

Base de Données Structurées et Non Structurées

Partie 11: Algèbre relationnel



Prof. Badr-Eddine BOUDRIKI SEMLALI
Email: badreddine.boudrikisemlali@uae.ac.ma

2024/2025

Plan de cours

1. Introduction aux bases de données et méthode de conception MERISE
2. Modélisation des traitements : MCC, MCT, MOT
3. Modélisation des données: (MCD) → modèle « E/A »
4. Dépendance fonctionnelles (Construction du MCD)
5. Normalisation (5 formes de normalisation)
6. Le Modèle Logique de Données (relationnel) : MLD
7. Le Modèle Physique de Données : MPD
8. Le Modèle relationnel: Concepts de bases
9. Structure de base de données
10. Règles d'intégrité structurelle
11. Algèbre relationnel

Algèbre relationnelle

- ✍️ Puisque le modèle relationnel est basé sur une organisation des données sous forme de tables (relations), **alors** la manipulation de ces données (tables) se fait selon le concept mathématique de relation de la théorie des ensembles, **l'Algèbre relationnelle**.
- ✍️ **L'Algèbre Relationnelle** à été introduit pour formaliser des opérations formelles sur les relations.
- ✍️ Une **opération relationnelle** conduit à construire une nouvelle relation (Table).
- ✍️ **L'Algèbre Relationnelle** est la base théorique sur laquelle la construction des **langages d'interrogation de base de données relationnelles** s'est construite. Il existe plusieurs langages mais aujourd'hui le plus utilisé est **SQL**. Sous certains SGBDR grand public comme Access, les utilisateurs débutants utilisent aussi **QBE** (query by exemple), qui est un langage de requête visuel, basé sur l'algèbre relationnelle.

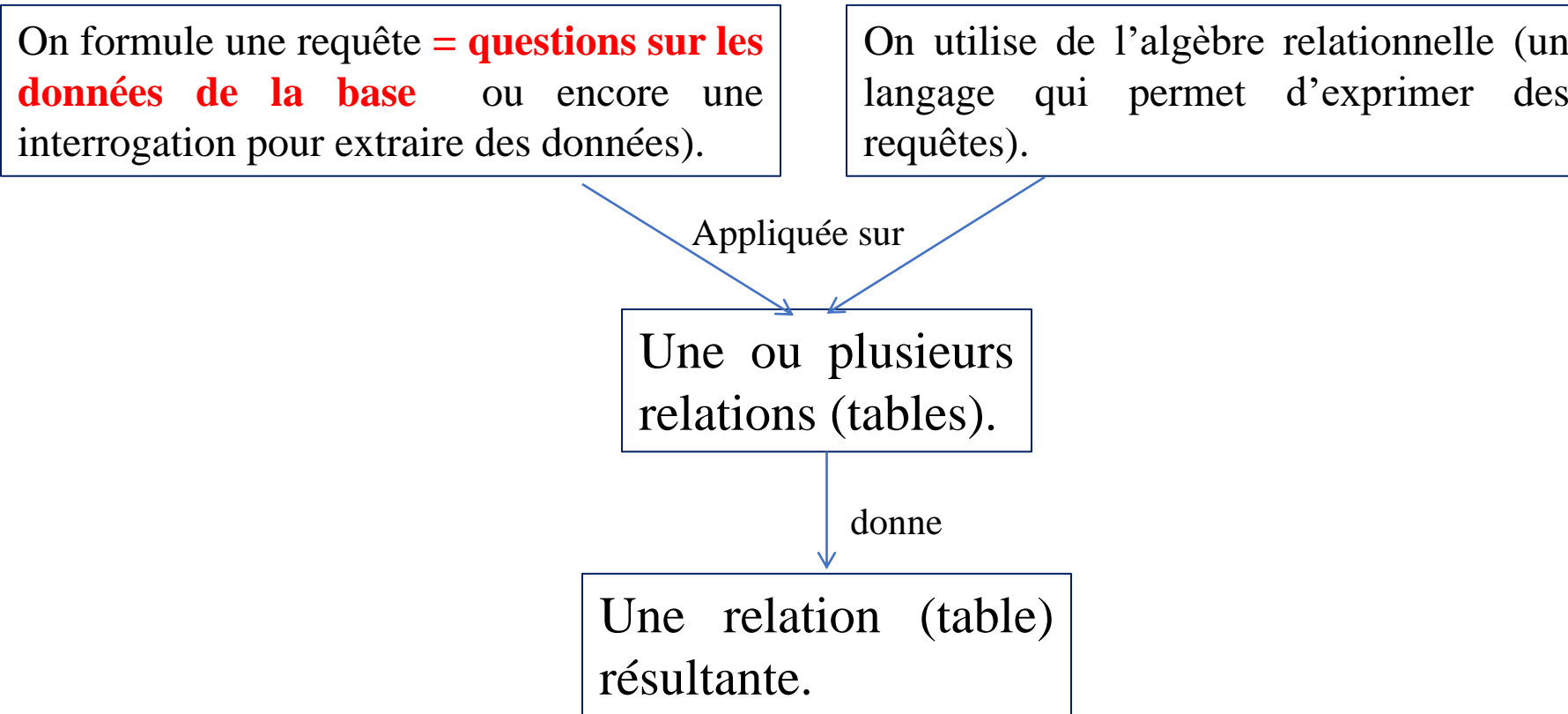
Algèbre relationnelle

Langage algébrique :

- ✍ Collection d'opérations formelles qui agissent sur des relations pour produire d'autres relations.
- ✍ Modèle relationnel basé sur la théorie des ensembles :
 - utilisation d'opérateurs classiques de manipulation des ensembles (union, intersection, différence, produit cartésien).
- ✍ Introduction d'opérateurs propres aux bases de données.

Algèbre relationnelle

✍ L'algèbre relationnelle, c'est un langage d'interrogation des bases de données relationnelles.



L'algèbre relationnelle permet de manipuler les données des tables d'une base de données à l'aide de **requêtes** (query en anglais). Elle prépare la conception de requêtes qui seront traduites en SQL.

Algèbre relationnelle

Présentation: Types d'opérateurs

On peut distinguer **huit opérateurs** initiaux en trois familles d'opérateurs relationnels :

- ✍ **Les opérateurs unaires** : Sélection, Projection.
- ✍ **Les opérateurs binaires ensemblistes** : Union ($R1 \cup R2$), Intersection ($R1 \cap R2$), Différence ($R1 - R2$).
- ✍ **Les opérateurs binaires ou n-aires** : Produit Cartésien ($R1 \times R2$), Jointure, Division.
- ✍ Algèbre relationnelle est un ensemble d'opérateurs qui s'appliquent aux relations.
- ✍ **Résultat** : nouvelle relation (table résultante).

Les opérateurs unaires: Sélection

Soit l'Exemple de Sélection suivant :

Soit la table Livre (N°Livre, Titre, Année)

- On veut savoir quels sont les livres qui sont sortis en 2000.
- $R_2 = \text{Restriction}(\text{Livre}; \text{Année} = 2000)$

Les opérateurs unaires: Sélection

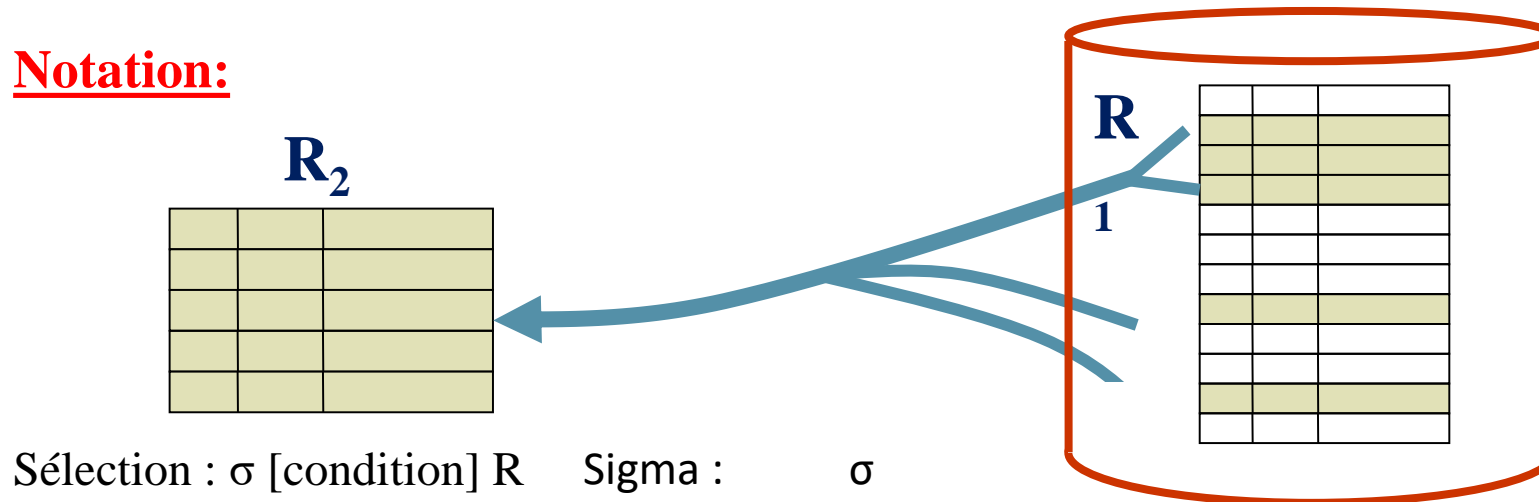
Opération unaire portant sur R_1 qui produit une relation R_2 de même schéma ayant pour t-uples ceux vérifiant la condition donnée en opérande.

Notation: $R_2 = \sigma_{\text{condition}}(R_1)$.

$R_2 = \text{Sélection}(R_1; \langle \text{condition_de_restriction} \rangle)$

1. **Condition_simple** : opérande1 α opérande2. α est un opérateur à prendre parmi $\{=, \sigma, >, <, >=, <= \}$
2. **Condition_composée** : conditions simples liées par les opérateurs logiques **et**, **ou**, **non**
 R_2 contient les lignes de R_1 qui vérifient la condition de restriction

Notation:



Les opérateurs unaires: Sélection

Exemple 2:

Soit la table Patient(nom, prénom, dateNaissance)

- On veut savoir quels sont les patient qui portent le nom Jamal
- $R_2 = \sigma_{\text{nom}=\text{Jamal}}(\text{Patient})$

| Patient | nom | prénom | dateNaissance |
|---------|-------|----------|---------------|
| | Jamal | Alami | 21/08/1999 |
| | Kamal | Benani | 28/08/2000 |
| | Jamal | Hassani | 23/06/2001 |
| | Taha | Bensouda | 07/09/2003 |

Résultat :

| nom | prénom | dateNaissance |
|-------|---------|---------------|
| Jamal | Alami | 21/08/1999 |
| Jamal | Hassani | 23/06/2001 |

Les opérateurs: Union ($R_1 \cup R_2$)

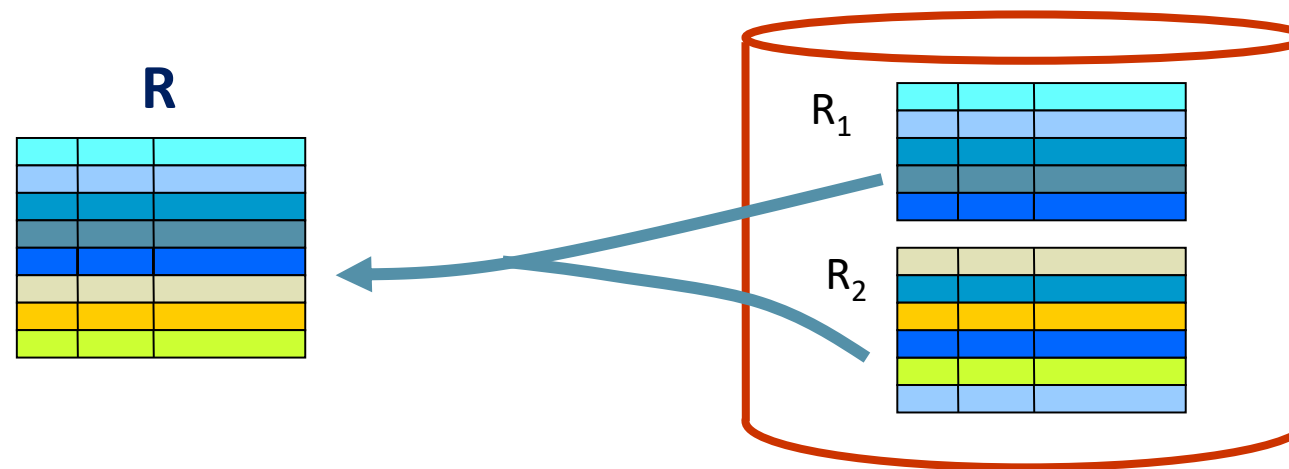
L'union de deux tables est l'ensemble des **occurrences qui appartiennent soit à la première table, soit à la deuxième, soit aux deux tables**. C'est la traduction du **OU** logique.

- Les deux tables doivent avoir le même schéma
- Le résultat est une nouvelle table contenant l'ensemble des lignes contenues dans les deux tables, après élimination des doublons.

$$R = \text{UNION} (R_1 ; R_2) = R_1 \cup R_2$$

R contient les lignes de R1 et celles de R2, Les doublons sont éliminés.

R1 et R2 doivent avoir les mêmes champs.



Les opérateurs: Union ($R1 \cup R2$)

Exemple d'union :

Soit les tables:

- Etudiant(N_Etd, Nom, Adresse)
- Enseignant(N_Ens, Nom, Adresse)

On veut avoir la liste des personnes qui accèdent à l'université (étudiants ou Enseignant)

Union(Etudiant ; Enseignant)

Les opérateurs: Intersection ($R1 \cap R2$)

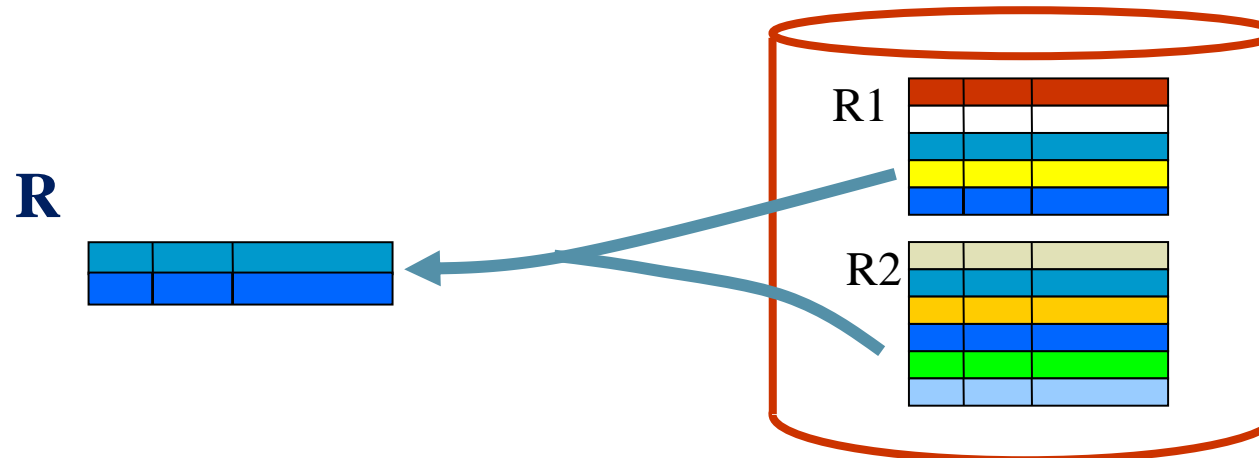
L'intersection de 2 relations est l'ensemble des occurrences qui sont présentes dans les deux relations . C'est la traduction du ET logique.

- Les deux tables doivent avoir le même schéma
- Le résultat est une nouvelle table contenant uniquement les communs aux deux tables, après élimination des doublons.

$$R = \text{Intersection}(R1; R2) = R1 \cap R2$$

R contient les lignes de R1 qui sont dans R2

R1 et R2 doivent avoir les mêmes champs.



Les opérateurs: Intersection ($R1 \cap R2$)

Exemple d'Intersection :

Soit les tables:

Etudiant(NSS, Nom, Adresse)

Employé(NSS, Nom, Adresse)

On veut avoir la liste des personnes qui sont en même temps étudiants et "salariés".

Intersection(Etudiant ; Employé)

Les opérateurs: Différence (R1-R2)

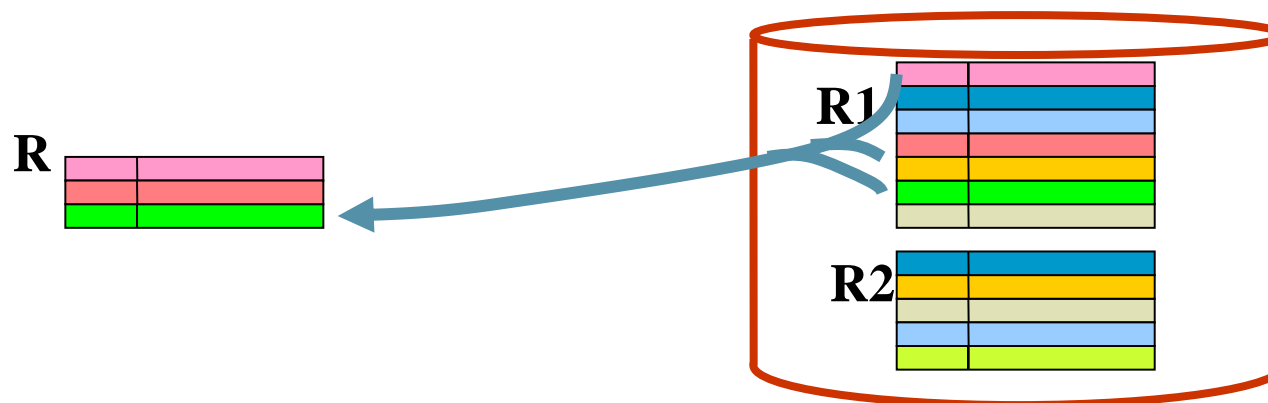
La différence entre deux table est l'ensemble des **occurrences qui appartiennent à une table sans appartenir à la seconde** .

- Les deux tables doivent avoir le même schéma
- Le résultat est une nouvelle **table contenant les lignes de la 1^{ère} table sauf celles contenus dans la 2^{ème}**

$R = \text{DIFFERENCE}(R1; R2) = R1 - R2$

R contient les lignes de R1 qui ne sont pas dans R2.

R1 et R2 doivent avoir le même schéma.



Les opérateurs: Différence (R1-R2)

Exemple de Différence :

Soit les tables:

Etudiant(NSS, Nom, Adresse)

Employé(NSS, Nom, Adresse)

On veut avoir la liste des personnes qui sont salariés et qui ne sont pas en même temps des étudiants.

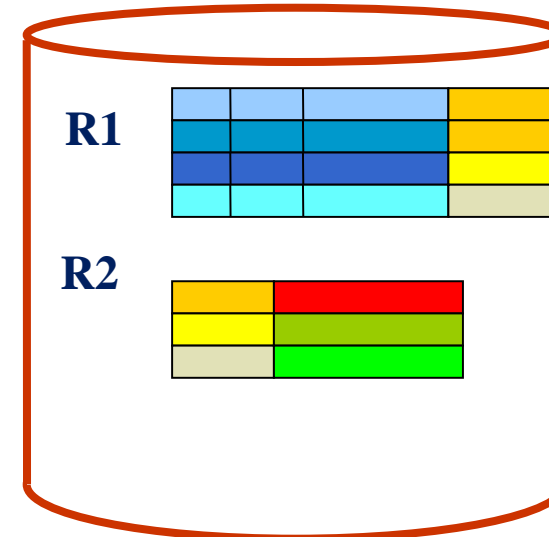
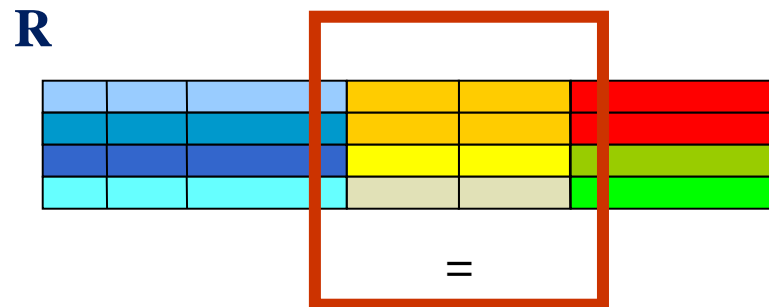
Différence(Employé; Etudiant)

Les opérateurs: Jointure

Operation binaire portant sur R_1 et R_2 consistant a rapprocher les t-uples des 2 relations selon un critere C pour former une relation R_3 qui contient l'ensemble des t-uples du produit cartesien $R_1 \times R_2$ selon le critere C .

$$R = \text{JOINTURE}(R_1; R_2) = R_1 \bowtie R_2$$

R contient les combinaisons des lignes de R_1 avec les lignes de R_2 qui ont la même valeur sur les champs communs.



Les opérateurs: Jointure

Exemple 1:

Soit les tables

Livre(N°Livre, Titre, Année, N°Auteur)

Auteur(N°Auteur, Nom, AnnéeNaissance)

On veut savoir pour chaque livre, l'auteur qui l'a écrit (on suppose qu'il n'y a qu'un auteur par livre)

R1= Jointure (Livre ; Auteur)

Exemple 2:

Soit les tables

Livre(N°Livre, Titre, Année, N°Auteur)

Auteur(N°Auteur, Nom, AnnéeNaissance)

On veut avoir le titre des livres écrits par un auteur qui s'appelle Najib

R1= Jointure (Livre ; Auteur)

R2= Restriction(R1; Nom='Najib')

R3= Projection(R2 ; Titre)

Algèbre relationnelle: Conclusion

L'algèbre relationnelle est un langage de requêtes d'interrogation des données. C'est un **langage théorique comme l'algorithmique** : il ne peut pas être compris directement par les SGBDR. Il faut le **traduire** dans un **langage supporté par le SGBD** tel que **SQL** (le langage standard pour tous les SGBDR) ou QBE sur Access.

Merci pour votre motivation.
Fin des Cours du module de SIBD