

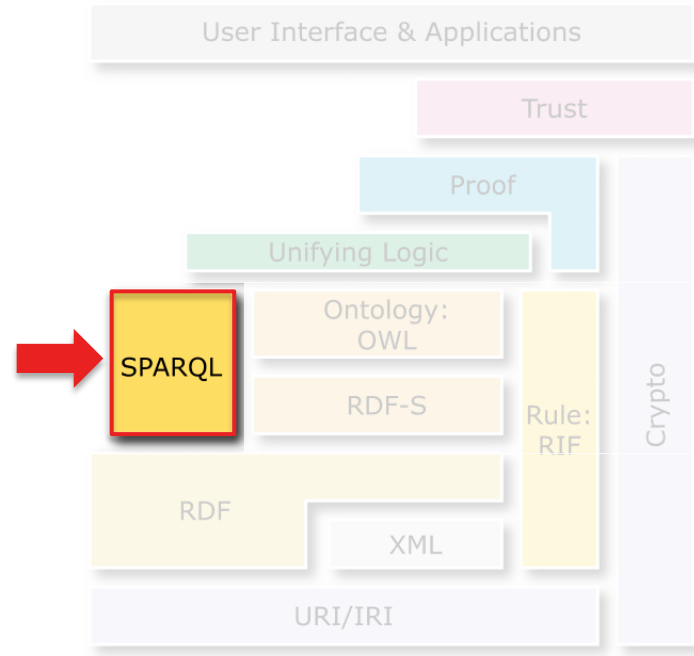
Web Sémantique

Semaine 03 : le langage de requête SPARQL

« accéder aux sources de données du Web »

SEMAINE 03 : le langage de requête SPARQL

interroger des bases RDF publiées sur le Web



SEMAINE 03 : le langage de requête SPARQL

1. Appariement de graphes RDF
2. Filtres, contraintes et fonctions
3. Pré et post traitements
4. Différentes formes de requêtes
5. Formats et protocole de retours
6. Modification des bases

SEMAINE 03 : le langage de requête SPARQL

- 1. Appariement de graphes RDF**
2. Filtres, contraintes et fonctions
3. Pré et post traitements
4. Différentes formes de requêtes
5. Formats et protocole de retours
6. Modification des bases

SPARQL Protocol And RDF Query Language

1. Langage de requête (syntaxe Turtle)

- SPARQL 1.1 Query Language - W3C REC 21 Mar. 2013

SPARQL Protocol And RDF Query Language

1. Langage de requête (syntaxe Turtle)

- SPARQL 1.1 Query Language - W3C REC 21 Mar. 2013
- SPARQL 1.1 Update - W3C REC 21 Mar. 2013

SPARQL Protocol And RDF Query Language

1. Langage de requête (syntaxe Turtle)

- SPARQL 1.1 Query Language - W3C REC 21 Mar. 2013
- SPARQL 1.1 Update - W3C REC 21 Mar. 2013

2. Langage de présentation des résultats

- SPARQL Query Results XML Format - W3C REC 21 Mar. 2013

interroger avec SPARQL

| | |
|---------------|------------------------------|
| SELECT | <i>ce que vous voulez</i> |
| FROM | <i>où vous voulez</i> |
| WHERE | <i>{ comme vous voulez }</i> |

les triplets en SPARQL

- syntaxe Turtle avec des points d'interrogation pour les **variables**:

```
?x    rdf:type    ex:Person
```

les triplets en SPARQL

- syntaxe Turtle avec des points d'interrogation pour les **variables**:

```
?x    rdf:type    ex:Person
```

- décrire des **patrons de graphes** à trouver:

```
SELECT ?subject ?property ?value  
WHERE { ?subject ?property ?value }
```

les triplets en SPARQL

- syntaxe Turtle avec des points d'interrogation pour les **variables**:

```
?x    rdf:type    ex:Person
```

- décrire des **patrons de graphes** à trouver:

```
SELECT ?subject ?property ?value  
WHERE { ?subject ?property ?value }
```

- un patron est par défaut une **conjonction** de triplets:

```
SELECT ?x WHERE  
{ ?x    rdf:type    ex:Person .  
  ?x    ex:name     ?name . }
```

mêmes abréviations qu'en Turtle

- triplets ayant un sujet commun :

```
SELECT ?name ?fname
WHERE {?x a ex:Person;
       ex:name ?name ;
       ex:firstname ?fname ;
       ex:author ?y . }
```



```
SELECT ?name ?fname
WHERE{?x rdf:type ex:Person.
      ?x ex:name ?name .
      ?x ex:firstname ?fname .
      ?x ex:author ?y . }
```

mêmes abréviations qu'en Turtle

- triplets ayant un sujet commun :

```
SELECT ?name ?fname
WHERE {?x a ex:Person;
       ex:name ?name ;
       ex:firstname ?fname ;
       ex:author ?y . }
```



```
SELECT ?name ?fname
WHERE{?x rdf:type ex:Person.
      ?x ex:name ?name .
      ?x ex:firstname ?fname .
      ?x ex:author ?y . }
```

- plusieurs valeurs

```
?x ex:firstname "Fabien", "Lucien" .
```

mêmes abréviations qu'en Turtle

- triplets ayant un sujet commun :

```
SELECT ?name ?fname
WHERE { ?x a ex:Person ;
        ex:name ?name ;
        ex:firstname ?fname ;
        ex:author ?y . }
```



```
SELECT ?name ?fname
WHERE { ?x rdf:type ex:Person .
        ?x ex:name ?name .
        ?x ex:firstname ?fname .
        ?x ex:author ?y . }
```

- plusieurs valeurs

```
?x ex:firstname "Fabien", "Lucien" .
```

- ressource anonyme

```
[ ex:firstname "Fabien" ] ou
```

```
[ ] ex:firstname "Fabien" .
```

déclarer les préfixes des espaces de noms

- déclarer des **préfixes** pour les vocabulaires utilisés dans la requête:

```
PREFIX mit: <http://www.mit.edu#>  
SELECT ?student  
WHERE {  
  ?student mit:registeredAt ?x .  
}
```

déclarer les préfixes des espaces de noms

- déclarer des **préfixes** pour les vocabulaires utilisés dans la requête:

```
PREFIX mit: <http://www.mit.edu#>
```

```
SELECT ?student
```

```
WHERE {
```

```
  ?student <http://www.mit.edu#registeredAt> ?x .
```

```
}
```


déclarer les préfixes des espaces de noms

- déclarer des **préfixes** pour les vocabulaires utilisés dans la requête:

```
PREFIX mit: <http://www.mit.edu#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?student
WHERE {
  ?student mit:registeredAt ?x .
  ?x foaf:homepage <http://www.mit.edu> .
}
```

déclarer les préfixes des espaces de noms

- déclarer des **préfixes** pour les vocabulaires utilisés dans la requête:

```
PREFIX mit: <http://www.mit.edu#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?student
WHERE {
  ?student mit:registeredAt ?x .
  ?x foaf:homepage <http://www.mit.edu> .
}
```

- déclarer un espace de **base** par défaut i.e. pour les URI relatifs

```
BASE <...>
```

spécifier la langue et le type des littéraux

pour obliger à respecter un attribut : @fr , ^^xsd:integer

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
```

```
SELECT ?x ?f WHERE {
```

```
  ?x foaf:name "Fabien"@fr ; foaf:knows ?f .
```

```
}
```

spécifier la langue et le type des littéraux

pour obliger à respecter un attribut : @fr , ^^xsd:integer

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
```

```
SELECT ?x ?f WHERE {
```

```
  ?x foaf:name "Fabien"@fr ; foaf:knows ?f .
```

```
}
```

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
```

```
SELECT ?x WHERE {
```

```
  ?x foaf:name "Fabien"@fr ;
```

```
    foaf:age "21"^^xsd:integer .
```

```
}
```

déclarer un motif optionnel

une partie du patron de graphe n'est pas obligatoire.

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?person ?name
WHERE {
  ?person foaf:homepage <http://fabien.info> .
  OPTIONAL { ?person foaf:name ?name . }
}
```

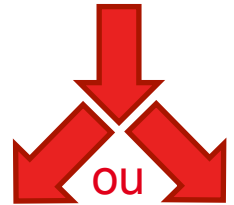


La variable **?name** est potentiellement « unbound »

déclarer deux patrons comme alternatifs

faire l'union des résultats de deux motifs de graphes

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?person ?name
WHERE {
  ?person foaf:name ?name .
  {
    ?person foaf:homepage <http://fabien.info> .
  }
  UNION
  {
    ?person foaf:homepage <http://bafien.org> .
  }
}
```



soustraire un motif des résultats

supprimer des résultats correspondants à un pattern

```
PREFIX ex: <http://www.example.abc#>
```

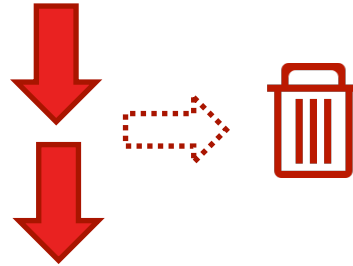
```
SELECT ?x
```

```
WHERE {
```

```
    ?x a ex:Person
```

```
    MINUS { ?x a ex:Man }
```

```
}
```



spécifier des valeurs prédéfinies pour les variables

fournir des résultats ou une partie du *binding* est prédéfinie

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
```

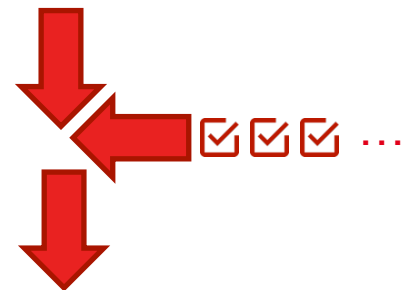
```
SELECT ?person ?name
```

```
WHERE {
```

```
  ?person foaf:name ?name .
```

```
}
```

```
VALUES ?name { "Peter" "Pedro" "Pierre" }
```

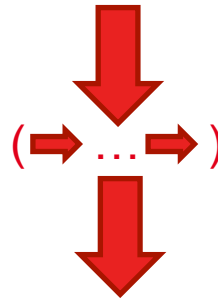


motifs de chemins dans les graphes

des expressions régulières sur des chemins entre ressources

| | | | |
|---|------------------|---|-------------------|
| / | : sequence | | : alternative |
| + | : one or several | * | : zero or several |
| ? | : optional | ^ | : reverse |
| ! | : negation | | |

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?friend WHERE {
  ?x foaf:name "Fabien Gandon" ;
  foaf:knows+ ?friend .
}
```



ne garder que les résultats distincts

un seul exemplaire des réponses ayant les mêmes valeurs pour les mêmes variables

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>  
SELECT DISTINCT ?name  
WHERE { ?person foaf:name ?name . }
```