

Bases de Données

(G3SI2BD)

AIR1

Manel Zarrouk
zarrouk@lipn.univ-paris13.fr

Bases de Données

Plan du cours

Part 1 - Introduction

Part 2 - L'algèbre relationnelle

Part 3 - SQL

Part 4 - Modélisation

Part 5 - Programmes et transactions

Bases de Données

Sources des transparents:

MOOC Bases de données relationnelles : apprendre pour utiliser

Philippe Rigaux et Serge Abiteboul

Lien:

<https://www.fun-mooc.fr/courses/course-v1:CNAM+01041+session01/about>

Je vous encourage fortement à regarder les vidéos de ce MOOC car ils sont complémentaires aux transparents

Bases de Données

Part 2 - L'algèbre relationnelle

- Projection, Selection, Renommage
- Jointure
- Les opérations ensemblistes
- La notion de composition
- Des opérations redondantes
- Évaluation et optimisation

Bases de Données

Part 2 - L'algèbre relationnelle

- Projection, Selection, Renommage
- Jointure
- Les opérations ensemblistes
- La notion de composition
- Des opérations redondantes
- Évaluation et optimisation

Projection, Sélection, Renommage

On va prendre une BD "Simple"

R

A	B	C	D
1	0	2	2
4	1	2	2
6	0	6	3
7	1	7	3
1	0	7	0
1	1	6	0

S

A	B
1	2
4	2
6	6
7	7
1	7
1	6

S'

B	C
2	3
2	5
9	1
6	8

T

A	B
1	2
5	4
1	6
8	5
7	7

Projection, Sélection, Renommage

Des requêtes élémentaires

R

A	B	C	D
1	0	2	2
4	1	2	2
6	0	6	3
7	1	7	3
1	0	7	0
1	1	6	0

Requête : R

retourne la relation
R de Simple

Pareil pour S, S', T

Projection, Sélection, Renommage

Algèbre relationnelle : des opérations

Une boîte à outils d'opérations (fonctions)

- Input : relations
- Output : relation

Donc on peut les composer

$op1 ((R \text{ } op2 \text{ } S) \text{ } op2 \text{ } (op1 \text{ } T))$



Note : c'est pour cela qu'on parle **d'algèbre**

Projection, Sélection, Renommage

Les outils pour une seule Table

Que peut-on faire avec une seule table ?

Sélection

- Supprimer des lignes

Projection

- Supprimer des colonnes

Renommage

- Renommer les attributs

R

A	B	C	D
1	0	2	2
4	1	2	2
6	0	6	3
7	1	7	3
1	0	7	0
1	1	6	0

Projection, Sélection, Renommage

SÉLECTION $\sigma_{A=1}(R)$

Garder les nuplets de R qui satisfont la condition A=1

R

A	B	C	D
1	0	2	2
4	1	2	2
6	0	6	3
7	1	7	3
1	0	7	0
1	1	6	0

Supprimer des
lignes (nuplets)
qui ne satisfont
pas la condition

$\sigma_{A=1}(R)$

A	B	C	D
1	0	2	2
1	0	7	0
1	1	6	0

Projection, Sélection, Renommage

SÉLECTION $\sigma_C(R) = \{ t \in R \mid t \text{ satisfait } C \}$

$\sigma_{condition}(R)$

- La condition porte sur les attributs de R
 - attribut = valeur ou attribut = attribut
 - attribut \neq valeur ou attribut \neq attribut
 - Exemple $A = 1$, $A = B$, $A \neq 1$, $A \neq B$
- $\text{type}(\sigma_{condition}(R)) = \text{type}(R)$

Evaluation : facile

- Temps linéaire
- Mieux avec un index

Projection, Sélection, Renommage

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

Garder les colonnes A et C

R

A	B	C	D
1	0	1	2
4	1	2	2
6	0	6	3
7	1	1	3
1	0	1	0
1	1	1	0

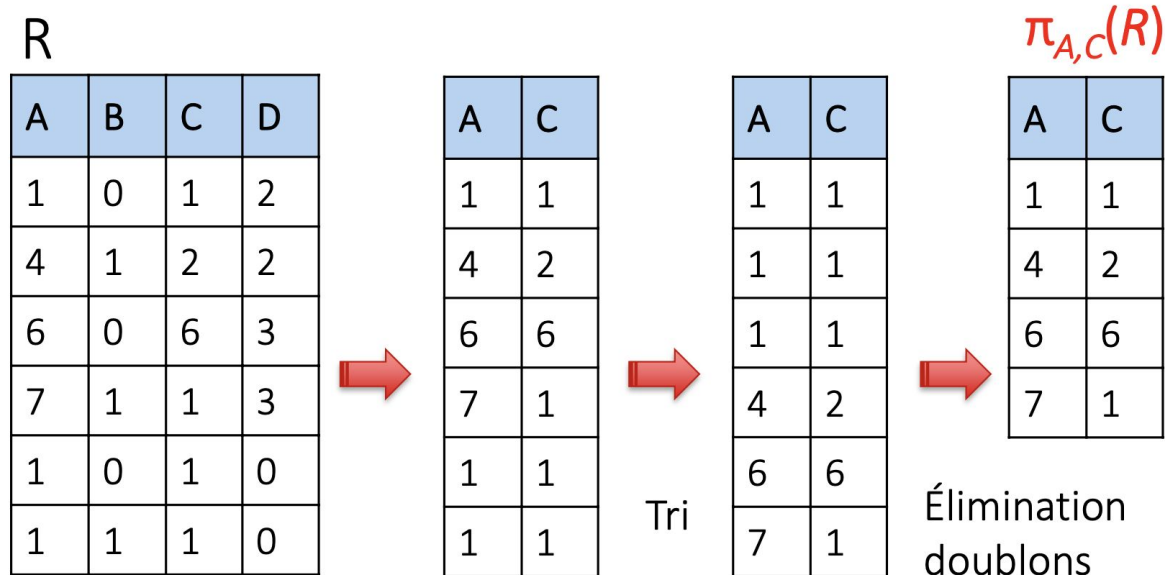
Supprimer les
colonnes
(attributs) qu'on
ne garde pas

Attention : une relation
est un **ensemble** de
nuplets
Pas de doublon

???

Projection, Sélection, Renommage

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



Projection, Sélection, Renommage

PROJECTION $\pi_{A1,...,An}(R)=\{t \mid_{A1,...,An} \mid t \in R$

$\pi_{A1,...,An}(R)$

- $\{A1,...,An\}$: sous-ensemble des attributs de R
- $Type(\pi_{A1,...,An}(R)) = \{A1,...,An\}$

Efficace à évaluer

- Tri : $n \log(n)$
- Élimination des doublons en temps linéaire

Projection, Sélection, Renommage

Retour à la composition

“Donnez moi les noms et lieux des hôtels”

Logement

id	nom	capacité	type	lieu
pi	U Pinzuttu	10	Gîte	Corse
ta	Tabriz	34	Hôtel	Alsace
be	Benbow	45	Auberge	Cévennes
re	Réforme	134	Hôtel	Alpes
sh	Tashilhunpo	28	Monastère	Bretagne

Projection, Sélection, Renommage

Retour à la composition

“Donnez moi les noms et lieux des hôtels”

$\sigma_{type="H\hat{o}tel"}(Logement)$

id	nom	capacité	type	lieu
pi	U Pinzuttu	10	Gîte	Corse
ta	Tabriz	34	Hôtel	Alsace
be	Benbow	45	Auberge	Cévennes
re	Réforme	134	Hôtel	Alpes
sh	Tashilhunpo	28	Monastère	Bretagne

Projection, Sélection, Renommage

Retour à la composition

“Donnez moi les noms et lieux des hôtels”

$\pi_{nom, lieu}(\sigma_{type = 'H\hat{o}tel'}(Logement))$

id	nom	capacité	type	lieu
pi	U Pinzuttu	10	Gîte	Corse
ta	Tabriz	34	Hôtel	Alsace
be	Benbow	45	Auberge	Cévennes
re	Réforme	134	Hôtel	Alpes
sh	Tashilhunpo	28	Monastère	Bretagne

Projection, Sélection, Renommage

Retour à la composition

“Donnez moi les noms et lieux des hôtels”

$\pi_{nom, lieu}(\sigma_{type='H\hat{o}tel'}(Logement))$

nom	lieu
Tabriz	Alsace
Réforme	Alpes

Projection, Sélection, Renommage

RENOMMAGE $\delta_{A \rightarrow A', D \rightarrow E}(R)$

Renommage

- attribut \rightarrow attribut

Exemple : $A \rightarrow A', D \rightarrow E$

Modifier les noms
d'attributs

$\delta_{A \rightarrow A', D \rightarrow E}(R)$

A'	B	C	E
1	0	2	2
4	1	2	2
6	0	6	3
7	1	7	3
1	0	7	0
1	1	6	0

Projection, Sélection, Renommage À retenir

Les opérations unaires de l'algèbre

- Projection : supprimer des colonnes
- Sélection : supprimer des lignes
- Renommage : renommer des colonnes

Bases de Données

Part 2 - L'algèbre relationnelle

- Projection, Selection, Renommage
- Jointure
- Les opérations ensemblistes
- La notion de composition
- Des opérations redondantes
- Évaluation et optimisation

Jointure

Combiner des informations des deux tables

Jointure naturelle ⌘

JOINTURE (NATURELLE) ⌘

Deux nuplets qui ont la même valeur pour la partie commune (ici B) peuvent être combinés pour obtenir un nouvel nuplet

S		S'	
A	B	B	C
1	2	2	3
4	2	2	5
6	6	9	1
7	7	6	8
1	7		
1	6		

A	B	C
4	2	5

La jointure ⋈

$type(S) = V ; type(S') = W$

$type(S \bowtie S') = V \cup W$

S		S'	
A	B	B	C
1	2	2	3
4	2	2	5
6	6	9	1
7	7	6	8
1	7		
1	6		

Perte d'information
Redondance

$S \bowtie S'$

A	B	C
1	2	3
1	2	5
4	2	3
4	2	5
1	6	8
6	6	8

Jointure

Définition formelle de la jointure

$$S \bowtie S' = \{ t \text{ sur } V \cup W \mid \exists v \in S \text{ et } w \in S', \\ v|_{V \cap W} = w|_{V \cap W}, t|_V = v, t|_W = w \}$$

S			S'		
A	B		B	C	
1	2	→	2	3	←
4	2		2	5	
6	6		9	1	
7	7		6	8	
1	7				
1	6				

=

Et on itère

Choisir v

Choisir w

Si $v|_{V \cap W} = w|_{V \cap W}$

cela donne un t

A	B	C
1	2	3

Jointure

Evaluation de la jointure

Relativement efficace à évaluer

1. Trier chaque relation suivant les attributs communs
2. Les fusionner en les parcourant en parallèle

Dans le pire des cas le nombre de nuplets du résultat est le produit des nombres de nuplets des relations en input

Jointure

Comment joindre sur des noms d'attributs différents

S1		S2	
B	A	A	C
2	1	2	3
2	4	2	5
5	6	9	1
7	7	6	8
7	1		
6	1		

$S1 \bowtie \delta_{B \rightarrow A}(S2)$

B	A	C
5	6	8

Utiliser un renommage

Jointure

Un cas limite de la jointure : le produit cartésien X

$$\text{type}(Q) \cap \text{type}(Q') = \emptyset$$

$$\text{Produit cartésien } Q \times Q' = Q \bowtie Q'$$

$$\text{taille}(Q \bowtie Q') = \text{taille}(Q) \times \text{taille}(Q')$$

$$Q = \sigma_{A=B}(S)$$

A	B
6	6
7	7

$$Q' = \delta_{B \rightarrow B'}(\sigma_{B \neq 2}(S'))$$

B'	C
9	1
6	8

$$Q \times Q'$$

A	B	B'	C
6	6	9	1
6	6	6	8
7	7	9	1
7	7	6	8

Jointure À retenir

Le cœur de l'algèbre, une opération binaire :
la jointure

Permet de faire des « ponts » entre des
nuplets de deux tables

Bases de Données

Part 2 - L'algèbre relationnelle

- Projection, Selection, Renommage
- Jointure
- Les opérations ensemblistes
- La notion de composition
- Des opérations redondantes
- Évaluation et optimisation

Les opérations ensemblistes

Q et Q' des requêtes algébriques qui retournent des ensembles de nuplets du même type

Alors on peut calculer leur \cup , $-$, \cap

Attention : seulement si les deux ensembles ont le même type

« on ne compare pas des pommes et des oranges »

Les opérations ensemblistes

L'UNION $S \cup T$

$$\text{type}(S \cup T) = \text{type}(S) = \text{type}(T)$$

S

A	B
1	2
4	2
6	6
7	7
1	7
1	6

T

A	B
1	2
5	4
1	6
8	5
7	7

$S \cup T$

A	B
4	2
6	6
1	7
1	2
7	7
1	6
5	4
8	5

Les opérations ensemblistes

La DIFFERENCE S - T

$$\text{type}(S - T) = \text{type}(S) = \text{type}(T)$$

S		T		S - T	
A	B	A	B	A	B
1	2	1	2	4	2
4	2	5	4	6	6
6	6	1	6	1	7
7	7	8	5		
1	7	7	7		
1	6				

Les opérations ensemblistes

L'INTERSECTION : on l'a déjà

$\text{type}(S) = \text{type}(T)$

$$S \cap T = S \bowtie T$$

S

A	B
1	2
4	2
6	6
7	7
1	7
1	6

T

A	B
1	2
5	4
1	6
8	5
7	7

$S \cap T$

A	B
1	2
7	7
1	6

Une autre façon

$$S \cap T = (T - (T - S))$$

Les opérations ensemblistes

Evaluation de l'UNION et la DIFFERENCE

Trier les deux relations

Les parcourir dans l'ordre en gardant

- Les nuplets des deux côtés pour l'union
- Les nuplets du premier qui ne sont pas dans le second pour l'union

Dans la même idée que ce qu'on a fait pour la jointure (intersection)

Le coût est à chaque fois celui du tri

Les opérations ensemblistes

Evaluation de l'UNION et la DIFFERENCE

Trier les deux relations

Les parcourir dans l'ordre en gardant

- Les nuplets des deux côtés pour l'union
- Les nuplets du premier qui ne sont pas dans le second pour l'union

Dans la même idée que ce qu'on a fait pour la jointure (intersection)

Le coût est à chaque fois celui du tri

Les opérations ensemblistes

À retenir

Deux opérations binaires :

- L'union
- La différence

Et nous savons faire l'intersection

Bases de Données

Part 2 - L'algèbre relationnelle

- Projection, Selection, Renommage
- Jointure
- Les opérations ensemblistes
- La notion de composition
- Des opérations redondantes
- Évaluation et optimisation

La notion de composition

Une requête plus complexe

Quels sont les pays pour lesquels nous avons des clients qui sont allés en Corse mais qui ne sont jamais allés en gîte ?



La notion de composition

Étape 0 : relier Client et Logement

Client (id, nom, prénom, ville, pays)

Séjour (id, idClient, idLogement, début, fin)

Logement (id, nom, capacité, type, lieu)

$$Q0 = \delta_{id \rightarrow idClient, nom \rightarrow nomClient}(\text{Client}) \bowtie \\ (\text{Séjour} \bowtie \delta_{id \rightarrow idLogement}(\text{Logement}))$$

La notion de composition

Étape 1

Q1 : liste des Clients qui ont déjà fait un séjour en Corse et le pays d'où ils viennent

$$Q0 = \delta_{id \rightarrow idClient, nom \rightarrow nomClient}(Client) \bowtie \\ (Séjour \bowtie \delta_{id \rightarrow idLogement}(Logement))$$

$$Q1 = \pi_{pays, idClient}(\sigma_{lieu='Corse'}(Q0))$$

La notion de composition

Étape 2

Q2 : la liste des clients qui sont déjà allés dans un gîte

$$Q0 = \delta_{id \rightarrow idClient, nom \rightarrow nomClient}(Client) \bowtie \\ (Séjour \bowtie \delta_{id \rightarrow idLogement}(Logement))$$

$$Q2 = \pi_{idClient}(\sigma_{type='gite'}(Q0))$$

La notion de composition

Étape 3

Q3 : liste des clients qui ne sont jamais allés dans un gîte

$\delta_{id \rightarrow idclient}(\text{Client}) - Q2$

La notion de composition

Étape finale

Quels sont les pays pour lesquels nous avons des clients qui sont allés en Corse mais qui ne sont jamais allés en gîte ?

Pour répondre à la question

$$\begin{aligned} & \pi_{\text{pays}}(Q1 \bowtie Q3) \\ &= \pi_{\text{pays}}(\pi_{\text{pays}, \text{idClient}}(\sigma_{\text{lieu}='Corse'}(Q0))) \bowtie (\delta_{\text{id} \rightarrow \text{idclient}}(\text{Client}) - \pi_{\text{idClient}}(\sigma_{\text{type}='gite'}(Q0))) \end{aligned}$$

(On pourra s'amuser à remplacer Q0 par sa valeur)

La notion de composition

À retenir

La puissance de l'algèbre : la composition

On peut créer la requête par petits bouts
(résoudre d'abord des sous problèmes)

Bases de Données

Part 2 - L'algèbre relationnelle

- Projection, Selection, Renommage
- Jointure
- Les opérations ensemblistes
- La notion de composition
- Des opérations redondantes
- Évaluation et optimisation

Des opération redondantes

Les sélections plus compliquées

La division cartésienne

La semi-jointure

La θ -jointure

Des opération redondantes

Les SELECTIONS plus compliqués

La sélection de l'algèbre ne permet que des sélections très simples

On peut imaginer des sélections plus compliquées

- $\sigma_F(R)$ avec

$F : (A=1 \text{ ou } A=B) \text{ et } ((\text{non} (B=2 \text{ et } C=D)) \text{ et } A=D)$

- $\sigma_F(R) = \{ u \in R \mid u \text{ satisfait } F \}$

Facile à réaliser :

$$((\sigma_{A=1}(R) \cup \sigma_{A=B}(R)) - (\sigma_{B=2}(R) \bowtie \sigma_{C=D}(R))) \bowtie \sigma_{A=D}(R)$$

Des opération redondantes

La division cartésienne

Quels sont les clients qui sont allés dans tous les types de logement

Rappel : $Q0 = \delta_{id \rightarrow idClient, nom \rightarrow nomClient}(Client) \bowtie$
 $(Séjour \bowtie \delta_{id \rightarrow idLogement}(Logement))$

1. Qui est allé dans quel type de logement ?

- $H1 = \pi_{idClient, type}(Q0)$

2. Quels sont les types de logement ?

- $H2 = \pi_{type}(Logement)$

3. $type(H1 \div H2) = type(H1) - type(H2)$

4. $H1 \div H2 = \{ u \mid \forall v \text{ in } H2, uv \text{ in } H1 \}$

On verra en exercice comment réaliser cela avec les opérations de l'algèbre déjà apprises

Des opération redondantes

La semi jointure

$$\text{type}(H3 \bowtie H4) = \text{type}(H3)$$

$$H3 \ltimes H4 = \{ u \text{ in } H3 \mid u \text{ joint avec un nuplet de } H4 \}$$

Très simple à réaliser :

$$H3 \ltimes H4 = \pi_{\text{type}(H3)}(H3 \bowtie H4)$$

Utile dans les bases de données distribuées

Des opération redondantes

La Θ -jointure

(déjà vu : joindre sur des attributs différents)

Les débuts de séjour en Corse

$Q1 = \sigma_{\text{lieu}='Corse'}(\text{Logement})$

$Q2 = \pi_{\text{idlogement}, \text{début}}(\text{Séjour})$

$\pi_{\text{début}}(Q1 \bowtie_{\text{id=idlogement}} Q2)$

Des opération redondantes

À retenir

On peut faire des tas d'autres choses avec des relations juste avec les opérateurs de l'algèbre relationnelle

Bases de Données

Part 2 - L'algèbre relationnelle

- Projection, Selection, Renommage
- Jointure
- Les opérations ensemblistes
- La notion de composition
- Des opérations redondantes
- Évaluation et optimisation

Évaluation et optimisation

Évaluation de requêtes

Pour chaque opération de l'algèbre, on peut trouver un algorithme (en fait plusieurs) pour l'évaluer

A partir de là, pour chaque requête, on peut trouver un programme pour l'évaluer (en fait plusieurs)

Dans un SGBD, **le compilateur**

Requête de l'algèbre ➡ programme, pour l'évaluer

Dans le SGBD, **l'optimiseur**

➡ Recherche d'un programme plus efficace

Évaluation et optimisation

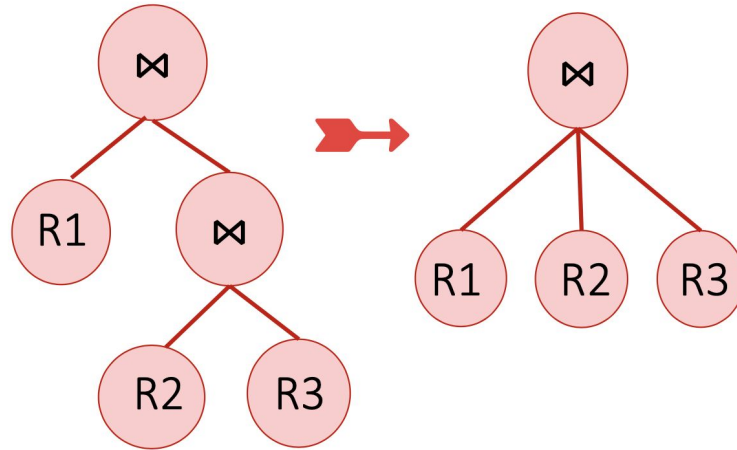
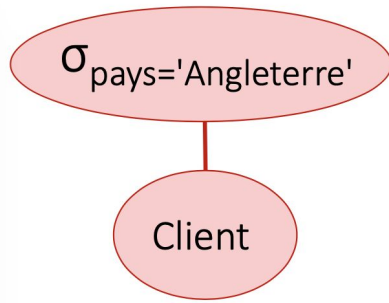
Comment optimiser?

Utilisation d'une représentation des requêtes
sous forme d'arbre

Utilisation de propriétés algébriques

Évaluation et optimisation

Une représentation graphique sous forme d'arbre

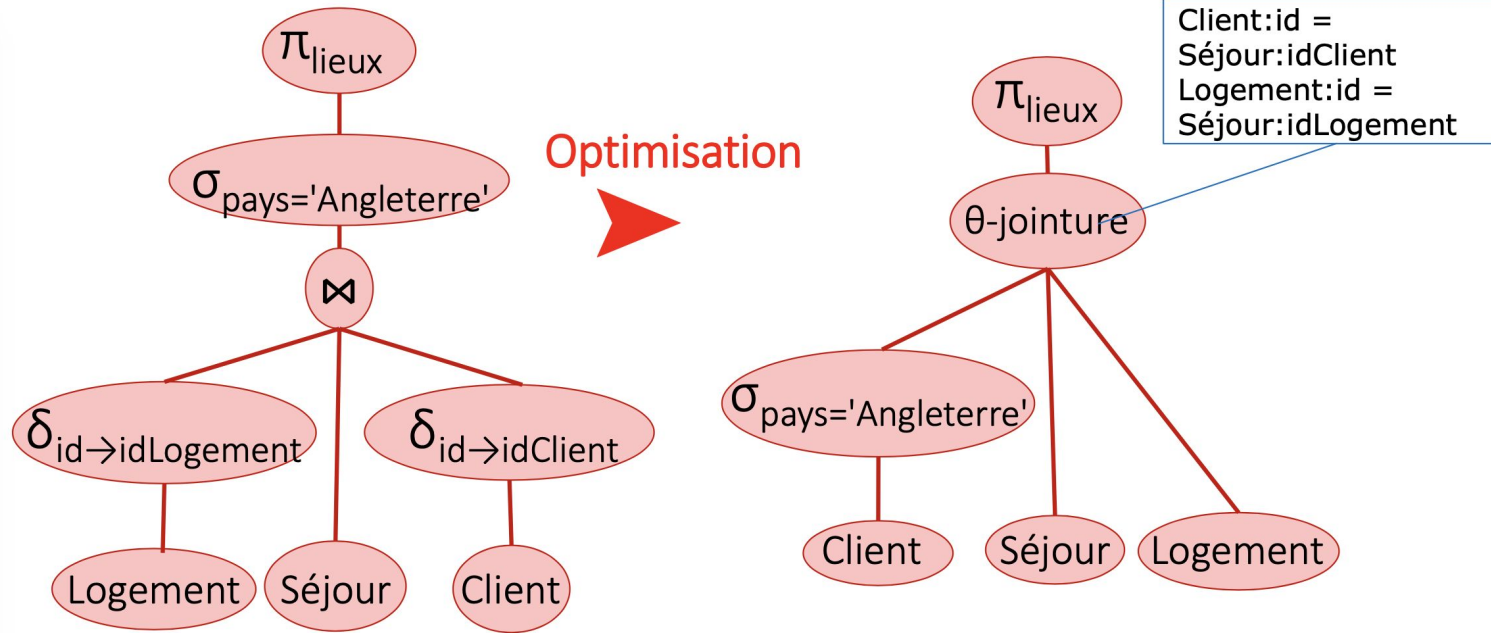


La jointure est associative

$$R1 \bowtie (R2 \bowtie R3) = (R1 \bowtie R2) \bowtie R3$$

Évaluation et optimisation

“Les lieux fréquentés par des Anglais”



Évaluation et optimisation

Utilisation de propriétés algébriques

$$\sigma_c(\sigma_{c'}(E)) = \sigma_{c'}(\sigma_c(E)) = \sigma_{c \text{ et } c'}(E)$$

composition

$$\pi_U(\pi_V(E)) = \pi_U(E)$$

observez U

$$\subseteq V$$

$$Q \bowtie (Q' \bowtie Q'') = (Q \bowtie Q') \bowtie Q''$$

associativité

$$Q \bowtie Q' = Q' \bowtie Q$$

commutativité

$$\sigma_c(Q \bowtie Q') = \sigma_c(Q) \bowtie Q'$$

« pousser σ »

...

Évaluation et optimisation

À retenir

On sait évaluer les requêtes algébriques

On sait le faire de façon efficace grâce à
l'optimisation

Bases de Données

Part 2 - L'algèbre relationnelle

À retenir

3 opérations unaires

- Sélection σ
- Projection π
- Renommage δ

3 opérations binaires

- Jointure \bowtie
- Union \cup
- Différence $-$

Et **la composition**