Langages de Requête: Calcul Positif

Jef Wijsen

November 15, 2006

1 Introduction

1.1 Le Schéma

On prend le schéma “aéroport” où on a enlevé les propriétaires.

```
PILOTE(Pnom, Adresse, Tel, Brevet)
   PK(Pnom)

TYPE(Tnom, Constructeur, Puissance, #Places)
   PK(Tnom)

MECANICIEN(Mnom, Adresse, Tel)
   PK(Mnom)

AVION(Immatriculation, AvionType, DateAchat)
   PK(Immatriculation)
   FK(AvionType) REFS TYPE

INTERVENTION(Numero, Objet, Date, Duree, Avion, Reparateur, Verificateur)
   PK(Numero)
   FK(Avion) REFS AVION
   FK(Reparateur) REFS MECANICIEN
   FK(Verificateur) REFS MECANICIEN

PHAB(Pilote, AvionType, #Vols)
   PK(Pilote, AvionType)
   FK(Pilote) REFS PILOTE
   FK(AvionType) REFS TYPE

MHAB(Mecanicien, AvionType)
   PK(Mecanicien, AvionType)
   FK(Mecanicien) REFS MECANICIEN
   FK(AvionType) REFS TYPE
```

1.2 Quelques Requêtes

- Qui est habilité à réparer un avion de type “B747”?

$$\{\langle x\rangle \mid \texttt{MHAB}(x, \text{“B747”})\}$$

- Qui est habilité à piloter un avion de type “B747”?

$$\{\langle x\rangle \mid \exists z(\texttt{PHAB}(x, \text{“B747”}, z))\}$$

- Avion “ABC123” est en panne. Donner les noms des mécaniciens habilités à intervenir.

$$\{\langle x\rangle \mid \exists y(\texttt{MHAB}(x, y) \wedge \exists z(\texttt{AVION}(\text{“ABC123”}, y, z)))\}$$

L'idée devrait être claire: donner le nom x de tout mécanicien qui peut intervenir sur le type y de l'avion “ABC123”. Noter que z n'est qu'un *bouche-trou* pour la colonne `DateAchat`.

MHAB

Mecanicien	AvionType
x	y

AVION

Immatriculation	AvionType	DateAchat
“ABC123”	y	z

- Donner les pairs de mécaniciens qui peuvent collaborer pour réaliser une intervention. Noter qu'une collaboration est exclue si les mécaniciens n'ont aucune habilitation en commun.

$$\{\langle x, y\rangle \mid \exists z(\texttt{MHAB}(x, z) \wedge \texttt{MHAB}(y, z))\}$$

Pour la table :

MHAB

Mecanicien	AvionType
Ed	B747
Ed	A380
Tim	B747
Eric	A380

, on obtient la réponse :

1	2
Ed	Tim
Ed	Eric
Tim	Ed
Eric	Ed
Ed	Ed
Tim	Tim
Eric	Eric

.

Noter que les deux premières rangées de la réponse auraient suffi. Il est toutefois clair que Tim et Eric ne pourront jamais intervenir ensemble.

- Faire une liste contenant les noms, adresses et numéros d'appel de tous les mécaniciens et pilotes.

$$\{\langle x, y, z\rangle \mid \texttt{MECANICIEN}(x, y, z) \vee \exists w(\texttt{PILOTE}(x, y, z, w))\}$$

Même liste, mais sans adresses.

$$\{\langle x, z\rangle \mid \exists y(\texttt{MECANICIEN}(x, y, z)) \vee \exists y'\exists w(\texttt{PILOTE}(x, y', z, w))\}$$

On peut résoudre cette dernière question aussi comme suit:

$$\{\langle x, z\rangle \mid \exists y(\texttt{MECANICIEN}(x, y, z) \vee \exists w(\texttt{PILOTE}(x, y, z, w)))\}$$

2 Calcul Positif du Premier Ordre

2.1 Syntaxe

Un *terme* est soit une variable, soit une constante. Les variables utilisées sont u, v, w, x, y, z. Les constantes apparaîtront entre guillemets, par exemple, “a”, “b”, “c” et “B747”.

- Si R est le nom d'une relation avec n colonnes, et $t_1, t_2, \ldots, t_n$ sont des termes, alors $R(t_1, t_2, \ldots, t_n)$ est une formule. Les variables dans cette formule sont toutes *libres*.
- Si φ_1 et φ_2 sont deux formules, alors $\varphi_1 \wedge \varphi_2$ est une formule. Les variables qui étaient libres dans φ_1 ou dans φ_2, restent libres dans $\varphi_1 \wedge \varphi_2$.
- Si φ_1 et φ_2 sont deux formules avec les mêmes variables libres, alors $\varphi_1 \vee \varphi_2$ est une formule. Les variables libres de $\varphi_1 \vee \varphi_2$ sont les variables libres de φ_1 (ou de φ_2, les deux formules ayant les mêmes variables libres).
- Si φ est une formule contenant une variable libre x, alors $\exists x(\varphi)$ est une formule. La variable x, qui était libre dans φ, n'est plus libre dans la formule $\exists x(\varphi)$.

La vérité (le fait d'être vrai (Vrai) ou faux (Faux)) d'une formule par rapport à une base de données dépend des valeurs que l'on donne aux variables libres. Supposons la table `MHAB` comme suit:

`MHAB`	`Mecanicien`	`AvionType`
	Ed	B747
	Ed	A380
	Tim	B747
	Eric	A380

Dans la formule φ suivante, x et y sont libres, mais z ne l'est pas. Suivant les valeurs que l'on donne aux variables x et y, la formule est vraie ou fausse.

$$\varphi(x, y) \quad = \quad \exists z(\texttt{MHAB}(x, z) \wedge \texttt{MHAB}(y, z))$$

Par exemple,[1]

$$\begin{aligned}
\varphi(\text{“Ed”}, \text{“Tim”}) \quad &= \quad \exists z(\texttt{MHAB}(\text{“Ed”}, z) \wedge \texttt{MHAB}(\text{“Tim”}, z)) \;\text{ égale Vrai;} \\
\varphi(\text{“Tim”}, \text{“Eric”}) \quad &= \quad \exists z(\texttt{MHAB}(\text{“Tim”}, z) \wedge \texttt{MHAB}(\text{“Eric”}, z)) \;\text{ égale Faux.}
\end{aligned}$$

2.2 Sémantique

La vérité d'une formule sans variables libres est définie comme suite (toujours par rapport à une base de données):

- $R(\text{“}a_1\text{”}, \text{“}a_2\text{”}, \ldots, \text{“}a_n\text{”})$ égale Vrai si $\langle a_1, a_2, \ldots, a_n\rangle$ est un tuple de la relation R dans la base de données.
- $\varphi_1 \wedge \varphi_2$ égale Vrai si à la fois φ_1 égale Vrai et φ_2 égale Vrai.
- $\varphi_1 \vee \varphi_2$ égale Vrai dès que φ_1 est Vrai ou φ_2 est Vrai.
- $\exists x(\varphi(x))$ est Vrai s'il existe une constante a telle que $\varphi(a)$ est Vrai.

[1] Vous connaissez les notations $\varphi(x, y)$ et $\varphi(\text{“Ed, “Tim”})$ de vos cours de mathématiques. Par exemple, $f(x, y) = x^2 + y^2$ indique que la valeur de f dépend des valeurs que l'on donne à x et y; on écrira $f(3, 4) = 3^2 + 4^2$.

2.3 Langage de Requête

Une *requête* est une expression de la forme

$$\{\langle x_1, x_2, \ldots, x_k \rangle \mid \varphi(x_1, x_2, \ldots, x_k)\} \ ,$$

où φ est une formule avec les variables libres $x_1, x_2, \ldots, x_k$.

La *réponse* à cette requête par rapport à une base de données est la relation contenant tout tuple $\langle a_1, a_2, \ldots, a_n \rangle$ tel que $\varphi(\text{“}a_1\text{”}, \text{“}a_2\text{”}, \ldots, \text{“}a_n\text{”})$ égale Vrai.

2.4 Note Concernant la Disjonction ($\vee$)

Dans la section 2.1, on a exigé que deux formules reliées par un “ou” logique ($\vee$) aient les mêmes variables libres. La requête suivante est donc erronée:

Requête erronée: $\{\langle x \rangle \mid \exists y \exists z (\texttt{MECANICIEN}(x, y, z)) \vee \texttt{MHAB}(\text{“Ed”}, \text{“B747”})\}$

Il y a une bonne raison pour exclure cette requête: si $\langle \text{Ed}, \text{B747} \rangle$ est un tuple dans la table `MHAB`, alors la formule $\exists y \exists z (\texttt{MECANICIEN}(x, y, z)) \vee \texttt{MHAB}(\text{“Ed”}, \text{“B747”})$ est Vrai indépendamment des valeurs que l'on choisit pour la variable libre x. Par exemple, si $\langle \text{Ed}, \text{B747} \rangle$ est un tuple dans la table `MHAB`, alors

$$\exists y \exists z (\texttt{MECANICIEN}(\text{“Superman”}, y, z)) \vee \texttt{MHAB}(\text{“Ed”}, \text{“B747”})$$

est Vrai, et Superman serait une réponse…

3 Exercices

Formuler les requêtes suivantes.

1. L'accident de l'avion ABC123 pourrait être causé par une réparation effectuée à la date du 13 novembre 2006. Donner le nom et l'adresse du mécanicien qui devait vérifier cette réparation.

2. Donner les interventions (s'il y en a) où la réparation et la vérification ont été effectuées par la même personne.

3. Donner le nom de chaque personne qui peut piloter à la fois un avion construit par Boeing et un d'Airbus.

4. Donner les types d'avion sur lesquels “Jean Dupont” est intervenu en tant que réparateur ou vérificateur.