
L'algèbre relationnelle



-
1. Introduction
 2. Les opérateurs unaires
 1. La sélection σ
 2. La projection π
 3. Le renommage ρ
 3. Opérateurs binaires ensemblistes
 1. L'union \cup
 2. L'intersection \cap
 3. La différence $-$
 4. Le produit cartésien \times
 4. Les opérateurs binaires
 1. La jointure \bowtie
 2. La division \div
 5. Les fonctions d'agrégation

Introduction

Les types d'opérateurs relationnels

On distingue quatre types d'opérations relationnels :

- Les opérateurs unaires : ce sont les opérateurs les plus simples, ils permettent de produire une nouvelle table à partir d'une autre table.
 - La sélection σ , la projection π et le renommage ρ
- Les opérateurs binaires ensembliste : ces opérateurs permettent de produire une nouvelle relation à partir de deux relations de même degré et de même domaine.
 - L'union \cup , L'intersection \cap et la différence $-$
- Les opérateurs binaires ou n-aires : ils permettent de produire une nouvelle table à partir de deux ou plusieurs autres tables.
 - Le produit cartésien \times , la jointure \bowtie et la division \div
- Les opérations d'agrégation : ce sont des calculs mathématiques sur un ensemble de valeurs extraites d'une ou de plusieurs relations.
 - Moyenne, Somme, Maximum, Minimum et Comptage.

La sélection σ

Notation

$\sigma_F(R)$ avec R une relation (table) et F une condition de sélection.

Définition formelle

Soit R une relation de schéma S, $A \in S$ et $a \in \text{Domaine}(A)$, on appelle sélection de R selon une condition F ($A=a$) la table obtenue en sélectionnant dans R uniquement les lignes qui satisfont cette condition.

$$\sigma_F(R) = \{e \in R \mid e.A = a\}$$

- Sélection des lignes qui satisferont la condition .
- Coupe horizontale de la relation.
- le résultat de la sélection est aussi une relation

La sélection σ

Condition(s)

La condition de la sélection sur une relation R peut être une comparaison simple (CS) sous la forme :

- $A=a$
- $A < a$
- $A=B$

où A, B sont deux attributs de la table R et $a \in \text{domaine}(A)$

La condition de la sélection sur une relation R peut être une expression composée sous la forme

$$\text{CS1 op CS2 op op CSN}$$

où op est un opérateur qui désigne un connecteur logique : et, ou, non

• Exemples

- $A = a \text{ et } B < a$
- $A = B \text{ ou } A > a$

La sélection σ

Exemple

On considère la relation Voiture suivante :

Voiture			
Marque	Pays	Employés	Chiffre Affaires
OPEL	Allemagne	14345	4.290.000,00
PEUGEOT	France	20267	3.720.000,00
TOYOTA	Japon	21379	5.420.000,00
RENAULT	France	13780	4.001.290,00

On veut les marques dont le nombre des employés est inférieur à 15000

→ Algèbre relationnelle: Petites-marques = $\sigma_{(\text{Employés} \leq 15000)}(\text{Voiture})$

→ Langage SQL : `select * from Voiture where Employés <= 15000`



La sélection σ

Exemple 2

On considère la relation Livre suivante :

Livre			
Titre	Auteur	Année	Prix
La guerre	Philippe Gesard	2010	35.00
Python	Vincent le Goff	2017	26.00
Maths MP	Daniel Guinin	2008	50.00
Physique	Leonard Susskind	2015	27.50

On veut les livres dont l'année est inférieure à 2016 et le prix est supérieur à 30.00

→ Algèbre relationnelle : $R2 = \sigma_{(Année \leq 2016 \text{ et } Prix \geq 30.00)}(Livre)$

→ Langage SQL : `select * from Livre where Année <= 2016 and Prix >= 30.00`

Livre			
Titre	Auteur	Année	Prix
La guerre	Philippe Gesard	2010	35.00
Maths MP	Daniel Guinin	2008	50.00



La projection π

Notation

$\pi_{A1,A2,\dots,A_n}(R)$ avec R une relation, et $(A1, \dots, A_n)$ des attributs de R .

Définition formelle

Soit R une relation de schéma S , et A un ensemble d'attributs tels que $A \subseteq R$. On appelle projection de R selon A la relation

$$\pi_A(R) = \{e(A) | e \in R\}$$

- Supprime les attributs qui ne sont pas présent dans la liste de projection
 $A=(A1,\dots, A_n)$
- Coupe verticale de la relation
- Le schéma de $\pi_A(R)$ est donc A .

La projection π

Exemple

On considère la relation Voiture suivante :

Voiture			
Marque	Pays	Employés	Chiffre Affaires
OPEL	Allemagne	20345	4.290.000,00
PEUGEOT	France	10267	3.720.000,00
TOYOTA	Japon	21379	5.420.000,00
RENAULT	France	13780	4.001.290,00

On veut les noms des marques et leurs pays

→ Algèbre relationnelle : $\text{Marque_Pays} = \pi_{(\text{Marque}, \text{Pays})}(\text{Voiture})$

→ Langage SQL : `select Marque, Pays from Voiture`

Marque_Pays	
Marque	Pays
OPEL	Allemagne
PEUGEOT	France
TOYOTA	Japon
RENAULT	France

La projection π

Exemple

On considère la relation Etudiant suivante :

Etudiant			
IdEtudiant	Nom	Age	Note
1	Sadir	19	14.50
2	Khalil	21	8.00
3	Wahbi	20	11.00
4	Nassiri	22	13.00
5	Kamil	22	6.50

On veut les noms des étudiants, leurs âges et leurs notes

→ Algèbre relationnelle : $\text{Etudiant_bis} = \pi_{(\text{Nom}, \text{Age}, \text{Note})}(\text{Etudiant})$

→ Langage SQL : `select Nom, Age, Note from Etudiant`

Nom	Age	Note
Sadir	19	14.50



La projection π et la sélection

Exemple

L'ordre des deux opérateurs σ et π est important, cela signifie que :

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

La projection π et la sélection

Pays_Petites-marques = $\pi_{(pays)}(\sigma_{(Employés \leq 15000)}(Voiture)) \Rightarrow$

Pays
France
France

Pays-petites-marques = $(\sigma_{(Employés \leq 15000)})(\pi_{(pays)}(Voiture))$

Cette dernière requête, donne une erreur, car l'attribut Employé n'est plus dans la relation résultat $\pi_{(pays)}(Voiture)$.

Le renommage ρ

Notation

$\rho_{A \rightarrow B}(R)$ avec R une relation, A attribut de R et B n'est pas un attribut de R ($B \notin \text{schéma}(R)$).

Définition formelle

Soit R une relation de schéma S , tel que $S = (A_1, A_2, \dots, A_n)$ et $B \notin \text{schéma}(R)$. Le renommage d'un attribut A_i avec un nouveau attribut B est une relation R' avec, $\text{schéma}(R') = (\text{schéma}(R) - \{A_i\}) \cup \{B\}$ définie par :

$$\rho_{A_i \rightarrow B}(R) = \{t \mid \exists u \in R, t[\text{schéma}(R) - \{A_i\}] = u[A] \text{ et } t[B] = u[A]\}$$

- Pour lever une ambiguïté entre les noms des attributs de deux relations d'une opération binaire.
- Permet de forcer ou d'éviter des jointures naturelles.
- On peut renommer aussi une relation

Le renommage ρ

Exemple

On considère la relation Etudiant suivante :

Etudiant			
IdEtudiant	Nom	Age	Note
1	Sadir	19	14.50
2	Khalil	21	8.00
3	Wahbi	20	11.00
4	Nassiri	22	13.00
5	Kamil	22	6.50

On renomme IdEtudiant en IdStudent et nom en Name, on obtient :

→ Algèbre relationnelle : $\text{Student} = \rho_{(\text{IdEtudiant}, \text{Nom} \rightarrow \text{IdStudent}, \text{Name})}(\text{Etudiant})$

→ Langage SQL : `select IdEtudiant as IdStudent, Nom as Name, Age, Note from Etudiant ;`

Student			
IdStudent	Name	Age	Note
1	Sadir	19	14.50
2	Khalil	21	8.00
3	Wahbi	20	11.00
4	Nassiri	22	13.00
5	Kamil	22	6.50

L'union \cup

Notation

$R1 \cup R2$ avec $R1$ et $R2$ deux relations, **ayant le même schéma**.

Définition formelle

Soient $R1$ et $R2$ deux relations de même schéma. On appelle l'union de $R1$ et $R2$ l'ensemble des tuples appartenant à $R1$ ou à $R2$.

$$R1 \cup R2 = \{t \mid t \in R1 \text{ ou } t \in R2\}$$

- $\text{schéma}(R1) = \text{schéma}(R2)$
- $\text{Schéma}(R1 \cup R2) = \text{schéma}(R1) = \text{schéma}(R2)$
- Les nom des attributs doivent être identiques, pour appliquer l'opérateur UNION

Exemple

On considère les deux relations suivantes :

Biblio1		
Titre	Auteur	Editeur
<i>La guerre</i>	<i>Philippe Gesard</i>	<i>Ellipse</i>
<i>Python</i>	<i>Vincent le Goff</i>	<i>Dunod</i>
<i>Maths MP</i>	<i>Daniel Guinin</i>	<i>hachette</i>

Biblio2		
Titre	Auteur	Editeur
<i>La guerre</i>	<i>Philippe Gesard</i>	<i>Ellipse</i>
<i>Algorithmique</i>	<i>Thomas Cormen</i>	<i>Dunod</i>
<i>Physique MPSI</i>	<i>Jean Brebec</i>	<i>hachette</i>

On veut l'ensemble des livres de deux bibliothèques Biblio1 et Biblio2

→ Algèbre relationnelle : tous-les-livres = Biblio1 \cup Biblio2

→ Langage SQL : `select * from Biblio1 UNION select * from Biblio2 ;`

tous-les-livres		
Titre	Auteur	Editeur
<i>La guerre</i>	<i>Philippe Gesard</i>	<i>Ellipse</i>
<i>Python</i>	<i>Vincent le Goff</i>	<i>Dunod</i>
<i>Maths MP</i>	<i>Daniel Guinin</i>	<i>hachette</i>
<i>Algorithmique</i>	<i>Thomas Cormen</i>	<i>Dunod</i>
<i>Physique MPSI</i>	<i>Jean Brebec</i>	<i>hachette</i>

L'intersection \cap

Notation

$R1 \cap R2$ avec $R1$ et $R2$ deux relations, **ayant le même schéma.**

Définition formelle

Soient $R1$ et $R2$ deux relations de même schéma. On appelle l'intersection de $R1$ et $R2$ l'ensemble des tuples (t) appartenant à $R1$ et à $R2$.

$$R1 \cap R2 = \{t \mid t \in R1 \text{ et } t \in R2\}$$

- $\text{schéma}(R1) = \text{schéma}(R2)$
- $\text{Schéma}(R1 \cap R2) = \text{schéma}(R1) = \text{schéma}(R2)$

L'intersection \cap

Exemple

On considère les deux relations suivantes :

Biblio1		
Titre	Auteur	Editeur
<i>La guerre</i>	<i>Philippe Gesard</i>	<i>Ellipse</i>
<i>Python</i>	<i>Vincent le Goff</i>	<i>Dunod</i>
<i>Maths MP</i>	<i>Daniel Guinin</i>	<i>hachette</i>

Biblio2		
Titre	Auteur	Editeur
<i>La guerre</i>	<i>Philippe Gesard</i>	<i>Ellipse</i>
<i>Algorithmique</i>	<i>Thomas Cormen</i>	<i>Dunod</i>
<i>Physique MPSI</i>	<i>Jean Brebec</i>	<i>hachette</i>

On veut les livres communs entre deux bibliothèques Biblio1 et Biblio2

→ ALgèbre relationnelle : livres-communs = Biblio1 \cap Biblio2

→ Langage SQL : select * from Biblio1 Intersect select * from Biblio2 ;

livres_communs		
Titre	Auteur	Editeur
<i>La guerre</i>	<i>Philippe Gesard</i>	<i>Ellipse</i>

La différence –

Notation

$R1 - R2$ avec $R1$ et $R2$ deux relations, ayant le même schéma.

Définition formelle

Soient $R1$ et $R2$ deux relations de même schéma. On appelle la différence entre deux relations $R1$ et $R2$, l'ensemble des tuples appartenant à $R1$ et qui ne sont pas dans $R2$.

$$R1 - R2 = \{t | t \in R1 \text{ et } t \notin R2\}$$

- $\text{schéma}(R1) = \text{schéma}(R2)$
- $\text{Schéma}(R1 - R2) = \text{schéma}(R1) = \text{schéma}(R2)$

La différence –

Exemple

On considère les deux relations suivantes :

Biblio1		
Titre	Auteur	Editeur
<i>La guerre</i>	<i>Philippe Gesard</i>	<i>Ellipse</i>
<i>Python</i>	<i>Vincent le Goff</i>	<i>Dunod</i>
<i>Maths MP</i>	<i>Daniel Guinin</i>	<i>hachette</i>

Biblio2		
Titre	Auteur	Editeur
<i>La guerre</i>	<i>Philippe Gesard</i>	<i>Ellipse</i>
<i>Algorithmique</i>	<i>Thomas Cormen</i>	<i>Dunod</i>
<i>Physique MPSI</i>	<i>Jean Brebec</i>	<i>hachette</i>

On veut les livres de Biblio1 qui ne sont pas dans Biblio2

→ ALgèbre relationnelle : différence= Biblio1 – Biblio2

→ Langage SQL : select * from Biblio1 Except select * from Biblio2 ;

Différence		
Titre	Auteur	Editeur
<i>Python</i>	<i>Vincent le Goff</i>	<i>Dunod</i>
<i>Maths MP</i>	<i>Daniel Guinin</i>	<i>hachette</i>



Le produit cartésien ×

Notation

$R1 \times R2$ avec $R1$ et $R2$ deux relations.

Définition formelle

Soient $R1$ et $R2$ deux relations, on appelle le produit cartésien de $R1$ et $R2$, toutes les combinaisons de lignes possibles à partir de ces deux relations.

$$R1 \times R2 = \{(v_1, \dots, v_n, v'_1, \dots, v'_m) \mid (v_1, \dots, v_n) \in R1 \text{ et } (v'_1, \dots, v'_m) \in R2\}$$

- $\text{schéma}(R1 \times R2) = \text{schéma}(R1) \cup \text{schéma}(R2)$
- Comme l'union ou l'intersection, la notion de produit cartésien est issue des opérateurs ensemblistes
- Le produit cartésien peut être utilisé comme opérateur n-aires
- Si les deux relations ont un attribut de même nom, on renomme cet attribut.

Le produit cartésien ×

Exemple

On considère les deux relations suivantes : la relation élève et la relation enseignant :

élève		
idEleve	nomEleve	age
1	Tahiri	21
2	Wahbi	18
3	Jebbar	20

enseignant	
idEnseignant	nomEnseignant
100	Wardi
120	Saadan

On veut le produit cartésien des deux relations élève×enseignant

→ Algèbre relationnelle : élève × enseignant

→ Langage SQL : select * from élève,enseignant ;

élève×enseignant				
idEleve	nomEleve	age	idEnseignant	nomEnseignant
1	Tahiri	21	100	Wardi
1	Tahiri	21	120	Saadan
2	Wahbi	18	100	Wardi

La jointure \bowtie

Notation

$R1 \bowtie R2$ avec $R1$ et $R2$ deux relations de schéma $S1$ et $S2$.

Définition formelle

Soient $R1$ et $R2$ deux relations ayant deux schémas $S1$ et $S2$ respectivement, et $A \in S1$, $B \in S2$ tels que $\text{dom}(A)=\text{dom}(B)$.

La jointure de $R1$ et $R2$ est une relation sur un schéma S avec $S=S1 \cup S2$ définie par :

$$R1 \bowtie R2 = \{e \in R1 \times R2 \mid e.A = e.B\} = \sigma_{A=B} R1 \times R2$$

- $\text{schéma}(R1 \bowtie R2) = \text{schéma}(R1) \cup \text{schéma}(R2)$
- La jointure naturelle comporte toutes les combinaisons de lignes de $R1$ et de $R2$ qui ont la même valeur pour les attributs de mêmes noms.
- On doit avoir au moins un attribut commun entre les deux relations.
- Les attributs de mêmes noms n'apparaissent qu'une seule fois dans la requête



La jointure ⋈

Exemple

On considère les deux relations suivantes : la relation livre et la relation auteur :

livre	
Titre	Nom-auteur
<i>Madame Bovary</i>	<i>Flaubert</i>
<i>Le pere Goriot</i>	<i>Balzac (de)</i>

auteur	
Nom	Prénom
<i>Flaubert</i>	<i>Gustave</i>
<i>Balzac (de)</i>	<i>Honore</i>
<i>Proust</i>	<i>Marcel</i>

On veut tous les auteurs qui ont écrits des livres.

→ Algèbre relationnelle : livre ⋈ auteur

→ Langage SQL : `Select * from livre join auteur on Nom-auteur=Nom ;`

livre ⋈ auteur			
Titre	Nom-auteur	Nom	Prénom
<i>Madame Bovary</i>	<i>Flaubert</i>	<i>Flaubert</i>	<i>Gustave</i>
<i>Le pere Goriot</i>	<i>Balzac (de)</i>	<i>Balzac (de)</i>	<i>Honore</i>

La division cartésienne \div

Notation

$R1 \div R2$ avec $R1$ et $R2$ deux relations de schéma $S1$ et $S2$.

Définition formelle

Soient $R1$ et $R2$ deux relations de schéma $S1$ et $S2$ respectivement, tels que $S1 = \{A_1, A_2, \dots, A_n\}$, $S2 = \{A_1, A_2, \dots, A_m\}$ avec $m < n$.

La division cartésienne de $R1$ et $R2$ est une relation sur un schéma $S3$ avec $S3 = \{A_{m+1}, A_{m+2}, \dots, A_n\}$ définie par :

$$R1 \div R2 = \{(a_{m+1}, a_{m+2}, \dots, a_n) \mid \forall (a_1, \dots, a_m) \in R2, \exists (a_1, \dots, a_m, a_{m+1}, \dots, a_n) \in R1\}$$

- La division cartésienne entre deux relations \Leftrightarrow la division euclidienne (entiers).
- La division permet de traiter les requêtes de style "donner tous les x tels que pour tout $y \dots$ "

La division ÷

Exemple

On considère les deux relations suivantes : la relation étudiant et la relation département :

etudiant	
etud	dep
1	1
1	2
1	3
1	4
2	1
2	2
3	2
4	2
4	4

departement
dep
1
2
3
4

Quel sont les étudiants qui sont inscrits dans tous les départements ?

etudiant ÷ departement
etud
1

Les fonctions d'agrégation

Objectif

Les fonctions d'agrégation permettent de répondre aux requêtes du type :

- "Quel est le nombre d'étudiants inscrits dans chaque département" ?
- "Quel est l'âge moyen des étudiants" ?

Définition

Soit R une relation, A , B deux attributs de R et f une fonction d'agrégation sur R .
On note $\pi_{A, f(B)}(R)$ la relation obtenue :

- en regroupant les valeurs de R qui sont identiques sur l'attribut A
- en définissant un nouveau attribut $f(B)$ pour ces valeurs regroupées, par application de la fonction d'agrégation f sur chacun de ces agrégats sur l'attribut B .

- Une fonction d'agrégation f_A en fonction d'attribut A sur une relation R est une fonction calculable qui pour un ensemble fini de tuples sur R renvoie une valeur entière ou réelle.
- L'agrégation va servir à regrouper les lignes ayant la même valeur pour l'attribut A .

Les fonctions d'agrégation

Quelques fonctions d'agrégation

Les fonctions d'agrégat vont permettre de calculer une valeur simple à partir d'un ensemble de valeurs provenant d'un même attribut mais de plusieurs tuples d'une relation.

Les fonctions d'agrégat les courantes, sont les suivantes :

- COMPTE : compter les valeurs d'un attribut d'une relation,
→ En SQL COMPTE :COUNT
- SOMME : additionner les valeurs d'un attribut d'une relation,
→ En SQL SOMME :SUM
- MOYENNE : effectuer la moyenne des valeurs d'un attribut d'une relation,
→ En SQL MOYENNE :AVG
- MAXIMUM : chercher la valeur maximale d'un attribut d'une relation,
→ En SQL MAXIMUM :MAX
- MINIMUM : chercher la valeur minimale d'un attribut d'une relation.
→ En SQL MINIMUM :MIN

Les fonctions d'agrégation

Exemple

Considérons la relation relevé suivante :

relevé		
classe	élève	note
MP1	Wahbi	17, 5
MP3	Sadir	7, 75
MP1	Tijani	9, 25
MP3	Abro	14, 0
MP3	Khilali	11, 5

On veut calculer la moyenne sur chaque classe :

→ Algèbre relationnelle :

classe $\gamma_{\text{moyenne}(\text{note})}$ (relevé)

→ Langage SQL :

SELECT classe, AVG(note) FROM relevé
GROUP BY classe ;

Le résultat est :

$\text{classe} \gamma_{\text{moyenne}(\text{note})}(\text{relevé})$	
classe	moyenne(note)
MP1	13.38
MP3	11.08

On veut calculer la note maximale de tous les élèves

→ ALgèbre relationnelle :

.....

→ Langage SQL :

.....