

Chapitre 8

Comparaison des requêtes algébriques, en calcul relationnel et requêtes SQL

I. Introduction

Nous avons vu dans les chapitres précédents qu'il existait un certain nombre de langage pour traduire une requête portant sur une base de données préalablement créée. Ces langages étaient les suivants :

- Le langage algébrique en utilisant l'algèbre relationnelle.
- Le calcul relationnel à variable domaine ou n-uplets.
- Le langage SQL dont l'interface est la plus répandue pour gérer une base de données.

Nous allons voir dans les paragraphes suivants quelles équivalences il existe entre ces différents langages afin de pouvoir passer de l'un à l'autre, sans se limiter à un langage précis.

II. Comparaisons des différents opérateurs

Pour les différents opérateurs, on considère deux schémas de relations $R(r_1, \dots, r_n)$ et $S(s_1, \dots, s_m)$, et deux relations r et s définies respectivement sur R et S . On note X et Y deux schémas de relations.

1. **Union**

Hypothèse : $R = S$.

Algèbre relationnelle :

$$r \cup s.$$

Calcul relationnel à variable domaine :

$$\{ (a_1, \dots, a_n) \mid \exists (r_1, \dots, r_n, s_1, \dots, s_n) R(r_1, \dots, r_n) \wedge S(s_1, \dots, s_n) \wedge [[(a_1 = r_1) \wedge \dots \wedge (a_n = r_n)] \vee [(a_1 = s_1) \wedge \dots \wedge (a_n = s_n)]] \}$$

Calcul relationnel à variable n-uplets :

$$\{ a \mid \exists r, s R(r) \wedge S(s) \wedge [(a = r) \vee (a = s)] \}$$

Langage SQL :

SELECT * FROM R UNION SELECT * FROM S;

2. **Intersection**

Hypothèse : $R = S$.

Algèbre relationnelle :

$$r \cap s.$$

Calcul relationnel à variable domaine :

$$\{ (a_1, \dots, a_n) \mid R(a_1, \dots, a_n) \wedge S(a_1, \dots, a_n) \}$$

Calcul relationnel à variable n-uplets :

$$\{ t \mid R(t) \wedge S(t) \}$$

Langage SQL :

SELECT * FROM R INTERSECT SELECT * FROM S;

3. Différence

Hypothèse : $R = S$.

Algèbre relationnelle :

$$r - s = r \setminus s.$$

Calcul relationnel à variable domaine :

$$\{ (a_1, \dots, a_n) \mid \exists (s_1, \dots, s_n) \ R(a_1, \dots, a_n) \wedge S(s_1, \dots, s_n) \wedge [(a_1 \neq s_1) \vee \dots \vee (a_n \neq s_n)] \}$$

Calcul relationnel à variable n-uplets :

$$\{ a \mid \exists s \ R(a) \wedge S(s) \wedge (a \neq s) \}$$

Langage SQL :

SELECT * FROM R MINUS SELECT * FROM S;

4. Produit cartésien

Hypothèse : $R \cap S = \emptyset$.

Algèbre relationnelle :

$$r \times s.$$

Calcul relationnel à variable domaine :

$$\{ (r_1, \dots, r_n, s_1, \dots, s_n) \mid R(r_1, \dots, r_n) \wedge S(s_1, \dots, s_n) \}$$

Calcul relationnel à variable n-uplets :

$$\{ r, s \mid R(r) \wedge S(s) \}$$

Langage SQL :

SELECT R.r₁, ..., R.r_n, S.s₁, ..., S.s_m FROM R, S;

5. Sélection

Hypothèse : F est une condition définie sur R .

Algèbre relationnelle :

$$\sigma_F(r).$$

Calcul relationnel à variable domaine :

$$\{ (r_1, \dots, r_n) \mid R(r_1, \dots, r_n) \wedge F(r_1, \dots, r_n) \}$$

Calcul relationnel à variable n-uplets :

$$\{ r \mid R(r) \wedge F(r) \}$$

Langage SQL :

SELECT * FROM R WHERE F;

6. Projection

Hypothèse : $X \subseteq R$.

Algèbre relationnelle :

$$\Pi_X(r).$$

Calcul relationnel à variable domaine :

$$\{ (r_1, \dots, r_p) \mid R(r_1, \dots, r_p) \}$$

Calcul relationnel à variable n-uplets :

$$\{ r' \mid R(r', r'') \wedge F(r) \}$$

Langage SQL :

SELECT R.x₁, ..., R.x_p FROM R;

7. Jointure

Hypothèse : $R \cap S \neq \emptyset$.

Algèbre relationnelle :

$$r \bowtie s.$$

Calcul relationnel à variable domaine :

$$\{ r_1, \dots, r_n, s_1, \dots, s_m \mid \exists (s_1, \dots, s_{p-1}) R(r_1, \dots, r_n) \wedge S(s_1, \dots, s_m) \wedge (r_1 = s_1) \wedge \dots \wedge (r_{p-1} = s_{p-1}) \}$$

Calcul relationnel à variable n-uplets :

$$\{ r, p, s \mid R(r, p) \wedge S(p, s) \}$$

Langage SQL :

SELECT R.r₁, ..., R.r_n, S.s₁, ..., S.s_m **FROM** R, S **WHERE** R.r₁ = S.s₁ **AND** ... **AND** R.r_{p-1} = S.s_{p-1};

8. Division

Hypothèse : $Y = R - S$.

Algèbre relationnelle :

$$r / s = r \div s.$$

Calcul relationnel à variable domaine :

$$\{ r_1, \dots, r_{p-1} \mid \exists (r_p, \dots, r_n) R(r_1, \dots, r_n) \wedge S(r_p, \dots, r_n) \}$$

Calcul relationnel à variable n-uplets :

$$\{ r \mid R(r, p) \wedge S(p) \}$$

Langage SQL :

SELECT A.r₁, ..., A.r_{p-1} **FROM** R A **WHERE NOT EXISTS**
 (**SELECT** * **FROM** S **WHERE NOT EXISTS**
 (**SELECT** * **FROM** R B **WHERE** (A.r₁ = B.r₁ **AND** ... **AND** A.r_{p-1} = B.r_{p-1})
AND (B.r_p = S.r_p **AND** ... **AND** B.r_n = S.r_n)))
GROUP BY A.r₁, ..., A.r_{p-1}
HAVING COUNT(*) = (**SELECT** COUNT (DISTINCT S.r_p, ..., S.r_n) **FROM** S);