

Partie « web sémantique »

Exercice 3 (Questions de cours – 2 pts)

Au sujet de la ressource DBpedia

3.1) quelle est la nature de cette ressource ? à partir de quelle information est-elle construite et comment ? (5 lignes maxi)

Une BC, un graphe RDF, un ensemble de données ouvertes liées. Construite par extraction d'information à partir des info-box des pages Wikipedia, et de la hiérarchie des catégories de Wikipedia.

3.2) DBpedia serait-elle une ressource adaptée pour annoter un corpus d'articles scientifiques du domaine de la médecine ? Pourquoi ? (5 lignes maxi)

Non. DBpedia est un vocabulaire général, alors que les articles scientifiques contiennent un vocabulaire technique et spécialisé. Utiliser plutôt MESH ou UMLS

Exercice 4 (4 pts)

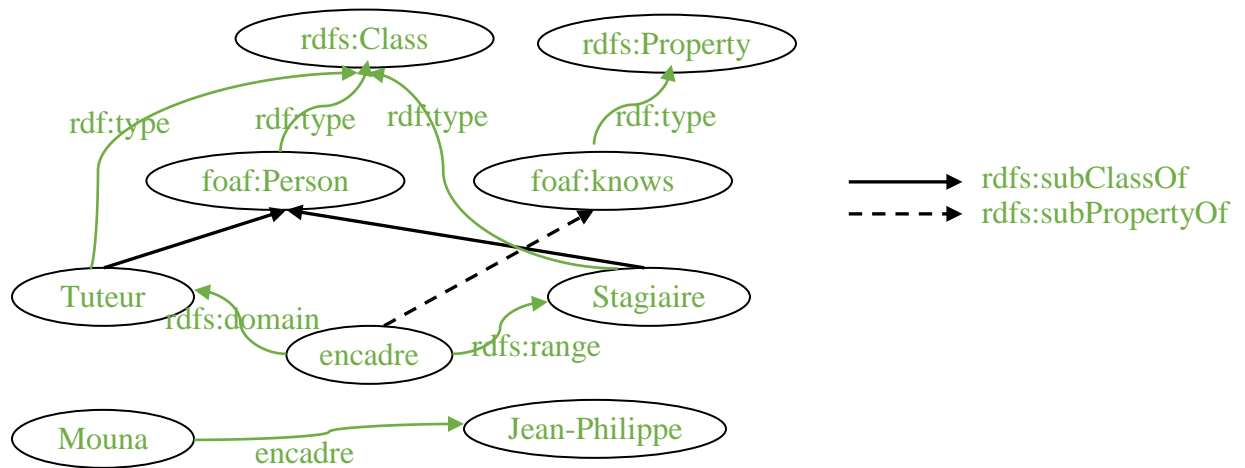
Voici un « fichier » qui décrit un vocabulaire et des ressources

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:foaf="http://xmlns.com/foaf/0.1/#"
xmlns:ex3="http://www.irit.fr/MELODI/ontologie/ex3#"
xml:base="http://www.irit.fr/ MELODI/ontologie/ex3">
<rdfs:Class rdf:ID="Stagiaire">
<rdfs:subClassOf rdf:resource="foaf:person"/>
</rdfs:Class>
<rdfs:Class rdf:ID="Tuteur">
<rdfs:subClassOf rdf:resource="foaf:person"/>
</rdfs:Class>
<rdf:Property rdf:ID="encadre">
<rdfs:domain rdf:resource="#Tuteur"/>
<rdfs:range rdf:resource="#Stagiaire"/>
<rdfs:subPropertyOf
rdf:resource="http://xmlns.com/foaf/0.1/knows"/>
</rdf:Property>
<rdf:Description rdf:ID="Mouna">
<ex3:encadre rdf:resource="#Jean-Philippe"/>
</rdf:Description>
```

4.1) Quel est la notation utilisée ici ? Quels sont les langages utilisés ? (1 pt en tout)

Notation RDF/XML. Les langages utilisés sont RDF, RDFS .

4.2) Donner sous la forme d'un graphe les 9 triplets décrits dans ce fichier. (1,5pt)



4.3) Quels sont les 5 triplets supplémentaires que l'on peut inférer d'après la sémantique de `rdfs:subClassOf` et `rdfs:subPropertyOf` ? (1,5 pt)

Mouna `rdfs:type` #Tuteur et Mouna `rdfs:type` foaf:Person
 Jean-Philippe `rdfs:type` #Stagiaire et Jean-Philippe `rdfs:type` foaf:Person
 Encadre `rdfs:type` rdfs:Property

Exercice 5 (4 pts)

Considérons la représentation OWL et RDF dans le « fichier » ci-dessous :

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:exo1="http://www.irit.fr/MELODI/ontologie/ex3#"
  xml:base="http://www.irit.fr/ MELODI/ontologie/ex3">
  <rdfs:Class rdf:ID="RoleSocial">
  <rdfs:Class rdf:ID="Stagiaire">
    <rdfs:subClassOf rdf:resource="#RoleSocial"/>
  </rdfs:Class>
  <rdfs:Class rdf:ID="Tuteur">
    <rdfs:subClassOf rdf:resource="#RoleSocial"/>
  </rdfs:Class>
  <rdf:Property rdf:ID="exerce">
    <rdfs:domain rdf:resource="foaf:Person "/>
    <rdfs:range rdf:resource="#RoleSocial"/>
  </rdf:Property>
  <rdf:Property rdf:ID="encadre">
    <rdfs:domain rdf:resource="#Tuteur"/>
    <rdfs:range rdf:resource="#Stagiaire"/>
```

```

<rdfs:subPropertyOf
  rdf:resource="http://xmlns.com/foaf/0.1/knows"/>
</rdf:Property>

<owl:Class rdf:ID="PersonneTuteur">
  <rdfs:subClassOf rdf:resource="foaf:Person" />
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#exerce" />
      <owl:someValuesFrom rdf:resource="#Tuteur" />
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="PersonneStagiaire">
  <rdfs:subClassOf rdf:resource="foaf:Person" />
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#exerce" />
      <owl:someValuesFrom rdf:resource="#Stagiaire" />
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<rdf:Description rdf:ID="Mouna">
  <ex3:encadre rdf:resource="#Jean-Philippe"/>
</rdf:Description>

<rdf:Description rdf:ID="Nathalie">
  <ex3:encadre rdf:resource="#Julien"/>
  <ex3:encadre rdf:resource="#Laurent"/>
</rdf:Description>

```

5.1) Ecrire une requête SPARQL qui retourne toutes les personnes qui sont des stagiaires. (1pt)

```

SELECT ?p WHERE
{ ?p rdf:type ex3:personneStagiaire }

```

5.2) Que retourne cette requête à partir du fichier ci-dessus ? (0,5pt)

Rien car les définitions de la BC conduisent à définir

#Mouna rdf:type #Tuteur (et non #personneTuteur)

#Nathalie rdf:type #Tuteur

#Jean-Philippe rdf:type #Stagiaire

#Julien rdf:type #Stagiaire

#Laurent rdf:type #Stagiaire

5.3) Soient les 2 requêtes SPARQL suivantes portant sur les connaissances précédentes

```

SELECT ?p1 ?p2
WHERE

```

```
{ ?p1 ex3:encadre ?p2 } GROUP BY ?p1
```

```
SELECT ?p1 ?p2
```

```
WHERE
```

```
{ ?p1 rdf:type foaf:Person .
```

```
?p2 rdf:type foaf:Person .
```

```
?p1 ex3:encadre ?p2 } GROUP BY ?p1
```

Que retournent les deux requêtes à partir des connaissances de l'énoncé ? (2pt)

La première retourne

?p1	?p2
-----	-----

#Mouna	#Jean-Philippe
--------	----------------

#Nathalie	#Julien
-----------	---------

#Nathalie	#Laurent
-----------	----------

La deuxième retourne un tableau vide

?p1	?p2
-----	-----