



Chapitre III: Base de données relationnelle

Introduction:

Une base de données relationnelle est une base de données structurée suivant les principes de l'algèbre relationnelle (Modèle relationnel).

Le père des bases de données relationnelles est Edgar Frank Codd. Chercheur chez IBM à la fin des années 1960, il étudiait alors de nouvelles méthodes pour gérer de grandes quantités de données car les modèles et les logiciels de l'époque ne le satisfaisait pas.

Après avoir validé le MCD, le niveau logique d'abstraction s'impose, pour traduire le MCD en MLD suivant des règles bien définies. Le Modèle Logique de Données est un passage du Modèle Conceptuel de Données validé vers l'implantation physique des données. Il se situe alors entre le M.C.D. et le M.P.D. (Modèle Physique des Données). Le MLD ajoute au MCD la notion d'organisation et indique comment les données seront organisées.

A ce stade, le modèle est encore indépendant des choix matériels et logiciels. Cette étape consiste à transformer le modèle conceptuel de données en modèle logique, selon un formalisme adapté à un type général de système de gestion de données.

Pour ce faire, on dispose de quatre types de modèles : le modèle hiérarchique, le modèle réseau, le modèle relationnel enfin le modèle objet. Plus récemment, sont apparus les modèles orientés objets et même des SGBD orientés objets. Pourtant les SGBD relationnels restent extrêmement majoritaire.

1. Modèle relationnel

1.1 Présentation

Le modèle relationnel est aux fondements des SGBDR. Il a été, et continue d'être, le modèle théorique dominant pour la représentation logique des bases de données, même si le mouvement NoSQL propose des alternatives.

Le modèle relationnel permet de reformuler le modèle conceptuel dans un formalisme – le tableau – beaucoup plus proche de l'implémentation informatique.

Le modèle relationnel inclut des concepts pour la description de données, ainsi que des concepts pour la manipulation de données.

Représenter le monde en tables :

Le modèle relationnel permet de représenter les données que l'on va gérer à l'aide d'un nombre de concepts:

- Les relations ou tables : des lignes et des colonnes
- Les domaines de valeurs : chaque case d'une table prend une unique valeur dans un domaine prédéfini.
- Les clés : il existe des cases dont les valeurs doivent être uniques et non nulles.
- Les clés étrangères : il existe des cases qui doivent prendre une valeur existante dans les cases d'une autre table.

1. Modèle relationnel

1.2 Table, lignes et colonnes.

Lorsque les données ont la même structure (comme par exemple les renseignements relatifs aux clients), on peut les organiser en **Table** dans laquelle:

- Les **colonnes** décrivent les champs en commun.
- Les **lignes** contiennent les valeurs de ces champs pour chaque enregistrement.

CinClient	Nom	Prénom	Adresse
J14785	Alaoui	Mustapha	13, Rue 515 ...
JB14796	Filali	Farid	12, Bloc 306 ...
C236902	Alaoui	Laila	13, Cité nahda...
...	

Contenu de la Table (Relation) Client, avec en première ligne les intitulés des colonnes

L'élément de base du modèle relationnel est la **relation** ou **table**. La relation est désignée par son nom.

1. Modèle relationnel

1.3 Concepts de base

a. Domaine

Définition : Ensemble, caractérisé par un nom, dans lequel des données peuvent prendre leurs valeurs.

Remarque : Un domaine peut-être défini en intension (c'est à dire en définissant les propriétés caractéristiques des valeurs du domaine, on parle aussi de compréhension) ou en extension (c'est à dire en énumérant toutes les valeurs du domaine)

- Les entiers
- Les réels inférieur à 5
- Les booléen (vrai ou faux)
- Toutes les chaînes de 1 à 255 caractères
- Les valeurs monétaires, définie comme des décimaux avec deux chiffres après la virgule
- Les dates, définies comme des chaînes de 10 caractères comprenant des chiffres et des tirets selon le patron "00-00-0000"
- Les salaires, définis comme des valeurs monétaires compris entre 15.000 et 100.000

**Domaines
définis en
intension :**

**Domaines
définis en
extension :**

- Couleur : {Bleu, Vert, Rouge, Jaune, Blanc, Noir}
- SGBD : {Hiérarchique, Réseau, Relationnel, Objet, Relationnel-Objet}

1. Modèle relationnel

1.3 Concepts de base

b. Attribut

Définition : Un attribut est un identificateur décrivant une information stockée dans une base.

Exemples : l'âge d'une personne, le nom d'une personne, le numéro de sécurité sociale.

Un attribut est caractérisé par un nom et un domaine dans lequel il prend ses valeurs.

Synonymes : Champs, Propriété, Colonne.

c. Enregistrement

Définition : On appelle enregistrement d'une relation, une ligne de cette relation. Un enregistrement prend une valeur pour chaque attribut de la relation.

Synonymes : Tuple, N-uplet, Vecteur, Ligne

1. Modèle relationnel

1.3 Concepts de base

d. Relation

Définition : Une relation est un sous-ensemble du produit cartésien de n domaines d'attributs ($n > 0$). Une relation est représentée sous la forme d'un tableau à deux dimensions dans lequel les n attributs correspondent aux titres des n colonnes. On l'appelle aussi TABLE.

Relation : Client

CinClient	Nom	Prénom	Adresse
J14785	Alaoui	Mustapha	13, Rue 515 ...
JB14796	Filali	Farid	12, Bloc 306 ...
C236902	Alaoui	Laila	13, Cité nahda...
...	

Attributs

Enregistrements

1. Modèle relationnel

1.4 La gestion des clés

a. Clé primaire.

Les lignes d'une table doivent être uniques, cela signifie qu'une colonne (au moins) doit servir à les identifier ➔ Il s'agit de la clé primaire de la table.

Propriétés requises de la clé primaire :

- La valeur vide (NULL) est interdite.
- La valeur de la clé primaire d'une ligne ne devrait pas changer au cours du temps.

b. Clé candidate.

On appelle clés candidates l'ensemble des clés d'une relation qui n'ont pas été choisies comme clé primaire (elles étaient candidates à cette fonction).

1. Modèle relationnel

1.4 la gestion des clés

c. Clé artificielle.

S'il est impossible de trouver une clé primaire, ou que les clés candidates sont trop complexes, il est possible de faire appel à une clé artificielle. Une clé artificielle est un attribut supplémentaire ajouté au schéma de la relation, qui n'est lié à aucune signification, et qui sert uniquement à identifier de façon unique les enregistrements et/ou à simplifier les références de clés étrangères.

Au niveau de l'implémentation physique, il est courant que des clés artificielles soient utilisées de façon systématique.

- Du point de vue de **l'évolutivité** de la BD, il existe toujours un risque qu'une clé non-artificielle perde sa propriété d'unicité ou de non-nullité.
- Du point de vue de **la maintenance** de la BD, il existe toujours un risque qu'une clé non-artificielle voit sa valeur modifiée et dans ce cas, la répercussion de ce changement pour mettre à jour toutes les références peut poser problème.
- Du point de vue de **la performance** de la BD, les clés non-artificielles ne sont pas en général optimisées en terme de type et de taille, et donc peuvent limiter les performances dans le cadre des jointures.

1. Modèle relationnel

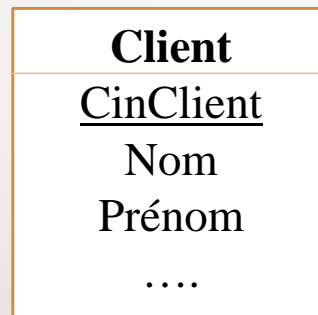
1.4 La gestion des clés

d. Clé étrangère.

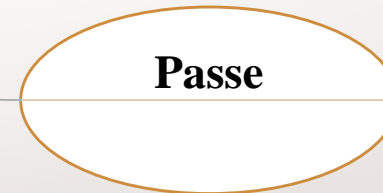
Une clé étrangère est un attribut ou un groupe d'attributs d'une relation R1 devant apparaître comme clé primaire dans une relation R2 afin de matérialiser une référence entre les tuples de R1 et les tuples de R2.

Une clé étrangère d'un tuple (enregistrement) référence une clé primaire d'un autre tuple.

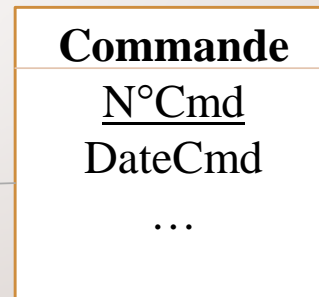
Considérant l'exemple suivant:



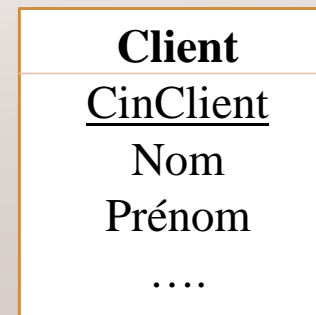
(0,n)



(1,1)



La transformation du MCD ci-dessus en MLD (Modèle Logique de données) permet d'obtenir le résultat suivant :



1. Modèle relationnel

1.4 La gestion des clés

d. Clé étrangère.

La table : Client

CinClient	Nom	Prénom	Adresse
J14785	Alaoui	Mustapha	13, Rue 515 ...
JB14796	Filali	Farid	12, Bloc 306 ...
C236902	Alaoui	Laila	13, Cité nahda...
...	

La table : Commande

N°Cmd	CinClient	DateCmd
1	J14785	01/01/2020
2	J14785	10/10/2020
3	JB14796	05/02/2020
...

Dans la table « Commande », Les valeurs de la colonne *CinClient* ne doivent contenir que des valeurs prises par la colonne *CinClient* (Identifiant) de la table « Client ».



On dit alors que la colonne *CinClient* de la table « Commande » est une **clé étrangère**.

1. Modèle relationnel

1.4 La gestion des clés

e. Contrainte d'intégrité référentielle.

Une clé étrangère respecte la contrainte d'intégrité référentielle si sa valeur est effectivement existante dans la clé primaire d'un tuple de la relation référencée, ou si sa valeur est null.

Une clé étrangère qui ne respecte pas la contrainte d'intégrité référentielle exprime un lien vers un tuple qui n'existe pas et donc n'est pas cohérente.

Exemple :

CinClient	Nom	Prénom	Adresse
J14785	Alaoui	Mustapha	13, Rue 515 ...
JB14796	Filali	Farid	12, Bloc 306 ...
C236902	Alaoui	Laila	13, Cité nahda...
...	

N°Cmd	CinClient	DateCmd
1	J14785	01/01/2020
2	A251478	10/10/2020
3	JB14796	05/02/2020
...



Ce tuple ne respecte pas la contrainte d'intégrité référentielle. ➔ Incohérence des données

1. Modèle relationnel

1.5 Le schéma relationnel.

Le schéma d'une relation (d'une table) est l'ensemble constitué du nom de la relation, suivi de la liste de tous les attributs sur lesquels elle est définie.

Par convention, on souligne les clés primaires et on fait précéder les clés étrangères d'un dièse # dans la description des colonnes de la table.

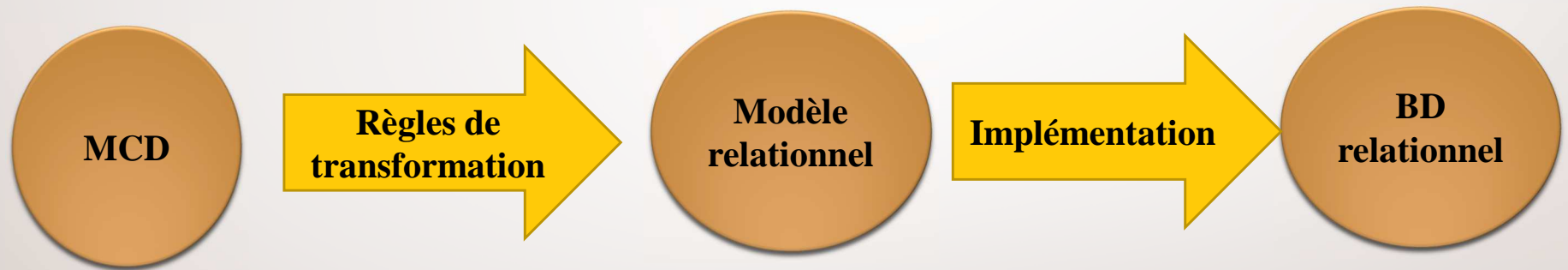
Exemple :

- Client (CinClient, NomClient, PrénomClient, AdresseClient, VilleClient)
- Commande (N°Cmd, #CinClient, DateCmd)

1. Modèle relationnel

1.6 Passage MCD vers MLD

La traduction du **MCD** en modèle logique relationnel s'effectue directement par la transformation des entités (objets) conceptuelles en relations, en fonction des règles de passage précises.



Le passage du modèle conceptuel des données au modèle logique des données s'effectue en appliquant des règles s'appuyant sur les cardinalités des couples entité - association

1. Modèle relationnel

1.6 Passage MCD vers MLD

Règles de transformation :

Règle 1 : Un objet conceptuel (entité) se transforme en relation (Table). Chaque propriété se transforme en attribut. L'identifiant de l'entité devient la clé primaire de la relation.

Client
<u>N°Client</u>
NomClient
PrenomClient
AdresseClient
VilleClient

Schéma relationnel :

Client (N°Client, NomClient, PrenomClient, AdresseClient, VilleClient)

1. Modèle relationnel

1.6 Passage MCD vers MLD

Règles de transformation :

Règle 2 (Association Père-Fils) :

Une association binaire (ou réflexive) ayant des cardinalités (1,1) - (1,n) ou (1,1) - (0,n) se traduit par l'immigration de l'identifiant de l'entité forte (ayant cardinalité (1,n) ou (0,n)) vers l'entité faible (cardinalité (1,1)).

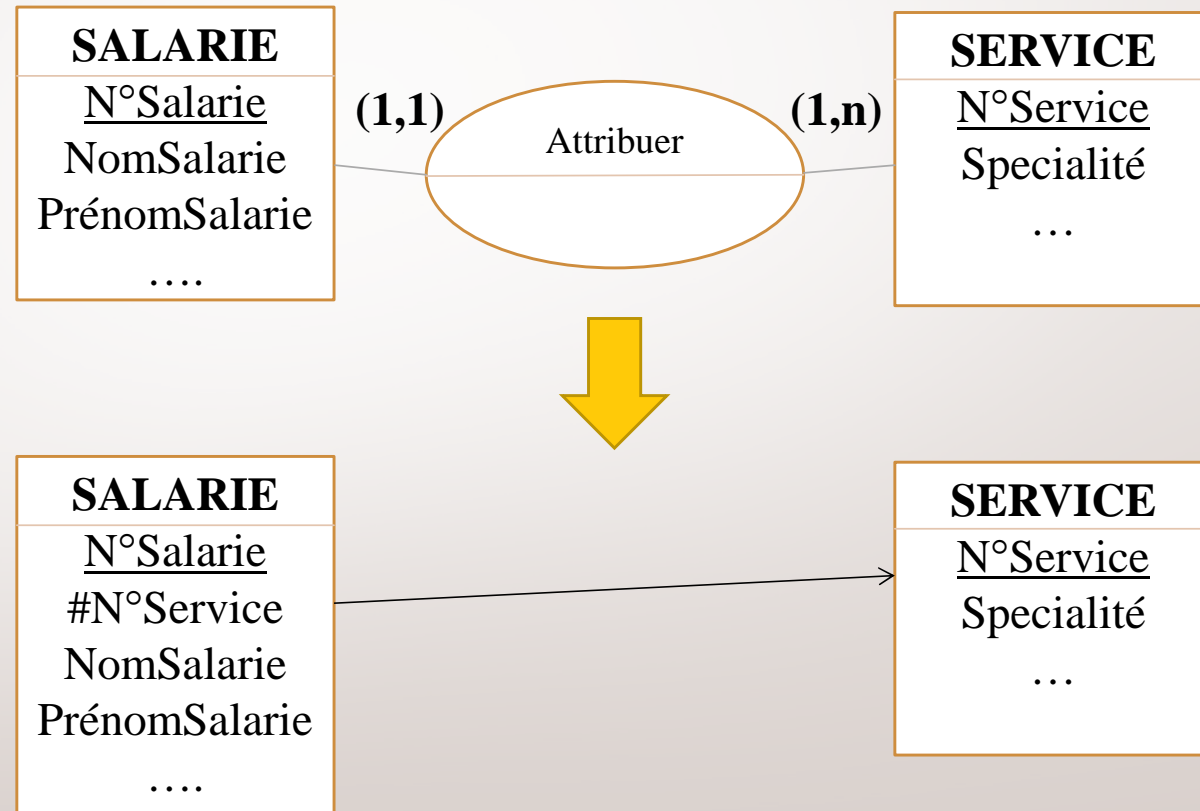


Schéma relationnel :

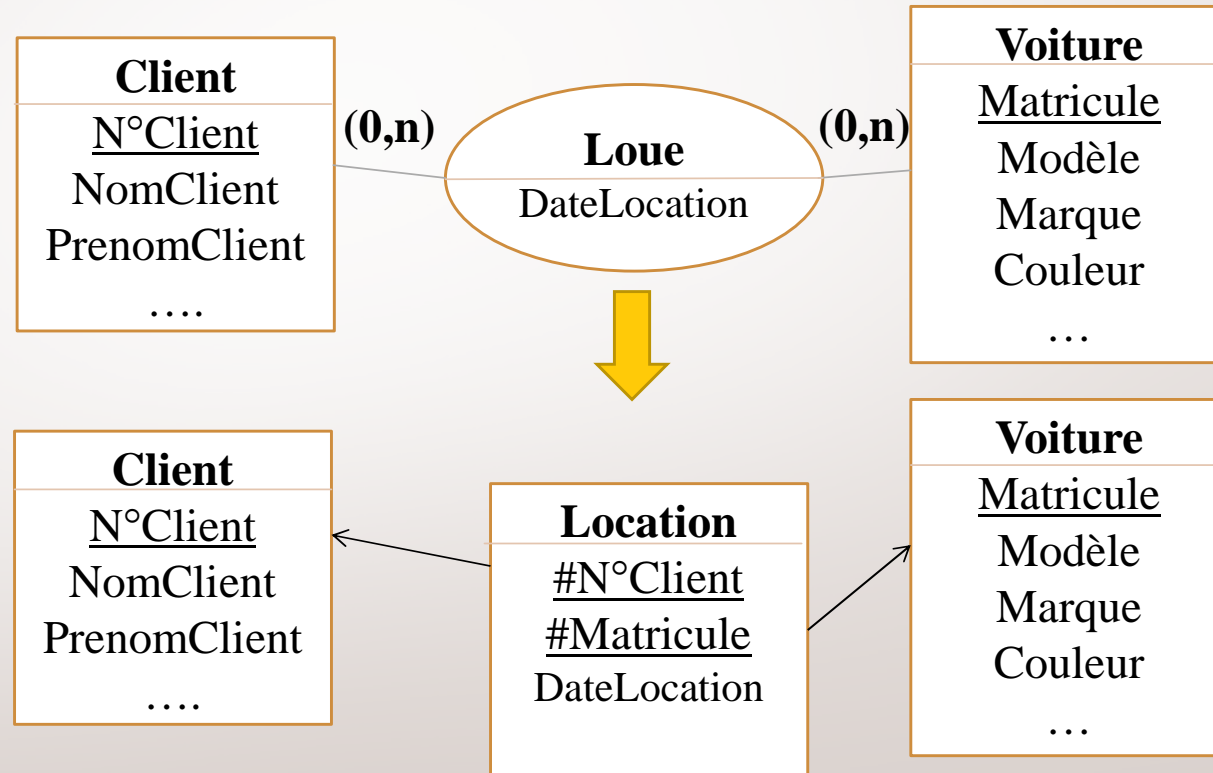
- **SALARIE** (N°Salarie, NomSalarie, PrénomSalarie, #N°Service)
- **SERVICE** (N°Service, Spécialité)

1. Modèle relationnel

1.6 Passage MCD vers MLD

Règles de transformation :

Règle 3 (Association Père-Père) :
Une association binaire de type n:n devient une table supplémentaire dont la clé primaire est composée de la concaténation des deux clés étrangères (qui référencent les deux clés primaires des deux tables en association). Les propriétés de l'association deviennent des colonnes (Attributs) de cette nouvelle table.



- Schéma relationnel :**
- **Client** (N°Client, NomClient, PrenomClient, ...)
 - **Voiture** (Matricule, Modele, Marque, Couleur, ...)
 - **Location** (#N°Client, #Matricule, DateLocation)

1. Modèle relationnel

1.6 Passage MCD vers MLD

Règles de transformation :

Règle 4 (Association Fils-Fils) :

La solution la plus simple et la plus générale pour transformer une association 1:1 consiste à traiter cette association 1:1 comme une association 1:N.

Dans ce cas l'association disparaît, la clé primaire de fils 1 devienne clé étrangère de fils 2 ou la clé primaire de fils 2 devienne clé étrangère de fils 1.

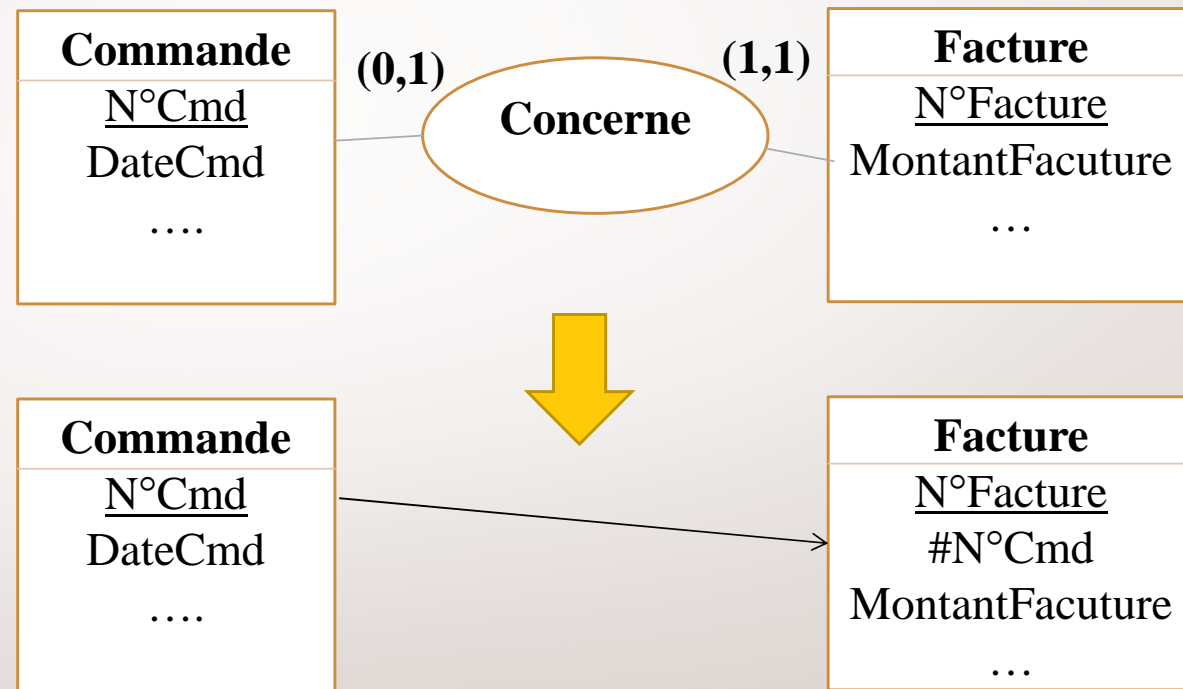


Schéma relationnel :

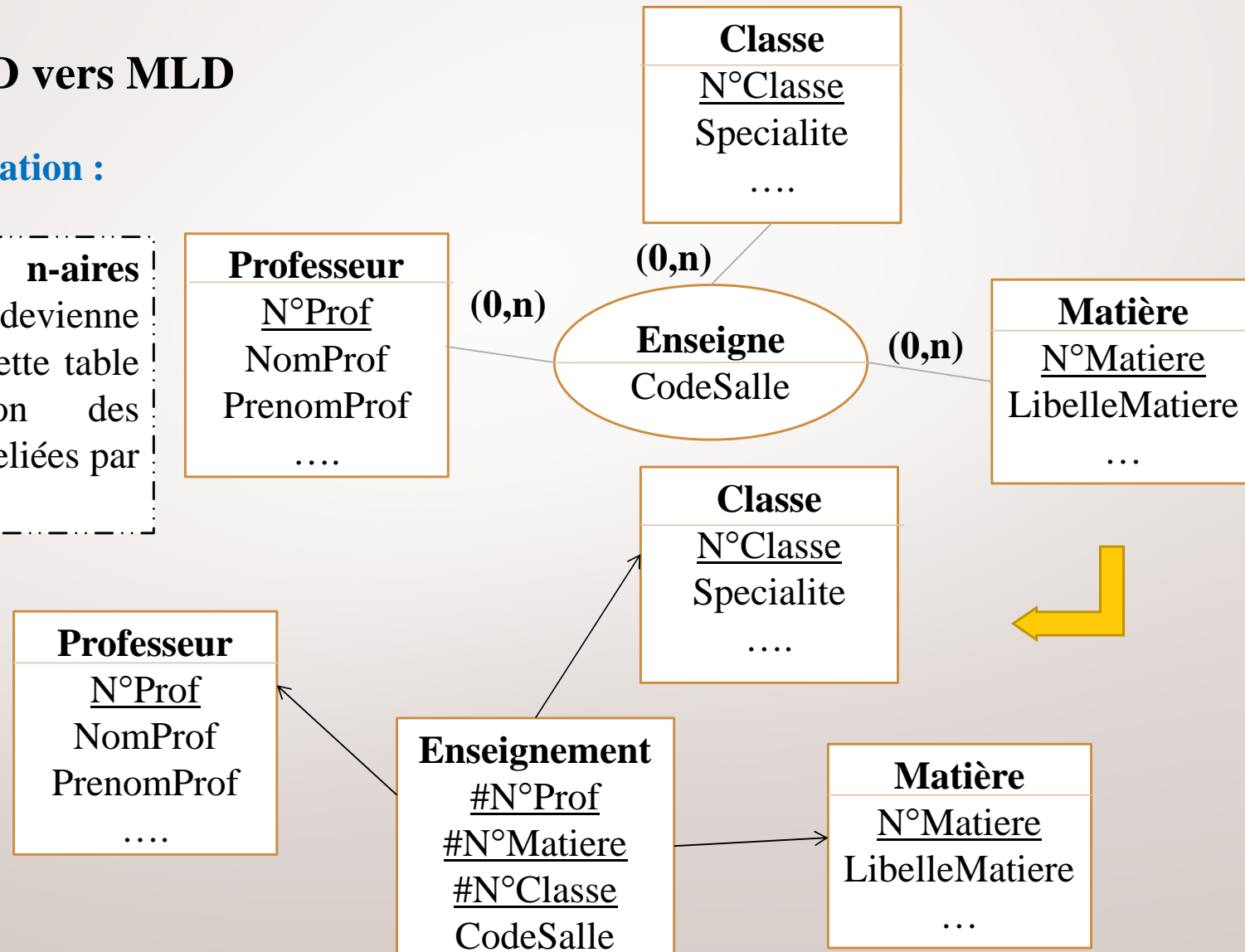
- **Commande** (N°Cmd, DateCmd)
- **Facture** (N°Facture, MontantFacture, #N°Cmd)

1. Modèle relationnel

1.6 Passage MCD vers MLD

Règles de transformation :

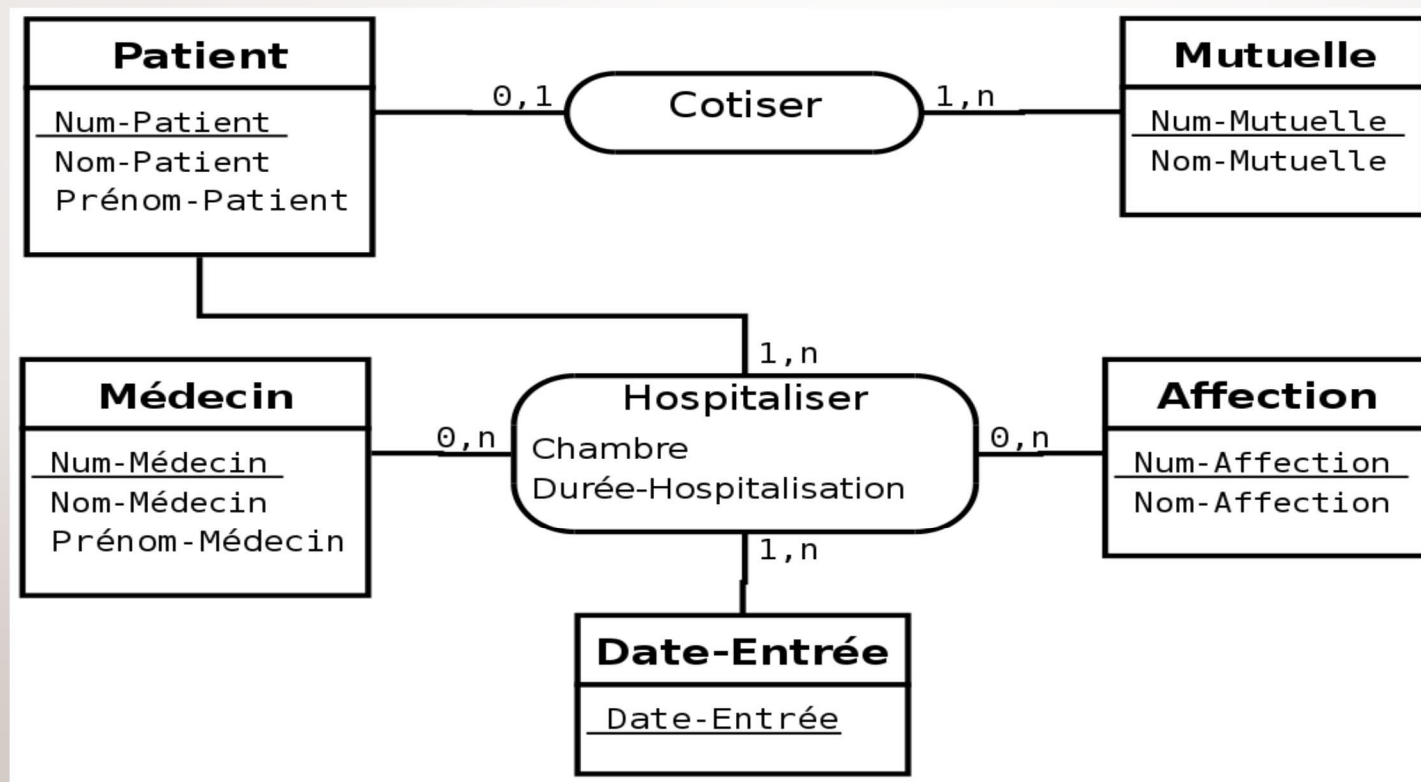
Règle 5 (Association n-aires [n>2]) : L'association devient une table et la clé de cette table est la concaténation des identifiants des entités reliées par l'association.



1. Modèle relationnel

1.6 Passage MCD vers MLD

Exercice d'application : Donner le MLD correspondant au MCD suivant





1. Modèle relationnel

1.6 Passage MCD vers MLD

Voir exercice 3 TD2

1. Modèle relationnel

1.7 Normalisation

L'application systématique des règles de vérification sur les éléments du modèle logique de données permet de s'assurer qu'il est conforme à ce que l'on attend, et donc apte à générer le modèle correspondant à la réalité à décrire.

Le modèle relationnel de données doit être raffiné par vérification et application des règles précises (**Formes Normales**).

Les formes normales ont pour objectif de permettre la décomposition de relation sans perte d'information, à partir de **la notion de DF**. L'objectif de cette décomposition est d'aboutir à un modèle performant représentant les données pertinentes du système.

1. Modèle relationnel

1.7 Normalisation

A. Première forme normale (1FN)

Une relation (Table) est en première forme normale (1FN) si :

- ❖ La relation possède une clé
- ❖ Tout attribut est atomique (élémentaire)

NumClient	Nom	Ville
001	Alaoui Salma	Rabat
002	Zerhouni Rachid	Tanger
003	Hafssi Omar	Fes
...



NumClient	Nom	Prénom	Ville
001	Alaoui	Salma	Rabat
002	Zerhouni	Rachid	Tanger
003	Hafssi	Omar	Fes
...

Si on considère le *Nom* comme un Nom Complet composé de 2 parties (Nom et Prénom).
La relation « Client » n'est pas conforme à la 1FN.

1. Modèle relationnel

1.7 Normalisation

B. Deuxième forme normale (2FN)

Une relation (Table) est en première forme normale (2FN) si :

- ❖ Elle est en première forme normale (1FN)
- ❖ Tout attribut non clé ne dépend pas d'une partie de cette clé.

On considère la relation suivante :

NumSalarie	NomSalarie	PrénomSalarie	NumProjet	NbHeures
001	Alaoui	Salma	1	20
002	Zerhouni	Rachid	2	15,5
003	Hafssi	Omar	2	18
...

Soit (NumSalarie ,
NumProjet) clé primaire
de la relation

NumSalarie + NumProjet → NbHeures

NumSalarie + NumProjet  NomSalarie

NumSalari → NomSalarie

Alors il faut scinder la table en 2 ...

1. Modèle relationnel

1.7 Normalisation

B. Deuxième forme normale (2FN)

NumSalarie	NomSalarie	PrénomSalarie	NumProjet	NbHeures
001	Alaoui	Salma	1	20
002	Zerhouni	Rachid	2	15,5
003	Hafssi	Omar	2	18
...



NumSalarie	NomSalarie	PrénomSalarie
001	Alaoui	Salma
002	Zerhouni	Rachid
003	Hafssi	Omar
...	...	

NumSalarie	NumProjet	NbHeures
001	1	20
002	2	15,5
003	2	18

1. Modèle relationnel

1.7 Normalisation

C. Troisième forme normale (3FN)

3FN permet d'assurer l'élimination de redondances dues aux dépendances transitives.

Une relation est en troisième forme normale (3FN) si :

- ❖ Elle est en 2FN
- ❖ Tout attribut non clé n'est en dépendance fonctionnelle que de la clé (pas de transitivité)

On considère la table sous le schéma relationnel suivant :

Matiere (CodeMatiere, LibelleMatiere, Langue, MatriculeProf, NomProf)

CodeMatiere	LibelleMatiere	Langue	MatriculeProf	NomProf
M-1	Informatique	Anglaise	P01	Mamoni
M-2	Chimie	Française	P02	Rachdi
M-3	Physique	Française	P03	Folam
...

Alors il faut
scinder la table
en 2 ...

La table n'est pas en 3FN : CodeMatiere → MatriculeProf → NomProf

1. Modèle relationnel

1.7 Normalisation

C. Troisième forme normale (3FN)

CodeMatiere	LibelleMatiere	Langue	MatriculeProf	NomProf
M-1	Informatique	Anglaise	P01	Mamoni
M-2	Chimie	Française	P02	Rachdi
M-3	Physique	Française	P03	Folam
...

CodeMatiere	LibelleMatiere	Langue	MatriculeProf
M-1	Informatique	Anglaise	P01
M-2	Chimie	Française	P02
M-3	Physique	Française	P03
...



MatriculeProf	NomProf
P01	Mamoni
P02	Rachdi
P03	Folam
...	...

1. Modèle relationnel

1.7 Normalisation

D. Forme Normale de Boyce Codd

Une relation est en forme normale de BOYCE-CODD (BCNF) si, et seulement si, elle est en troisième forme normale et si les seules dépendances fonctionnelles élémentaires sont celles dans lesquelles une clé détermine un attribut.

Cette forme normale permet de renforcer certaines lacunes de la troisième forme normale.

Soit, par exemple, le schéma relationnel décrivant l'enseignement d'une matière donnée à une classe par un enseignant :

- Matière(nom-matière)
- Classe(num-classe)
- Enseignant(nom-enseignant)
- Enseignement(nom-enseignant, num-classe, nom-matière)

RG: On suppose, qu'une matière n'est enseignée qu'une seule fois dans une classe et que par un seul enseignant, et qu'un enseignant n'enseigne qu'une seule matière.

1. Modèle relationnel

1.7 Normalisation

D. Forme Normale de Boyce Codd

Chacune des relations respecte bien la troisième forme normale.

- Matière(nom-matière)
- Classe(num-classe)
- Enseignant(nom-enseignant)
- Enseignement(nom-enseignant, num-classe, nom-matière)

Cependant, dans la relation Enseignement, nous avons les dépendances fonctionnelles élémentaires suivantes :

1. nom-matière, num-classe \rightarrow nom-enseignant
2. nom-enseignant \rightarrow nom-matière

Il existe donc des dépendances fonctionnelles élémentaires dont la source n'est pas la clé de la relation.

FN1: Chaque relation possède une clé et tout attribut est atomique (élémentaire).

FN2: FN1+ Tout attribut non clé ne dépend pas d'une partie de cette clé.

FN3: FN2 + Tout attribut non clé n'est en dépendance fonctionnelle que de la clé (pas de transitivité).

1. Modèle relationnel

1.7 Normalisation

D. Forme Normale de Boyce Codd

nom-enseignant	num-classe	nom-matière
George	5	Physique
George	6	Physique
George	7	Physique
George	8	Physique
Michael	5	Mathématiques
Michael	6	Mathématiques
Michael	7	Mathématiques
Michael	8	Mathématiques

Exemple de relation présentant une redondance due au non-respect de la forme normale de BOYCE-CODD.

Le non-respect de la forme normale de BOYCE-CODD entraîne une redondance illustrée par la table: pour chaque nom-enseignant identifiant un enseignant, il faut répéter le nom-matière identifiant la matière qu'il enseigne.

Pour normaliser la relation Enseignement, il faut la décomposer pour aboutir au schéma relationnel suivant :

- Matière(nom-matière)
- Classe(num-classe)
- Enseignant(nom-enseignant, nom-matière)
- Enseigner(nom-enseignant, num-classe)

1. Modèle relationnel

1.7 Normalisation

E. Forme Normale 4 et 5

Pour aller plus loin que le cours : quatrième et cinquième forme normale

Dans la pratique, la plupart des problèmes de conception peuvent être résolus en appliquant les concepts de troisième forme normale et de forme normale de BOYCE-CODD. Les quatrième et cinquième formes normales traitent encore d'autres cas de redondance, mais qui ne sont pas expliqués par des dépendances fonctionnelles.

1. Modèle relationnel

1.8 Le schéma d'une base de données relationnelle

Le schéma d'une base de données est défini par l'ensemble des schémas des relations qui la composent.

Base de données : **Bibliothèque**

- Adherent (**N°Adh**, sexe_adh, nom_adh, prenom_adh, date_naissance, adresse_adh, tel_adh)
- Auteur (**N°Aut**, nom_aut, prenom_aut, pays_aut)
- Rayon (**N°Rayon**, intitule)
- Ouvrage (**Ref**, #N°Auteur, #N°Rayon, titre, date_edition)
- Emprunt (**#N°Adh**, **#Ref**, date_emp)



1. Modèle relationnel

Voir exercice 4 TD2

2. Algèbre relationnelle

L'algèbre relationnelle est un support mathématique cohérent sur lequel repose le modèle relationnel.

La représentation d'information sous forme relationnelle est intéressante car les fondements mathématiques du relationnel, outre qu'ils permettent une modélisation logique simple et puissante, fournissent également un ensemble de concepts pour manipuler formellement l'information modélisée.

L'algèbre relationnelle fournit les opérateurs de base pour manipuler les extensions des relations d'une BD relationnelle. Autrement dit, dans le modèle relationnel, tout ce qui concerne la manipulation des relations est appelé **l'algèbre relationnelle**.

On peut distinguer trois familles d'opérateurs relationnels :

- **Les opérateurs unaires (Sélection, Projection)** : ce sont les opérateurs les plus simples, ils permettent de produire une nouvelle table à partir d'une autre table.
- **Les opérateurs binaires ensemblistes (Union, Intersection Différence)** : ces opérateurs permettent de produire une nouvelle relation à partir de deux relations de même degré (nombre d'attributs) et de même domaine.
- **Les opérateurs n-aires (Produit cartésien, Jointure, Division)** : ils permettent de produire une nouvelle table à partir de deux ou plusieurs autres tables.

2. Algèbre relationnelle

2.1 Les opérateurs unaires :

A. Sélection

La sélection (parfois appelée restriction) génère une relation regroupant exclusivement toutes les occurrences de la relation R qui satisfont l'expression logique E, on la note $\sigma_{(E)}R$.

En d'autres termes, la sélection permet de choisir (sélectionner) des lignes dans le tableau. Le résultat de la sélection est donc une nouvelle relation qui a les mêmes attributs que R.

Exemple:

Numéro	Nom	Prénom
5	Durand	Caroline
1	Germain	Stan
12	Dupont	Lisa
3	Germain	Rose-Marie

Exemple de relation *Personne*

Numéro	Nom	Prénom
5	Durand	Caroline
12	Dupont	Lisa

Exemple de sélection sur la relation *Personne*
 $\sigma (\text{Numéro} \geq 5) \text{ Personne}$

2. Algèbre relationnelle

2.1 Les opérateurs unaires : Ces opérations portent sur une seule relation

B. Projection

La projection, c'est simplement le fait de sélectionner les attributs (A_1, \dots, A_n) d'une relation R que l'on souhaite, en éliminant les autres., on la note $\Pi_{(A_1, \dots, A_n)} R$.

En d'autres termes, la projection permet de choisir des colonnes dans le tableau.

Exemple:

Numéro	Nom	Prénom
5	Durand	Caroline
1	Germain	Stan
12	Dupont	Lisa
3	Germain	Rose-Marie

Exemple de relation *Personne*

Nom
Durand
Germain
Dupont

Exemple de projection sur la relation *Personne*
 $\Pi (\text{Nom}) \text{ Personne}$

2. Algèbre relationnelle

2.2 Les opérateurs binaires ensemblistes :

A. Union

L'union est une opération portant sur deux relations R_1 et R_2 ayant le même schéma et construisant une troisième relation constituée des n-uplets appartenant à chacune des deux relations R_1 et R_2 sans doublon, on la note $R_1 \cup R_2$.

R_1 et R_2 doivent avoir les mêmes attributs et si une même occurrence existe dans R_1 et R_2 , elle n'apparaît qu'une seule fois dans le résultat de l'union. Le résultat de l'union est une nouvelle relation qui a les mêmes attributs que R_1 et R_2 .

Exemple:

Relation R_1	
Nom	Prénom
Durand	Caroline
Germain	Stan
Dupont	Lisa
Germain	Rose-Marie

Relation R_2	
Nom	Prénom
Dupont	Lisa
Juny	Carole
Fourt	Lisa

Relation R	
Nom	Prénom
Durand	Caroline
Germain	Stan
Dupont	Lisa
Germain	Rose-Marie
Juny	Carole
Fourt	Lisa

Exemple d'union : $R = R_1 \cup R_2$

2. Algèbre relationnelle

2.2 Les opérateurs binaires ensemblistes :

B. Intersection

L'intersection est une opération portant sur deux relations R_1 et R_2 ayant le même schéma et construisant une troisième relation dont les n-uplets sont constitués de ceux appartenant aux deux relations, on la note $R_1 \cap R_2$.

R_1 et R_2 doivent avoir les mêmes attributs. Le résultat de l'intersection est une nouvelle relation qui a les mêmes attributs que R_1 et R_2 .

Exemple:

Relation R_1	
Nom	Prénom
Durand	Caroline
Germain	Stan
Dupont	Lisa
Germain	Rose-Marie
Juny	Carole

Relation R_2	
Nom	Prénom
Dupont	Lisa
Juny	Carole
Fourt	Lisa
Durand	Caroline

Relation R	
Nom	Prénom
Durand	Caroline
Dupont	Lisa
Juny	Carole

Exemple d'intersection : $R = R_1 \cap R_2$

2. Algèbre relationnelle

2.2 Les opérateurs binaires ensemblistes :

C. Différence

La différence est une opération portant sur deux relations R_1 et R_2 ayant le même schéma et construisant une troisième relation dont les n-uplets sont constitués de ceux ne se trouvant que dans la relation R_1 ; on la note $R_1 - R_2$.

R_1 et R_2 doivent avoir les mêmes attributs. Le résultat de la différence est une nouvelle relation qui a les mêmes attributs que R_1 et R_2 .

Exemple:

Relation R_1	
Nom	Prénom
Durand	Caroline
Germain	Stan
Dupont	Lisa
Germain	Rose-Marie
Juny	Carole

Relation R_2	
Nom	Prénom
Dupont	Lisa
Juny	Carole
Fourt	Lisa
Durand	Caroline

Relation R	
Nom	Prénom
Germain	Stan
Germain	Rose-Marie

Exemple différence : $R = R_1 - R_2$

2. Algèbre relationnelle

2.3 Les opérateurs n-aires:

A. Produit cartésien

Le produit cartésien est une opération portant sur deux relations R1 et R2 et qui construit une troisième relation regroupant exclusivement toutes les possibilités de combinaison des occurrences des relations R1 et R2, on la note $R1 \times R2$.

Le résultat du produit cartésien est une nouvelle relation qui a tous les attributs de R1 et tous ceux de R2. Le nombre d'occurrences de la relation qui résulte du produit cartésien est le nombre d'occurrences de R1 multiplié par le nombre d'occurrences de R2.

Exemple:

Relation Amie	
Nom	Prénom
Fourt	Lisa
Juny	Carole

Relation Cadeau	
Article	Prix
livre	45
poupée	25
montre	87

Relation R			
Nom	Prénom	Article	Prix
Fourt	Lisa	livre	45
Fourt	Lisa	poupée	25
Fourt	Lisa	montre	87
Juny	Carole	livre	45
Juny	Carole	poupée	25
Juny	Carole	montre	87

Exemple Produit cartésien : $R = \text{Amie} \times \text{Cadeau}$

2. Algèbre relationnelle

2.3 Les opérateurs n-aires:

B. Jointure

La jointure est une opération portant sur deux relations R1 et R2 qui construit une troisième relation regroupant exclusivement toutes les possibilités de combinaison des occurrences des relations R1 et R2 qui satisfont l'expression logique E. La jointure est notée $R1 \bowtie_E R2$.

En fait, la jointure n'est rien d'autre qu'un produit cartésien suivi d'une sélection : $R1 \bowtie_E R2 = \sigma_{(E)}(R1 \times R2)$

Exemple:

Relation Famille			Relation Cadeau			Relation R					
Nom	Prénom	Age	AgeC	Article	Prix	Nom	Prénom	Age	AgeC	Article	Prix
Fourt	Lisa	6	99	livre	30	Fourt	Lisa	6	99	livre	30
Juny	Carole	42	6	poupée	60	Fourt	Lisa	6	20	baladeur	45
Fidus	Laure	16	20	baladeur	45	Fourt	Lisa	6	10	déguisement	15
			10	déguisement	15	Juny	Carole	42	99	livre	30
						Fidus	Laure	16	99	livre	30
						Fidus	Laure	16	20	baladeur	45

Exemple jointure : $R = \text{Famille} \bowtie_{((\text{Age} \leq \text{AgeC}) \wedge (\text{Prix} < 50))} \text{Cadeau}$

2. Algèbre relationnelle

2.3 Les opérateurs n-aires:

B. Jointure

On distingue selon l'opérateur de l'expression logique :

- ✓ Opérateur $\{<, >, \leq, \geq, \neq\}$: **théta-jointure**
- ✓ Opérateur $=$: **équi-jointure**

➤ Théta- jointure

Une theta-jointure est une jointure dans laquelle l'expression logique E est une simple comparaison entre un attribut A1 de la relation R1 et un attribut A2 de la relation R2. La theta-jointure est notée $R1 \bowtie_E R2$.

Exemple Théta- jointure : $R = \text{Famille} \bowtie_{(\text{Age} < \text{AgeC})} \text{Cadeau}$

➤ Equi jointure

Une equi-jointure est une theta-jointure dans laquelle l'expression logique E est un test d'égalité entre un attribut A1 de la relation R1 et un attribut A2 de la relation R2. L'equi-jointure est notée $R1 \bowtie_E R2$.

Exemple Equi-jointure : $R = \text{Famille} \bowtie_{(\text{Age} = \text{AgeC})} \text{Cadeau}$

2. Algèbre relationnelle

2.3 Les opérateurs n-aires:

B. Jointure

➤ Jointure naturelle

Une jointure naturelle est une equi-jointure dans laquelle les attributs des relations R_1 et R_2 portent le même nom A . Dans la relation construite, l'attribut A n'est pas dupliqué mais fusionné en un seul attribut. La jointure naturelle est notée $R_1 \bowtie R_2$.

Le résultat de la jointure naturelle est une nouvelle relation qui a tous les attributs de R_1 et tous ceux de R_2 sauf A . Il est en fait indifférent d'éliminer l'attribut A de la relation R_1 ou R_2 .

Exemple:

Relation <i>Famille</i>		
Nom	Prénom	Age
Fourt	Lisa	6
Juny	Carole	40
Fidus	Laure	20
Choupy	Emma	6

Relation <i>Cadeau</i>		
Age	Article	Prix
40	livre	45
6	poupée	25
20	montre	87

Relation <i>R</i>				
Nom	Prénom	Age	Article	Prix
Fourt	Lisa	6	poupée	25
Juny	Carole	40	livre	45
Fidus	Laure	20	montre	87
Choupy	Emma	6	poupée	25

Exemple de jointure naturelle : $R = \text{Famille} \bowtie \text{Cadeau}$

2. Algèbre relationnelle

2.3 Les opérateurs n-aires:

C. Division

La division est une opération portant sur deux relations R1 et R2, telles que le schéma de R2 est strictement inclus dans celui de R1, qui génère une troisième relation regroupant toutes les parties d'occurrences de la relation R1 qui sont associées à toutes les occurrences de la relation R2 ; on la note $R1 \div R2$.

Autrement dit, la division de R1 par R2 ($R1 \div R2$) génère une relation qui regroupe tous les n-uplets qui, concaténés à chacun des n-uplets de R2, donne toujours un n-uplet de R1.

La relation R2 ne peut pas être vide. Tous les attributs de R2 doivent être présents dans R1 et R1 doit posséder au moins un attribut de plus que R2 (inclusion stricte). Le résultat de la division est une nouvelle relation qui a tous les attributs de R1 sans aucun de ceux de R2.

Relation Enseignement	
Enseignant	Etudiant
Germain	Dubois
Fidus	Pascal
Robert	Dubois
Germain	Pascal
Fidus	Dubois
Germain	Durand
Robert	Durand

Relation Etudiant
Nom
Dubois
Pascal

Relation R
Enseignant
Germain
Fidus

Exemple de division : $R = \text{Enseignement} \div \text{Etudiant}$.

La relation R contient donc tous les enseignants de la relation Enseignement qui enseignent à tous les étudiants de la relation Etudiant.

2. Algèbre relationnelle

Exercice d'application

A. On considère la relation *Client* suivante :

<u>Id</u>	Nom	Prénom	Courriel
3	Bernard	Alain	ab@gmail.com
23	Perrier	Charles	cp@yahoo.ca
2	Labbé	Caroline	cl@hotmail.com
8	Barrette	Patricia	pb@bell.ca

Donner le résultat des opérations suivantes :

- $\Pi (\text{Nom}) \text{ Client}$
- $\Pi (\text{Nom}, \text{Prénom}) \text{ Client}$
- $\sigma (\text{Id} > 4) \text{ Client}$
- $\sigma (\text{Id} \leq 2 \cup \text{Id} \geq 20) \text{ Client}$

2. Algèbre relationnelle

Client

<u>Id</u>	Nom	Prénom	Courriel
3	Bernard	Alain	ab@gmail.com
23	Perrier	Charles	cp@yahoo.ca
2	Labbé	Caroline	cl@hotmail.com
8	Barrette	Patricia	pb@bell.ca

Π (Nom) *Client*

<u>Nom</u>
Bernard
Perrier
Labbé
Barrette

Π (Nom, Prénom) *Client*

<u>Nom</u>	<u>Prénom</u>
Bernard	Alain
Perrier	Charles
Labbé	Caroline
Barrette	Patricia

2. Algèbre relationnelle

Client

<u>Id</u>	Nom	Prénom	Courriel
3	Bernard	Alain	ab@gmail.com
23	Perrier	Charles	cp@yahoo.ca
2	Labbé	Caroline	cl@hotmail.com
8	Barrette	Patricia	pb@bell.ca

$\sigma (Id > 4) \text{ Client}$

<u>Id</u>	Nom	Prénom	Courriel
23	Perrier	Charles	cp@yahoo.ca
8	Barrette	Patricia	pb@bell.ca

$\sigma (Id \leq 2 \cup Id = 20) \text{ Client}$

<u>Id</u>	Nom	Prénom	Courriel
23	Perrier	Charles	cp@yahoo.ca
2	Labbé	Caroline	cl@hotmail.com

2. Algèbre relationnelle

Exercice d'application

B. On considère les deux relations *Client* et *Employé* suivantes :

Employé

Nom	Prénom
Prévost	Charlotte
Richter	Karl
Barette	Patricia

Client

Nom	Prénom
Bernard	Alain
Perrier	Charles
Labbé	Caroline
Barrette	Patricia

Donner le résultat des opérations suivantes :

- $Client \cup Employé$
- $Client \cap Employé$
- $Client - Employé$
- $Employé - Client$

2. Algèbre relationnelle

Client U Employé

Nom	Prénom
Prévost	Charlotte
Richter	Karl
Barette	Patricia
Bernard	Alain
Perrier	Charles
Labbé	Caroline

Employé

Nom	Prénom
Prévost	Charlotte
Richter	Karl
Barette	Patricia

Client

Nom	Prénom
Bernard	Alain
Perrier	Charles
Labbé	Caroline
Barrette	Patricia

Client \cap Employé

Nom	Prénom
Barette	Patricia

2. Algèbre relationnelle

Employé - Client

Nom	Prénom
Prévost	Charlotte
Richter	Karl

Client - Employé

Nom	Prénom
Bernard	Alain
Perrier	Charles
Labbé	Caroline

Employé

Nom	Prénom
Prévost	Charlotte
Richter	Karl
Barette	Patricia

Client

Nom	Prénom
Bernard	Alain
Perrier	Charles
Labbé	Caroline
Barrette	Patricia

2. Algèbre relationnelle

Exercice d'application

C. On considère les deux relations *Producteur* et *Produit* suivantes :

Donner le résultat des opérations suivantes :

- $\text{Producteur} \times \text{Produit}$
- $\text{Producteur} \bowtie_{(\text{Produit} = \text{Id})} \text{Produit}$
- $\text{Producteur} \bowtie \text{Produit}$

Produit

Id	Nom
4	Fraise
7	Mais

Producteur

Nom	Prénom	Produit
Bernard	Alain	4
Perrier	Charles	2
Labbé	Caroline	4

2. Algèbre relationnelle

Produit

Id	Nom
4	Fraise
7	Mais

Producteur

Nom	Prénom	Produit
Bernard	Alain	4
Perrier	Charles	2
Labbé	Caroline	4

Producteur x Produit

Nom	Prénom	Produit	Id	Nom
Bernard	Alain	4	4	Fraise
Bernard	Alain	4	7	Mais
Perrier	Charles	2	4	Fraise
Perrier	Charles	2	7	Mais
Labbé	Caroline	4	4	Fraise
Labbé	Caroline	4	7	Mais

2. Algèbre relationnelle

Produit

Id	Nom
4	Fraise
7	Mais

Producteur

Nom	Prénom	Produit
Bernard	Alain	4
Perrier	Charles	2
Labbé	Caroline	4

$\text{Producteur} \bowtie_{(\text{Produit} = \text{Id})} \text{Produit}$

Nom	Prénom	Produit	Id	Nom
Bernard	Alain	4	4	Fraise
Labbé	Caroline	4	4	Fraise

$\text{Producteur} \bowtie \text{Produit}$

Nom	Prénom	IdProduit	Nom
Bernard	Alain	4	Fraise
Labbé	Caroline	4	Fraise

2. Algèbre relationnelle

2.4 Les fonctions d'agrégation :

En algèbre relationnel on a la possibilité d'effectuer des calculs sur les attributs ce calcul peut être effectué en ligne par des expressions arithmétiques sur les attributs ou en colonne par les fonctions agrégatives.

Exemple:

Ventes

client	produit	prix	quantité
Ali	PC	3000	2
Ahmed	Clavier	50	10
Rachid	Souris	20	50

$R = \Pi \text{ client, produit, prix} * \text{quantité (ventes)}$

R		
client	produit	prix*quantité
Ali	PC	6000
Ahmed	Clavier	500
Rachid	Souris	1000

2. Algèbre relationnelle

2.4 Les fonctions d'agrégation :

Fonctions agrégatives : Les expressions de calcul permettent d'effectuer des opérations de calcul en ligne, sur des attributs de relations. Le concept d'**agrégat** permet de telles opérations.

Définition : Agrégat

Partitionnement horizontal d'une relation en fonction des valeurs d'un groupe d'attributs, suivi d'un regroupement par application d'une fonction de calcul sur ensemble.

Notation :

R = AGREGAT(RELATION ; ATTRIBUT1 ; FONCTION{ATTRIBUT2})

2. Algèbre relationnelle

2.4 Les fonctions d'agrégation :

Exemple: Soit la relation « notes » suivante :

Classe	Elève	Note
MPSI	Meyer	17,5
PCSI	Martin	7,75
MPSI	Bernard	9,25
PCSI	Robert	14,0
PCSI	Dubois	11,5

R1 = AGREGAT(notes ; classe ; MOYENNE{Note})

Cet expression sert à regrouper les élèves d'une même classe (ce groupe de valeurs est appelé un *agrégat*) et à effectuer une opération sur chacun des agrégats. Ici par exemple, on pourrait calculer la moyenne sur chaque classe, ce qui produit la relation suivante :

Le résultat R1 est :

Classe	MOYENNE(Note)
MPSI	13.38
PCSI	11.08

2. Algèbre relationnelle

2.4 Les fonctions d'agrégation :

Exemple: Soit la relation « notes » suivante :

notes		
Classe	Elève	Note
MPSI	Meyer	17,5
PCSI	Martin	7,75
MPSI	Bernard	9,25
PCSI	Robert	14,0
PCSI	Dubois	11,5

R2 = AGREGAT(notes ;MAXIMUM{Note})

Le résultat R2 est :

MAXIMUM (Note)
17,5

R2 = AGREGAT(notes ;COMPTE{Note})

Le résultat R2 est :

COMPTE (Note)
5

2. Algèbre relationnelle

2.4 Les fonctions d'agrégation :

Les fonctions de calcul sur ensemble les plus souvent proposées sont :

- SOMME (**SUM**) permettant de calculer la somme des éléments d'un ensemble ;
- MOYENNE (**AVG**) permettant de calculer la moyenne des éléments d'un ensemble ;
- MINIMUM (**MIN**) permettant de sélectionner l'élément minimum d'un ensemble ;
- MAXIMUM (**MAX**) permettant de sélectionner l'élément maximum d'un ensemble ;
- COMPTE (**COUNT**) permettant de compter les éléments d'un ensemble.



Voir TD 3