

# Bases de données

DIU

Sofian MAABOUT

maabout@labri.fr

# Au menu

- C'est quoi une Base de Données (BD) ?
- C'est quoi un Système de Gestion de Bases de Données (SGBD) ?
- Le cas particulier des BD's *relationnelles*
  - Structure
  - Interrogation
    - Algèbre relationnelle (langage théorique)
    - SQL (langage utilisé en pratique)
    - Aspects "avancés"
      - Mise à jour des données
      - Définition/création d'une BD
      - Contraintes sur les données
      - Accès concurrents à une BD

# Au menu

- C'est quoi une Base de Données (BD) ?
- C'est quoi un Système de Gestion de Bases de Données (SGBD) ?
- Le cas particulier des BD's *relationnelles*
  - Structure
  - Interrogation
    - Algèbre relationnelle (langage théorique)
    - SQL (langage utilisé en pratique)
    - Aspects "avancés"
      - Mise à jour des données
      - Définition/création d'une BD
      - Contraintes sur les données
      - Accès concurrents à une BD

# Base de données: que sa quo

- Pas de définition précise
- Tentative: une collection de données ayant une structure particulière
  - A la limite
    - Le système de fichiers d'un système d'exploitation est une BD: *Collection* de fichiers dans une *structure* arborescente
    - Le WEB: *collection* de sites avec une *structure* en réseau (graphe orienté)
    - Classeur Excel: *collection* de cellules *structurées* en Feuilles de calcul et tableaux
    - ...

# Bases de données: que sa quo

- Communément et historiquement, on parle de bases de données dès lors que l'on définit:
  - Un modèle ou une structure de données
  - Un langage de manipulation de ces données
    - Interrogation et
    - mise à jour
- Les modèles les plus connus (ordre chronologique):
  - Modèles hiérarchique et réseaux (60's): Codasyl, Socrates, Cobol
  - *Modèle relationnel: 70's*
  - Modèle orienté objet : Fin 80's, début 90's
  - Modèle déductif : Fin 80's, début 90's
  - Modèle semi-structuré: Mi-90's (XML, JSON, ...)
  - Modèles No-SQL: ~-05's.: objet-document, graphe, ...

# Modèle relationnel

- Modèle théorique proposé par Ted Codd (IBM) en 1970.
- Premier système de gestion de bases de données relationnel (SGBD) commercial en 1979 par Relational Software (Oracle)
- Dernière version d'Oracle: 19.c sortie en 2019
- Dans le Top 10 mondial des sociétés éditrices de logiciel:
  - IBM (DB2, ...), Oracle (Oracle), SAP (SAP)

# Modèle relationnel: structure

- Une BD relationnelle est constituée d'un ensemble de relations (ou tables)
- Une relation est un tableau
  - qui a un nom
  - Où chaque colonne (appelé aussi *attribut* ou *champ*) et a un nom
  - Où chaque ligne est *un enregistrement* (ou *tuple*).
  - Deux relations dans la même BD ne peuvent pas avoir le même nom
  - Deux attributs dans la même relation ne peuvent pas avoir le même nom

## Exemple: Relation « Livre »

Livre	N°Livre	Titre	NomAuteur	Editeur
	137	La peste	Camus	Gallimard
	11	Le monde selon Garp	Irving	Folio
	46	L'assomoire	Zola	Livre de Poche
	12	L'étranger	Camus	Gallimard



## Exemple: Base de données « Bibliothèque »

- Livre(N°Livre, Titre, Auteur, Editeur)
- Exemple(N°Livre, N°Exemple, Disponible)
- Adhérent(N°Adhérent, Nom, Prénom, Adresse, Mail, Phone)
- Prêt (N°Livre, N°Exemple, N°Adhérent, DateEmprunt, DateRetour)

# En fait, pourquoi l'appeler modèle « relationnel » ?

- Basé sur la notion mathématique de relation.
- Soient  $A_1, \dots, A_m$  des ensembles de valeurs
- $A_1 \times A_2 \times \dots \times A_m$  désigne le produit cartésien de ces ensembles
- Une relation sur ces  $m$  ensembles est un sous-ensemble de leur produit cartésien
- Exemple:
  - $N^{\circ}\text{Livre} = \{ 11, 46, 137\}$ ,  $\text{NomAuteur} = \{\text{Camus, Zola, Irving}\}$ ,  $\text{Titre} = \{\text{La Peste, Le monde selon Garp, L'assommoire, L'étranger}\}$ ,  $\text{Editeur} = \{\text{Gallimard, Poche, Folio}\}$
  - La relation Livre est un sous-ensemble du produit de ces 4 ensembles
    - Produit : toutes les associations possibles
    - Relation: seulement les associations correctes
    - Une association est un quadruplet ou simplement un tuple ou bien encore enregistrement

# Interrogation d'un BD relationnelle (BDR)

- A l'origine, 2 langages « théoriques » ont été proposés pour les BDR:
  - L'algèbre relationnelle
  - Le calcul relationnel
- Algèbre: basée sur un ensemble d'opérateurs
- Calcul: basé sur la logique de premier ordre (calcul de prédicats)
  
- Langage pratique: SQL
  - C'est le langage standard utilisé par tout SGBD relationnel

# Algèbre relationnelle

- Opérateurs de base:
  - Projection
  - Sélection
  - Renommage
  - Union, Intersection, Différence
  - Produit cartésien
- Opérateurs supplémentaires:
  - Jointure (naturelle)
  - Jointure externe
  - Division
  - ...

# Algèbre relationnelle: Projection

- Notation :  $\pi_{Attributs}(relation)$
- Résultat: Seules les attributs de la relation qui sont mentionnés, sont retournés dans le résultat

R	A	B	C
	a1	b1	c1
	a2	b2	c2
	a1	b3	c1
	a3	b2	c2

$$\pi_{A,C}(R) =$$

A	C
a1	c1
a2	c2
a3	c2

- Le résultat est une relation « sans nom »
- Relation = ensemble donc pas de doublons:  $\langle a1, c1 \rangle$  n'apparaît qu'une seule fois
- $\pi_{NomAuteur}(Livres) = \{Zola, Irving, Camus\}$  Pas besoin d'afficher 2 fois Camus

# Algèbre relationnelle: sélection

- Notation :  $\sigma_{condition}(relation)$
- Résultat: Seuls les enregistrements de la relation qui satisfont la condition sont retournés dans le résultat

R	A	B	C
	a1	b1	c1
	a2	b2	c2
	a1	b3	c1
	a3	b2	c2

$$\sigma_{C=c2}(R) =$$

A	B	C
a2	b2	c2
a3	b2	c2

Les livres écrits par Camus :  $\sigma_{NomAuteur=Camus}(Livres)$

# Algèbre relationnelle: composition d'opérations

- Résultat d'une opération est une relation  $\Rightarrow$  On peut appliquer un opérateur à ce résultat
- Le titre des livres de Camus  $\pi_{Titre}(\sigma_{NomAuteur=Camus}(Livres))$
- L'algèbre est un langage fermé
- Grâce à la composition, on peut exprimer des requêtes plus ou moins sophistiquées

# Algèbre relationnelle: Renommage

- Notation :  $\rho_{Attribut \rightarrow Attribut'}(relation)$
- Résultat: Le contenu de la relation est retourné. Seul le nom d'un attribut est modifié  $\rho_{Attribut \rightarrow Attribut'}(relation)$

R	A	B	C
	a1	b1	c1
	a2	b2	c2
	a1	b3	c1
	a3	b2	c2

$$\rho_{A \rightarrow D}(R) =$$

R	D	B	C
	a1	b1	c1
	a2	b2	c2
	a1	b3	c1
	a3	b2	c2

Les titres de Camus:  $\rho_{Titre \rightarrow TitreCamus}(\sigma_{NomAuteur=Camus}(Livre))$

Dans le résultat, on aura une colonne qui s'appellera « TitreCamus » et non pas « Titre »



# Algèbre relationnelle: Union, Intersection, Différence

- Opérations ensemblistes classiques
- Notations:  $R1 \cup R2$ ,  $R1 \cap R2$ ,  $R1 - R2$
- Résultat: Les enregistrements qui sont dans R1 OU dans R2, ceux qui sont dans R1 ET R2, ceux qui sont dans R1 mais PAS dans R2
- Il faut que R1 et R2 aient le même schéma (mêmes attributs nommés de la même manière).

# Algèbre relationnelle: Union, Intersection, Différence

R1	A	B	C
	a1	b1	c1
	a2	b2	c2
	a1	b3	c1
	a3	b2	c2

R2	A	B	C
	a1	b1	c1
	a2	b3	c2
	a1	b3	c1

$R1 \cup R2 =$

A	B	C
a1	b1	c1
a2	b2	c2
a1	b3	c1
a3	b2	c2
a2	b3	c2

$R1 \cap R2 =$

A	B	C
a1	b1	c1
a1	b3	c1

$R1 - R2 =$

A	B	C
a2	b2	c2
a3	b2	c2

# Algèbre relationnelle: Produit Cartésien

- Notation :  $R1 \times R2$
- Résultat : chaque ligne de R1 est « composée » avec TOUTES les lignes de R2
- Condition: Il faut que les attributs de R1 soient distincts de ceux de R2
- Les attributs dans le résultat sont ceux de R1 et ceux de R2

R1	A	B	C
	a1	b1	c1
	a2	b2	c2

R2	D	E
	d1	e1
	d2	e2
	d3	e3

$R1 \times R2 =$

A	B	C	D	E
a1	b1	c1	d1	e1
a2	b2	c2	d1	e1
a1	b1	c1	d2	e2
a2	b2	c2	d2	e2
a1	b1	c1	d3	e3
a2	b2	c2	d3	e3

# Algèbre relationnelle: exemple

- Prêt(N°Adhérent, N°Livre, ...)                      Livre(N°Livre, Titre, ...)
- Les titres des livres empruntés par l'adhérent 154.
  - On compose chaque Prêt à chaque livre: Prêt x Livre
    - Mais pas possible car on a N°Livre qui est attribut commun
  - On renomme d'abord le N°Livre dans une des deux relations
    - $\rho_{N^{\circ}\text{Livre} \rightarrow \text{NL}}(\text{Livre})$
  - On compose le résultat avec Prêt : Prêt x  $\rho_{N^{\circ}\text{Livre} \rightarrow \text{NL}}(\text{Livre})$
  - On ne garde que les prêts associés au livre qui les concerne
    - $\sigma_{N^{\circ}\text{Livre}=\text{NL}}(\text{Prêt} \times \rho_{N^{\circ}\text{Livre} \rightarrow \text{NL}}(\text{Livre}))$
  - On ne garde que les prêts qui concernent l'adhérent 154
    - $\sigma_{N^{\circ}\text{Adhérent}=154}(\sigma_{N^{\circ}\text{Livre}=\text{NL}}(\text{Prêt} \times \rho_{N^{\circ}\text{Livre} \rightarrow \text{NL}}(\text{Livre})))$
  - On ne retourne que l'attribut Titre dans le résultat
    - $\pi_{\text{Titre}}(\sigma_{N^{\circ}\text{Adhérent}=154}(\sigma_{N^{\circ}\text{Livre}=\text{NL}}(\text{Prêt} \times \rho_{N^{\circ}\text{Livre} \rightarrow \text{NL}}(\text{Livre}))))$

# Algèbre relationnelle: quelques remarques

- Les opérations ne sont généralement pas commutatives
  - $\pi_A (\sigma_{B=b_1} (R))$  n'est pas équivalente à :  $\sigma_{B=b_1} (\pi_A (R))$
  - Observer qu'une fois qu'on a projeté sur A, il devient impossible de vérifier B=b1
- Pour une même requête, il peut exister plusieurs expressions équivalentes:
  - $\pi_{\text{Titre}} (\sigma_{N^{\circ}\text{Adhérent}=154} (\sigma_{N^{\circ}\text{Livre}=NL} (\text{Prêt} \times \rho_{N^{\circ}\text{Livre} \rightarrow NL} (\text{Livre}))))$
  - $\pi_{\text{Titre}} (\sigma_{N^{\circ}\text{Livre}=NL} (\sigma_{N^{\circ}\text{Adhérent}=154} (\text{Prêt}) \times \rho_{N^{\circ}\text{Livre} \rightarrow NL} (\text{Livre})))$
- Certains opérateurs sont unaires: projection, sélection et renommage, d'autres sont binaires: ensemblistes et produit cartésien.

# Algèbre relationnelle: Jointure

- Opération qui simplifie l'écriture de requêtes nécessitant un renommage, un produit et une sélection:
- Notation :  $R1 \bowtie R2$
- Résultat : chaque tuple de R1 est composé à tout tuple de R2 *avec qui il partage les mêmes valeurs sur les attributs communs aux deux tables*

# Algèbre relationnelle: jointure

R1	A	B	C
	a1	b1	c1
	a2	b2	c2

R2	C	D
	c1	d1
	c1	d2
	c3	d2

$R1 \bowtie R2 =$

A	B	C	D
a1	b1	c1	d1
a1	b1	c1	d2

Prêt(N°Adhérent, N°Livre, ...)

Livre(N°Livre, Titre, ...)

Les titres des livres empruntés par l'adhérent 154.

$\pi_{\text{Titre}} (\sigma_{\text{N}^\circ\text{Adhérent}=154}(\text{Livre} \bowtie \text{Prêt}))$

# Algèbre relationnelle: jointure

- En pratique, la jointure est une opération très fréquemment utilisée dès lors que l'on a besoin de combiner 2 ou plusieurs tables
- Le produit cartésien est quant à lui rarement utilisé
- Si les deux tables qu'on veut joindre n'ont aucun attribut commun, alors la jointure revient à réaliser leur produit



# Algèbre relationnelle: remarques générales

- Trouver l'expression d'une requête en algèbre revient à
- Identifier les tables qu'on doit mettre en jeu
  - Soit parce qu'elles contiennent des attributs qu'on veut afficher
  - Soit parce qu'elles contiennent des attributs sur lesquels on veut exprimer des conditions
  - Soit parce qu'elles permettent d'établir des liens entre tables
- Faire attention à l'enchaînement des opérations

# Algèbre relationnelle: table de liaison

- Adhérent(N°Adhérent, Nom, ...),
- Afficher le titre des livres empruntés par un adhérent dont le nom est « Dupont »
  - On utilisera Livre car c'est elle qui contient l'attribut Titre qu'on doit afficher (retourner)
  - On utilisera Adhérent car on doit poser une condition sur le nom.
  - Il n'y a pas de lien direct entre ces deux tables permettant de dire que tel adhérent a emprunté tel livre.
  - Pour établir le lien entre Livre et Adhérent, on utilise la table Prêt.
- $\pi_{\text{Titre}} (\sigma_{\text{NomAdhérent=Dupont}} (\text{Livre} \bowtie \text{Prêt}) \bowtie \text{Adhérent})$

# Algèbre relationnelle

- Malgré la simplicité de ses opérations, l'algèbre permet d'exprimer des requêtes plus ou moins sophistiquées
- Même si on n'utilise jamais l'algèbre en pratique, il est important de la comprendre car cela est très utile pour exprimer les requêtes en SQL
- Certaines requêtes simples inexprimables en algèbre: afficher le nombre total de livres
- La puissance d'expression de l'algèbre est équivalente à celle du Calcul relationnel (pas vu). Ex: Nom et prénom des adhérents ayant emprunté un livre
  - $\{(N, P) \mid \exists x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8 \text{ Prêt}(x_1, x_2, x_3, x_4, x_5) \text{ AND Adhérent}(x_1, N, P, x_6, x_7, x_8)\}$

# Conclusion

- Quoi retenir ?
  - Structure d'une BD relationnelle
  - Expression de requêtes en algèbre
- Généralement, 2 types d'exos:
  - Requête en français à exprimer en algèbre
  - Requête en algèbre et
    - Donner son résultat étant donnée une instance de BD (petit jeu de données)
    - Donner ce qu'elle exprime en français.