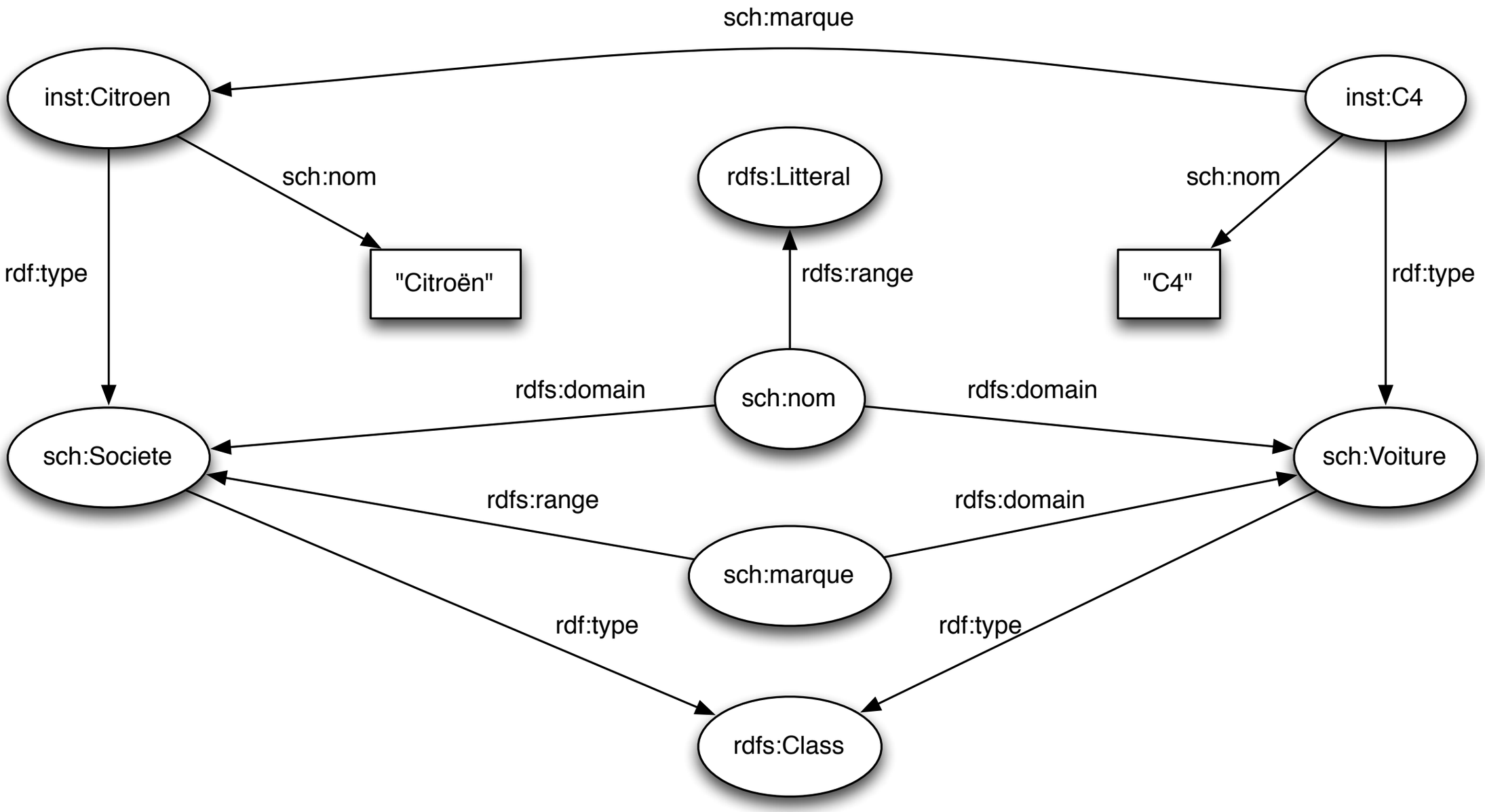
LABD

Master Info M1 2017-2018

LABD 10 : Le langage de requêtes SPARQL

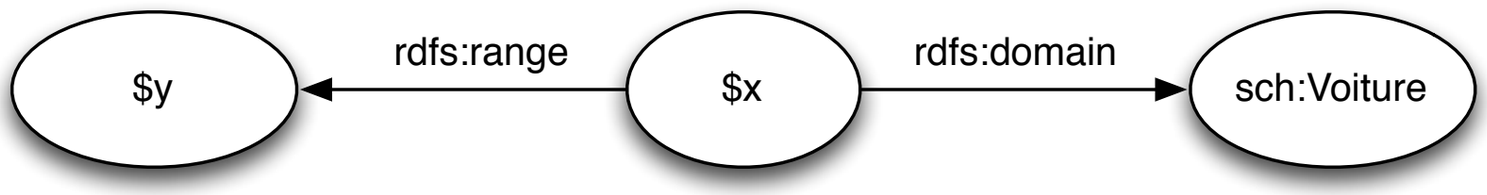
Modèle de graphe



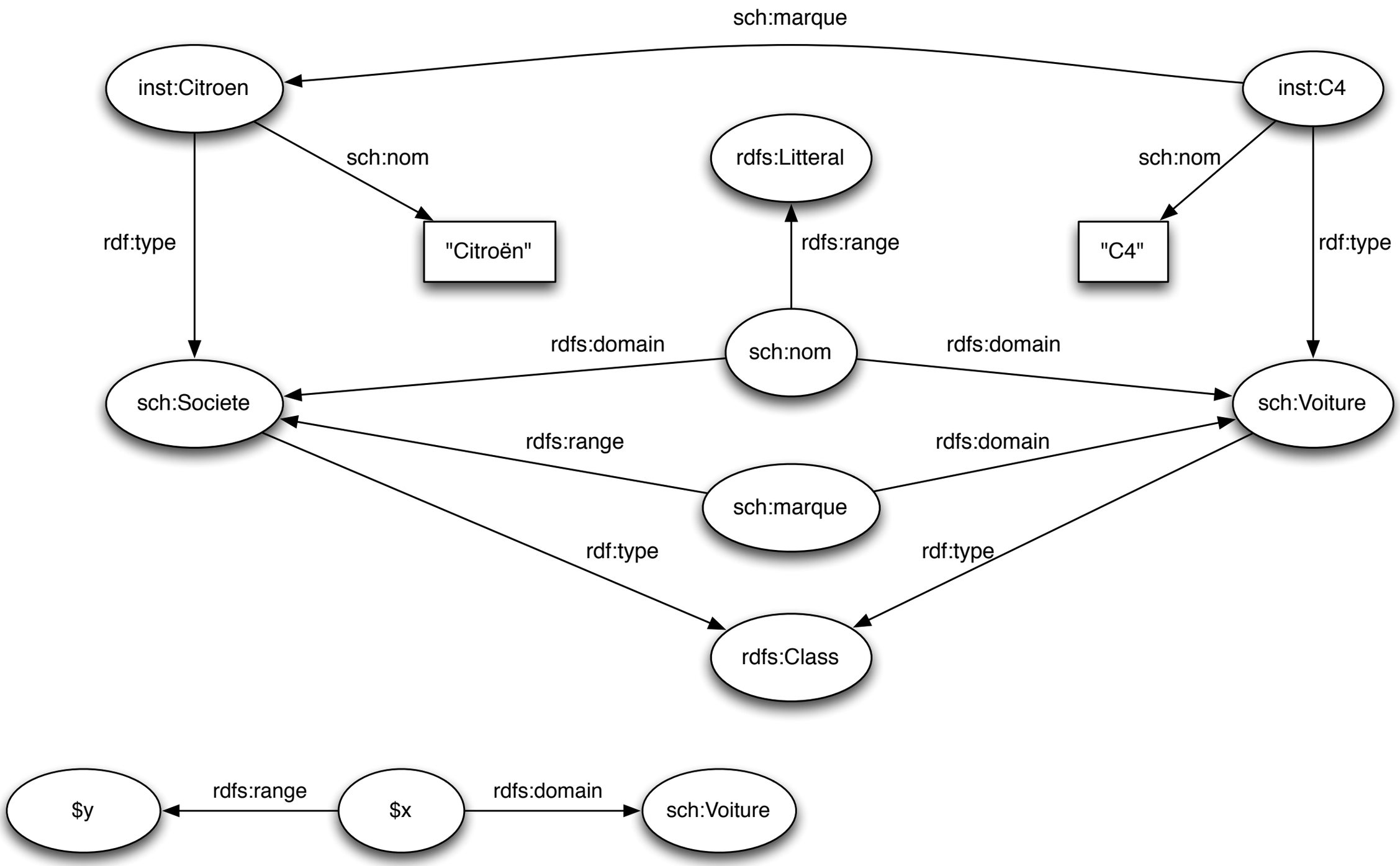
un graphe, réponse à la question 2 de l’exercice 1 du dernier TP

Interroger avec des motifs de graphes

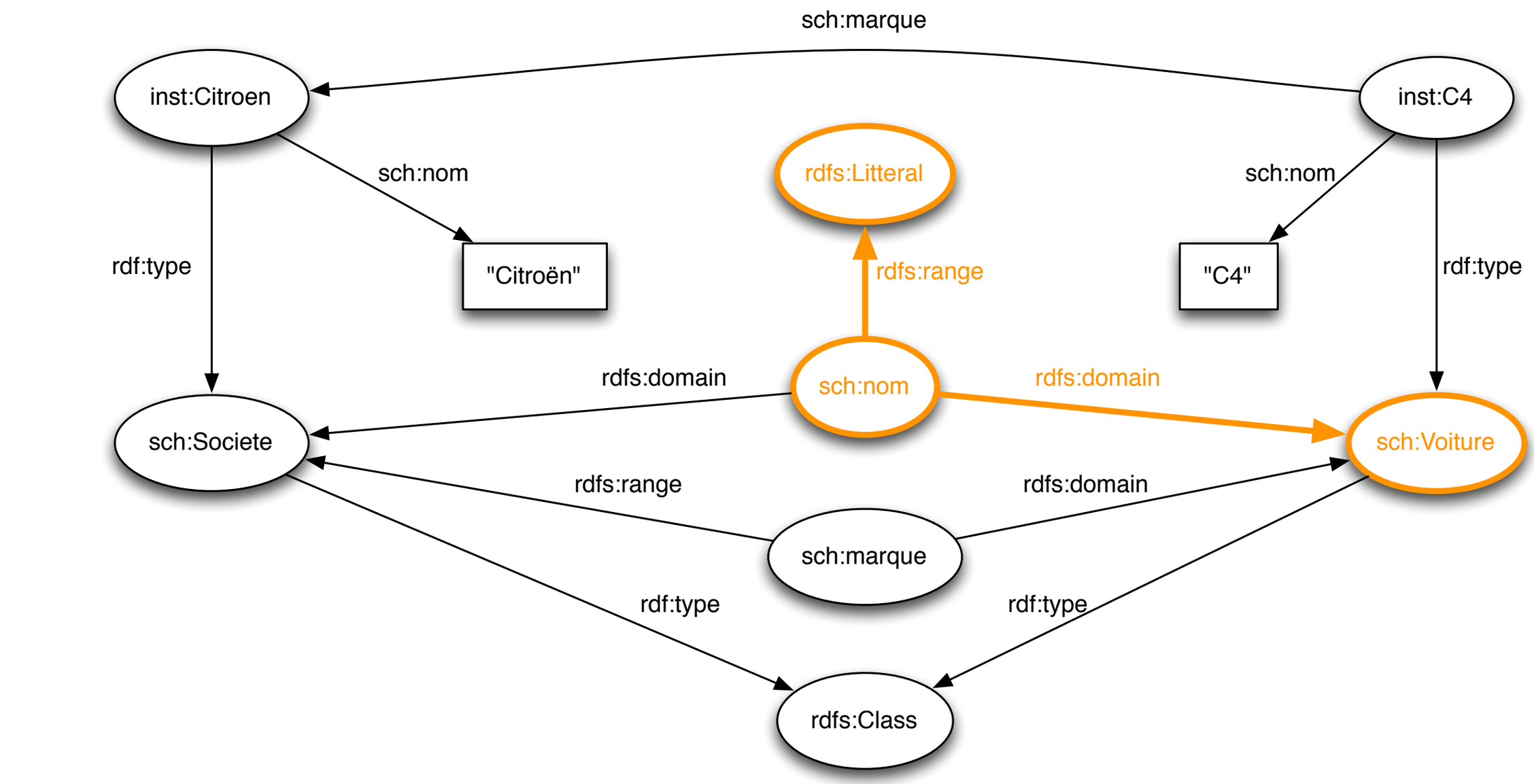
Trouver toutes les propriétés concernant les voitures ainsi que les co-domaines de ces propriétés

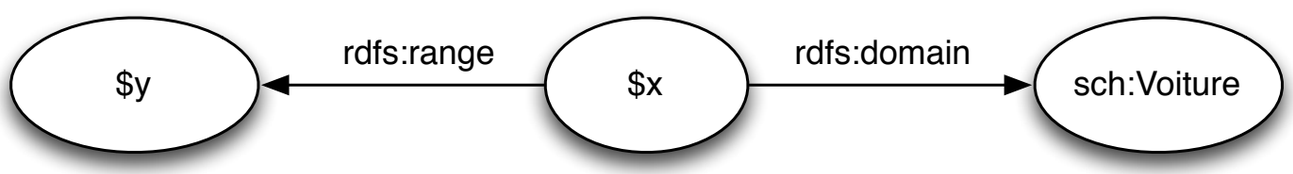


Interroger : motifs de graphes



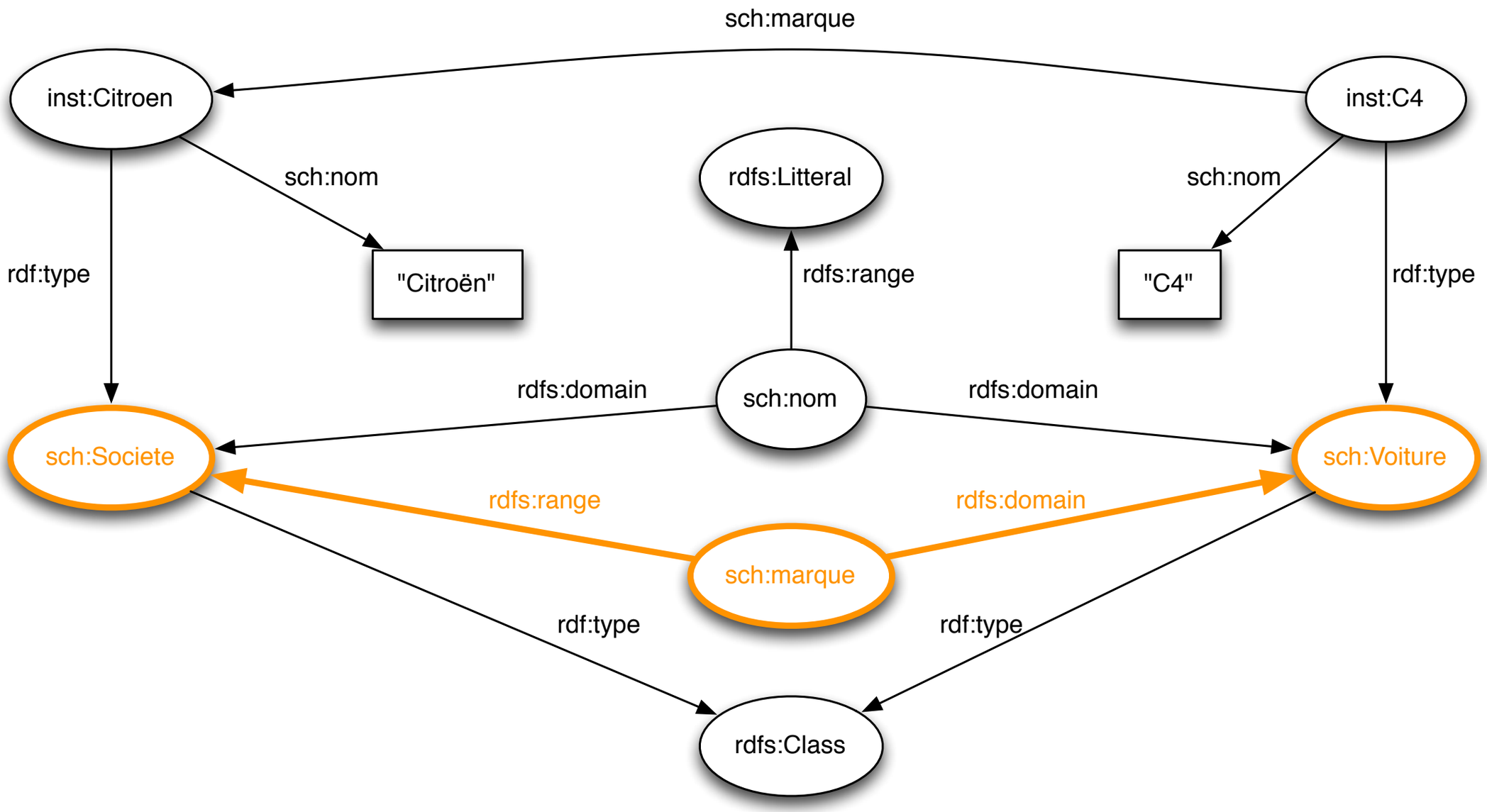
Interroger : motifs de graphes

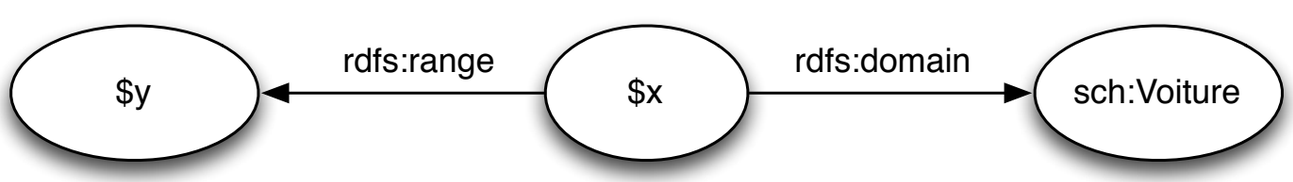





```
$x = sch:nom, $y = rdfs:Litteral
```

Interroger : motifs de graphes

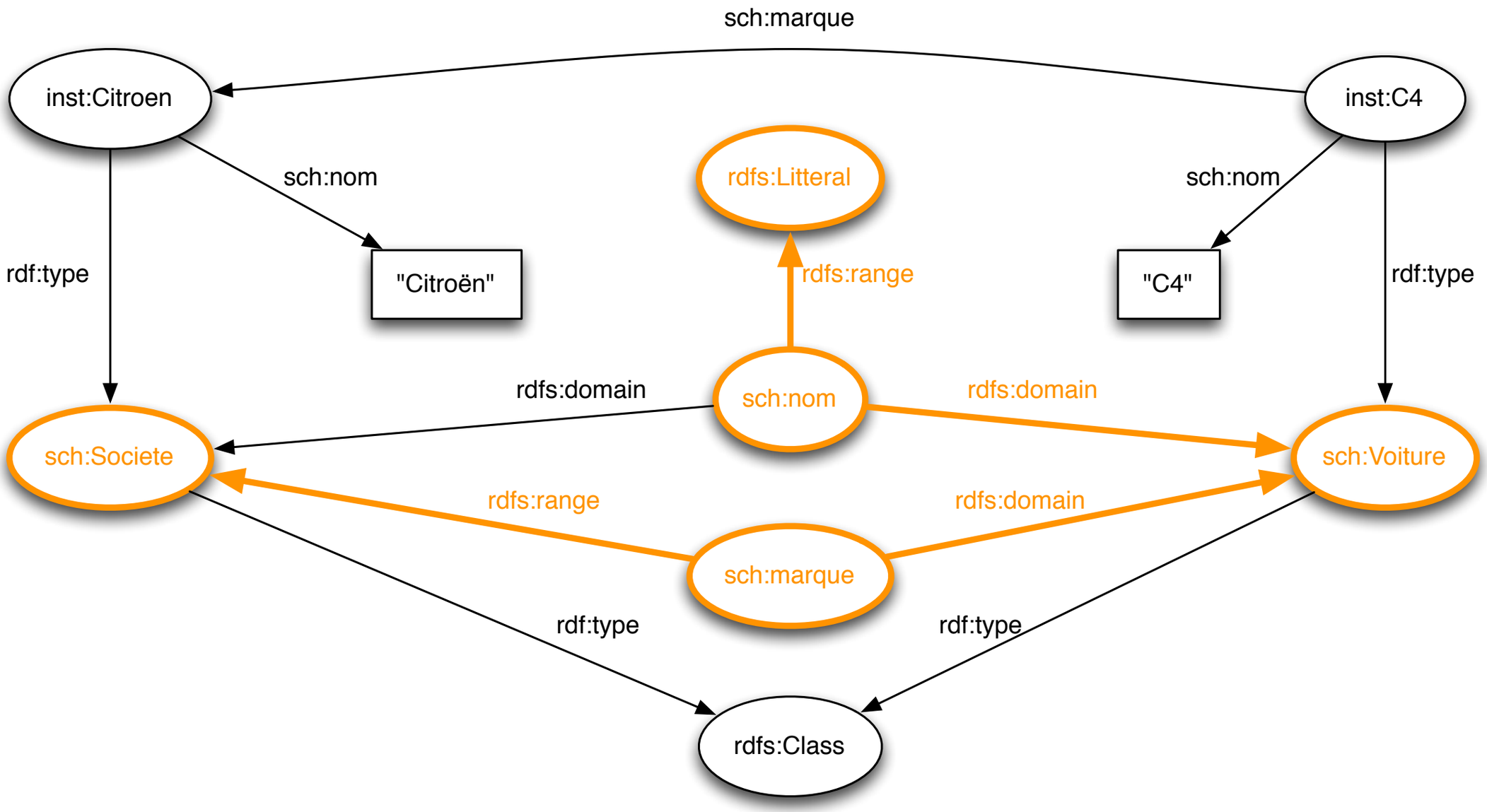


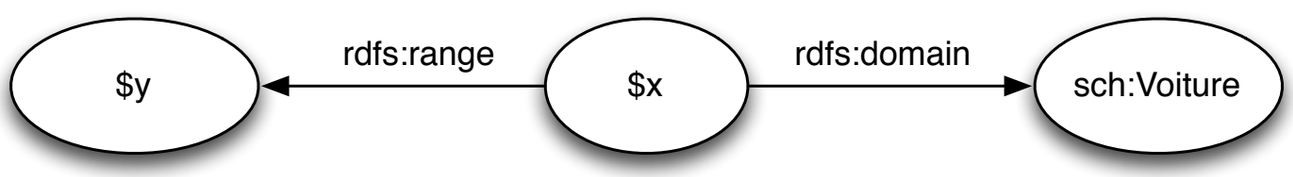



```
$x = sch:nom, $y = rdfs:Litteral

$x = sch:marque, $y = sch:Societe
```

Interroger : motifs de graphes

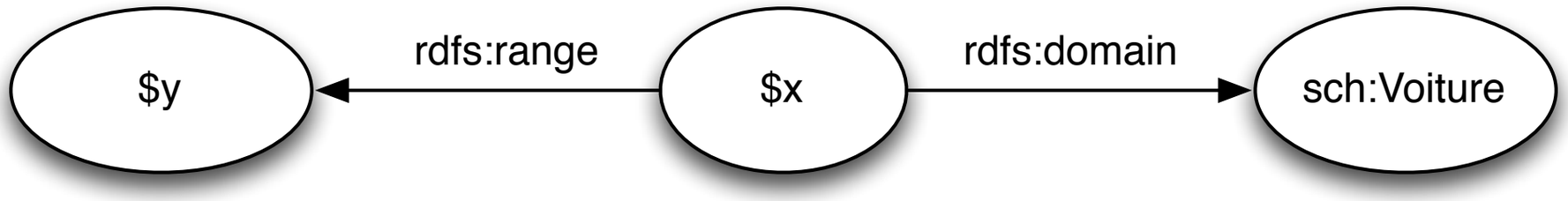




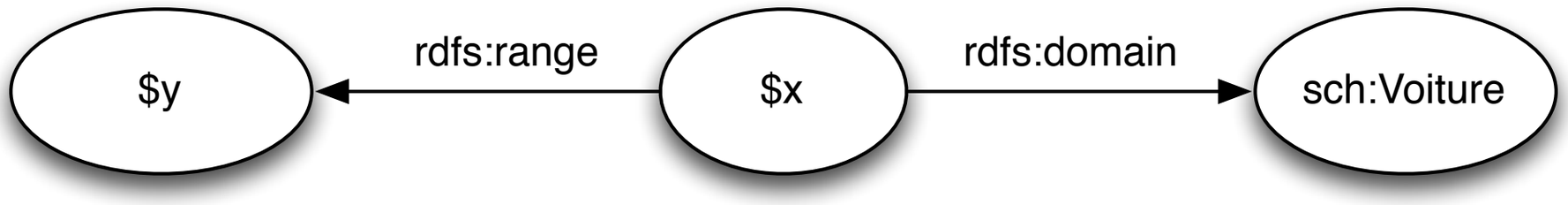

```
$x = sch:nom, $y = rdfs:Litteral

$x = sch:marque, $y = sch:Societe
```

Interroger : motifs de graphes



Interroger : SPARQL




```
SELECT $x $y
WHERE {
  $x rdfs:domain sch:Voiture
  $x rdfs:range $y
}
```

Interroger : SPARQL

```
SELECT $x $y
WHERE {
  $x rdfs:domain sch:Voiture
  $x rdfs:range $y
}
```

Interroger : SPARQL

```
PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
SELECT $x $y
WHERE {
  $x rdfs:domain sch:Voiture
  $x rdfs:range $y
}
```

Interroger : SPARQL

```
PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
SELECT ?x ?y
WHERE {
  ?x rdfs:domain sch:Voiture
  ?x rdfs:range ?y
}
```

variables résultat
liées à des valeurs obtenues
par toutes les façons
possibles de satisfaire le
motif de graphe

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

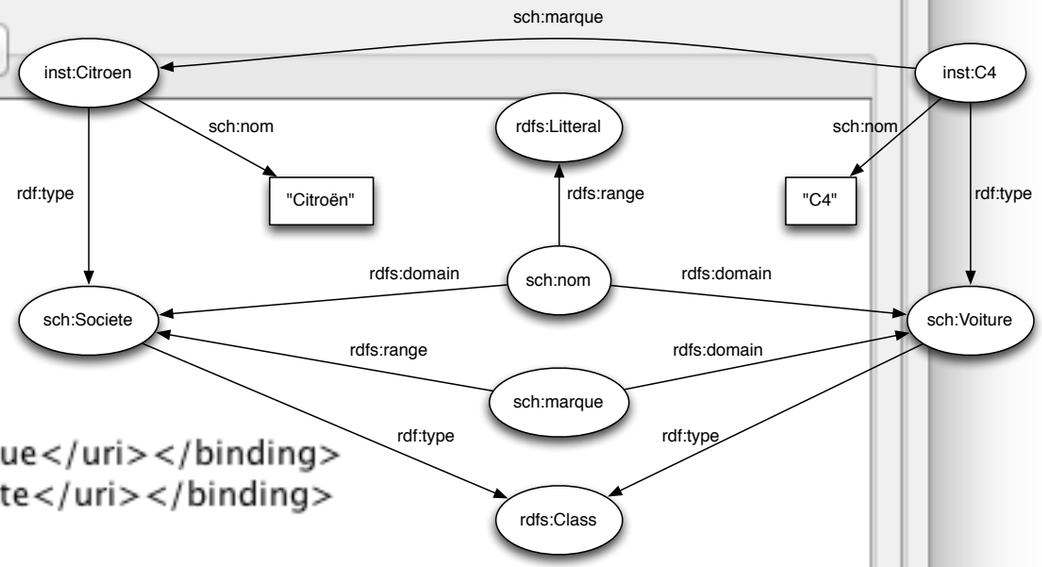
System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT ?x ?y
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?x rdfs:range ?y
6  }
```

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='y'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='y'><uri>http://www.labd.org/2015/exercice-1/schema#Societe</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='y'><uri>http://www.w3.org/2000/01/rdf-schema#Literal</uri></binding>
</result>
</results>
</sparql>
```



Corese/KGRAM 3.1 – Wimmics Inria I3S – 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

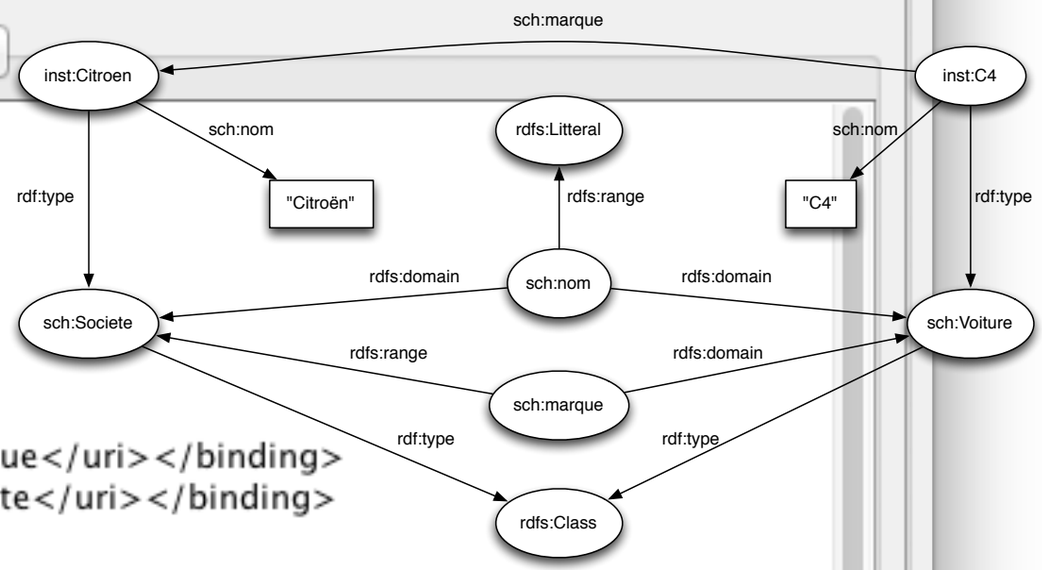
Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT ?x ?y
3  WHERE {
4              ?x rdfs:domain ?z
5              ?x rdfs:range ?y
6  }
```

il peut y avoir plus de variables dans le **`where`** que dans le **`select`**

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='y'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='y'><uri>http://www.labd.org/2015/exercice-1/schema#Societe</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='y'><uri>http://www.w3.org/2000/01/rdf-schema#Literal</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='y'><uri>http://www.w3.org/2000/01/rdf-schema#Literal</uri></binding>
</result>
</results>
```



Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

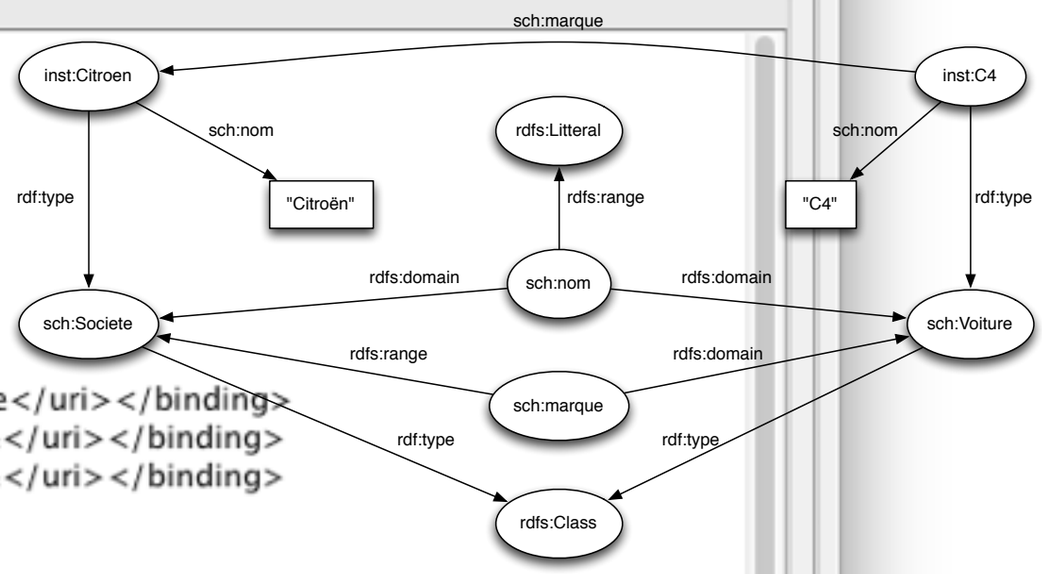
System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT *
3  WHERE {
4              ?x rdfs:domain ?z
5              ?x rdfs:range ?y
6  }
```

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='z'/>
<variable name='y'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/schema#Voiture</uri></binding>
<binding name='y'><uri>http://www.labd.org/2015/exercice-1/schema#Societe</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/schema#Societe</uri></binding>
<binding name='y'><uri>http://www.w3.org/2000/01/rdf-schema#Literal</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/schema#Voiture</uri></binding>
<binding name='y'><uri>http://www.w3.org/2000/01/rdf-schema#Literal</uri></binding>
</result>
</results>
```



Corese/KGRAM 3.1 – Wimmics Inria I3S – 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

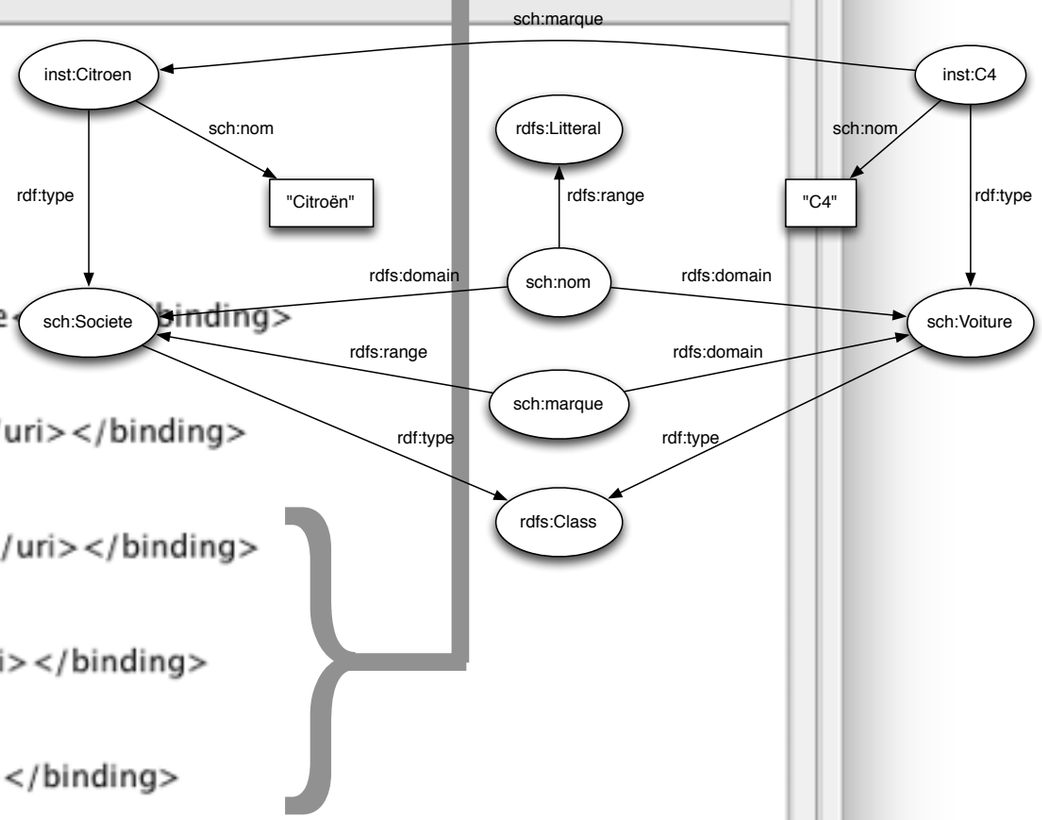
Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT *
3  WHERE {
4              ?x rdf:type rdf:Property
5  }
```

le logiciel charge automatiquement les triplets RDF et RDFS

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.w3.org/1999/02/22-rdf-syntax-ns#type</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.w3.org/2000/01/rdf-schema#domain</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.w3.org/2000/01/rdf-schema#range</uri></binding>
</result>
</results>
</sparql>
```



Graphe interrogé

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ns0: <http://www.labd.org/2015/exercice-1/schema#> .

<http://www.labd.org/2015/exercice-1/schema#Voiture> a rdfs:Class .

<http://www.labd.org/2015/exercice-1/schema#Societe> a rdfs:Class .

<http://www.labd.org/2015/exercice-1/schema#nom>
  rdfs:domain <http://www.labd.org/2015/exercice-1/schema#Voiture>,
    <http://www.labd.org/2015/exercice-1/schema#Societe> ;
  rdfs:range rdfs:Literal .

<http://www.labd.org/2015/exercice-1/schema#marque>
  rdfs:domain <http://www.labd.org/2015/exercice-1/schema#Voiture> ;
  rdfs:range <http://www.labd.org/2015/exercice-1/schema#Societe> .

<http://www.labd.org/2015/exercice-1/instance#C4>
  a <http://www.labd.org/2015/exercice-1/schema#Voiture> ;
  ns0:nom "C4" ;
  ns0:marque <http://www.labd.org/2015/exercice-1/instance#Citroen> .

<http://www.labd.org/2015/exercice-1/instance#Citroen>
  a ns0:Societe ;
  ns0:nom "Citroën" .
```

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT *
3  WHERE {
4              ?x rdf:type rdf:Property
5  }
```

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.w3.org/1999/02/22-rdf-syntax-ns#type</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.w3.org/2000/01/rdf-schema#domain</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.w3.org/2000/01/rdf-schema#range</uri></binding>
</result>
</results>
</sparql>
```

obtenus par inférence

SPARQL

SPARQL 1.0 : 2008

SPARQL 1.1 : 2013

- expressions possibles dans le `SELECT`
- opérateurs de négation
- agrégats (`COUNT` , `SUM`, …, `GROUP BY`,…)
- sous-requêtes dans la partie `WHERE` pour définir des nouvelles variables
- expressions régulières de chemins pour les motifs de graphes

Structure d'une requête SPARQL

1. un prologue optionnel
   (`PREFIX`,…)
2. une section qui définit la sortie
   (`SELECT ?x`,…)
3. une section optionnelle qui définit les sources de données à interroger
   (`FROM`)
4. une section qui définit les contraintes à vérifier par les variables
   (`WHERE`)
5. une section optionnelle qui définit l'ordre des résultats en sortie
   (`ORDER BY`)
6. une section optionnelle de regroupement de résultats
   (`GROUP BY`)
7. une section optionnelle qui précise un intervalle de sélection de résultats
   (`LIMIT`, `OFFSET`)

1. Prologue : `base  prefix`

```
PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
SELECT ?x ?y
WHERE {
   ?x rdfs:domain sch:Voiture
   ?x rdfs:range ?y
}
```

```
PREFIX : <http://www.labd.org/2015/exercice-1/schema#>
SELECT ?x ?y
WHERE {
   ?x rdfs:domain :Voiture
   ?x rdfs:range ?y
}
```

```
BASE <http://www.labd.org/2015/exercice-1/schema>
SELECT ?x ?y
WHERE {
   ?x rdfs:domain <#Voiture>
   ?x rdfs:range ?y
}
```

1. Prologue : `base  prefix`

```
PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
SELECT ?x ?y
WHERE {
   ?x rdfs:domain sch:Voiture
   ?x rdfs:range ?y
}
```

```
PREFIX : <http://www.labd.org/2015/exercice-1/schema#>
SELECT ?x ?y
WHERE {
   ?x rdfs:domain :Voiture
   ?x rdfs:range ?y
}
```

```
BASE <http://www.labd.org/2015/exercice-1/schema>
SELECT ?x ?y
WHERE {
   ?x rdfs:domain <#Voiture>
   ?x rdfs:range ?y
}
```

Structure d’une requête SPARQL

1. un prologue optionnel
   (`PREFIX`,…)
2. **une section qui définit la sortie**
   (`SELECT ?x`,…)
3. une section optionnelle qui définit les sources de données à interroger
   (`FROM`)
4. une section qui définit les contraintes à vérifier par les variables
   (`WHERE`)
5. une section optionnelle qui définit l’ordre des résultats en sortie
   (`ORDER BY`)
6. une section optionnelle de regroupement de résultats
   (`GROUP BY`)
7. une section optionnelle qui précise un intervalle de sélection de résultats
   (`LIMIT`, `OFFSET`)

2. Définition de la sortie

- `SELECT [DISTINCT] ?x1 ?x2 …`
  `SELECT [DISTINCT] *`
  `SELECT (`expression d'agrégation`)`
  projection (comme en relationnel) ou agrégation

- `ASK`
  requête booléenne, répond à « le motif de graphe est-il satisfiable? »

- `CONSTRUCT {` graphe donné par des triplets `}`
  retourne un graphe RDF

- `DESCRIBE` IRI ou variables
  donne une description sous forme de graphe RDF des ressources correspondantes

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

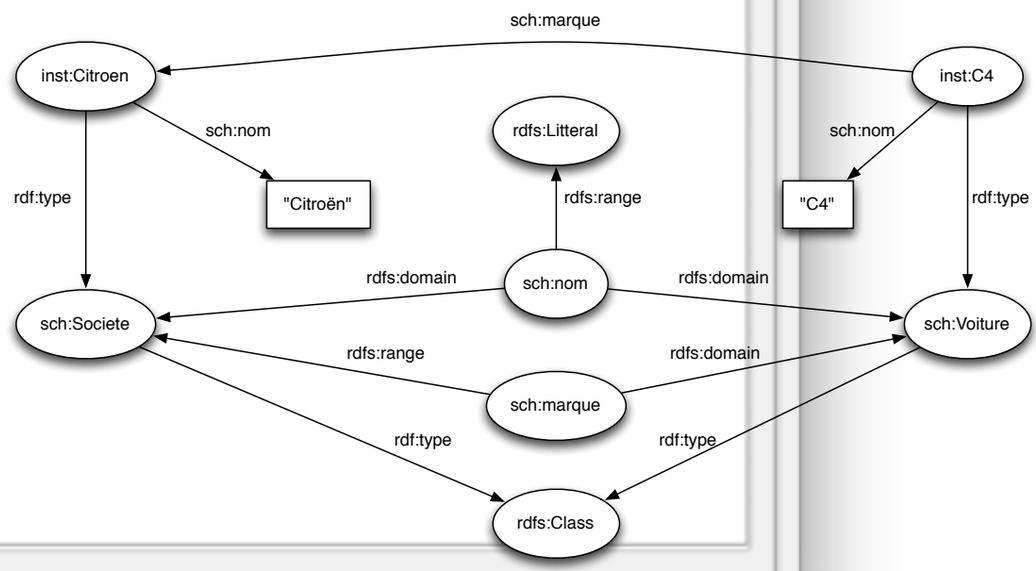
Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT (COUNT(?x) AS ?nx)
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?x rdfs:range ?y
6  }
```

On doit obligatoirement ranger la valeur dans une variable

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='nx'/>
</head>
<results>
<result>
<binding name='nx'><literal datatype='http://www.w3.org/2001/XMLSchema#integer'>2</literal></binding>
</result>
</results>
</sparql>
```



Corese/KGRAM 3.1 – Wimmics Inria I3S – 2014-01-29

File Edit Engine Debug ?

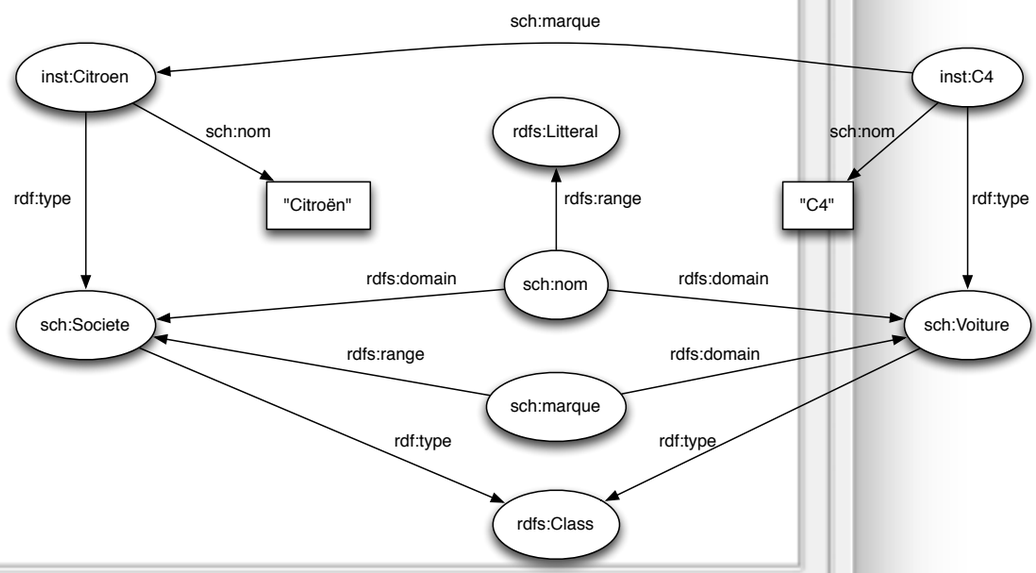
System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  ASK
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?x rdfs:range ?y
6  }
```

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
</head>
<boolean>true</boolean>
</sparql>
```



Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

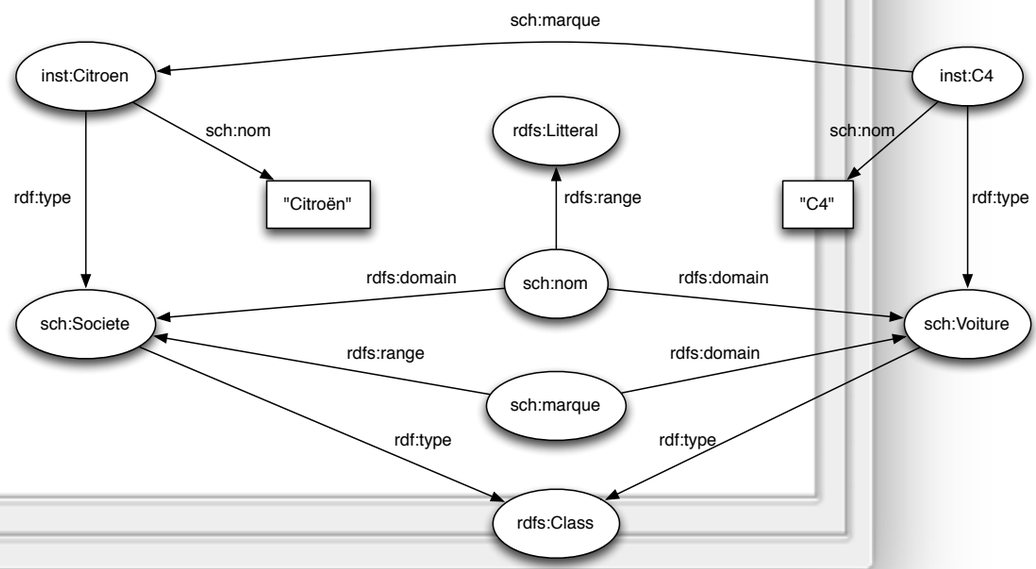
System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  ASK {
3              ?x rdfs:domain sch:Voiture
4              ?x rdfs:range ?y
5  }
```

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
</head>
<boolean>true</boolean>
</sparql>
```



Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

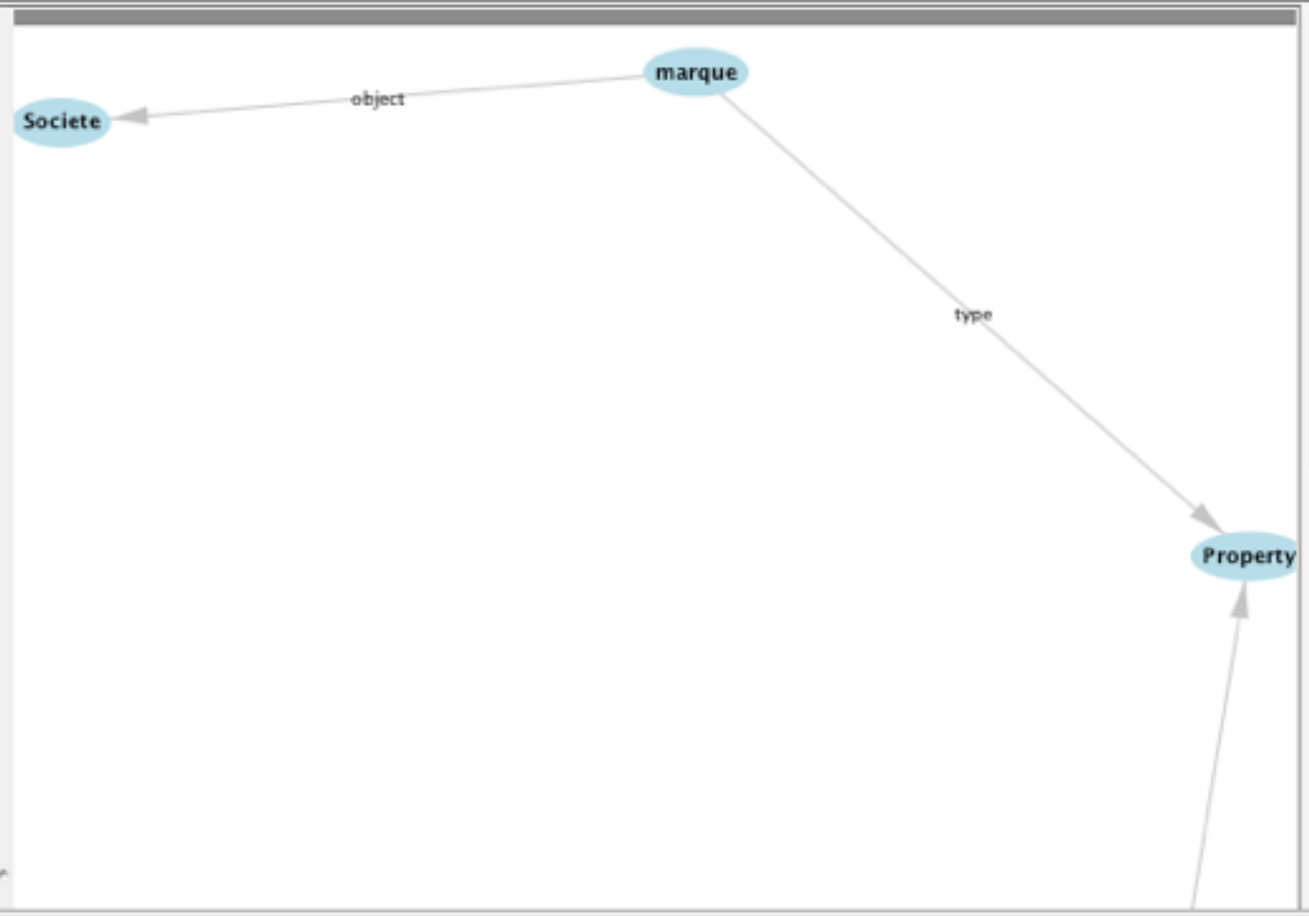
System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1
2  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
3  PREFIX new: <http://www.labd.org/2015/construct#>
4  CONSTRUCT {
4                  ?x new:object ?y
5                  ?x new:type ?z
6  }
7  WHERE {
8                  ?x rdfs:domain sch:Voiture
9                  ?x rdfs:range ?y
10                 ?x rdf:type ?z
11 }
```

Graph | XML | Validate

```
1  graph {
2                  color:grey;
3  }
4  node {
5                  text-size:9;
6                  text-color:black;
7                  text-style:bold;
8                  text-align:center;
9                  width:17;
10                 color:lightblue;
11                 node-shape:text-ellipse;
12 }
13 node.Literal {
14                 text-size:9;
15                 text-color:black;
16                 text-style:bold;
17                 text-align:center;
18                 width:17;
19                 color:orange;
20                 node-shape:text-box;
21 }
22 node.Blank {
23                 text-size:9;
24                 text-color:black;
25
26
```



Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

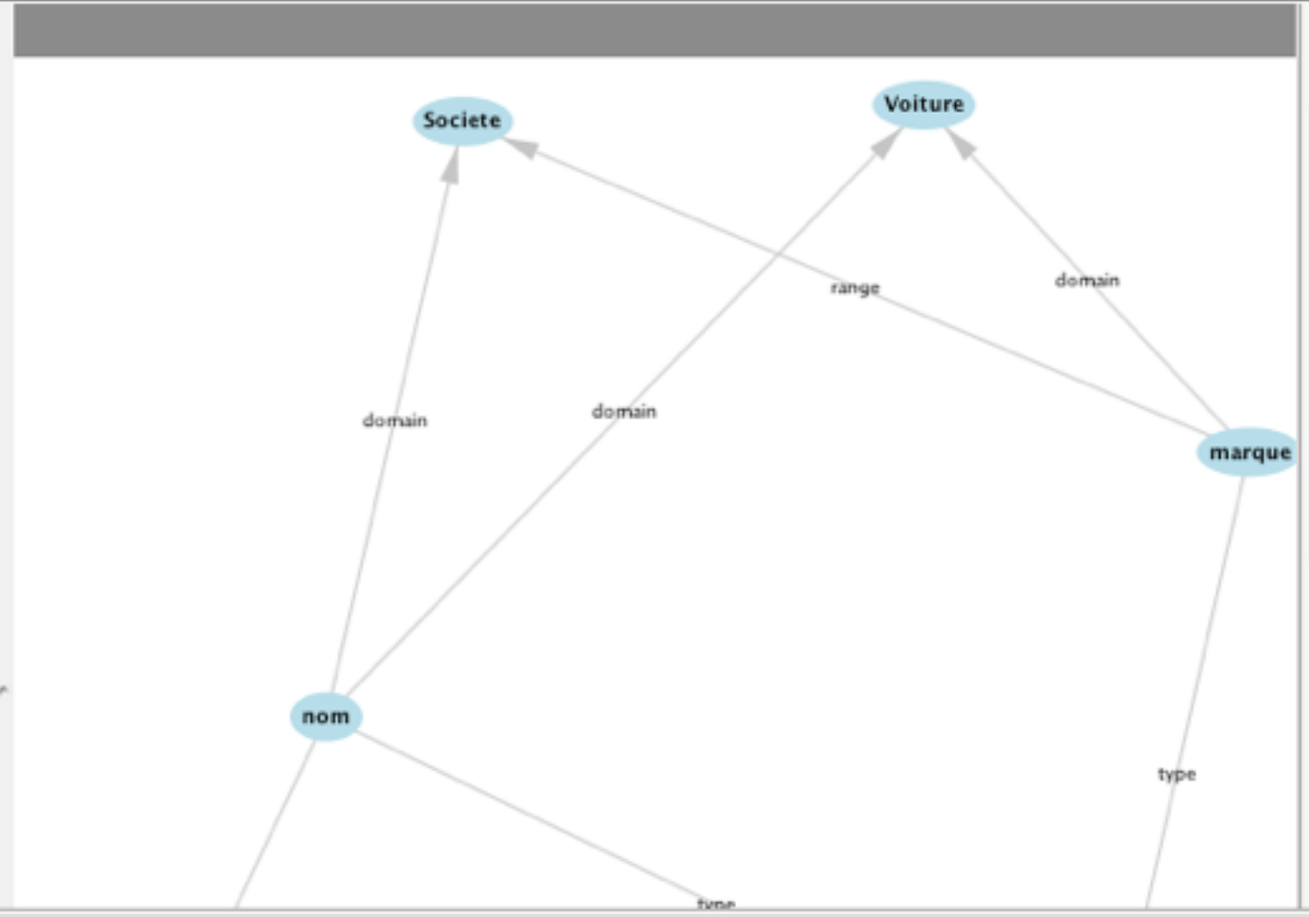
System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  DESCRIBE ?x
3  WHERE {
4                  ?x rdfs:domain sch:Voiture
5                  ?x rdfs:range ?y
6  }
```

Graph | XML | Validate

```
6                  text-color:black;
7                  text-style:bold;
8                  text-align:center;
9                  width:17;
10                 color:lightblue;
11                 node-shape:text-ellipse;
12 }
13 node.Literal {
14                 text-size:9;
15                 text-color:black;
16                 text-style:bold;
17                 text-align:center;
18                 width:17;
19                 color:orange;
20                 node-shape:text-box;
21 }
22 node.Blank {
23                 text-size:9;
24                 text-color:black;
25                 text-style:bold;
26                 text-align:center;
27                 width:17;
28                 color:yellow;
29                 node-shape:text-ellipse;
30
31
```



Structure d'une requête SPARQL

1. un prologue optionnel
   (`PREFIX`,…)
2. une section qui définit la sortie
   (`SELECT ?x`,…)
3. **une section optionnelle qui définit les sources de données à interroger**
   (`FROM`)
4. une section qui définit les contraintes à vérifier par les variables
   (`WHERE`)
5. une section optionnelle qui définit l'ordre des résultats en sortie
   (`ORDER BY`)
6. une section optionnelle de regroupement de résultats
   (`GROUP BY`)
7. une section optionnelle qui précise un intervalle de sélection de résultats
   (`LIMIT`, `OFFSET`)

3. Sources de données à interroger : **`FROM`** uri

```
PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
SELECT ?x ?y
FROM <file:///Users/yroos/Documents/fil/labd/labd-11/ex1.xml>
FROM <http://www.w3.org/2000/01/rdf-schema>
WHERE {
   ?x rdfs:domain sch:Voiture
   ?x rdfs:range ?y
}
```

Structure d'une requête SPARQL

1. un prologue optionnel
   (`PREFIX`,…)
2. une section qui définit la sortie
   (`SELECT ?x`,…)
3. une section optionnelle qui définit les sources de données à interroger
   (`FROM`)
4. **une section qui définit les contraintes à vérifier par les variables**
   (`WHERE avec UNION, FILTER, OPTIONAL, MINUS, NOT EXIST`)
5. une section optionnelle qui définit l'ordre des résultats en sortie
   (`ORDER BY`)
6. une section optionnelle de regroupement de résultats
   (`GROUP BY`)
7. une section optionnelle qui précise un intervalle de sélection de résultats
   (`LIMIT`, `OFFSET`)

4. Contraintes à vérifier par les variables : `WHERE`

```
PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
SELECT ?x ?y
WHERE {
   ?x rdfs:domain sch:Voiture
   ?x rdfs:range ?y
}
```

juxtaposition de triplets = conjonction

4. Disjonction de motifs : `UNION`

```
PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
SELECT ?x ?y
WHERE {
    {
        ?x rdfs:domain sch:Voiture
        ?x rdfs:range ?y
    } UNION {
        ?x rdfs:domain sch:Societe
        ?x rdfs:range ?y
    }
}
```

groupes de motifs

Il s’agit bien d’une union ensembliste

4. Ajouter des contraintes : `FILTER`

`FILTER (` expression booléenne où apparaissent des variables `)`
`FILTER regex(` variable , expression régulière `)`

- ne garde que les liaisons de variables qui satisfont le filtre

- la portée du filtre est le groupe de motif où apparaît le filtre

- on dispose des opérateurs booléens (`|| && !`), des comparaisons (`= != < <= > >=`), des opérateurs arithmétiques (`+ - * /`)

- on dispose de fonctions spécifiques : `isURI(?x)`, `isBlank(?x)`, `isLiteral(?x)`, `bound(?x)`

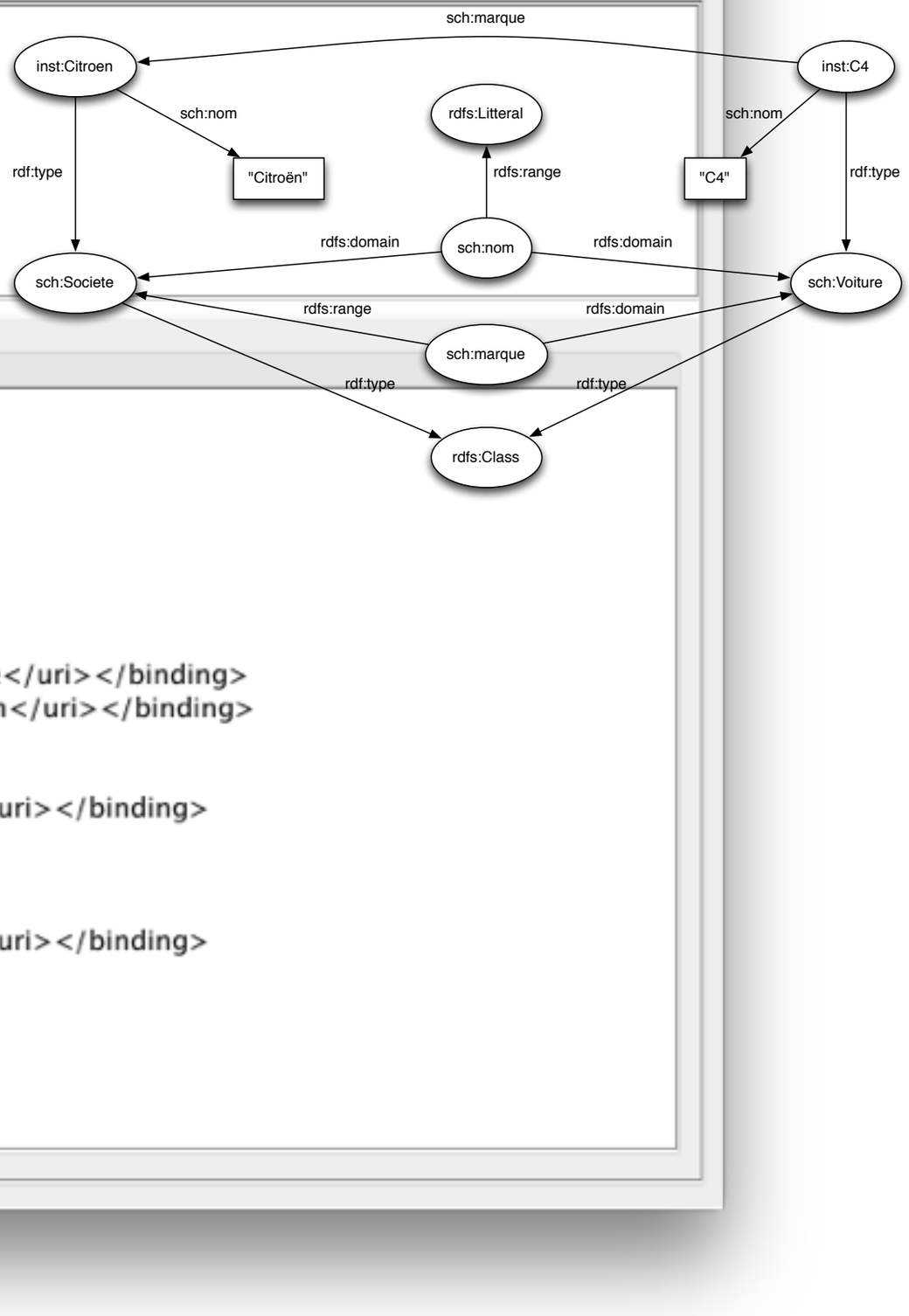
Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System Query1 × +

Query to SPIN to SPARQL Prove Trace Search Refresh stylesheet Default stylesheet

```
1  PREFIX sch:<http://www.labd.org/2015/exercice-1/schema#>
2  SELECT ?x ?z
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?v ?x ?z
6  }
```



Graph XML Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='z'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/instance#Citroen</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>Citroën</literal></binding>
</result>
</results>
</sparql>
```

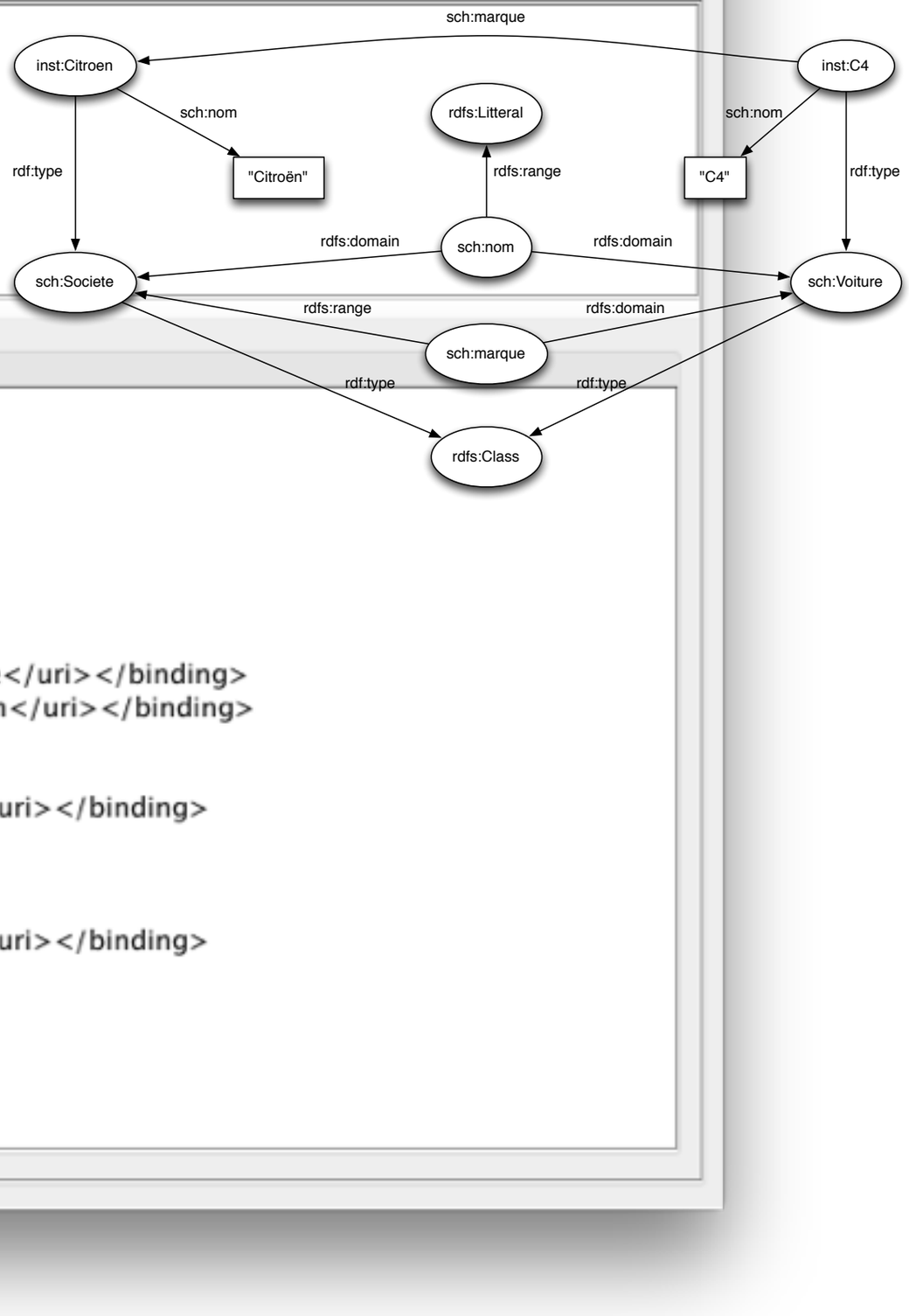
Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch:<http://www.labd.org/2015/exercice-1/schema#>
2  SELECT ?x ?z
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?v ?x ?z
6              FILTER regex(?z , "C.*")
7  }
```



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='z'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/instance#Citroen</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>Citroën</literal></binding>
</result>
</results>
</sparql>
```

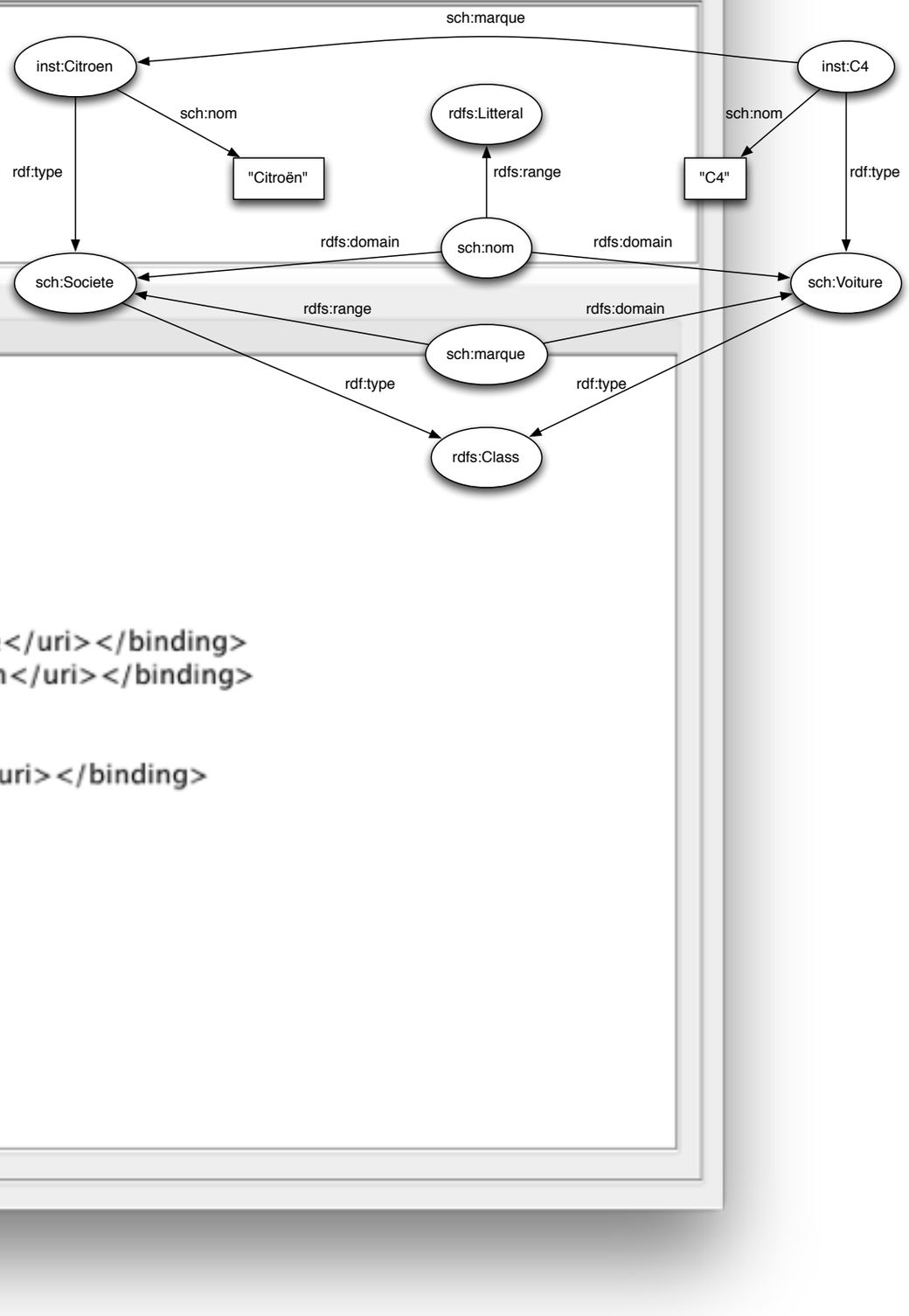
Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | Query2 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch:<http://www.labd.org/2015/exercice-1/schema#>
2  SELECT ?x ?z
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?v ?x ?z
6              FILTER regex(?z,"Ci.*")
7  }
```



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='z'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/instance#Citroen</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>Citroën</literal></binding>
</result>
</results>
</sparql>
```

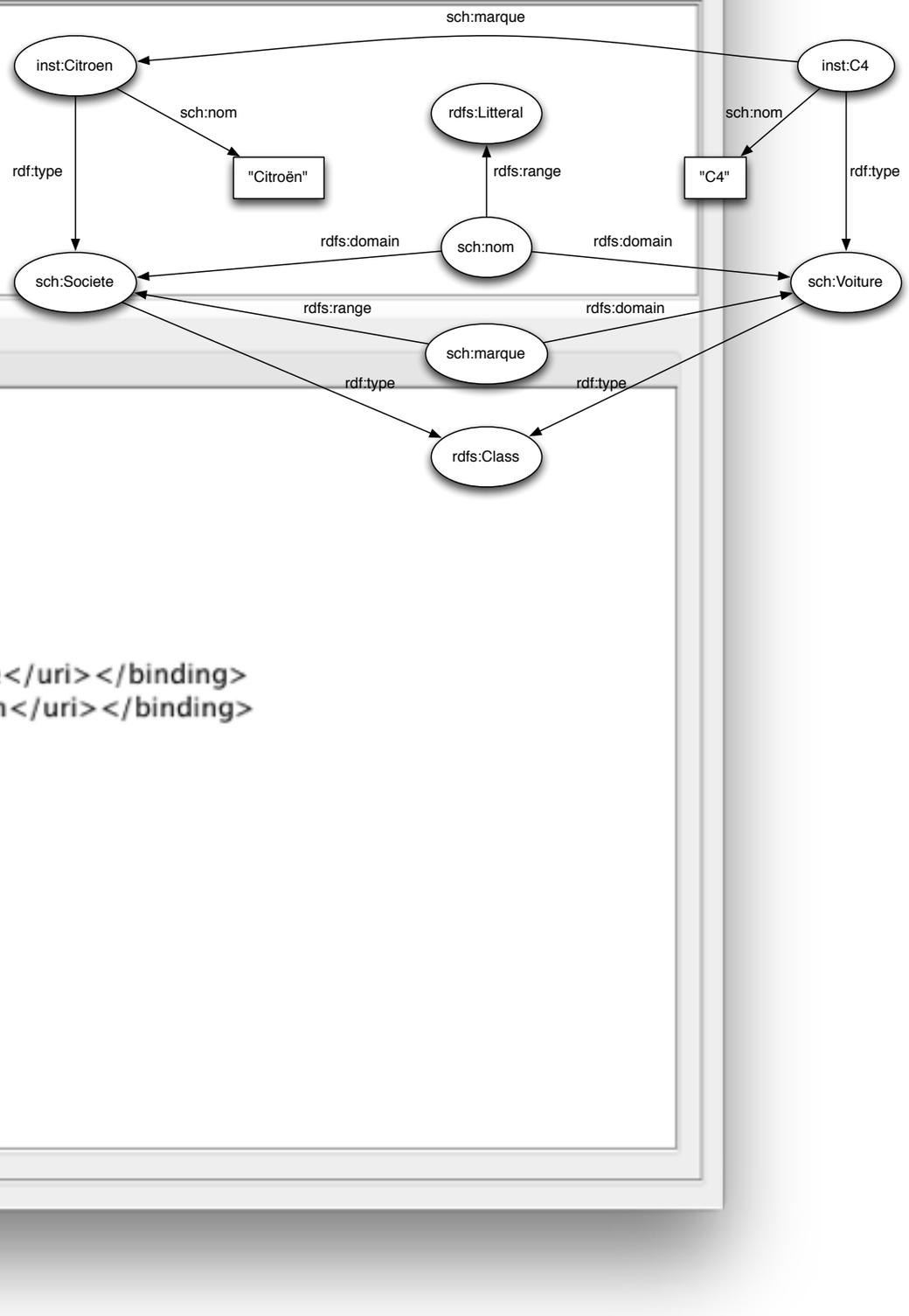
Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch:<http://www.labd.org/2015/exercice-1/schema#>
2  SELECT ?x ?z
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?v ?x ?z
6              FILTER isURI(?z)
7  }
```



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='z'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/instance#Citroen</uri></binding>
</result>
</results>
</sparql>
```

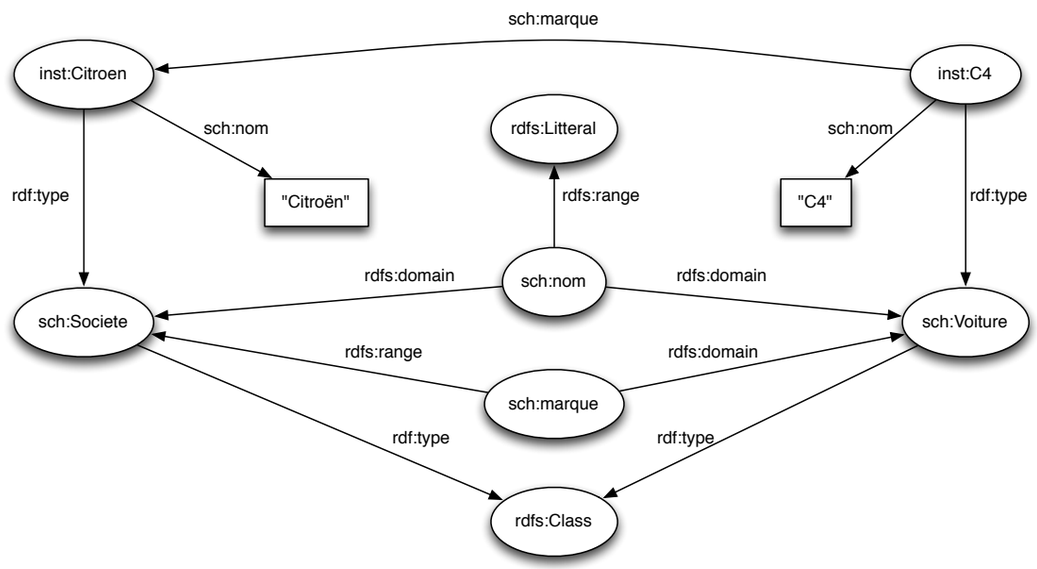
Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT distinct ?x ?v
3  WHERE
4  {
5     ?x rdfs:domain sch:Voiture
6     ?z ?x ?v
7     FILTER (isLiteral(?v))
8  }
```



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='v'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>Citro&#65533;n</literal></binding>
</result>
</results>
</sparql>
```

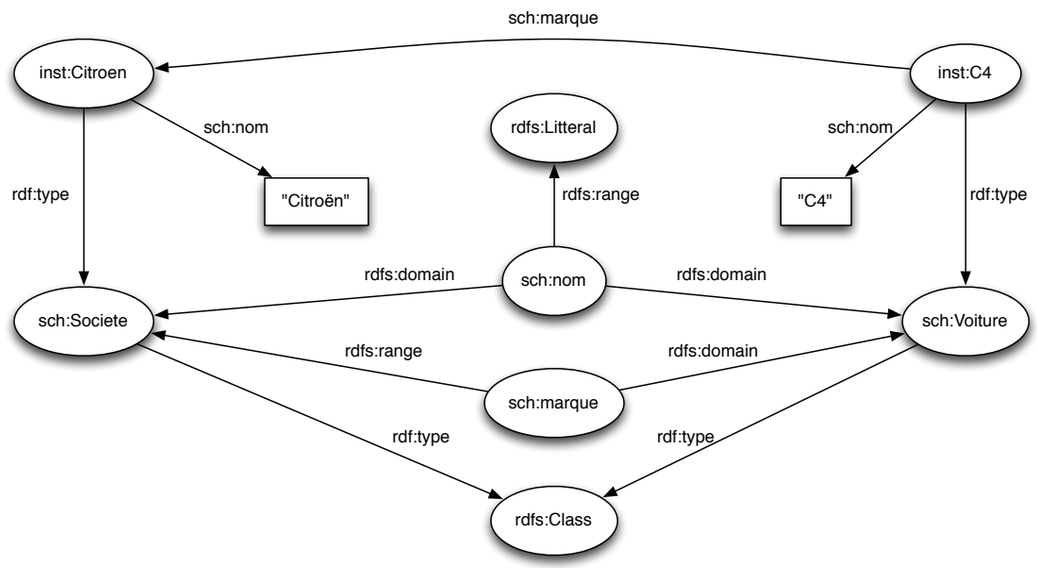
Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT distinct ?x ?v
3  WHERE
4  {
5     ?x rdfs:domain sch:Voiture
6     ?z ?x ?v
7     ?x rdfs:range ?y
8     FILTER (?y = rdfs:Literal)
9  }
```



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='v'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>Citro&#65533;n</literal></binding>
</result>
</results>
</sparql>
```

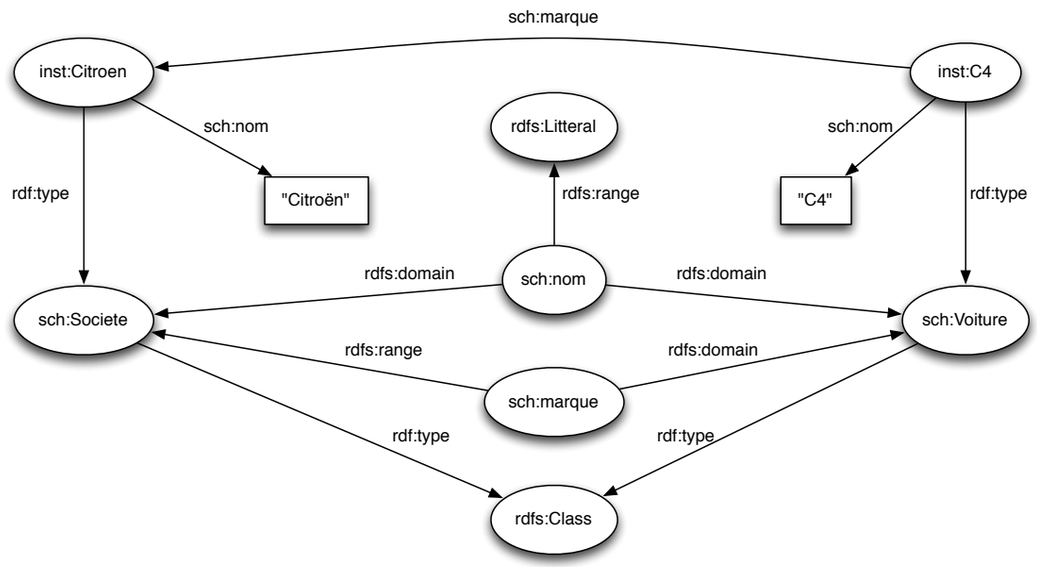
Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT distinct ?x ?v
3  WHERE
4  {
5    {
6      ?x rdfs:domain sch:Voiture
7      ?z ?x ?v
8    } UNION {
9      ?x rdfs:domain sch:Societe
10     ?z ?x ?v
11   }
12 }
```



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='v'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='v'><uri>http://www.labd.org/2015/exercice-1/instance#Citroen</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>Citro&#65533;n</literal></binding>
</result>
</results>
</sparql>
```

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

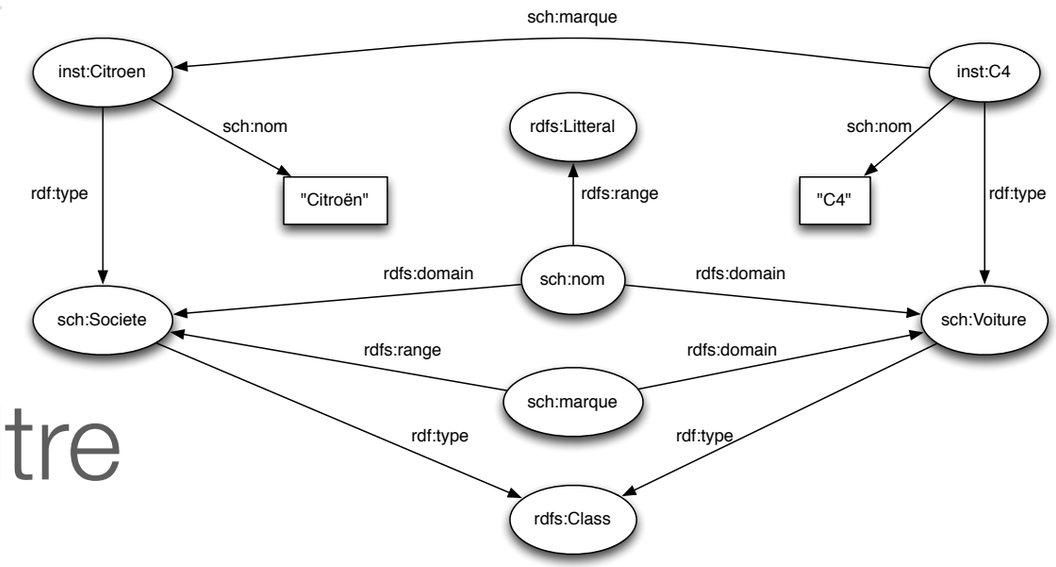
File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT distinct ?x ?v
3  WHERE
4  {
5    {
6      ?x rdfs:domain sch:Voiture
7      ?z ?x ?v
8    } UNION {
9      ?x rdfs:domain sch:Societe
10     ?z ?x ?v
11     FILTER (isLiteral(?v))
12   }
13 }
```

portée du filtre



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='v'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='v'><uri>http://www.labd.org/2015/exercice-1/instance#Citroen</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>Citro&#65533;n</literal></binding>
</result>
</results>
</sparql>
```

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

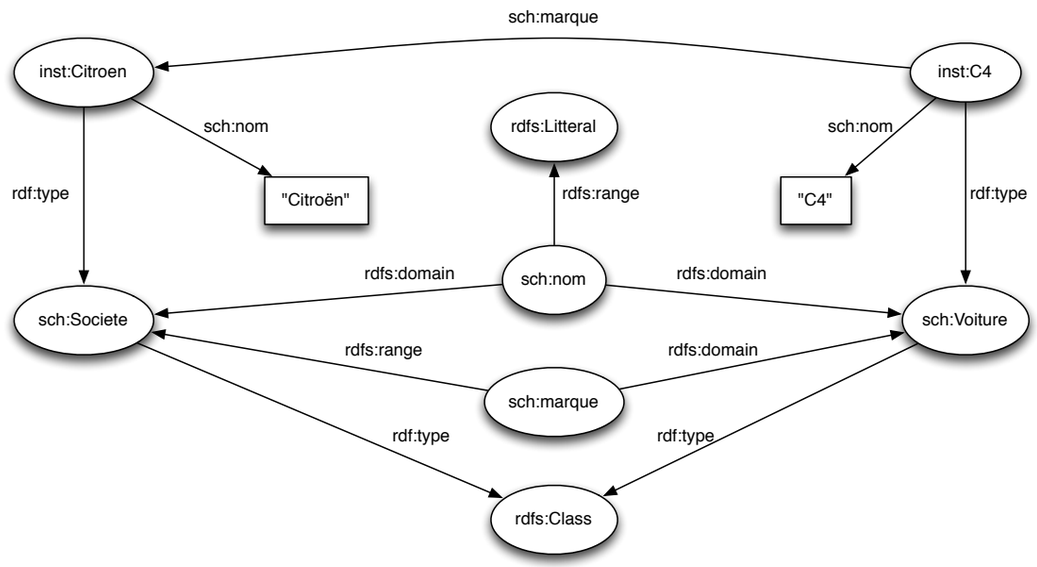
File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT distinct ?x ?v
3  WHERE
4  {
5    {
6      ?x rdfs:domain sch:Voiture
7      ?z ?x ?v
8    } UNION {
9      ?x rdfs:domain sch:Societe
10     ?z ?x ?v
11   }
12   FILTER (isLiteral(?v))
13 }
```

portée du filtre



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='v'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>Citro&#65533;n</literal></binding>
</result>
</results>
</sparql>
```

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

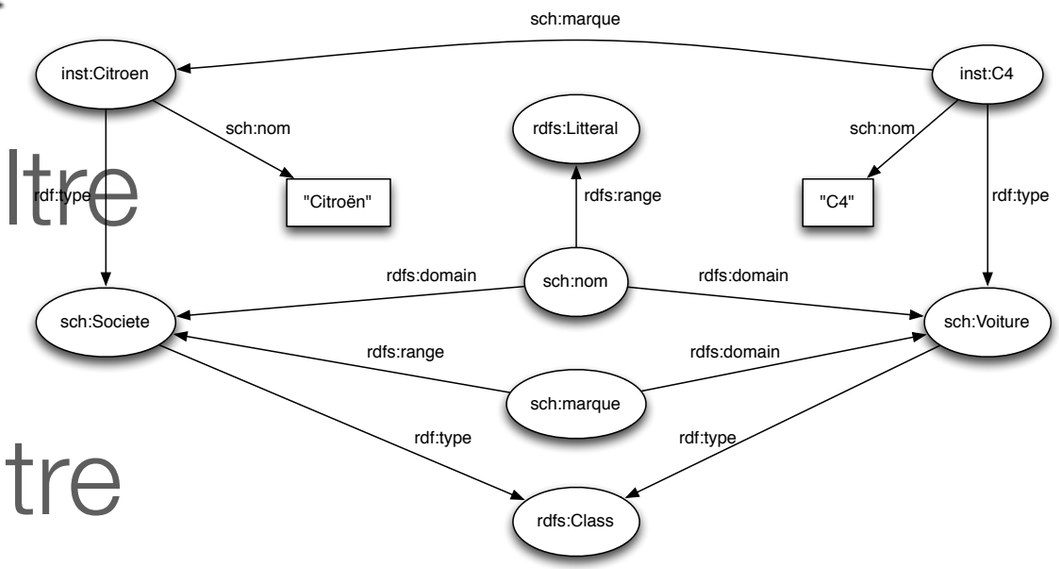
System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT distinct ?x ?v
3  WHERE
4  {
5    {
6      ?x rdfs:domain sch:Voiture
7      ?z ?x ?v
8      FILTER (isLiteral(?v))
9    } UNION {
10     ?x rdfs:domain sch:Societe
11     ?z ?x ?v
12     FILTER (isLiteral(?v))
13   }
14 }
```

portée du filtre

portée du filtre



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='v'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>Citro&#65533;n</literal></binding>
</result>
</results>
</sparql>
```

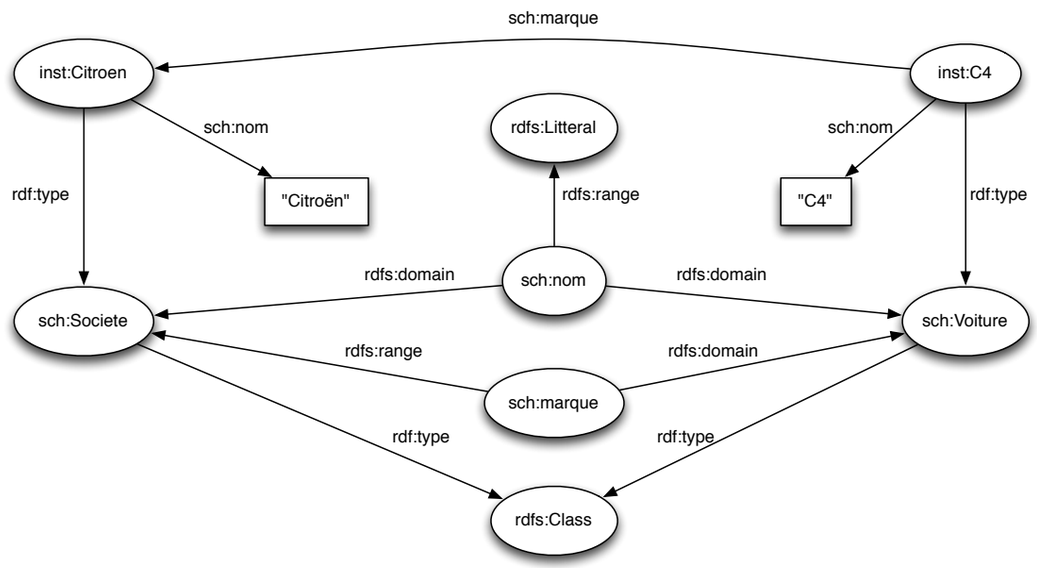
Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT distinct ?x ?v
3  WHERE
4  {
5    {
6      ?x rdfs:domain sch:Voiture
7      FILTER (isLiteral(?v))
8      ?z ?x ?v
9    } UNION {
10     FILTER (isLiteral(?v))
11     ?x rdfs:domain sch:Societe
12     ?z ?x ?v
13   }
14 }
```



Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='v'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='v'><literal>Citro&#65533;n</literal></binding>
</result>
</results>
</sparql>
```

4. Relâcher des contraintes : `OPTIONAL`

- Normalement, quand une partie de graphe satisfait un motif alors toutes les variables du motif sont liées à des valeurs.

- On peut définir des parties optionnelles dans le motif avec la clause `OPTIONAL`

- Les motifs apparaissant dans cette clause peuvent ne pas être satisfaits par une solution. En particulier certaines variables apparaissant dans la clause `OPTIONAL` peuvent n'être liées à aucune valeur dans une solution.

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT DISTINCT ?x ?z
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?v ?x ?z
6              OPTIONAL {?z ?s ?t}
7  }
```

Graph | XML | Validate

```
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/instance#Citroen</uri></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>Citroën</literal></binding>
</result>
</results>
</sparql>
```

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT DISTINCT ?x ?z
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?v ?x ?z
6              ?z ?s ?t
7  }
```

Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='z'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#marque</uri></binding>
<binding name='z'><uri>http://www.labd.org/2015/exercice-1/instance#Citroen</uri></binding>
</result>
</results>
</sparql>
```

4. Et la négation ?

- Tester la non-existence d'un motif avec `FILTER NOT EXISTS { motif }`

  Une solution ne sera gardée qu'à condition que le motif de la clause `FILTER NOT EXISTS { motif }` n'est pas satisfait. Il existe aussi `FILTER`

- Enlever des solutions satisfaisant un motif avec `MINUS { motif }`

  La sémantique du `MINUS` n'est pas toujours très claire. Il est conseillé d'utiliser plutôt `FILTER NOT EXISTS`

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System | Query1 × | +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2  SELECT DISTINCT ?x ?z
3  WHERE {
4              ?x rdfs:domain sch:Voiture
5              ?v ?x ?z
6              FILTER NOT EXISTS {?z ?s ?t}
7  }
```

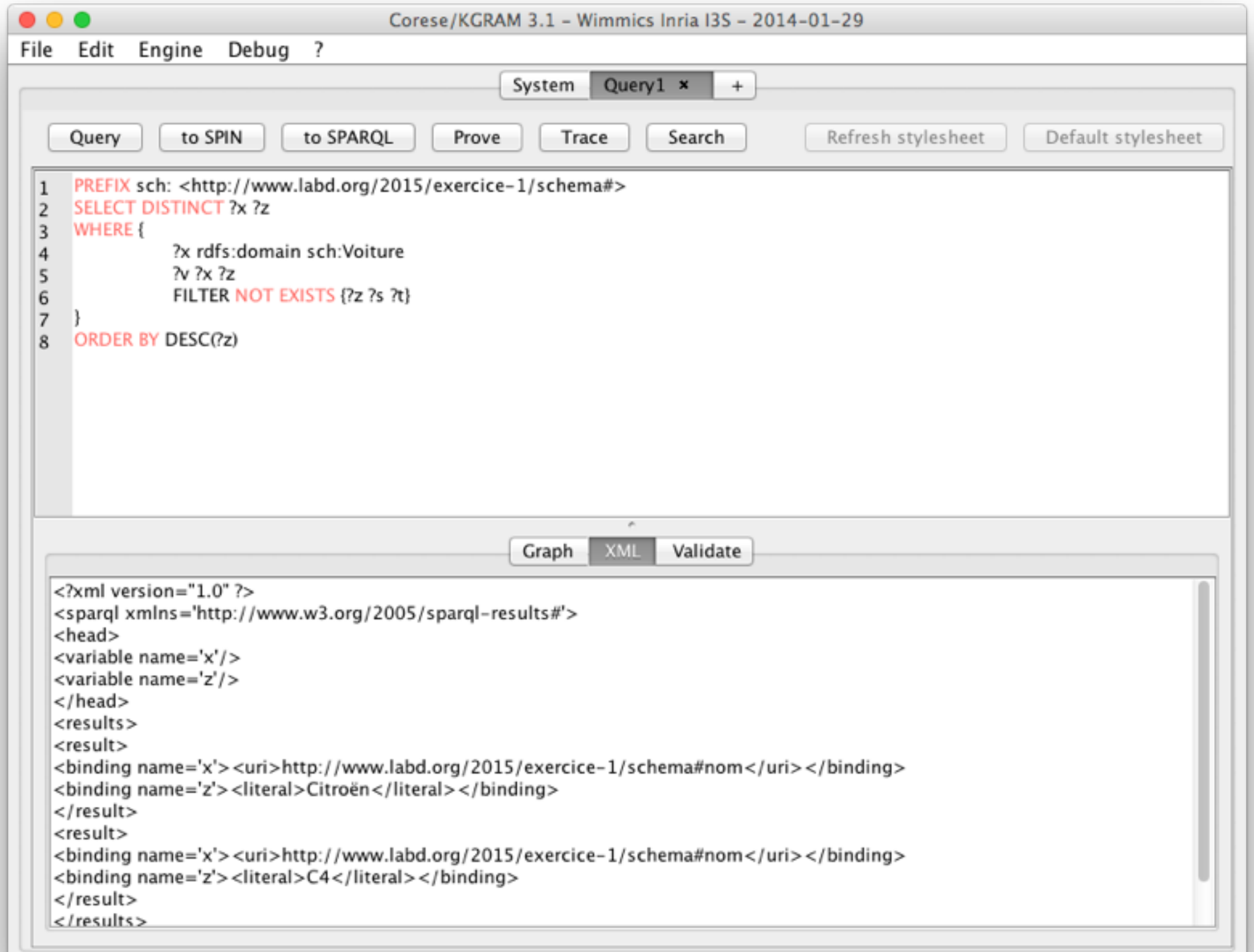
Graph | XML | Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='z'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>C4</literal></binding>
</result>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='z'><literal>Citroën</literal></binding>
</result>
</results>
```

Structure d'une requête SPARQL

1. un prologue optionnel
   (`PREFIX`,…)
2. une section qui définit la sortie
   (`SELECT ?x`,…)
3. une section optionnelle qui définit les sources de données à interroger
   (`FROM`)
4. une section qui définit les contraintes à vérifier par les variables
   (`WHERE`)
5. **une section optionnelle qui définit l'ordre des résultats en sortie**
   (`ORDER BY`)
6. une section optionnelle de regroupement de résultats
   (`GROUP BY`)
7. une section optionnelle qui précise un intervalle de sélection de résultats
   (`LIMIT`, `OFFSET`)

5. Ordre des résultats en sortie : `ORDER BY`



Structure d'une requête SPARQL

1. un prologue optionnel
   (`PREFIX`,…)
2. une section qui définit la sortie
   (`SELECT ?x`,…)
3. une section optionnelle qui définit les sources de données à interroger
   (`FROM`)
4. une section qui définit les contraintes à vérifier par les variables
   (`WHERE`)
5. une section optionnelle qui définit l'ordre des résultats en sortie
   (`ORDER BY`)
6. **une section optionnelle de regroupement de résultats**
   (`GROUP BY`)
7. une section optionnelle qui précise un intervalle de sélection de résultats
   (`LIMIT`, `OFFSET`)

6. Agrégation des données

- Par défaut, un ensemble de solutions consiste en un groupe de solutions

- On peut, à l'aide de `GROUP BY` partitioner la solution en plusieurs groupes sur lesquels on peut appliquer une fonction d'agrégation.

- Les fonctions d'agrégation sont `COUNT`, `SUM`, `MIN`, `MAX`, `AVG`, `GROUP_CONCAT`, `SAMPLE`

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System Query1 × +

Query to SPIN to SPARQL Prove Trace Search Refresh stylesheet Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2
3  SELECT ?x (GROUP_CONCAT(?z," I ") as ?gcx)
4  WHERE
5  {
6   ?x rdfs:domain sch:Voiture
7   ?v ?x ?z
8  }
9  GROUP BY ?x
```

On doit obligatoirement ranger la valeur d'agrégat dans une variable

Graph XML Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='gcx'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='gcx'><literal datatype='http://www.w3.org/2001/XMLSchema#string'>C4 I  Citro&#65533;n I </literal></binding>
</result>
</results>
</sparql>
```

Corese/KGRAM 3.1 - Wimmics Inria I3S - 2014-01-29

File Edit Engine Debug ?

System Query1 × +

Query | to SPIN | to SPARQL | Prove | Trace | Search | Refresh stylesheet | Default stylesheet

```
1  PREFIX sch: <http://www.labd.org/2015/exercice-1/schema#>
2
3  SELECT ?x (GROUP_CONCAT(?z," | ") as ?gcx)
4  WHERE
5  {
6   ?x rdfs:domain sch:Voiture
7   ?v ?x ?z
8  }
9  GROUP BY ?x
10 HAVING (count (?x)>1)
```

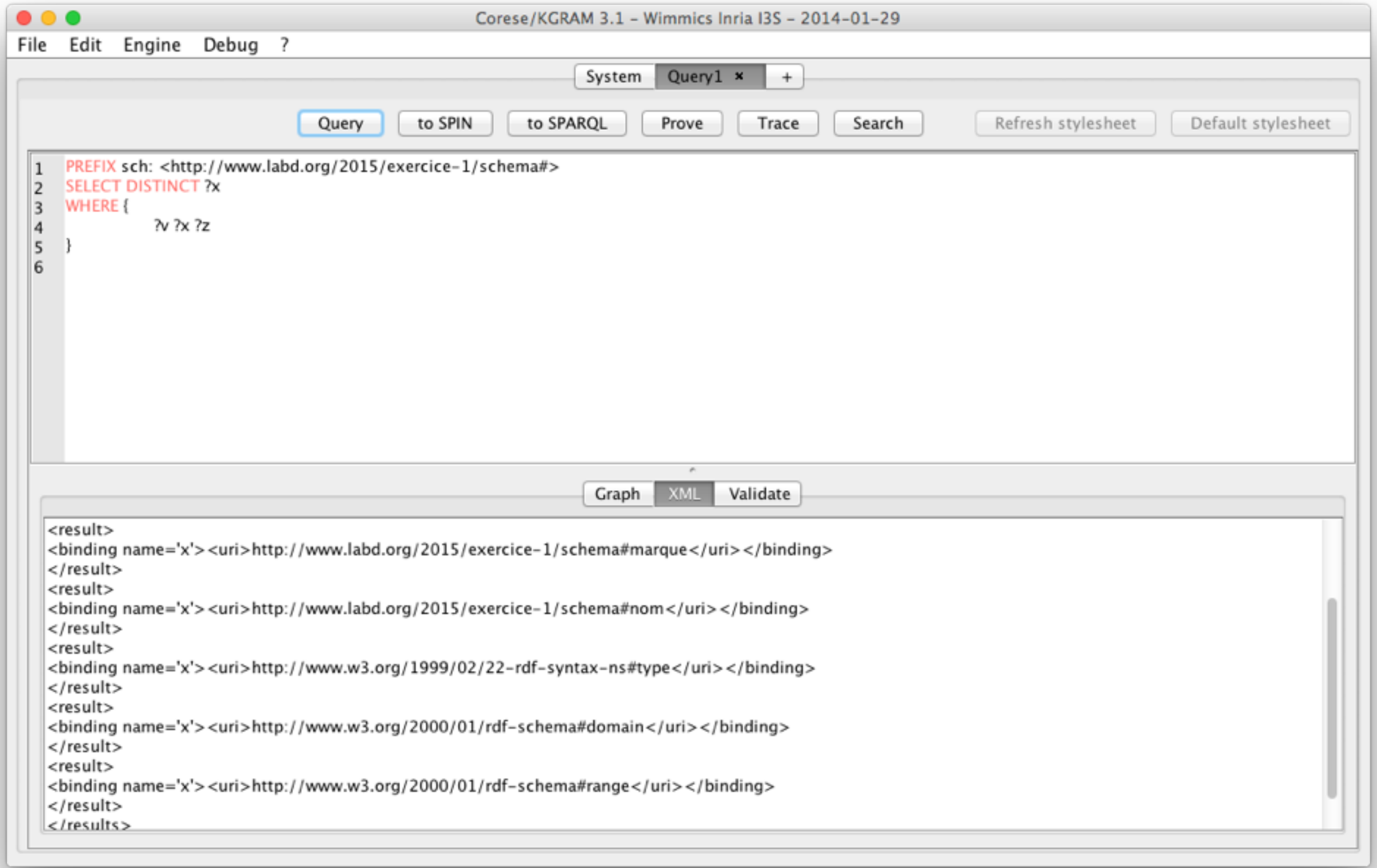
Graph XML Validate

```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='x'/>
<variable name='gcx'/>
</head>
<results>
<result>
<binding name='x'><uri>http://www.labd.org/2015/exercice-1/schema#nom</uri></binding>
<binding name='gcx'><literal datatype='http://www.w3.org/2001/XMLSchema#string'>C4 |  Citro&#65533;n | </literal></binding>
</result>
</results>
</sparql>
```

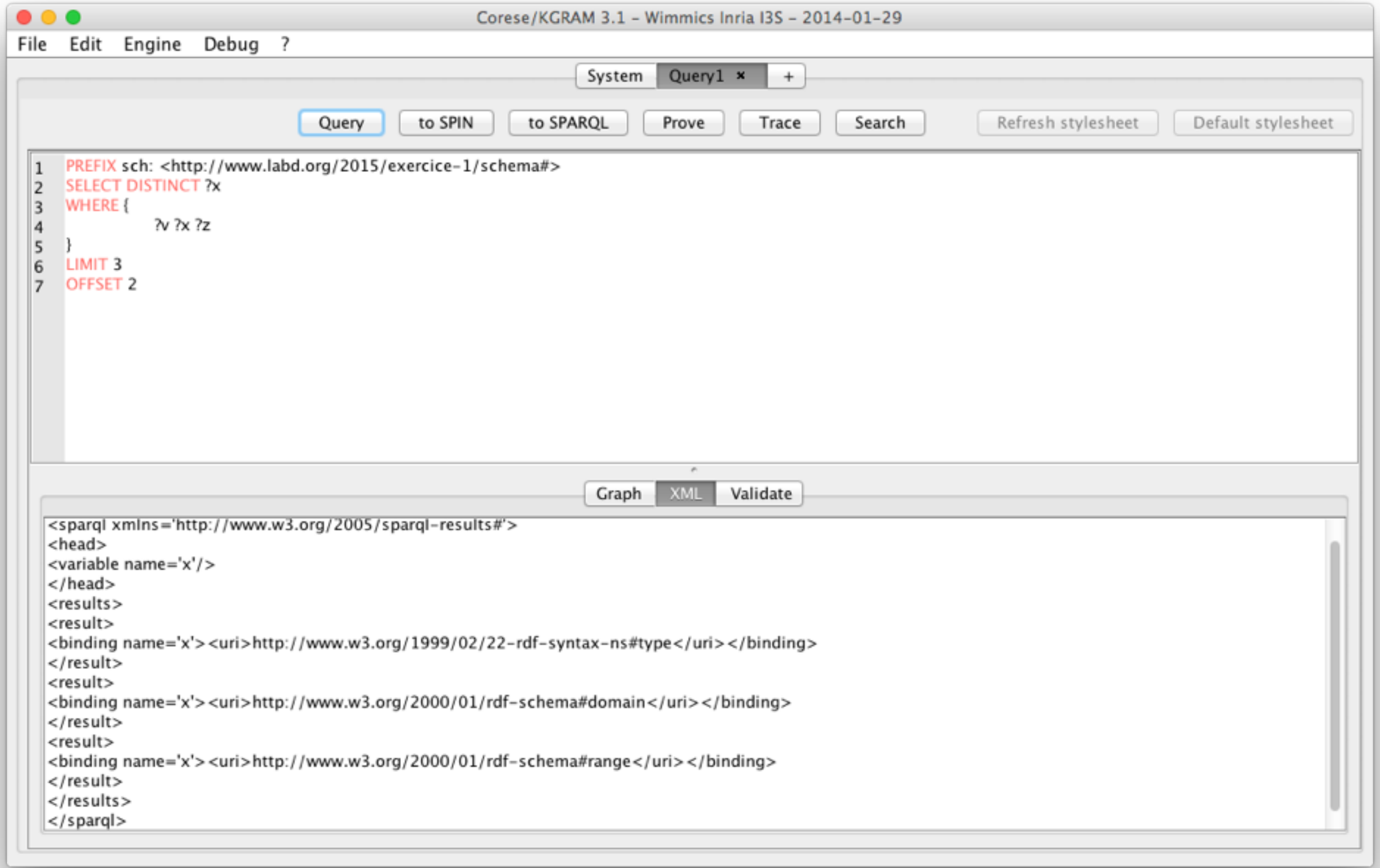
Structure d'une requête SPARQL

1. un prologue optionnel
   (`PREFIX`,…)
2. une section qui définit la sortie
   (`SELECT ?x`,…)
3. une section optionnelle qui définit les sources de données à interroger
   (`FROM`)
4. une section qui définit les contraintes à vérifier par les variables
   (`WHERE`)
5. une section optionnelle qui définit l'ordre des résultats en sortie
   (`ORDER BY`)
6. une section optionnelle de regroupement de résultats
   (`GROUP BY`)
7. **une section optionnelle qui précise un intervalle de sélection de résultats (`LIMIT`, `OFFSET`)**

7. Intervalle de résultats. `LIMIT` et `OFFSET`



7. Intervalle de résultats. `LIMIT` et `OFFSET`



Pour quelques motifs de plus…

- Expressions régulières de chemins : permet de décrire un motif de chemin de longueur variable existant entre deux ressources (et pas seulement des triplets). Un tel chemin est une expression formée de noms de propriétés ou des variables et utilisant des opérateurs d'expressions régulières

```
{?a (ex:motherOf|ex:fatherOf)+ <#me>}
```

- Définir de nouvelles variables et les lier : la clause `BIND` permet de définir de nouvelles variables et de leur affecter une valeur.

```
{   ?x ex:price ?p
    ?x ex:discount ?d
    BIND (?p * (1 - ?d) AS ?price)
    FILTER (?price < 20)
}
```

Pour quelques motifs de plus…Sous-Requêtes

```
@prefix : <http://people.example/> .

:alice :name "Alice", "Alice Foo", "A. Foo".
:alice :knows :bob, :carol .
:bob :name "Bob", "Bob Bar", "B. Bar" .
:carol :name "Carol", "Carol Baz", "C. Baz".
```

```
PREFIX : <http://people.example/>
PREFIX : <http://people.example/>
SELECT ?y ?minName
WHERE {
 :alice :knows ?y
 { SELECT ?y (MIN(?name) AS ?minName)
   WHERE {?y :name ?name}
   GROUP BY ?y
 }
}
```

```
<http://people.example/bob>   "B. Bar"
<http://people.example/carol> "C. Baz"
```