Interrogation de données médiatisée par des connaissances : le cas des key-value stores

Olivier Rodriguez
<olivier.rodriguez@inria.fr>

Lundi 17 décembre 2018

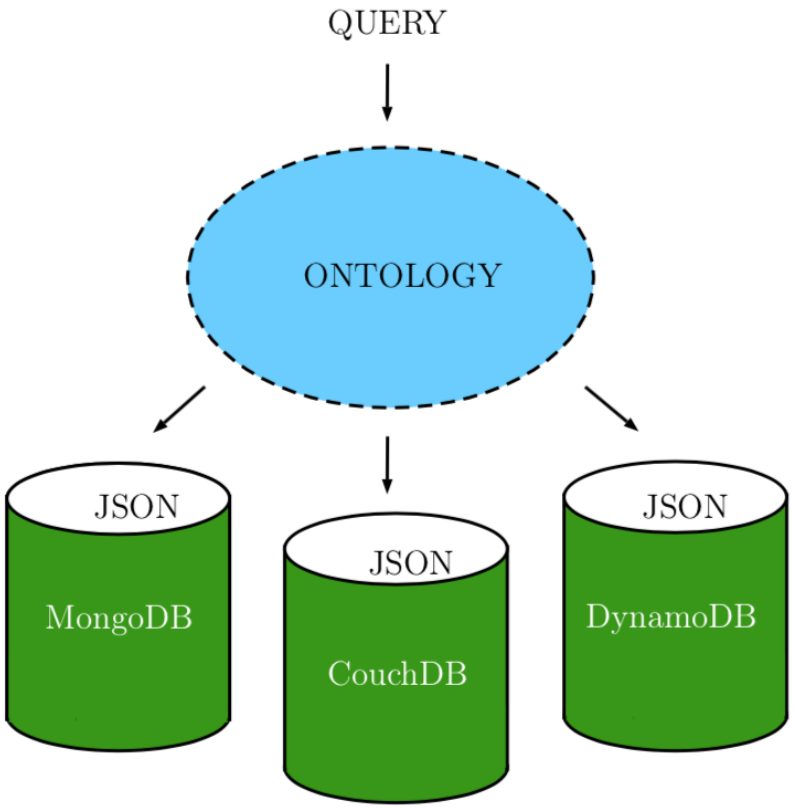
Données arborescentes

Pourquoi s'y intéresser ?

- Données très utilisées
- Systèmes de stockages efficaces & récents
- Spécialisation des graphes

Cadre de travail

Ontology-Mediated Query Answering (OMQA)



Avantages du cadre :

- Enrichir le vocabulaire d'interrogation
- Gérer de l'information incomplète
- Vue unifiée pour des données hétérogènes

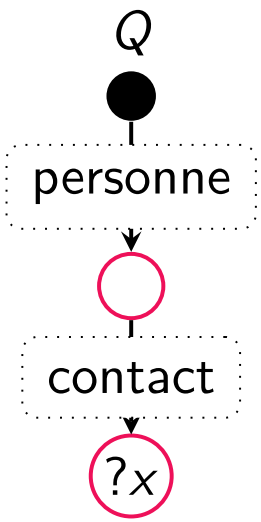
Problématiques de recherche :

- *Interrogation en présence d'ontologie*
- Interrogation fédérée

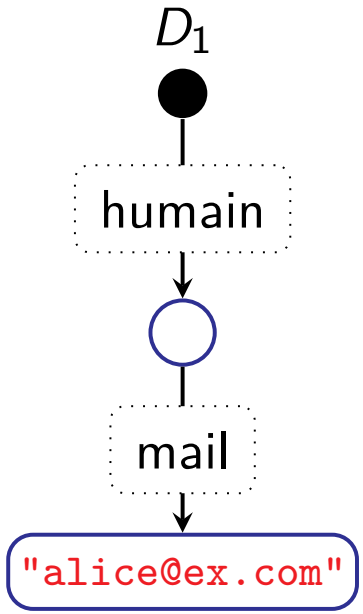
Interrogation en présence d'ontologie

Matérialisation

Requête



Base de données



Règles

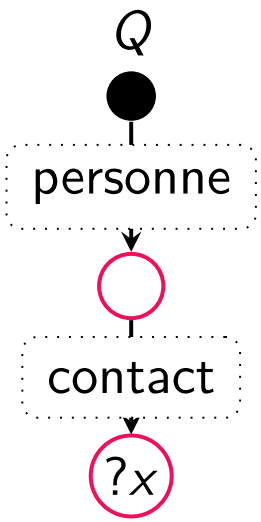
`humain(X,Y)` $\longrightarrow$ `personne(X,Y)`

`mail(X,Y)` $\longrightarrow$ `contact(X,Y)`

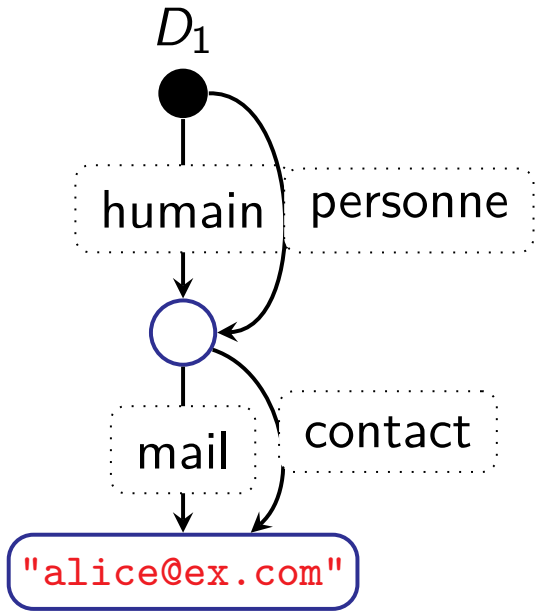
Interrogation en présence d'ontologie

Matérialisation

Requête



Base de données



Règles

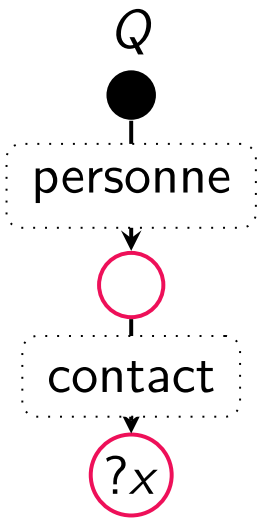
`humain(X,Y)` $\longrightarrow$ `personne(X,Y)`

`mail(X,Y)` $\longrightarrow$ `contact(X,Y)`

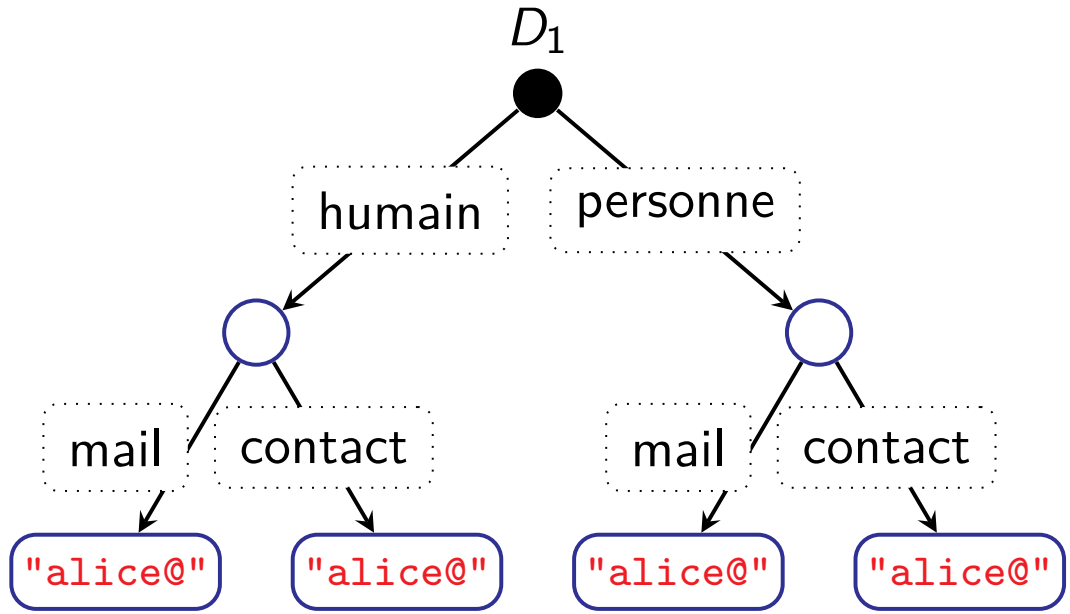
Interrogation en présence d'ontologie

Matérialisation

Requête



Base de données



Règles

`humain(X,Y)` $\longrightarrow$ `personne(X,Y)`

`mail(X,Y)` $\longrightarrow$ `contact(X,Y)`

Interrogation en présence d'ontologie

Matérialisation

Problèmes

- Explosion de l'espace de stockage
- Synchronisation avec les bases sources

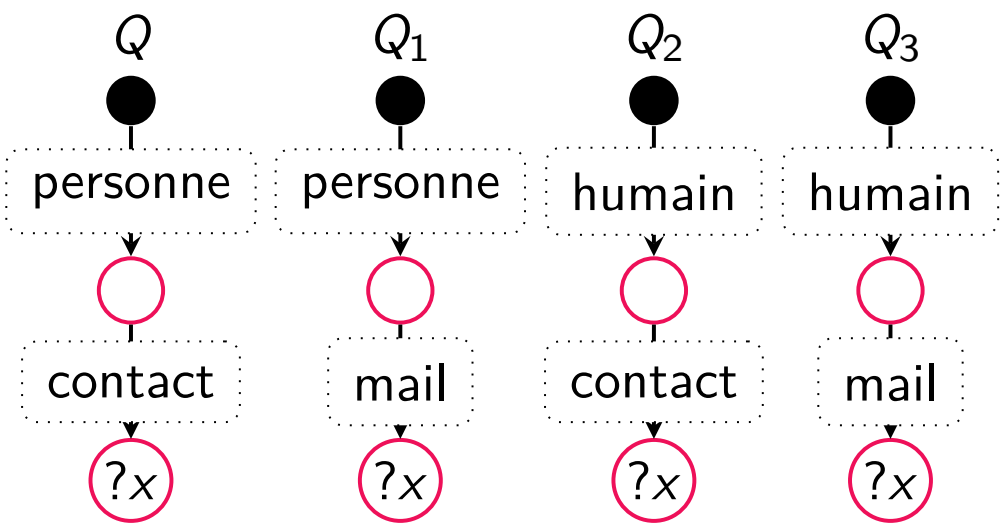
Axe de recherche

Réécriture des requêtes.

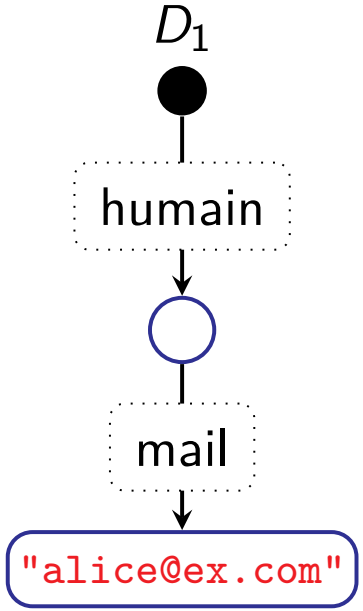
Interrogation en présence d'ontologie

Réécriture

Réécritures



Base de données



Règles

`humain(X,Y)` $\longrightarrow$ `personne(X,Y)`

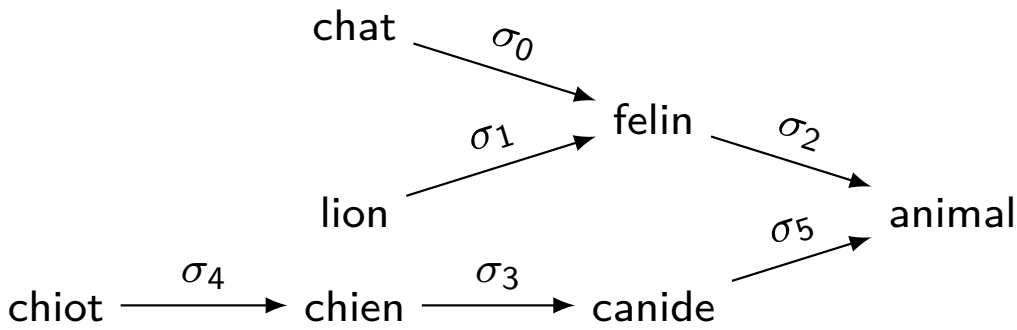
`mail(X,Y)` $\longrightarrow$ `contact(X,Y)`

Règles [MRU16]

$$\begin{array}{l} \texttt{humain(X,Y)} \longrightarrow \texttt{personne(X,Y)} \\ \texttt{personne(X,Y)} \longrightarrow \texttt{nom(X,Z)} \end{array} \Longrightarrow \begin{array}{l} \texttt{humain} \longrightarrow \texttt{personne} \\ \texttt{personne} \longrightarrow \exists\texttt{nom} \end{array}$$

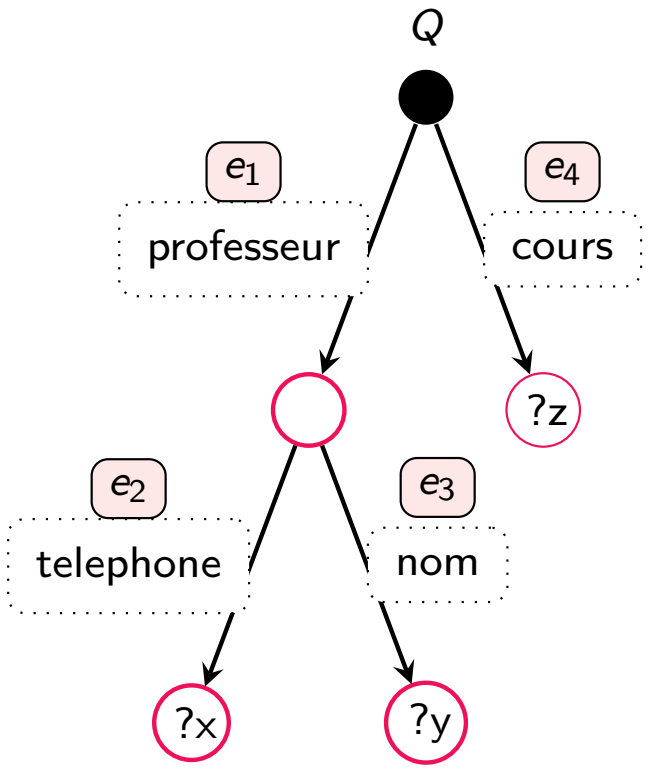
Graphe de dépendances des règles

$$\Sigma = \left\{ \begin{array}{lcl} \sigma_0 : \mathit{chat} & \to & \mathit{felin} \\ \sigma_1 : \mathit{lion} & \to & \mathit{felin} \\ \sigma_2 : \mathit{felin} & \to & \mathit{animal} \\ \sigma_3 : \mathit{chien} & \to & \mathit{canide} \\ \sigma_4 : \mathit{chiot} & \to & \mathit{chien} \\ \sigma_5 : \mathit{canide} & \to & \mathit{animal} \end{array} \right\}$$



$$\begin{array}{lcl} \mathit{app}(\mathit{animal}, \Sigma) & = & \Sigma \\ \mathit{app}(\mathit{felin}, \Sigma) & = & \{\sigma_0, \sigma_1\} \\ \mathit{app}(\mathit{canide}, \Sigma) & = & \{\sigma_3, \sigma_4\} \\ \mathit{app}(\mathit{chat}, \Sigma) & = & \emptyset \end{array}$$

Contextes de codages



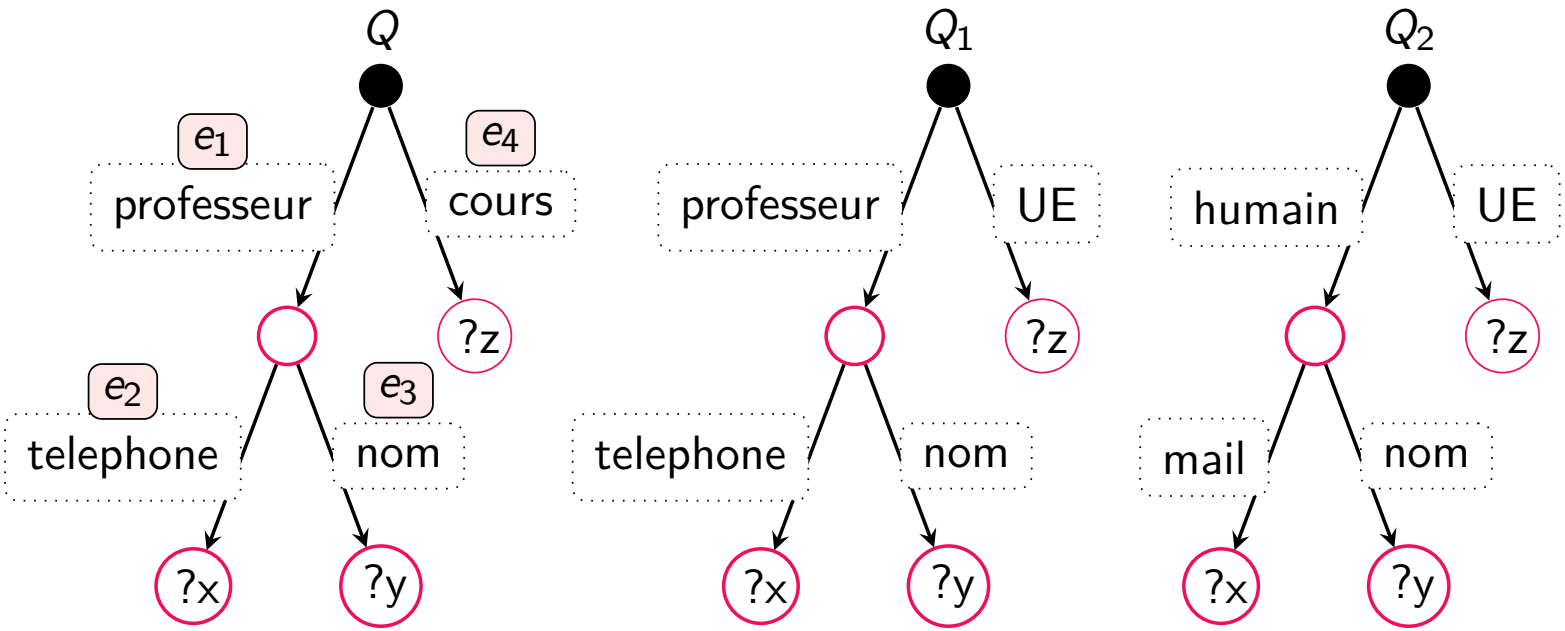
$$\Sigma = \left\{ \begin{array}{lcl} \sigma_1 : \mathit{personne} & \rightarrow & \mathit{professeur} \\ \sigma_2 : \mathit{humain} & \rightarrow & \mathit{personne} \\ \sigma_3 : \mathit{mail} & \rightarrow & \exists \mathit{telephone} \\ \sigma_4 : \mathit{UE} & \rightarrow & \mathit{cours} \end{array} \right\}$$

$$Cc = \left\{ \begin{array}{lcl} \mathrm{Cc}_1 & = & \{(0, \mathit{professeur}), (1, \mathit{personne}), (2, \mathit{humain})\} \\ \mathrm{Cc}_2 & = & \{(0, \mathit{telephone}), (1, \mathit{mail})\} \\ \mathrm{Cc}_3 & = & \{(0, \mathit{nom})\} \\ \mathrm{Cc}_4 & = & \{(0, \mathit{cours}), (1, \mathit{UE})\} \end{array} \right\}$$

Codes

$c = (e_4, e_3, e_2, e_1)$
$0000 = Q$
$1000 = Q_1$
$1012 = Q_2$

$$Cc = \left\{ \begin{array}{lcl} \mathrm{Cc}_1 & = & \{(0, \textit{professeur}), (1, \textit{personne}), (2, \textit{humain})\} \\ \mathrm{Cc}_2 & = & \{(0, \textit{telephone}), (1, \textit{mail})\} \\ \mathrm{Cc}_3 & = & \{(0, \textit{nom})\} \\ \mathrm{Cc}_4 & = & \{(0, \textit{cours}), (1, \textit{UE})\} \end{array} \right\}$$



Équivalence code/entier

$$c = (e_4, e_3, e_2, e_1) \qquad Cc = \left\{ \begin{array}{lcl} \mathrm{Cc}_1 & = & \{(0, \textit{professeur}), (1, \textit{personne}), (2, \textit{humain})\} \\ \mathrm{Cc}_2 & = & \{(0, \textit{telephone}), (1, \textit{mail})\} \\ \mathrm{Cc}_3 & = & \{(0, \textit{nom})\} \\ \mathrm{Cc}_4 & = & \{(0, \textit{cours}), (1, \textit{UE})\} \end{array} \right\}$$

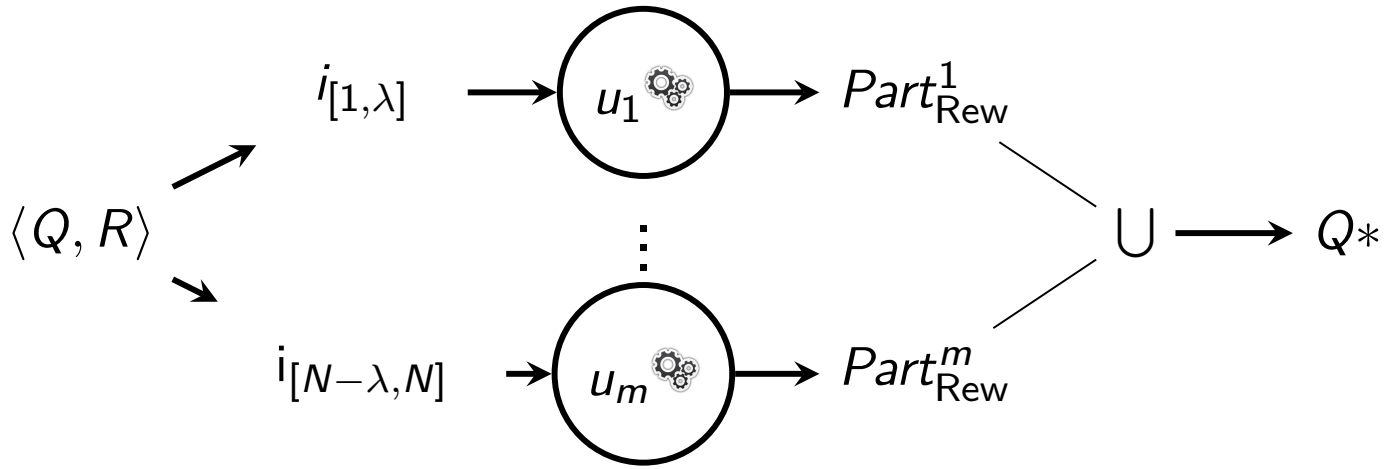
$$\begin{array}{ll} c & \equiv i \\ 0000 & \equiv 0 \\ 0001 & \equiv 1 \\ 0002 & \equiv 2 \\ 0010 & \equiv 3 \\ 0011 & \equiv 4 \\ & \vdots \\ 1012 & \equiv 11 \end{array}$$

$$\begin{array}{rcl} i = & & e_4 \times (|Cc_1| \times |Cc_2| \times |Cc_3|) \\ & + & e_3 \times (|Cc_1| \times |Cc_2|) \\ & + & e_2 \times (|Cc_1|) \\ & + & e_1 \end{array}$$

$$Q^* = [0, 11]$$

Parallélisation de la réécriture

- Idée : créer un partitionnement de l'espace des réécritures
- Unité de calcul associée avec un intervalle de taille $\lambda \approx \lceil N/m \rceil$



- Possibilité de paralléliser l'évaluation des réécritures sur les données.

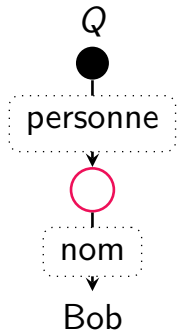
Traduction du langage de requête vers un langage cible

- Difficultés à assurer la correspondance entre le langage abstrait et un langage natif d'un key-value store (ex : mongodb)
- Non trivial : les key-value stores manquent de standardisation

Données

```
{
   "personne" : {
      "nom" : "Bob",
      "tel" : "12-34-56-78-90"
   }
}
```

Requête



Traduction MongoDB "intuitive" ne sélectionne pas le record

```
db.find( { "personne" : { "nom" : "Bob" } } )
```

(*)Equality matches on the embedded document require an exact match of the specified document, including the field order.

État des travaux

- Traduction des requêtes arborescentes vers le langage de MongoDB
- Article démonstration accepté à BDA 2017 (comité de lecture)

  « *Querying Key-Value Stores Under Simple Semantic Constraints : Rewriting and Parallelization* »

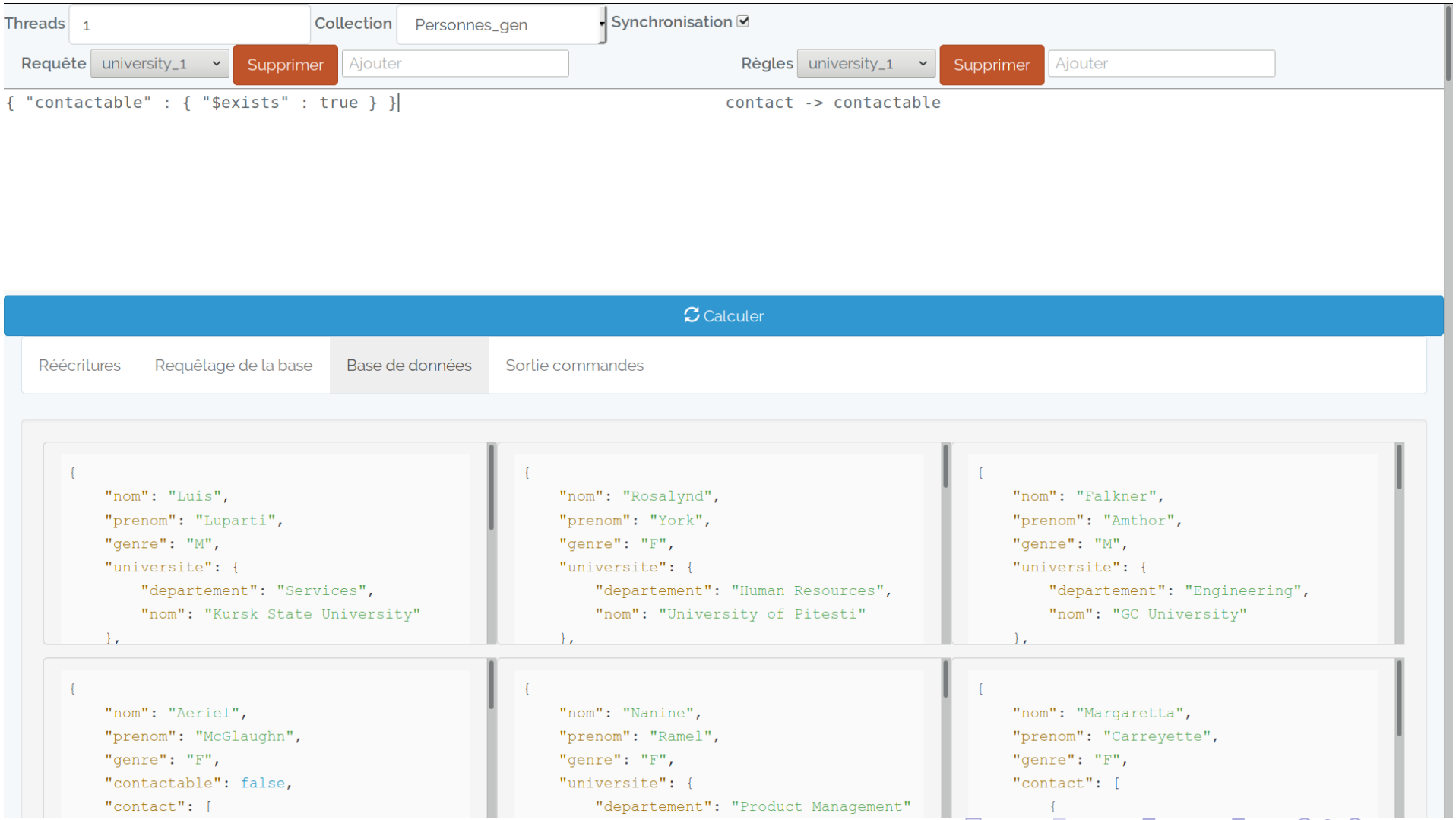
  O. Rodriguez, C. Colomier, C. Rivière, R. Akbarinia, F. Ulliana

  Démonstration présentée à Nancy en novembre 2017.

Premières expérimentations

Nombre de threads	1	2	3	4	5
Nombre de réécritures	Temps d'exécution (ms)				
18	10		1	1	1
36	1	2	1	1	1
108	15	3	3	2	2
324	6	4	3	3	3
648	6	5	4	3	3
1 296	9	9	7	11	7
2 592	25	15	14	18	9
18 432	180	186	144	152	34
93 312	3 662	1 234	883	525	590
124 416	1 407	1 055	1 516	678	930
186 624	12 144	2 475	1 490	2 363	1 103
209 952	1 370	826	1 293	1 045	523
248 832	7 400	5 452	3 577	1 141	1 107
746 496	32 478	14 393	13 268	17 206	13 660
839 808	58 435	30 302	24 687	20 361	20 004
1 259 712	34 084	22 343	13 602	19 425	11 567
5 971 968					
7 558 272					
22 674 816					
286 654 464					

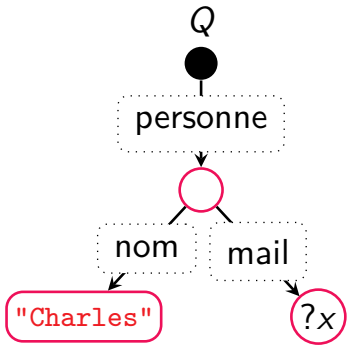
Système d'intégration de données arborescentes



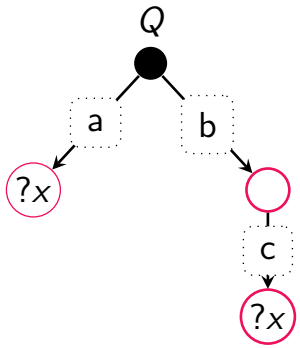
Objectifs à venir I

1) Vers un le langage d'interrogation plus expressif

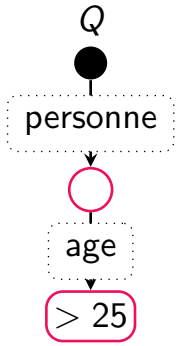
Actuellement



Jointure



Op. de comparaisons



2) Vers un langage de règles plus expressif

Actuellement

$(\forall)$ $\mathtt{A} \longrightarrow \mathtt{B}$

$(\exists)$ $\mathtt{A} \longrightarrow \exists\mathtt{B}$

Règles chemin ([MRU16, BBM+17])

$A_1 \ldots A_n \longrightarrow [\exists] B_1 \ldots B_m$

Point clé : la réécriture doit produire un ensemble fini de requêtes qui peuvent être évaluées dans le langage des sources

Objectifs à venir II

3) Étudier de nouvelles possibilités de traductions (pour le moment mongodb) pour confirmer que le langage de requête est concrètement utilisable sur les key-value stores.

- couchdb, dynamodb

4) Réfléchir aux optimisations en connaissant des contraintes sur les données (schéma).

5) Étudier la problématique de la fédération des bases de données.

Bibliographie I

Meghyn Bienvenu, Pierre Bourhis, Marie-Laure Mugnier, Sophie Tison, and Federico Ulliana.

Ontology-mediated query answering for key-value stores.

In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, pages 844–851. AAAI Press, 2017.

Marie-Laure Mugnier, Marie-Christine Rousset, and Federico Ulliana.

Ontology-mediated queries for nosql databases.

In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 1051–1057. AAAI Press, 2016.