples Calcul Relationnel

- Quel est le prenom et le nom de tous les étudiants

- $\{x = (x_1, x_2) \mid \exists x' \in Etudiants((x_1, x_2) = x' [PRENOMETUD, NOMETUD])\}$

- Quel est le prenom et le nom des enseignants qui sont PR

- $\{x = (x_1, x_2) \mid \exists x' \in Enseignants(x' [GRADE] = 'PR' \wedge (x_1, x_2) = x' [PRENOMENS, NOMENS])\}$

- Quel est le nom des enseignants qui encadrent l'étudiant 53 ?

- $\{x = (x_1) \mid \exists x' \in Encadre(x' [NUMETUD] = 107 \wedge \exists y' \in Enseignants(x' [NUMENS] = y' [NUMENS] \wedge y' [NOMENS] = x_1))\}$

- Quels est le num des étudiants qui n'ont pas d'encadrant.

- $\{x = (x_1) \mid \exists x' \in Etudiants(x' [NUMETUD] = x_1 \wedge \forall y' \in Encadre(x' [NUMETUD] \neq y' [NUMETUD]))\}$

- Lister le nom de tous les étudiants et enseignants.

- $\{x = (x_1) \mid \exists x' \in Etudiants(x' [NOMETUD] = x_1)\} \cup \{x = (x_1) \mid \exists x' \in Enseignants(x' [NOMENS] = x_1)\}$

# Structured Query Language - SQL

- Langage implémenté et universel d'interrogation d'une BD relationnel
- Traduit du calcul relationnel, donc **déclaratif**.
- Définit dans les années 80, dernière norme en 92
  - Beaucoup d'évolutions : UPSERT, Windows functions, Grouping Set, récursivité
- Se décompose en sous-ensembles :
  - DML : Manipulation (màj) et interrogation des données
  - DDL : Définition des données (au niveau du schéma)
  - DCL : Control des droits des utilisateurs
  - TCL : Control des transactions

## Langage très puissant souvent sous-exploité

- Apprenez en pratiquant sur l'excellent site <https://www.pgexercises.com/>
- Si pouvez le faire en SQL, alors ne le faites pas autrement.

# Exemples d'interrogations

- Quel est le prenom et le nom de tous les étudiants
  - `SELECT PRENOMETUD, NOMETUD FROM Etudiants`
- Quel est le prenom et le nom des enseignants qui sont PR
  - `SELECT PRENOMENS, NOMENS FROM Enseignants WHERE GRADE='PR'`
- Quel est le nom des enseignants qui encadrent l'étudiant 53 ?
  - `SELECT NOMENS FROM Enseignants JOIN Encadre ON Enseignants.NUMENS=Encadre.NUMENS WHERE Encadre.NUMETUD=107`
- Quels est le num des étudiants qui n'ont pas d'encadrant.
  - `SELECT NUMETUD FROM Etudiants EXCEPT SELECT NUMETUD FROM Encadre`
- Lister le nom de tous les étudiants et enseignants.
  - `SELECT NUMETUD AS 'NOM' FROM Etudiants UNION SELECT NUMENS AS 'NOM' FROM Enseignants`

- 1 Le modèle relationnel
  - Intuition
  - Structure et contraintes de clés
  - Traduction E/A vers Relationnel
  - Langages
  - Les transactions

# Définition des transactions

Transaction = ensemble de mises à jour

- Perçue comme atomique par l'utilisateur
- Il n'y a que deux états de fin possible
  - annulation : aucune trace, rien n'a été fait.
  - validation : modifications présentes et pérennes même en cas de panne
- modifications invisibles PENDANT la transaction.
  - Les requêtes concurrentes ne voient que l'état AVANT
- La base est cohérente AVANT et APRES la transaction
  - Contraintes pas forcément vérifiées PENDANT la transaction.

On résume par l'acronyme ACID : Atomicité, Cohérence, Isolation, Durabilité