

M1IF04 GDW - TD3

RDFS et inférence

Exercice 1:

On considère le graphe suivant (syntaxe n3):

```

1 @prefix ucbl: <http://univ-lyon1.fr#> .
2 @prefix ue:   <http://univ-lyon1.fr/ue#> .
3 @prefix etu:  <http://univ-lyon1.fr/etudiant#> .
4 @prefix form: <http://univ-lyon1.fr/formation#> .
5 @prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
6 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
7
8 ue:mif18 ucbl:formation form:mlif.
9 ue:mif17 ucbl:formation form:mlif;
10          ucbl:departement ucbl:informatique .
11 ue:mif16 ucbl:formation form:mlif;
12          ucbl:departement ucbl:informatique .
13 ue:mif13 ucbl:formation form:mlif .
14 etu:1234567 ucbl:inscrit ue:mif18;
15             ucbl:nom "Alice" .
16 etu:2345678 ucbl:inscrit ue:mif17 , ue:mif13 ;
17             ucbl:binome etu:3456789;
18             ucbl:nom "Basile" .
19 etu:3456789 ucbl:inscrit ue:mif17 ;
20             ucbl:nom "Charlotte" .
21 etu:4567890 ucbl:inscrit ue:mif13 , ue:mif16 ;
22             ucbl:binome etu:1234567 ;
23             ucbl:nom "Damien" .
24 form:mlif ucbl:departement ucbl:informatique .
25 form:m2ti ucbl:departement ucbl:informatique .
26 ucbl:inscrit rdfs:domain ucbl:etudiant .
27 ucbl:inscrit rdfs:range ucbl:ue .

```

et sa représentation graphique dans la figure 1

On considère les règles RDFS suivantes (nommées entre parenthèses):

$$\frac{P \text{ rdfs:domain } T \quad \text{et} \quad S \text{ } P \text{ } O}{S \text{ rdf:type } T} \quad (RDFS\text{Domain})$$

$$\frac{P \text{ rdfs:range } T \quad \text{et} \quad S \text{ } P \text{ } O}{O \text{ rdf:type } T} \quad (RDFS\text{Range})$$

ainsi que les règles métier suivantes:

$$\frac{E \text{ ucbl:binome } E2}{E2 \text{ ucbl:binome } E} \quad (Sym\text{Binome})$$

$$\frac{E \text{ ucbl:binome } E2 \quad \text{et} \quad E \text{ ucbl:inscrit } UE}{E2 \text{ ucbl:inscrit } UE} \quad (Binome\text{Inscrit})$$

1. Donner une/des règle(s) métier permettant de déduire qu'une UE dans une formation doit être dans le même département que cette formation.

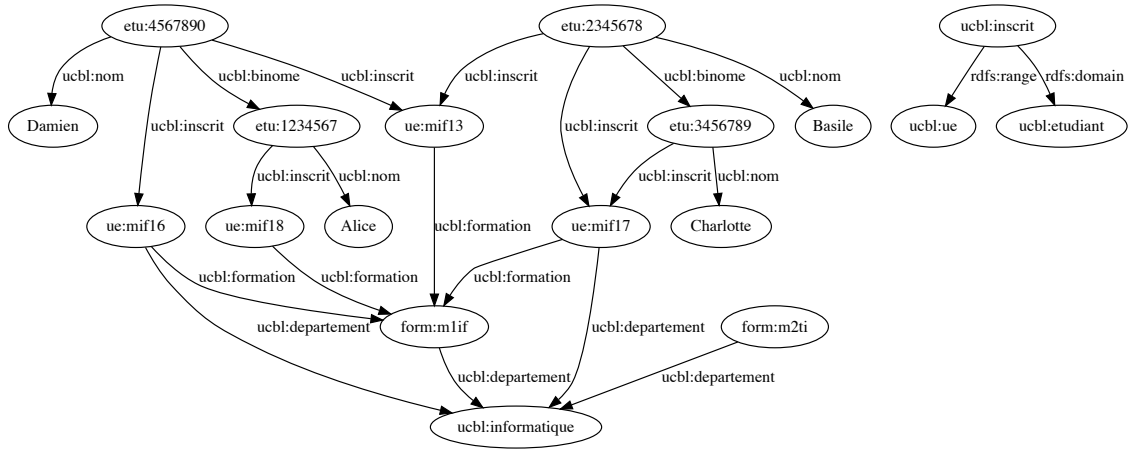


Figure 1: Représentation du graphe

2. Saturer le graphe en utilisant les règles pour déduire tous les nouveaux triplets possibles.

3. On considère la requête suivante:

```

1 PREFIX ucbl: <http://univ-lyon1.fr#> .
2 PREFIX ue:   <http://univ-lyon1.fr/ue#> .
3 PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
4
5 SELECT ?e WHERE {
6   ?e ucbl:inscrit ue:mif18 .
7   ?e rdf:type ucbl:etudiant .
8 }
```

- Répondre à la requête sur le graphe saturé précédent.
- Répondre à la requête sur le graphe de départ, mais en utilisant le chaînage arrière pour déduire à la volée les triplets intéressants via le chaînage arrière.
- Expliquer pourquoi, quand on interroge ce graphe, la ligne 7 est inutile dans cette requête.
- Réécrire la requête de façon à y répondre sur le graphe d'origine mais en prenant les inférences en compte.