

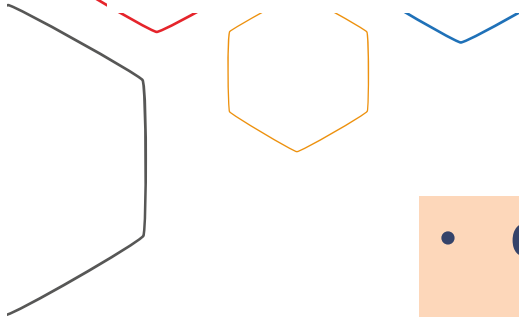


Représentation de connaissances



Interrogation de bases de connaissances

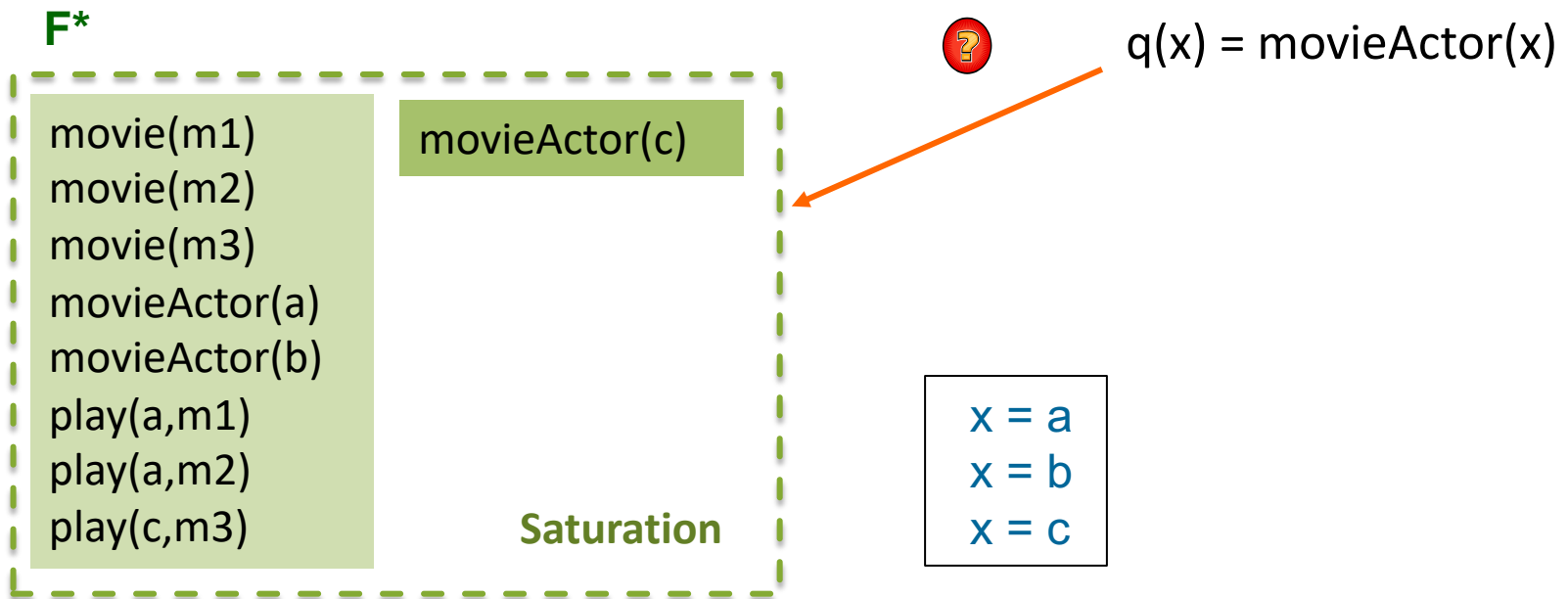
avec règles Datalog

- 
- Chaînage avant (Forward Chaining , « matérialisation »)
 - **Chaînage arrière classique (Backward Chaining)**
 - Variante de BC par réécriture de requête

EXAMPLE (FORWARD CHAINING)

R: $\text{play}(x,y), \text{movie}(y) \rightarrow \text{movieActor}(x)$

« find all movie actors »



Query Answering

1. **Forward chaining:** compute F^* (can be done offline)
2. Evaluate queries on F^* (based on homomorphisms to F^*)

EXAMPLE (BACKWARD CHAINING - CLASSICAL)

R: $\text{play}(x,y), \text{movie}(y) \rightarrow \text{movieActor}(x)$

$q(x) = \text{movieActor}(x)$

movie(m1)
movie(m2)
movie(m3)
movieActor(a)
movieActor(b)
play(a,m1)
play(a,m2)
play(c,m3)

Rewrite queries with the rules and the facts until obtaining empty sets of atoms
Compose substitutions to « trace » answers

- q unified with facts

$x = a$
 $x = b$

- q unified with head(R):

Renaming of R's variables (to avoid confusion)
R: $\text{play}(x',y'), \text{movie}(y') \rightarrow \text{movieActor}(x')$

$x \rightarrow x'$ unifier of q and head(R)

⇒ q1: $\text{play}(x',y'), \text{movie}(y')$

- part of q1 unified with fact $\text{play}(c,m3)$

$x' \rightarrow c, y' \rightarrow m3$

⇒ q2: $\text{movie}(m3)$

- q2 unified with fact $\text{movie}(m3)$

⇒ \emptyset

$x = c$

$x = a$
 $x = b$
 $x = c$

UNIFIERS

R1: $p(x1, y1), p(y1, z1) \rightarrow gp(x1, z1)$
R2: $mo(x2, y2) \rightarrow p(x2, y2)$
R3: $fa(x3, y3) \rightarrow p(x3, y3)$

$A = gp(x, a)$

A **unifier** u of atoms A and B is a substitution (of **variables**) such that $u(A) = u(B)$

A **most general unifier** (mgu) of A and B is a unifier u of A and B such that every other unifier of A and B can be written as $s \circ u$ where s is a substitution

$A = gp(x, a)$

$B = gp(x1, z1)$

$u1:$

$x \rightarrow x1$
 $z1 \rightarrow a$

$u2:$

$x1 \rightarrow x$
 $z1 \rightarrow a$

mgu

Il peut y avoir plusieurs mgu de A et B ,
mais ils sont « équivalents » :
on passe de l'un à l'autre par une substitution s qui
ne fait que renommer bijectivement les variables

$u2 = \{ x1 \rightarrow x \} \circ u1$

$u2 = \{ x \rightarrow x1 \} \circ u2$

$u3:$

$x \rightarrow a$
 $z1 \rightarrow a$
 $x1 \rightarrow a$

$u3 = \{ x1 \rightarrow a \} \circ u1$

DIRECT REWRITING

R1: $p(x1,y1), p(y1,z1) \rightarrow gp(x1,z1)$

R2: $mo(x2,y2) \rightarrow p(x2,y2)$

R3: $fa(x3,y3) \rightarrow p(x3,y3)$

Basic step: computation of a **direct rewriting** of a query q

1. look for a mgu u of an atom A in q and an atom in the head of a rule R
2. the direct rewriting of q with R according to u is

$$rew(q,R,u) = u(body(R)) \cup u(q \setminus A)$$

$q = gp(x,a)$

$head(R1) = gp(x1,z1)$

$u: x \rightarrow x1, z1 \rightarrow a$

$$rew(q,R1,u) = p(x1,y1), p(y1,a)$$

$q' = gp(x,a), p(x,b)$

$head(R1) = gp(x1,z1)$

$u: x \rightarrow x1, z1 \rightarrow a$

$$rew(q',R1,u) = p(x1,y1), p(y1,a), p(x1,b)$$

BACKWARD CHAINING POUR CQ BOOLÉENNE

Algorithme BC(K,Q)

// Donnée : $K = (F,R)$ et Q un but (liste d'atomes)

// Remarque : on voit un fait $p(a_1, \dots, a_k)$ comme une règle $\rightarrow p(a_1, \dots, a_k)$

// Résultat : vrai ssi K répond positivement à Q

Début

Si $Q = \emptyset$, retourner vrai

Pour toute règle $R = B \rightarrow H$ de **F** et **R**

telle que premier(Q) s'**unifie** avec H par un unificateur u

// ceci peut nécessiter de renommer les variables de R

// pour que les ensembles de variables de Q et R soient disjoints

$Q' \leftarrow u(B) \cup u(\text{reste}(Q))$

Si BC(K, Q') = vrai, retourner vrai

FinPour

Retourner faux

Fin

BACKWARD CHAINING POUR CQ BOOLÉENNE

Algorithme BC(K,Q)

// Donnée : $K = (F,R)$ et Q un but (liste d'atomes)

// Remarque : on voit un fait $p(a_1, \dots, a_k)$ comme une règle $\rightarrow p(a_1, \dots, a_k)$

// Résultat : vrai ssi K répond positivement à Q

Début

Si $Q = \emptyset$, retourner vrai

Pour toute règle $R = B \rightarrow H$ de **F** et **R**

telle que premier(Q) s'**unifie** avec H par un unificateur u

// ceci peut nécessiter de renommer les variables de R

// pour que les ensembles de variables de Q et R soient disjoints

$Q' \leftarrow u(B) \cup u(\text{reste}(Q))$

Si BC(K, Q') = vrai, retourner vrai

FinPour

Retourner faux

Fin

BACKWARD CHAINING POUR CQ

// $Q(x_1, \dots, x_k)$ est une CQ vue comme un ensemble d'atomes

// $S = \{(x_1, x_1), \dots, (x_k, x_k)\}$ initialement

Algorithme BC(K,Q,S)

// Donnée : $K = (F, R)$ et Q une CQ, S substitution de $\{x_1, \dots, x_k\}$

// Résultat : Ensemble des $\{(x_1, a_1), \dots, (x_k, a_k) \mid (a_1, \dots, a_k) \text{ est une réponse à } Q \text{ dans } K\}$

Début

$\text{Rep} \leftarrow \emptyset$ // sera le résultat

Pour toute règle $R = B \rightarrow H$ de **F** et **R** telle que $\text{premier}(Q)$ s'unifie avec H par u

 // si besoin, renommer les variables de R pour que $\text{variables}(Q) \cap \text{variables}(R) = \emptyset$

$Q' \leftarrow u(B) \cup u(\text{reste}(Q))$

$S \leftarrow u \circ S$ // mise à jour de (x_i, y) par (x_i, z) si $(y, z) \in u$

Si $Q' = \emptyset$, ajouter S à Rep

Sinon $\text{Rep} \leftarrow \text{Rep} \cup \text{BC}(K, Q', S)$

FinPour

Retourner Rep

Fin

EXEMPLE BC

R1: $p(x_1, y_1), p(y_1, z_1) \rightarrow gp(x_1, z_1)$

R2: $mo(x_2, y_2) \rightarrow p(x_2, y_2)$

R3: $fa(x_3, y_3) \rightarrow p(x_3, y_3)$

$Q(x) = gp(x, a)$

« Trouver les grand-parents de a »