Aspects logiques

Cours L3 – Bases de Données

Louis Jachiet

ENS

Slides adaptées depuis celle de Serge Abiteboul

Introduction

Un *modèle* de base de données c'est :

- un modèle de définition de données
- un langage de manipulation de données (langage de requête)

Nous allons voir aujourd'hui deux grandes familles de modèles :

- ceux similaires au modèle conjonctif
- ceux similaires au modèle relationnel

Et pour chaque :

- différent modèles de données : relations nommées ou non, faits
- différent modèles de langages de requêtes : algèbre, calcul, programmation logique

Et montrer que tous les modèles de chaque famille sont équivalents !

Rappels : le langage de définition de données relationnel

- dom ou domaine l'ensemble des valeurs contenues dans la BD
- att ou attribut un couple "nom de relation" - "type des valeurs"
- var ou variable
- $sort(R)$ l'ensemble des attributs de la relation R

Rappels : exemple

Cinémas		
Nom	Adresse	nbSalles
"La Nef"	"bd Édouard Rey"	7
"Le Mélies"	"caserne de Bonne"	3
"Le Club"	"rue Phalanstère"	3

Casting		
Film	Nom	Rôle
"Inception"	"Ellen Page"	Acteurice
"Inception"	"Leonardo DiCaprio"	Acteurice
"Inception"	"Christopher Nolan"	Réalisateurice
"Toy Story 3"	"Tom Hanks"	Acteurice
"Mamma Mia"	"Meryl Streep"	Acteurice
"Mamma Mia"	" Phyllida Lloyd"	Réalisateurice

Séances		
Titre	Date	Cinéma
"Inception"	12/08/2010 20h	"Le Mélies"
"Toy Story 3"	13/08/2010 17h	"Le Club"
"Toy Story 3"	13/08/2010 20h	"Le Club"
"Toy Story 3"	10/08/2010 17h	"Le Mélies"
"Akmareul boatda"	10/08/2010 16h	"Le Club"
"How to train your dragon"	12/03/2010 18h	"Pathé Chavant"

$$sort(Cinémas) = \{Nom, Adresse, nbSalles\}$$
$$sort(Séances) = \{Titre, Date, Cinéma\}$$
$$sort(Casting) = \{Film, Nom, Rôle\}$$

Rappels : schéma de BD

Schéma de BD

Schéma d'une BD c'est un ensemble de **relname** avec pour chaque un schéma de relation (le schéma d'un relation R c'est l'ensemble d'attributs $sort(R)$). Parfois on ajoute aussi le domaine D que peuvent prendre les valeurs. Formellement c'est donc un triplet $(D, R, Sort)$ (ou une paire $(R, Sort)$) avec R fini mais pas généralement D est infini.

Instance d'une relation

Étant donnée une relation R d'attributs $(c_1, t_1), \ldots, (c_n, t_n)$, une instance de R est un ensemble de n-uplets $(v_1^1, \ldots v_n^1), \ldots (v_1^k, \ldots v_n^k)$ tel que pour tout $1 \leq i \leq n, 1 \leq j \leq k$ on ait que $v_i^j \in t_i$

Instance d'une BD

Il s'agit d'une fonction qui associe à chaque **relname** R une instance de R compatible avec le schéma de R.

Variantes possibles

- Il est possible de supprimer la contrainte de typage !
- Il est aussi possible de supprimer les noms des attributs !
- Considérer simplement un ensemble de faits :
  $\{Cinémas(\text{"}La\ Nef\text{"}, bd\ Édouard\ Rey\text{"}, 7),$
  $Séances(\text{"}Inception\text{"}, \text{"}12/08/2010\ 20h\text{"}, \text{"}Le\ Mélies\text{"}),$
  $\ldots\}$

Requêtes conjonctives : intuition

Quels sont les noms et adresses des cinémas projetant un film du réalisateur Bergman ?

Schéma

- Cinémas(Nom,Adresse, nbSalles)
- Casting(Film,Nom,Rôle)
- Séances(Titre,Date,Cinéma)

Variables $\in dom^n$ (Tuple relational calculus [Cod72])

si $cine \in Cinémas$, $proj \in Séances$, $cast \in Casting$ sont tels que :
le *Rôle* dans *cast* est Réalisateurice et le *Nom* est "Bergman"
et le *Titre* dans *proj* et *Film* dans *cast* sont les mêmes
et *Cinéma* dans *proj* et *Nom* dans *cine* sont les mêmes
alors nous voulons la *Cinéma* de *proj* et *Adresse* de *cine*.

Requêtes conjonctives : intuition

Quels sont les noms et adresses des cinémas projetant un film du réalisateur Bergman ?

Schéma

- Cinémas(Nom,Adresse, nbSalles)
- Casting(Film,Nom,Rôle)
- Séances(Titre,Date,Cinéma)

variables $\in$ *dom* (Domain relational calculus [Lacroix, Pirotte])

si $\langle Film : x_{ti}, Nom : \text{“Bergman”}, Rôle : Réalisateurice\rangle \in Casting$,
et $\langle Cinéma : x_{sa}, Titre : x_{ti}, Date : x_{da}\rangle \in Séance$
et $\langle Nom : x_{sa}, Adresse : x_{ad}, nbSalles : x_{nb}\rangle \in Cinémas$,
alors inclure le n-uplet $\langle Cinéma : x_{sa},\ Adresse : x_{ad}\rangle$ dans la réponse, où $x_{ti}, x_{da}, \ldots$ sont des variables.

Requêtes conjonctives : intuition

Quels sont les noms et adresses des cinémas projetant un film du réalisateur Bergman ?

Schéma

- Cinémas(Nom,Adresse, nbSalles)
- Casting(Film,Nom,Rôle)
- Séances(Titre,Date,Cinéma)

En utilisant des règles logiques (Non-recursive conjunctive datalog)

$$\begin{aligned} \text{sol}(x_{sa},x_{ad}) \leftarrow\ & \text{Cinéma}(x_{sa},x_{ad},x_{nb}), \\ & \text{Casting}(x_{ti},\text{"Bergman"},\text{Réalisateurice}), \\ & \text{Séance}(x_{ti},x_{da},x_{sa}) \end{aligned}$$

Rêgles conjonctives (nrc-datalog)

Une *règle* conjonctive sur **R** est une expression q de la forme :

$$ans(u) \leftarrow R_1(u_1), ..., R_n(u_n)$$

- $n \geq 0$, $R_1, ..., R_n \in \mathbf{R}$
- $ans \notin \mathbf{R}$
- $u, u_1, ..., u_n$ n-uplets libres (i.e., variables + constantes) de bonnes arités ($|u_i| = |sort(R)|$)
- chaque variable apparaissant dans u doit aussi apparaître au moins une fois dans $u_1, ..., u_n$.

$R_1(u_1), \ldots, R_n(u_n)$: *corps*
$ans(u)$: *tête*.
Intuition : usine à fournir des faits

Requêtes conjonctives

Étant donné $\mathcal{V}$ un ensemble de variables, une *valuation* est une fonction ν de $\mathcal{V}$ dans **dom**. Pour chaque constante a, $\nu(a) = a$

Pour un schéma **S**, une requête q compatible avec **S** et une instance **I** de **S** on définit :

$$q(\mathbf{I}) \;=\; \{\nu(u) \mid \nu \text{ est une valuation sur } var(q) \text{ et } \nu(u_i) \in \mathbf{I}(R_i), \text{ pour chaque } i \in [1, n]\}.$$

Domaine actif : $adom(\mathbf{I})$ est l'ensemble des constantes apparaîssant dans **I** ; de la même manière $adom(q)$ est l'ensemble des constantes de q et
$adom(q, \mathbf{I}) = adom(q) \cup adom(\mathbf{I})$
$adom(q(\mathbf{I})) \subseteq adom(q, \mathbf{I})$ (donc $q(\mathbf{I})$ est fini)

Une autre syntaxe : Calcul conjonctif

Requête :

$$\begin{array}{lll} ans(x_{sa}, x_{ad}) & \leftarrow & Cinéma(x_{sa}, x_{ad}, x_{nb}), \\ & & Casting(x_{ti}, "Bergman", Réalisateurice), \\ & & Séance(x_{ti}, x_{da}, x_{sa}) \end{array}$$

Les virgules sont à lire comme des conjonctions ($\wedge$) les variables qui ne sont pas dans la tête sont supposées quantifiées existentiellement ($\exists$)
En calcul conjunctif :

$$\begin{array}{lll} \{x_{sa},\ x_{ad} \mid & \exists x_{ti} \exists x_{nb} \exists x_{da} & Cinéma(x_{sa}, x_{ad}, x_{nb}) \wedge \\ & & Casting(x_{ti}, "Bergman", Réalisateurice) \wedge \\ & & Séance(x_{ti}, x_{da}, x_{sa})\} \end{array}$$

Imbrication : Elle s'exprime aussi par :

$$\begin{array}{lll} \{x_{sa},\ x_{ad} \mid & \exists x_{ti} & (\exists x_{nb} Cinéma(x_{sa}, x_{ad}, x_{nb})) \wedge \\ & & Casting(x_{ti}, "Bergman", Réalisateurice) \wedge \\ & & (\exists x_{da} Séance(x_{ti}, x_{da}, x_{sa}))\} \end{array}$$

Algèbre PSJR

L'algèbre est un langage procédural (algèbre) (on décrit comment récupérer ce que l'on veut) et non plus déclaratif (on décrit ce que l'on veut). Mais relativement facile à implanter.

Atomes de l'algèbre :

1. R : une relation (existante dans le schéma)

Opérations de l'algèbre :

2. Projection Π_p : ne garder que certaines colonnes
3. Sélection σ_f : enlever des lignes suivant un critère (égalité entre deux valeurs ou entre une valeur et une constante)
4. Jointure $\bowtie$: faire un pont entre deux tables
5. Renommage $\rho_{a/b}$: renommer attributs d'une table

Pas de négation ou d'union dans le langage !

Algèbre PSJR

R

A	B
1	2
4	2
6	6
7	7
1	7
1	6

S

B	C
2	3
2	5
9	1
8	8

$R \bowtie S$

A	B	C
1	2	3
1	2	5
4	2	3
4	2	5

$\sigma_{A=1}(R)$

A	B
1	2
1	7
1	6

$\rho_{B/B',C/A}(S)$

B'	A
2	3
2	5
9	1
8	8

$\pi_A(R)$

A
1
4
6
7

Atomes

$$R(I) = R$$

Sélection

$\sigma_{A=c}$ et $\sigma_{A=B}$, $(A, B \in \mathbf{att},\ a \in \mathbf{dom})$

$$\sigma_{A=c}(I) = \{t \in I \mid t(A) = c\} \text{ et } \sigma_{A=B}(I) = \{t \in I \mid t(A) = t(B)\}.$$

Projection

$\pi_{A_1,\ldots,A_n}$ $(n \geq 0,\ A_1, \ldots, A_n \in \mathbf{att})$

$$\pi_{A_1,\ldots,A_n}(I) = \{\langle A_1 : t(A_1), \ldots, A_n : t(A_n)\rangle \mid t \in I\}$$

$$\pi_{A_1,\ldots,A_n}(I) = \{t|_{A_1,\ldots,A_n} \mid t \in I\}.$$

Algèbre PSJR

Jointure (Naturelle)

$sort(I) = V$, $sort(J) = W$, $sort(I \bowtie J) = V \cup W$

$$I \bowtie J \;= \{t \text{ sur } V \cup W \mid \text{pour } \; v \in I \text{ et } w \in J, t|_V = v \text{ et } t|_W = w\}$$

Quand $sort(I) = sort(J)$, alors $I \bowtie J = I \cap J$
Quand $sort(I) \cap sort(J) = \emptyset$ alors $I \bowtie J = I \times J$ (produit cartésien).

Renommage

utilise une fonction injective de $U = A_1 ... A_n$ dans **att** que l'on notera $A_1/B_1, \ldots, A_n/B_n$.

$$\rho_{A_1/B_1,\ldots,A_n/B_n}(I) \;= \{v \text{ sur } B_1, \ldots, B_n \mid \begin{array}{l} \text{pour } u \in I, \text{ et } v(B_i)) = u(A_i) \\ \text{pour tout } i \in [1..n] \end{array} \}$$

Théorème d'équivalence

Théorème de Codd

q est exprimable comme une rêgle conjonctive
ssi q est exprimable dans le calcul conjonctif
ssi q est exprimable dans l'algèbre PSJR

Preuve

- Rêgle $\Rightarrow$ Calcul : évident
- Calcul $\Rightarrow$ Algèbre : $\exists$ par projection, $\wedge$ par jointure
- Algèbre $\Rightarrow$ Rêgle : plus subtil, on se ramème à la forme

$$\pi_{j_1,\ldots,j_n}(\sigma_{\gamma_1}(\rho_{f_1}(R_1) \bowtie \ldots \bowtie \sigma_{\gamma_n}(\rho_{f_n}(R_n))))$$

en utilisant, des équivalences algébriques

Equivalences algébriques

1. pousser les renommages
2. pousser les filtres (attention aux conflits de constantes)
3. faire remonter les projections

Exemples

$\sigma_F(\sigma_{F'}(q))$	$\leftrightarrow$	$\sigma'_F(\sigma_F(q))$	
$\pi_X(\pi_Y(q))$	$\leftrightarrow$	$\pi_{X\cap Y}(q)$	
$\sigma_F(\pi_X(q))$	$\leftrightarrow$	$\pi_X(\sigma_F(q))$	si F porte sur des attributs de X
$q_1 \bowtie q_2$	$\leftrightarrow$	$q_2 \bowtie q_1$	
$\sigma_F(q_1 \bowtie q_2)$	$\rightarrow$	$\sigma_F(q_1) \bowtie q_2$	si F porte sur des attributs de q_1
$\sigma_{A=B}(q_1 \bowtie q_2)$	$\rightarrow$	$q_1 \bowtie \rho_{A/B}(q_2)$	si $A \in sort(q_1), B \in sort(q_2)$, $A \notin sort(q_2)$
$\pi_X(q_1) \bowtie \pi_X(q_2)$	$\rightarrow$	$\pi_X(q_1 \bowtie q_2)$	si $sort(q_1) \cap sort(q_2) \subseteq X$

Quelques propositions

- Les requêtes exprimables comme des rêgles conjonctives sans constantes sont satisfaisables : pour tout q, il existe i tel que $q(I) \neq \emptyset$.
  - Avec constantes : $\sigma_{A=0}(R) \bowtie \sigma_{A=1}(R)$ Non !
- Les requêtes conjonctives sont monotones : pour tout **I**,**J**, pour tout q,

$$\mathbf{I} \subseteq \mathbf{J} \Rightarrow q(\mathbf{I}) \subseteq q(\mathbf{J})$$

- Fermeture sous composition (vues)

$$\begin{array}{lcl} S_1(x,z) & \leftarrow & Q(x,y), R(y,z,w) \\ S_2(x,y,z) & \leftarrow & S_1(x,w), R(w,y,v), S_1(v,z) \\ S_3(x,z) & \leftarrow & S_2(x,u,v), Q(v,z) \end{array}$$

Peut être obtenu directement :

$$\begin{array}{lcl} S_3(x,z) & \leftarrow & Q(x,y_1), R(y_1,w,w_1), R(w,u,v_1), \\ & & Q(v,y_2), R(y_2,v,w_2), Q(v,z) \end{array}$$

Vues

La stabilité par composition permet de créer des vues.
Par exemple :

$$\begin{aligned} \mathit{Films} \quad = \quad & \rho_{Nom/Real}(\pi_{Titre,Nom}(\sigma_{rôle=réalisateurice}(\mathit{Casting}))) \\ & \bowtie \rho_{Nom/Act}(\pi_{Titre,Nom}(\sigma_{rôle=acteurice}(\mathit{Casting}))) \end{aligned}$$

avec $sort(\mathit{Films}) = \{\mathit{Titre}, \mathit{Real}, \mathit{Act}\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Soit le schéma ci-haut, écrire en algèbre, en calcul *n*-uplet et domaine les requêtes suivantes :

1. Où et quand peut on voir le film "Mad Max" ?
2. Quels sont les (titres de) films réalisés par Orson Welles ?
3. Quels sont les acteurs de "Ran" ?
4. Où peut-on voir un film dans lequel joue Signoret ?
5. Quels sont les acteurs qui ont produit un film ?

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Où et quand peut on voir le film “Mad Max” ?

- $\pi_{Nom,Horaire}(\sigma_{Titre=``MadMax''}(Salle))$
- $\{t.Nom, t.Horaire \mid t \in Salle \wedge t.Titre = ``MadMax''\}$
- $\{Nom, Horaire \mid \exists Titre\ (Nom, Horaire, Titre) \in Salle\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels sont les (titres de) films réalisés par Orson Welles ?

- $\pi_{Titre}\left(\sigma_{Réalisateur=``Orson\ Welles''}(Film)\right)$
- $\{t.titre \mid t \in Films \wedge t.Réalisateur = ``Orson\ Welles''\}$
- $\{Titre \mid \exists Acteur\ (Titre, ``Orson\ Welles'', Acteur) \in Film\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels sont les acteurs de "Ran" ?

- $\pi_{Acteur}(\sigma_{Titre=``Ran''}(Film))$
- $\{t.Acteur \mid t \in Films \wedge t.Titre = ``Ran''\}$
- $\{Acteur \mid \exists Real\ (``Ran'', Real, Acteur) \in Film\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Où peut-on voir un film dans lequel joue Signoret ?

- $\pi_{Nom}(\sigma_{Acteur=Signoret}(Film \bowtie Salle))$
- $\left\{ t.Nom \mid {}_{S \in Salle}^{F \in Films} \wedge F.Acteur = Signoret \wedge F.Titre = S.Titre \right\}$
- $\left\{ Nom \mid \exists Real, Titre, Horaire \; {}_{(Titre, Real, \text{``}Signoret\text{''}) \in Film}^{(Nom, Horaire, Titre) \in Salle} \right\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels sont les acteurs qui ont produit un film ?

- $\pi_{Acteur}(Film) \bowtie \rho_{Producteur/Acteur}(\pi_{Producteur}(Produit))$
- $\left\{ F.Acteur \mid \substack{F \in Film \\ P \in Produit} \wedge F.Acteur = P.Producteur \right\}$
- $\{x \mid (\exists a, b\ (a, b, x) \in Film) \wedge (\exists a\ (x, a) \in Produit)\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

6. Quels acteurs produisent un film dans lequel ils jouent ?
7. Quels sont les acteurs qui jouent dans les films de François Truffaut ?
8. Où peut on voir Marlon Brando après 16h ? (on suppose $\leq$ dans les filtres)
9. Quels acteurs jouent dans un film où Orson Welles joue ?
10. Quels producteurs sont producteurs de tous les films de Akira Kurosawa ?

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels sont les acteurs qui jouent dans les films de François Truffaut ?

- $\pi_{Acteur}(\sigma_{Réalisateur=\text{“}Francois\ \ Truffaut}(Film))$
- $\left\{F.Acteur \mid F \in Film \wedge F.Réalistateur = \text{“}Francois\ \ Truffaut''\right\}$
- $\left\{x \mid \exists t\ (t, \text{“}Francois\ \ Truffaut'', x) \in Film\right\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Où peut on voir Marlon Brando après 16h ?

- $\pi_{Nom}(\sigma_{Acteur=\text{``}Marlon\ Brando}(Film) \bowtie \sigma_{16h \leq Horaire}(Salle))$
- $\left\{F.Acteur \mid {}_{P \in Produit}^{F \in Film} \wedge F.Acteur = P.Producteur \wedge F.Titre = P.Titre\right\}$
- $\left\{x \mid \exists t\Big(\,(\exists a\ (t, a, x) \in Film) \wedge ((x, t) \in Produit)\Big)\right\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels acteurs jouent dans un film où Orson Welles joue ?

- $\pi_{Acteur}(\pi_{Titre}(\sigma_{Acteur=\text{“}Orson\ Welles\text{''}}(Film)) \bowtie Film)$
- $\left\{ F.Acteur \mid {}_{O \in Film}^{F \in Film} \wedge O.Acteur = \text{“}Orson\ Welles\text{''} \wedge F.Titre = O.Titre \right\}$
- $\left\{ a \mid \exists t \Big((\exists x\ (t, a, x) \in Film) \wedge (\exists r\ (t, \text{“}Orson\ Welles\text{''}, r) \in Film) \Big) \right\}$

Exercices

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels producteurs sont producteurs de tous les films de Akira Kurosawa ?

Cette requête n'est pas monotone ! Si on ajoute un nouveau film de Kurosawa sans acteur, on élimine tous les anciens résultats. Cette requête ne peut donc pas s'exprimer.

On rajoute union/disjonction

Où puis-je voir "Annie Hall" ou "Manhattan" ?

- Algèbre avec union (algèbre PSJRU)
  $\pi_{Cinéma}(\sigma_{Titre=\text{“Annie Hall”}}(Séances) \cup \sigma_{Titre=\text{“Manhattan”}}(Séances))$
- Plusieurs rêgles

$$\begin{array}{lcl} ans(x_t) & \leftarrow & Seances(x_t, \text{“Annie Hall”}, x_s) \\ ans(x_t) & \leftarrow & Seances(x_t, \text{“Manhattan”}, x_s) \end{array}$$

- Calcul avec disjunction $\vee$
  $\{x_t \mid \exists x_s(Séances(x_t, \text{“Annie Hall”}, x_s) \vee Séances(x_t, \text{“Manhattan”}, x_s))\}$

Autres possibilités :

- Sélection plus complexes
  $\pi_{Séances}(\sigma_{Titre=\text{“Annie Hall”} \vee Titre=\text{“Manhattan”}}(Séances))$
- Constantes plus complexes
  $\pi_{Cinema}(Seances \bowtie \{\langle Titre : \text{“Annie Hall”}\rangle, \langle Titre : \text{“Manhattan”}\rangle\})$

On rajoute l'union (2)

Le théorème d'équivalence reste vrai, les requêtes restent monotones ...

mais problème avec certains calculs, par exemple : $\{x, y \mid R(x) \vee R(y)\}$
Infini si on n'est pas prudent $\rightarrow$ formules sures

On rajoute la différence/négation

Dans quels films d'Hitchcock, n'a-t-il pas joué ?
Quels films passent au Gaumont Opéra mais pas au Réal ?

- Algèbre relationnelle : PSJRU + différence
- Calcul relationnel : $\exists, \wedge, \vee$ $\neg, \forall$
- Règles : négation dans le corps des règles

$\pi_{Titre}\sigma_{Real=\text{``Hitchcock''}}(Film) - \pi_{Titre}\sigma_{Act=\text{``Hitchcock''}}(Film)$

nr-datalog$^{\neg}$ par l'exemple

$$\begin{aligned} ans(x) \quad &\leftarrow \quad Film(x, \text{``Hitchcock''}, z), \\ &\qquad \neg Film(x, \text{``Hitchcock''}, \text{``Hitchcock''}) \end{aligned}$$

$$\begin{aligned} Hitch\text{-}Act(z) \quad &\leftarrow \quad , Réalisateurice), \\ &\qquad Casting(film, \text{``Hitchcock''}, z) \\ not\text{-}ans(x) \quad &\leftarrow \quad Film(x, y, z),\ \neg Hitch\text{-}Acteur(z) \\ ans(x) \quad &\leftarrow \quad Film(x, y, z),\ \neg not\text{-}ans(x) \end{aligned}$$

Calcul relationnel

On rajoute au calcul conjonctif $\neg$, $\vee$, $\forall$
$\varphi \vee \psi \equiv \neg(\neg\varphi \wedge \neg\psi)$, et $\forall x \varphi \equiv \neg\exists x \neg\varphi$
Terme : constante ou variable
Atome : $R(e_1, \ldots, e_n)$ (R nom de relation, $n = arit\acute{e}(R)$, chaque e_i est un terme)
Formules de base : atomes sur $\mathbf{R}$ + $e = e'$
Formules bien-formées :

(a) $\neg\varphi$
(b) $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$
(c) $\exists x\ \varphi$, $\forall x\ \varphi$

$$\begin{array}{rcl} \varphi \rightarrow \psi & \equiv & \neg\varphi \vee \psi \\ \varphi \leftrightarrow \psi & \equiv & (\varphi \wedge \psi) \vee (\neg\varphi \wedge \neg\psi) \end{array}$$

Calcul relationnel : indépendance au domaine

Variables libres et liées
litéral P : $\mathit{free}(P)$ = variables de P

$$\begin{array}{ll} \mathit{free}(\varphi \wedge \psi) = \mathit{free}(\varphi) \cup \mathit{free}(\psi) & \mathit{free}(\varphi \vee \psi) = \mathit{free}(\varphi) \cup \mathit{free}(\psi) \\ \mathit{free}(\neg \varphi) = \mathit{free}(\varphi) & \\ \mathit{free}(\exists x \varphi) = \mathit{free}(\varphi) - \{x\} & \mathit{free}(\forall x \varphi) = \mathit{free}(\varphi) - \{x\} \end{array}$$

Requêtes : $\{e_1, ..., e_n \mid \varphi\}$
variables de $e_1, ... e_n$ sont les variables libres de φ

$$\{x_t \mid \exists x_a \mathit{Film}(x_t, \text{``Hitchcock''}, x_a) \ \wedge \ \neg \mathit{Film}(x_t, \text{``Hitchcock''}, \text{``Hitchcock''})\}$$

$$\begin{array}{l} \{x_t \ \mid \ \exists x_d, x_a \ \mathit{Film}(x_t, x_d, x_a) \wedge \\ \qquad \forall y_a \ (\exists y_d \mathit{Film}(x_t, y_d, y_a) \\ \qquad\qquad \rightarrow \exists z_t \ \mathit{Film}(z_t, \text{``Hitchock''}, y_a))\} \end{array}$$

Sémantique du calcul

Problème : résultat fini ?

$$\begin{array}{lll} \textit{(non-sure-1)} & \{x & \mid \neg \textit{Films}(\text{“Cries and Whispers”}, \text{“Bergman”}, x)\} \\ \textit{(non-sure-2)} & \{x, y & \mid \textit{Films}(\text{“Cries and Whispers”}, \text{“Bergman”}, x) \\ & & \qquad \vee\ \textit{Films}(y, \text{“Bergman”}, \text{“Ullman”})\} \end{array}$$

Solution A : variables prennent leurs valeurs dans le domaine actif
Solution B : interdire les requêtes non-sures $\leftarrow$

$$\textit{(non-sure-3)} \quad \{\ x \mid \forall y\ R(x, y)\}$$

Résultats intermédiaires non finis

Calcul relationnel : sémantique domaine actif

valuations ν de *free*(φ) dans *adom*$(q, \mathbf{I})$

$\mathbf{I}$ *satisfait* φ pour ν (qui s'écrit $\mathbf{I} \models_{adom} \varphi[\nu]$) quand :

(a) $\varphi = R(u)$ et $\nu(u) \in \mathbf{I}(R)$;

(b) $\varphi = (s = s')$ et $\nu(s) = \nu(s')$;

(c) $\varphi = (\psi \wedge \xi)$ et ($\mathbf{I} \models_{adom} \psi[\nu]$ et $\mathbf{I} \models_{adom} \xi[\nu]$) ;

(d) $\varphi = (\psi \vee \xi)$ et ($\mathbf{I} \models_{adom} \psi[\nu]$ ou $\mathbf{I} \models_{adom} \xi[\nu]$) ;

(e) $\varphi = \neg\psi$ et $\mathbf{I} \not\models_{adom} \psi[\nu]$,

(f) $\varphi = \exists x\ \psi$ et pour quelque $c \in adom$, $\mathbf{I} \models_{adom} \psi[\nu \cup \{x/c\}]$; ou

(g) $\varphi = \forall x\ \psi$ et pour chaque $c \in adom$, $\mathbf{I} \models_{adom} \psi[\nu \cup \{x/c\}]$.

Sémantique d'une requête

$$q_{adom}(\mathbf{I}) = \{\nu(\langle e_1, \ldots, e_n\rangle) \mid \mathbf{I} \models_{adom} \varphi[\nu], \\ \nu \text{ est une valuation sur } \mathit{free}(\varphi) \\ \text{dont l'image} \subseteq \mathit{adom}\}.$$

Calcul relationnel : domaine indépendant

On normalise les formules en poussant les négations devant les $\exists$ et les atomes et en aplatissant les $\wedge$ et $\vee$:

1. $\neg\neg\varphi \to \varphi$
2. $\neg(\varphi_1 \wedge \cdots \wedge \varphi_n) \to \neg\varphi_1 \vee \cdots \vee \neg\varphi_n$
3. $\neg(\varphi_1 \vee \cdots \vee \varphi_n) \to \neg\varphi_1 \wedge \cdots \wedge \neg\varphi_n$

On calcule récursivement l'ensemble *rr* des variables *range-restricted* :

$$\begin{array}{rcl}
rr(R(e_1, \ldots, e_k)) & = & \{e_i \mid e_i \text{ variable}\} \\
rr(x = a) & = & \{x\} \\
rr(\varphi_1 \wedge \varphi_2) & = & rr(\varphi_1) \cup rr(\varphi_2) \\
rr(\varphi_1 \vee \varphi_2) & = & rr(\varphi_1) \cap rr(\varphi_2) \\
rr(\varphi \wedge x = y) & = & rr(\varphi) \oplus \{x, y\} \\
rr(\neg\varphi) & = & \emptyset \\
rr(\exists x \varphi) & = & rr(\varphi) \setminus \{x\}
\end{array}$$

Et on vérifie que $x \in rr(\varphi)$ pour chaque $\exists x\varphi$ puis que $free(q) \subseteq rr(q)$.

Exercice : range restriction

Les requêtes suivantes sont-elles *range-restricted* ?

- $R(x, y) \lor R(x, z)$
- $\Big(R_1(x, y) \lor R(z, y)\Big) \land R(z, x)$
- $\exists z\ P(z, y, x) \lor \Big((R(x, y) \land (S(z) \land \neg T(x, z))) \lor T(y, z)\Big)$

Exercice : la division

Montrer que la division est une opération redondante.
Soient I sur X, J sur Y $\subset$ X, on définit

$$I \div J \ = \ \{u \mid \forall v \in J, [u, v] \in I\}$$

avec $sort(I \div J) = sort(I) - sort(J)$, c'est à dire l'ensemble de n-uplets u de type $sort(I) - sort(J)$ tels que pour tout $v \in J$ on ait $u + v \in I$

Exemple : *Quels acteurs jouent dans tous les films de Hitchcock ?*

$$\boxed{\pi_{Titre,Acteur}(Films) \div \pi_{Titre}\left(\sigma_{Real=Hitchcock}(Films)\right)}$$

Exercice : la division (dans le cadre de l'exemple)

On calcule d'abord *PasActeurDansFilmHitchcock*(*Acteur*, *Titre*) qui contient les paires de *Titre* (de *Films* d'*Hitchcock*) et d'*Acteur* (de *Films*) qui ne sont pas dans *Films* :

$$\pi_{Titre}(\sigma_{Réalistateur=Hitchcock}(Films)) \bowtie \pi_{Acteur}(Films) - \pi_{Acteur,Titre}(Films)$$

Un *Acteur* a donc joué dans tous les films d'*Hitchcock* quand il n'apparaît pas dans la liste plus haut :

$$\pi_{Acteur}(Films) - \pi_{Acteur}(PasActeurDansFilmHitchcock)$$

Donc au total :

$$\pi_{Acteur}(Films) - \pi_{Acteur}\Big(\begin{array}{l} (\pi_{Titre}(\sigma_{Réalistateur=Hitchcock}(Films)) \\ \bowtie \pi_{Acteur}(Films)) \\ -\pi_{Acteur,Titre}(Films)\Big) \end{array}$$

Exercice : la division (dans le cadre général)

Pour $I \div J$ on calcule $IpasJ$ les paires $\bar{u}$ de type $sort(I \div J)$,$\bar{v}$ de type J telles que $\bar{u}, \bar{v}$ n'est pas dans I :

$$IpasJ = \pi_{sort(I \div J)}(I) \bowtie J - I$$

puis le résultat est l'ensemble de u de $\pi_{sort(I \div J)}(I)$ qui ne sont pas dans $\pi_{sort(I \div J)}(IpasJ)$:

$$\pi_{sort(I \div J)}(I) - \pi_{sort(I \div J)}(IpasJ)$$

$$\pi_{sort(I \div J)}(I) - \pi_{sort(I \div J)}(\pi_{sort(I \div J)}(I) \bowtie J - I)$$

Exercice : Complément de jointure

Montrer que le complément de jointure est une opération redondante.
Soient I sur X, J sur Y, X ∩ Y = Z

$$sort(I \overline{\bowtie} J) = X$$

$$I \overline{\bowtie} J = \{x \in I \mid \pi_Z(x) \notin \pi_Z(J)\}$$

Exemple : *Quels films contiennent des acteurs qui ne jouent dans aucun film d'Hitchcock ?*

$$\boxed{\begin{array}{c} \pi_{Titre,Acteur}\left(\sigma_{Real=Hitchcock}(Films)\right) \\ \\ \overline{\bowtie} \\ \pi_{Titre}\left(\sigma_{Real=Hitchcock}(Films)\right) \end{array}}$$

Exercice : Complément de jointure

Montrer que le complément de jointure est une opération redondante.
Soient I sur X, J sur Y, $X \cap Y = Z$

$$sort(I \overline{\bowtie} J) = X$$

$$I \overline{\bowtie} J = \{x \in I \mid \pi_Z(x) \notin \pi_Z(J)\}$$

$$I \overline{\bowtie} J = I - \pi_X(I \bowtie J)$$

Quelques propositions

- Théorème d'équivalence : algèbre $\leftrightarrow$ calcul $\leftrightarrow$ nr-datalog$^{\neg}$
- Non-monotonie (à cause de la différence)
- Fermeture sous composition (à cause de l'algèbre)
- Minimalité des opérateurs de l'algébre Projection Sélection Jointure Renommage Différence Union
- Faible complexité en *données* (LOGSPACE, PTIME, NC)
- Tests d'équivalence et d'inclusion sont indécidables - donc optimisation difficile
- Décidable pour les requêtes conjonctives - optimisation principalement pour RQ

Exercices (2)

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Soit le schéma ci-haut, écrire en algèbre, en calcul relationnel les requêtes suivantes :

11. Quels spectateurs voient tous les films ?
12. Quels sont les spectateurs qui aiment tous les films qu'ils voient ?
13. Quels films ne passent dans aucune salle ?
14. Qui produit un film qui ne passe dans aucune salle ?
15. Quels producteurs voient tous les films qu'ils produisent ?

Exercices (2)

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels spectateurs voient tous les films ?

- $Vu \div \pi_{Titre}(Film)$
- $\left\{ s \mid (\exists t\ (s,t) \in Vu) \wedge \neg \exists t, r, a \Big((t,r,a) \in Film \wedge \neg (s,t) \in Vu \Big) \right\}$

Exercices (2)

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels sont les spectateurs qui aiment tous les films qu'ils voient ?

- $\pi_{spectateur}(Vu) - \pi_{spectateur}(Vu - Aime)$
- $\{s \mid (\exists t\ (s,t) \in Vu) \wedge (\forall t \in (s,t) \in Vu \Rightarrow (s,t) \in Aime)\}$

Exercices (2)

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels films ne passent dans aucune salle ?

- $\pi_{Titre}(Film) - \pi_{Titre}(Salle)$
- $\{t \mid (\exists r, a\ (t, r, a) \in Film) \wedge \forall n, h\ \neg(n, h, t) \in Salle\}$

Exercices (2)

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Qui produit un film qui ne passe dans aucune salle ?

- $\pi_{Producteur}(Produit \bowtie (\pi_{Titre}(Film) - \pi_{Titre}(Salle)))$
- $\left\{p \mid \exists t \ (p, t) \in Produit \wedge \Big((\exists r, a \ (t, r, a) \in Film) \wedge \forall n, h \ \neg(n, h, t) \in Salle\Big)\right\}$

Exercices (2)

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

Quels producteurs voient tous les films qu'ils produisent ?

- $\pi_{Producteur}(Produit \ltimes (\pi_{Titre}(Film) - \pi_{Titre}(Salle)))$
- $\left\{p \mid \exists t\ (p,t) \in Produit \wedge \left((\exists r, a\ (t,r,a) \in Film) \wedge \forall n, h\ \neg(n,h,t) \in Salle\right)\right\}$

Exercices (2)

Schéma

```
Salle (Nom, Horaire, Titre)
Film (Titre, Réalisateur, Acteur)
Produit (Producteur, Titre)
Vu (Spectateur, Titre)
Aime (Spectateur, Titre)
```

16. Quels producteurs voient tous les films de Akira Kurosawa ?
17. Quels spectateurs aiment un film qu'ils n'ont pas vu ?
18. Qui n'aime aucun film ?
19. Qui ne produit aucun film de Jacques Doillon ?
20. Quels sont les producteurs qui ne voient que les films qu'ils produisent ?

De la théorie à la pratique

Algèbre et calcul relationnels sont équivalents
La traduction de l'un à l'autre est facile

Requete en calcul		*langage*
$\downarrow$		*assertionnel*
Compilation		
$\downarrow$		
Requete algebrique		*langage*
$\downarrow$		*procdural*
Optimisation		
$\downarrow$		
Langage		
machine	$\rightarrow$	*valuation*

En realité, on n'utilise pas le calcul relationnel, mais un langage hybride – SQL – avec des gadgets

Count, Avg, Sum, Order by, etc.

Quels sont les acteurs qui ont un nombre de Bacon ?

FIN

References

Serge Abiteboul.
Modèle relationnel (slides du cours de bases de données), 2009.

Serge Abiteboul, Richard Hull, and Victor Vianu.
Foundations of databases. 1995.
Addison-Wesley.

Edgar F Codd.
A relational model of data for large shared data banks.
Communications of the ACM, 13(6) :377–387, 1970.

Edgar F Codd.
Relational completeness of data base sublanguages.
IBM Corporation, 1972.