Introduction aux Bases de Données

L3

Céline Rouveirol

2010-2011

Plan

Algèbre relationnelle : introduction

- proposée par E. Codd en 1969
- collection d'opérations formelles dont les opérandes et les résultats sont des relations
- utilisée en général à l'intérieur de tout SGBD relationnel. Expressions algébriques procédurales donc faciles à optimiser par des transformations syntaxiques
- fondée sur 8 opérateurs : 4 ensemblistes (UNION, INTERSECTION, DIFFERENCE, PRODUIT) et 4 spécifiques (SELECTION, PROJECTION, JOINTURE, DIVISION)
- les opérateurs de base (*primitifs*) permettent de déduire d'autres opérations, dites opérations dérivées

Projection

Définition

La projection est utilisée pour extraire d'une instance de relation r de schéma R un sous-ensemble d'attributs. Soit X un ensemble d'attributs tels que $X \subseteq att(r) = R$, l'instance $\prod_X(r) = \{t[X] \mid t \in r\}$. Si $X = \emptyset$, alors $\prod_X(r) = \emptyset$.

Propriétés :

- $0 \leq card(\prod_X(r) \leq card(r)$
- $arite(\prod_X(R)) \leq arite(R)$

Projection, exemple

Vehicule	NV	Marque	Couleur	Puissance
	1234 GH 75	Renault	verte	7
	678 AZ 23	Citroen	blanche	6
	345 AZ 34	Renault	rouge	5

$\prod_{NV,Couleur}(Vehicule)$	NV	Couleur
	1234 GH 75	verte
	678 AZ 23	blanche
	345 AZ 34	rouge

$\prod_{Marque}(Vehicule)$	Marque
	Renault
	Citroen

Sélection

Définition

Etant donné une instance de relation r de schéma R, et ϕ une condition de sélection, i.e. une expression booléenne portant sur un ou plusieurs attributs de R, la sélection renvoie une l'instance notée $\sigma_\phi(r)$ telle que $\sigma_\phi(r) = \{t \mid t \in r \text{ et } \phi(t) = T\}$

- condition de sélection de la forme att_i comp att_j ou att_i comp *val*
- comp : $>, <, =, \neq, \geq, \leq$
- combinaison de condition par *OU*, *ET* et *NON*

Propriétés :

- $arite(\sigma_\phi(R)) = arite(R)$
- $0 \leq card(\sigma_\phi(r)) \leq card(r)$

Sélection, exemple

Vehicule	NV	Marque	Couleur	Puissance
	1234 GH 75	Renault	verte	7
	678 AZ 23	Citroen	blanche	6
	345 AZ 34	Renault	rouge	5

$\sigma_{Puissance>6}(Vehicule)$	NV	Marque	Couleur	Puissance
	1234 GH 75	Renault	verte	7

$\sigma_{Marque=Citroen}(Vehicule)$	NV	Marque	Couleur	Puissance
	678 AZ 23	Citroen	blanche	6

$\sigma_{Marque=Citroen\ ET\ Puissance>6}(Vehicule)$?

Renommage

Définition

Soit une instance de relation r, un ensemble d'attributs $X \subseteq att(r)$ et un ensemble d'attributs Z de même cardinalité que X ; le renommage du schéma de R étant donné Z est une bijection ρ de X dans Z notée $\rho_{X \to Z}(R)$, tel que X est remplacé par Z dans le schéma de R.

NB : Cette opération est purement syntaxique et n'a aucune incidence sur les instances de la relation ; elle sert à changer le nom (renommer) des attributs d'une relation.

Renommage, exemple

Vehicule	NV	Marque	Couleur	Puissance
	1234 GH 75	Renault	verte	7
	678 AZ 23	Citroen	blanche	6
	345 AZ 34	Renault	rouge	5

$\rho_{Marque \rightarrow Constructeur}(Vehicule)$	NV	Constructeur	Couleur	Puissance
	1234 GH 75	Renault	verte	7
	678 AZ 23	Citroen	blanche	6
	345 AZ 34	Renault	rouge	5

$\prod_{Constructeur}(\rho_{Marque \rightarrow Constructeur}(Vehicule))$	Constructeur
	Citroen
	Renault

Jointure

Définition

Soient r et s deux instances de relation ; la jointure de r et s renvoie une relation notée : $r \bowtie s = \{t \mid t[att(r)] \in r \text{ et } t[att(s)] \in s\}$ sur le schéma $att(r) \cup att(s)$

Les tuples de la relation résultat sont obtenus en concaténant chaque tuple de r avec ceux de s ayant mêmes valeurs pour les attributs de même nom.

Propriétés :

- opération commutative ($r \bowtie s = s \bowtie r$), associative ($r \bowtie (s \bowtie t)) = (r \bowtie (t \bowtie s))$
- $r(A, B) \bowtie s(B, C) = \prod_{A,B,C}(\sigma_{B=B'}((r \times \rho_{B \rightarrow B'}(s))))$

Jointure

Propriétés (suite) :

- si $att(r) \cap att(s) = \emptyset$ alors $r \bowtie s$ est un produit cartésien
- si $att(r) \subseteq att(s)$ alors $r \bowtie s$ est une sélection de “s par r”
- si $att(r) = att(s)$ alors $r \bowtie s$ est l’intersection de r et s

Jointure, exemple

r	A	B	C
	1	3	5
	2	4	6
	3	5	7
	4	6	8

s	A	D	E
	2	1	1
	1	2	2
	1	3	3
	3	3	4
	4	4	3

$r \bowtie s$	A	B	C	D	E
	1	3	5	2	2
	1	3	5	3	3
	2	4	6	1	1
	3	5	7	3	4
	4	6	8	4	3

Union - Différence

Définition :Union

Prend en argument 2 instances de relations r et s de même schéma, et renvoie une instance de même schéma définie par :
$r \cup s = \{t | t \in r \text{ ou } t \in s\}$

Définition :Différence

Prend en argument 2 instances de relations r et s de même schéma, et renvoie une instance de même schéma définie par :
$r \setminus s = \{t | t \in r \text{ et } t \notin s\}$

- opération non commutative

Produit Cartésien

Définition

Prend en argument 2 instances de relations r et s telles que $att(r) \cap att(s) = \emptyset$, et renvoie une instance définie sur $att(r) \cup att(s)$ par : $r \times s = \{t | t[att(r)] \in r \text{ et } t[att(s)] \in s\}$

Propriétés :

- opération commutative
- si $arite(R) = k_1$ et $arite(S) = k_2$ alors $arite(R \times S) = k_1 + k_2$
- $card(r \times s) = card(r) \times card(s)$

Opérations ensemblistes, exemples

Soient les deux instances de relation
$Etudiant\_MI1(IdEtudiant, Nom, Prenom, UE)$ et
$Etudiant\_MMI1(IdEtudiant, Nom, Prenom, UE)$

- on veut savoir les étudiants inscrits à l'UE BD :
  $\sigma_{UE='BD'}(Etudiant\_MI1) \cup \sigma_{UE='BD'}(Etudiant\_MMI1)$
- les UE propres à MI1 :

$$\prod_{UE}(Etudiants\_MI1) \setminus \prod_{UE}(Etudiants\_MMI1)$$

- le produite cartésien de ces 2 instances de relations est-il possible ?
- quelle est la jointure (naturelle) de ces 2 instances ?

Intersection

Définition

Prend en argument 2 instances de relations r et s de même schéma, et renvoie une instance de même schéma définie par :
$r \cap s = \{t | t \in r \text{ et } t \in s\}$

Cette opération peut s'exprimer à partir d'opérations de base :

- $r \cap s = r \setminus (r \setminus s) = s \setminus (s \setminus r)$
- $r \cap s = (r \cup s) \setminus [(r \setminus s) \cup (s \setminus r)]$

Jointure conditionnelle

Définition

Prend en argument 2 instances de relations r et s et une condition de sélection θ et renvoie une instance de relation définie sur $att(r) \cup att(s)$ par : $r \bowtie_\theta s = \sigma_\theta(r \times s)$. On a $0 \leq card(r \bowtie_\Theta s) \leq card(r) \times card(s)$

r	A	B	C
	1	3	5
	2	4	6
	3	5	7
	4	6	8

s	A	D	E
	2	1	1
	1	2	2
	1	3	3
	3	3	4
	4	4	3

$r \bowtie_{r.A<s.A} s$	r.A	B	C	s.A	D	E
	1	3	5	2	1	1
	1	3	5	3	3	4
	1	3	5	4	4	3
	2	4	6	3	3	4
	2	4	6	4	4	3
	3	5	7	4	4	3

Division

Définition

Prend en argument 2 instances de relations r et s telles que $att(s) \subseteq att(r)$ et renvoie une instance de relation définie sur $att(r) \setminus att(s)$ par : $r \div s = \{t | t \in \prod_X(r)$ et $\{t\} \times s \subseteq r\}$ avec $X = att(r) \setminus att(s)$.

- $r \div s = \prod_X(r) \setminus \prod_X(((\prod_X(r) \times s) \setminus r))$
- si $att(r) = \emptyset$ ou $att(s) = att(r)$, alors $r \div s = \emptyset$
- Cette opération permet de rechercher dans une relation les sous-tuples qui sont complétés par tous ceux d'une autre relation afin de répondre aux questions de la forme “pour tout x, trouver y”

Division, exemple

Soient l'instance $FILM(Titre, M-e-S, Acteur)$, et soit l'instance $CINE(NomCine, TitreFilm, Horaire, Salle)$. Quel cinéma passe tous les films de m_1 ?

FILM	Titre	M-e-S	Acteur
	t_1	m_1	a_1
	t_1	m_1	a_3
	t_2	m_1	a_2
	t_2	m_1	a_3
	t_3	m_2	a_1
	t_3	m_2	a_3

CINE	NomCine	Titre	Horaire	Salle
	$Cine_1$	t_1	14h	001
	$Cine_2$	t_1	14h	001
	$Cine_2$	t_2	16h	001
	$Cine_2$	t_3	14h	002
	$Cine_3$	t_1	14h	001
	$Cine_3$	t_3	14h	002

$$\prod_{NomCine, Titre}(CINE) \div \prod_{Titre}(\sigma_{M-e-S=m_1}(FILM))$$

Exercice

Soient les 3 instances :

- fréquente(PERSONNE,CAFE)
- sert(CAFE,JUSDEFRUIT)
- aime(PERSONNE,JUSDEFRUIT)

fréquente	P	C
	p1	c1
	p1	c2
	p1	c3
	p2	c2
	p2	c3
	p3	c4

sert	C	J
	c1	b1
	c1	b2
	c2	b3
	c3	b3
	c3	b2
	c4	b1

aime	P	J
	p1	b1
	p1	b3
	p2	b2
	p3	b2

Exercice (suite)

Ecrire des expressions algébriques pour les requêtes suivantes :

1. `doit_visiter(P,C)` qui liste tous les nuplets $< P, C >$) tel que le café C sert un jus de fruit que la personne P aime
2. `triste(P)` qui liste tous les nuplets $< P >$ telle que la personne P ne fréquente aucun café qui serve un jus de fruit qu'elle aime.
3. `très_heureux(P)` qui liste tous les nuplets $< P >$ si tous les cafés que P fréquente servent au moins un jus de fruit P aime.

On pourra supposer que toute personne fréquente au moins un café.