

# Bases de Données Relationnelles

## TD 2 : Calcul Relationnel des Tuples

### SI3 & MAM4

September 27, 2017

## 1 Algèbre relationnelle & langue naturelle

### 1.1 Formalisation en algèbre relationnelle des requêtes exprimées en langue naturelle

On considère le schéma de la base de données suivante:

```
marque(IdM, NomM, Classe, Pays, IdProp)
societe(IdS, Nom, Pays, Ville)
enreg(NumE, IdM, Pays, DateE, IdDeposant)
vente(NumV, IdM, DateV, Pays, IdVend, IdAch)
```

Exprimer (lorsque c'est possible) en algèbre relationnelle les requêtes suivantes (i.e., construire la formule algébrique qui les définit).

1. Les noms et pays des sociétés possédant au moins une marque.

---


$$\Pi_{Nom, Pays} [societe \bowtie (\delta_{IdProp \leftarrow IdS} (\Pi_{IdProp} marque))]$$


---

2. Les noms et villes des sociétés ayant au moins une marque dans la classe 24.

---


$$\Pi_{Nom, Ville} [societe \bowtie (\delta_{IdProp \leftarrow IdS} (\Pi_{IdProp} (\sigma_{Classe=24} marque)))]$$


---

3. Les noms des marques françaises enregistrées qui appartiennent au moins à deux classes distinctes.

---

On calcule d'abord la table  $\langle IdM, NomM, Classe \rangle$  qui contient toutes les marques françaises enregistrées, puis on fait une auto-jointure sur cette table.

$$A = \Pi_{IdM, NomM, Classe} (enreg \bowtie \Pi_{IdM, NomM, Classe} (\sigma_{Pays=Fr} marque))$$

$$B = \delta_{Classe \leftarrow C}(A)$$

$$\text{Ensemble recherché: } \Pi_{NomM} (\sigma_{Classe \neq C} A \bowtie B)$$


---

4. Les identifiants des marques enregistrées dans tous les pays

---


$$A = \Pi_{IdM, Pays}(enreg) \div \Pi_{Pays}(enreg)$$


---

5. Les noms des marques et les noms et pays de leurs propriétaires pour les marques enregistrées avant le 29 janvier 95.

---


$$\Pi_{NomM, Nom, Pays} [(\delta_{IdS \leftarrow IdProp} societe) \bowtie [(\Pi_{IdM} (\sigma_{DateE \leq 950129} enreg)) \bowtie (\Pi_{IdM, NomM, IdProp} (marque))]]$$


---

6. Les noms et pays des sociétés dont toutes les marques qu'elles possèdent sont dans la classe 14.

---


$$\Pi_{Nom, Pays} [(\delta_{IdS \leftarrow IdProp} societe) \bowtie [(\Pi_{IdProp} marque) \setminus (\Pi_{IdProp} (\sigma_{Classe \neq 14} marque))]]$$


---

7. Est-ce que toutes les marques ont été enregistrées ?

---


$$\text{Oui si } \Pi_{IdM} marque \setminus \Pi_{IdM} enreg = \emptyset$$


---

8. Les noms, villes et pays des propriétaires qui ont déposé eux-mêmes toutes les marques qu'ils possèdent et qui ont été enregistrées.

---

On recherche d'abord pour l'ensemble des marques enregistrées les propriétaires et les déposants, puis on élimine celles dont le propriétaire est différent du déposant.

$$A = (\Pi_{IdM, IdProp} marque) \bowtie (\Pi_{IdM, IdDeposant} enreg)$$

Ensemble recherché:

$$\Pi_{Nom, Ville, Pays} (\delta_{IdProp \leftarrow IdS} societe) \bowtie [\Pi_{IdProp} marque \setminus (\Pi_{IdProp} (\sigma_{IdProp \neq IdDeposant} A))]$$

Si on souhaite aussi afficher les sociétés qui ne possèdent aucune marque, il faut effectuer le calcul sur la table des sociétés.

---

9. Les noms des sociétés n'ayant vendu aucune des marques qu'elles possèdent.

---

On calcule l'ensemble de tous les propriétaires et on lui soustrait l'ensemble des propriétaires qui ont vendu une des marques qu'ils possèdent:

$$A = (\Pi_{IdProp} marque) \setminus (\Pi_{IdProp} (marque \bowtie (\delta_{IdVend \leftarrow IdProp, Pays \leftarrow PaysV} vente)))$$

Ensemble recherché:  $\Pi_{Nom}(\text{societe} \bowtie (\delta_{IdProp \leftarrow IdS} A))$

Si on souhaite aussi afficher les sociétés qui ne possèdent aucune marque, il faut effectuer le calcul sur la table des sociétés.

- 
10. L'avant-dernier propriétaire, s'il existe, de la marque "Chanel" enregistrée en France dans la classe 14.
- 

La difficulté vient du fait que le même propriétaire peut avoir vendu la même marque au propriétaire actuel. Il faut donc raisonner avec les tuples  $\langle NumV, IdVend \rangle$ . L'avant dernier propriétaire est le dernier vendeur. On va donc successivement :

- (a) Rechercher le propriétaire actuel de la marque channel enregistrée en france dans la classe 14:

$$P = \Pi_{IdM, IdProp} [(\sigma_{NomM=channel, Classe=14} \text{marque}) \bowtie (\Pi_{IdM} \sigma_{Pays=Fr} \text{enreg})]$$

- (b) Rechercher l'ensemble des vendeurs qui ont vendu cette marque à ce propriétaire :

$$V = \Pi_{NumV, IdVend} [(\delta_{IdProp \leftarrow IdAch} P) \bowtie \text{vente}]$$

- (c) Identifier le dernier vendeur:

$$NDV = \Pi_{NumV, IdVend} [\sigma_{NumV < NumV'} (V \bowtie (\delta_{NumV \leftarrow NumV', IdVend \leftarrow IdVend'} V))]$$

$$ADP = V \setminus NDV$$

Remarque : en général il est utile de dérouler ces requêtes sur un exemple !

---

Soit le schéma relationnel suivant :

```

employe(Nom, Prenom, DateDeNaissance, Adresse, NumeroSecuriteSociale, Salaire, NumeroDepartement,
Superieur)
departement( NomDepartement, NumeroDepartement, Directeur)
projet( NomProjet, NumeroProjet, Lieu, NumeroDepartement)
travaille( NumeroSecuriteSociale, NumeroProjet , Heures)

```

L'attribut Superieur d'un employé contient le numéro de sécurité sociale du supérieur direct de l'employé. Tout employé appartient à un département et travaille sur un nombre quelconque de projets. Chaque projet est rattaché à un département qui peut être différent de celui des employés travaillant sur ce projet.

La notation  $\text{rho}(a_1, a_2, \dots, a_n, b_1, \dots, b_k, a_{n+1}, \dots, a_m)$  indique que les attributs  $\{b_1, \dots, b_k\}$  constituent un identifiant unique de la relation rho.

Exprimer en algèbre relationnelles requêtes suivantes:

1. Date de naissance et adresse de Juliette Rochat.

---


$$\Pi_{DateDeNaissance, Adresse} [(\sigma_{Prenom=Juliette, Nom=Rochat} \text{employe})]$$


---

2. Nom et adresse des employés rattachés au département "recherche".
-

$$\prod_{Nom, Adresse} [employe \bowtie \sigma_{NomDepartement=recherche} departement]$$


---

3. Nom et prénom des employés dont le supérieur est Juliette Rochat.
- 

$$\prod_{Nom, Adresse} [employe \bowtie \delta_{NumeroSecuriteSociale \leftarrow Superieur} \prod_{NumeroSecuriteSociale} \sigma_{Prenom=Juliette, Nom=Rochat} employe]$$


---

4. Nom des employés qui travaillent plus de 10 heures sur un projet localisé à Sophia Antipolis.
- 

$$\prod_{Nom} [employe \bowtie \sigma_{Heures > 10} travaille \bowtie \sigma_{Lieu='Sophia Antipolis'} \prod_{NumeroProjet, Lieu} [projet]]$$


---

5. Nom des projets sur lesquels travaillent Jean Muller ou Annie Grandjean
- 

$$\prod_{NomProjet} [projet \bowtie travaille \bowtie (\prod_{NumeroSecuriteSociale} [\sigma_{(Nom=Muller, Prenom=Jean)} employe \cup \sigma_{(Nom=Grandjean, Prenom=Annie)} employe])]$$


---

6. Nom des projets sur lesquels travaillent a la fois Jean Muller et Annie Grandjean.
- 

$$\prod_{NomProjet} [projet \bowtie travaille \bowtie (\prod_{NumeroSecuriteSociale} [\sigma_{(Nom=Muller, Prenom=Jean)} employe])]$$

$$\cap$$

$$\prod_{NomProjet} [projet \bowtie travaille \bowtie (\prod_{NumeroSecuriteSociale} [\sigma_{(Nom=Grandjean, Prenom=Annie)} employe])]$$


---

7. Nom et prénom des employés qui ne travaillent sur aucun projet.
- 

$$\prod_{Nom, Prenom} [employe \setminus (employe \bowtie \prod_{NumeroSecuriteSociale} travaille)]$$


---

8. Numéro des projets qui ont au moins un participant de chaque département.
- 

$$ProDept = \prod_{NumeroProjet, NumeroDepartement} (travaille \bowtie employe)$$

$$Resultat = ProDept \div \prod_{NumeroDepartement} departement$$

Autre solution, sans utiliser la division.

Dans ce cas, il faut calculer l'inverse: les projets qui n'ont pas de participant dans chaque departement.

La relation P contient les couples (NumeroProjet, NumeroDepartement) tels que le projet NumeroProjet n'a pas de participant du departement NumeroDepartement:

$$P = (\prod_{NumeroProjet} projet) \times \prod_{NumeroDepartement} departement \setminus$$

$$\Pi_{NumeroProjet, NumeroDepartement} (travaille \bowtie employe)$$

$$Resultat = \Pi_{NumeroProjet} projet \setminus \Pi_{NumeroProjet} P$$


---

9. Nom des employés qui ne travaillent pas sur un projet localisé à Sophia Antipolis.
- 

On calcule d'abord les employés qui travaillent sur au moins un projet de Sophia-Antipolis (Sophia)

$$Sophia = \Pi_{Nom, NumeroSecuriteSociale} (employe \bowtie travaille \bowtie \Pi_{NumeroProjet} \sigma_{Lieu=Sophia} projet)$$

$$Resultat = \Pi_{Nom} ((\Pi_{Nom, NumeroSecuriteSociale} employe) - Sophia)$$


---

10. Nom des employes qui ne travaillent que sur des projets localisés à Sophia Antipolis.
- 

On calcule d'abord les employés qui travaillent sur au moins un projet qui n'est pas à Sophia Antipolis (NonSophia).

$$NonSophia = \Pi_{Nom, NumeroSecuriteSociale} (employe \bowtie travaille \bowtie \Pi_{NumeroProjet} \sigma_{Lieu \neq Sophia} projet)$$

$$Resultat = \Pi_{Nom} ((Sophia \setminus NonSophia))$$


---

## 1.2 Traduire en français les requêtes suivantes qui sont exprimées en algèbre relationnelle

1.  $\Pi_{(Nom, Prenom)} \sigma_{Superieur=X \wedge Salaire > Y} (Employe) \bowtie \delta_{NumeroSecuriteSocial \leftarrow X, Salaire \leftarrow Y} \Pi_{(NumeroSecuriteSociale, Salaire)} Employe$
- 

Noms et prénoms des employés qui gagnent plus que leur supérieur immédiat.

---

2.  $Projet - \Pi_{NomProjet, NumeroProjet, Lieu, NumeroDepartement} (Projet \bowtie Travaille \bowtie Employe)$
- 

Noms des projets sur lesquels ne travaille aucun employé du département du projet.

---

## 1.3 Traduire en français les requêtes suivantes qui sont exprimées en logique du premier ordre

1.  $\{ (e_1.Nom, e_1.Prenom) : \exists e_1, e_2 \in Employe, \exists t_1, t_2 \in Travaille \ e_2.Nom = "Rochat" \wedge e_2.Prenom = "Juliette" \wedge e_2.NumeroSecuriteSociale = t_2.NumeroSecuriteSociale \wedge e_1.NumeroSecuriteSociale = t_1.NumeroSecuriteSociale \wedge t_1.NumeroProjet = t_2.NumeroProjet \}$
- 

Noms et prénoms des employés qui travaillent sur un même projet que Juliette Rochat.

---

2.  $\{ (e.Nom, e.Prenom) : \exists e \in Employe, \forall p \in Projet (p.NumeroDepartement \neq e.NumeroDepartement \vee \exists t \in Travaille (t.NumeroSecuriteSociale = e.NumeroSecuriteSociale \wedge t.NumeroProjet = p.NumeroProjet)) \}$

---

Noms et prénoms des employés qui travaillent sur tous les projets de leur département.

---

## 2 Relations sur les expressions

1. Soit  $A \subseteq R$ , et soient  $r$  et  $s$  deux relations sur  $R$ . Quelles sont les relations d'inclusion ou d'égalité entre les expressions suivantes :

$\Pi_A(r \cap s)$  et  $\Pi_A(r) \cap \Pi_A(s)$

$\Pi_A(r \cup s)$  et  $\Pi_A(r) \cup \Pi_A(s)$

$\Pi_A(r \setminus s)$  et  $\Pi_A(r) \setminus \Pi_A(s)$

---

On a:

$\Pi_A(r \cap s) \subseteq \Pi_A(r) \cap \Pi_A(s)$  car les tuples qui diffèrent uniquement par les attributs de  $R \setminus A$  sont conservés si on effectue la projection avant l'intersection.

$\Pi_A(r \cup s) = \Pi_A(r) \cup \Pi_A(s)$  car contrairement à l'intersection, l'union n'élimine pas de relations.

$\Pi_A(r \setminus s) \supseteq \Pi_A(r) \setminus \Pi_A(s)$  car les tuples qui diffèrent uniquement par les attributs de  $R \setminus A$  sont éliminés si on effectue la projection avant la différence.

Remarque : en général un exemple est utile !

---

2. Exprimez  $\cap$  en fonction de  $\bowtie$

---

Les deux relations sont équivalentes si les tuples sont définis sur la même relation; incomparables autrement.

---

3. Soient  $r(R)$  et  $s(S)$  deux instances de relations. Quelles sont les relations d'inclusion existant entre  $r$ ,  $s$ ,  $r \bowtie s$ ,  $\Pi_R(r \bowtie s)$ ,  $\Pi_S(r \bowtie s)$  ?

---

$\Pi_R(r \bowtie s) \subseteq r$  et  $\Pi_S(r \bowtie s) \subseteq s$

---