



```
SELECT ?sujet ?objet  
WHERE {   ?sujet ?predicat ?objet }
```

```
PREFIX ex: <http://exemple.org/>  
SELECT ?predicat ?objet  
WHERE {ex:ma_constante ?predicat ?objet}
```

## Base de connaissance :

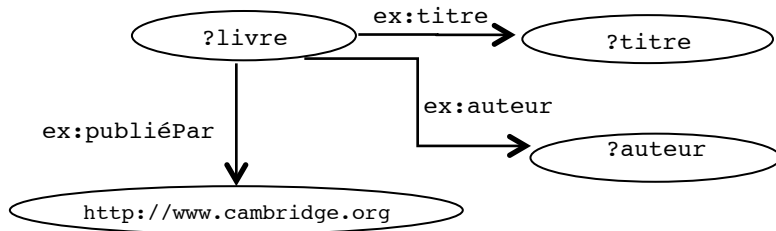
@prefix ex:	<http://exemple.org/> .	
ex:dlhb	ex:publiéPar	<http://www.cambridge.org/> ;
	ex:titre	"The Description Logic Handbook" ;
	ex:auteur	ex:fBaader ,
		ex:dCalvanese ;
	ex:auteur	ex:dMcGuinness.
ex:om	ex:publiéPar	<http://www.springer.com/>
	ex:titre	"Ontology Matching" ;
	ex:auteur	ex:jEuzenat ;
	ex:auteur	ex:pShvaiko.

Requête : "Quels sont les titres et noms d'auteurs des livres publiés par Cambridge Press ?"

## Base de connaissance :

@prefix ex:	<http://exemple.org/> .
ex:dlhb	ex:publiéPar <http://www.cambridge.org/> ;
	ex:titre "The Description Logic Handbook" ;
	ex:auteur ex:fBaader ,
	ex:dCalvanese ;
	ex:auteur ex:dMcGuinness.
ex:om	ex:publiéPar <http://www.springer.com/>
	ex:titre "Ontology Matching" ;
	ex:auteur ex:jEuzenat ;
	ex:auteur ex:pShvaiko.

Requête : "Quels sont les titres et noms d'auteurs des livres publiés par Cambridge Press ?"



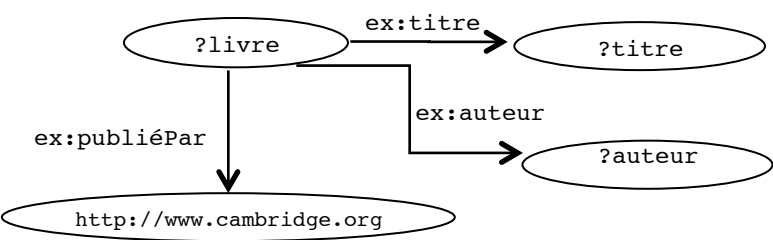
OU

```
PREFIX ex:<http://exemple.org/>
SELECT ?titre ?auteur
WHERE {
  ?livre    ex:publiéPar    <http://www.cambridge.org/>.
  ?livre    ex:titre       ?titre .
  ?livre    ex:auteur      ?auteur.
}
```

Base de connaissance :

@prefix ex:	<http://exemple.org/> .
ex:dlhb	ex:publiéPar <http://www.cambridge.org/> ;
	ex:titre "The Description Logic Handbook" ;
	ex:auteur ex:fBaader ,
	ex:dCalvanese ;
	ex:auteur ex:dMcGuinness.
ex:om	ex:publiéPar <http://www.springer.com/>
	ex:titre "Ontology Matching" ;
	ex:auteur ex:jEuzenat ;
	ex:auteur ex:pShvaiko.

Requête : "Quels sont les titres et noms d’auteurs des livres publiés par Cambridge Press ?"



OU

```
PREFIX ex:<http://exemple.org/>
SELECT ?titre ?auteur
WHERE {
  ?livre      ex:publiéPar    <http://www.cambridge.org/>.
  ?livre      ex:titre       ?titre .
  ?livre      ex:auteur      ?auteur.
}
```

Résultat :

Titre	Auteur
"The Description Logic Handbook"	http://exemple.org/fBaader
"The Description Logic Handbook"	http://exemple.org/dCalvanese
"The Description Logic Handbook"	http://exemple.org/dMcGuinness

## Groupes et options

Quels sont les livres, leurs titres et leurs auteurs ?

Livre	Titre	Auteur
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/fBaader">http://exemple.org/fBaader</a>
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/dCalvanese">http://exemple.org/dCalvanese</a>
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/dMcGuinness">http://exemple.org/dMcGuinness</a>
<a href="http://exemple.org/book2">http://exemple.org/book2</a>	"The Semantic Web"	<a href="#">_:a</a>
<a href="http://exemple.org/om">http://exemple.org/om</a>	"Ontology Matching"	<a href="http://exemple.org/jEuzenat">http://exemple.org/jEuzenat</a>
<a href="http://exemple.org/om">http://exemple.org/om</a>	"Ontology Matching"	<a href="http://exemple.org/pShvaiko">http://exemple.org/pShvaiko</a>
<a href="http://exemple.org/book4">http://exemple.org/book4</a>	"Formal Concept Analysis"	

## Groupes et options

Quels sont les livres, leurs titres et leurs auteurs ?

Livre	Titre	Auteur
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/fBaader">http://exemple.org/fBaader</a>
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/dCalvanese">http://exemple.org/dCalvanese</a>
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/dMcGuinness">http://exemple.org/dMcGuinness</a>
<a href="http://exemple.org/book2">http://exemple.org/book2</a>	"The Semantic Web"	_:a
<a href="http://exemple.org/om">http://exemple.org/om</a>	"Ontology Matching"	<a href="http://exemple.org/jEuzenat">http://exemple.org/jEuzenat</a>
<a href="http://exemple.org/om">http://exemple.org/om</a>	"Ontology Matching"	<a href="http://exemple.org/pShvaiko">http://exemple.org/pShvaiko</a>
<a href="http://exemple.org/book4">http://exemple.org/book4</a>	"Formal Concept Analysis"	

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre ?titre ?auteur
WHERE { ?livre      ex:titre      ?titre .}
```

## Groupes et options

Quels sont les livres, leurs titres et leurs auteurs ?

Livre	Titre	Auteur
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/fBaader">http://exemple.org/fBaader</a>
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/dCalvanese">http://exemple.org/dCalvanese</a>
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/dMcGuinness">http://exemple.org/dMcGuinness</a>
<a href="http://exemple.org/book2">http://exemple.org/book2</a>	"The Semantic Web"	_:a
<a href="http://exemple.org/om">http://exemple.org/om</a>	"Ontology Matching"	<a href="http://exemple.org/jEuzenat">http://exemple.org/jEuzenat</a>
<a href="http://exemple.org/om">http://exemple.org/om</a>	"Ontology Matching"	<a href="http://exemple.org/pShvaiko">http://exemple.org/pShvaiko</a>
<a href="http://exemple.org/book4">http://exemple.org/book4</a>	"Formal Concept Analysis"	

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre ?titre ?auteur
WHERE { ?livre      ex:titre      ?titre .
        ?livre      ex:auteur     ?auteur }
```



## Groupes et options

Quels sont les livres, leurs titres et leurs auteurs ?

Livre	Titre	Auteur
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/fBaader">http://exemple.org/fBaader</a>
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/dCalvanese">http://exemple.org/dCalvanese</a>
<a href="http://exemple.org/dlhb">http://exemple.org/dlhb</a>	"The Description Logic Handbook"	<a href="http://exemple.org/dMcGuinness">http://exemple.org/dMcGuinness</a>
<a href="http://exemple.org/book2">http://exemple.org/book2</a>	"The Semantic Web"	_:a
<a href="http://exemple.org/om">http://exemple.org/om</a>	"Ontology Matching"	<a href="http://exemple.org/jEuzenat">http://exemple.org/jEuzenat</a>
<a href="http://exemple.org/om">http://exemple.org/om</a>	"Ontology Matching"	<a href="http://exemple.org/pShvaiko">http://exemple.org/pShvaiko</a>
<a href="http://exemple.org/book4">http://exemple.org/book4</a>	"Formal Concept Analysis"	

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre ?titre ?auteur
WHERE { ?livre      ex:titre      ?titre .
        ?livre      ex:auteur     ?auteur }
```

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre ?titre ?auteur
WHERE {
    ?livre ex:titre      ?titre .
    OPTIONAL {?livre     ex:auteur      ?auteur}
}
```

## Groupes et unions

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre ?auteur
WHERE {
  ?livre ex:publiéPar <http://www.cambridge.com/> .
    {?livre    ex:auteur      ?auteur} UNION
    {?livre    ex:ecrivain    ?auteur}
}
```

## Groupes et unions 2

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre ?auteur ?nom
WHERE {
  ?livre ex:publiéPar <http://www.cambridge.com/> .
  {?livre ex:auteur ?auteur} UNION
  {?livre ex:ecrivain ?auteur} OPTIONAL
  {?auteur ex:nom ?nom}
}
```

Quelle interpretation ?

## Groupes et unions 2

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre ?auteur ?nom
WHERE {
  ?livre    ex:publiéPar    <http://www.cambridge.com/> .
    {?livre    ex:auteur      ?auteur} UNION
    {?livre    ex:ecrivain    ?auteur} OPTIONAL
    {?auteur    ex:nom        ?nom}
}
```

Quelle interpretation ?

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre ?auteur ?nom
WHERE {
  ?livre    ex:publiéPar    <http://www.cambridge.com/> .
    {
      {?livre    ex:auteur      ?auteur} UNION
      {?livre    ex:ecrivain    ?auteur}
    }
    OPTIONAL {?auteur ex:nom ?nom}
}
```

Une base de connaissance :

s1 p "test".

s2 p "test"^^xsd:string.

s3 p "test"@en.

s4 p "42"^^xsd:integer.

s5 p "test"^^<http://exemple.org/  
datatype1>.

Que retourne les requêtes suivante ?

{ ?subject p "test" }                   => ?

{ ?subject p "042"^^xsd:integer }       => ?

{ ?subject p 42 }                       => ?

## Tests et Filter

Faire des tests et des comparaisons ex : `WHERE price > 100`  
`WHERE title LIKE 'Handbook of%'`

`PREFIX ex:<http://exemple.org/>`

`SELECT ?livre`

```
WHERE {  
    ?livre      ex:publiéPar <http://www.springer.com/> .  
    ?livre      ex:prix      ?prix  
    FILTER ( ?prix < 100 )  
}
```

avec `FILTER`, on peut tester l'égalité (=) et utiliser les opérateurs arithmétiques +, -, \*, /.

## Tests et Filter

Faire des tests et des comparaisons ex : `WHERE price > 100`  
`WHERE title LIKE 'Handbook of%'`

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre
WHERE {
    ?livre      ex:publiéPar <http://www.springer.com/> .
    ?livre      ex:prix      ?prix
    FILTER ( ?price < 100 )
}
```

avec `FILTER`, on peut tester l'égalité (=) et utiliser les opérateurs arithmétiques +, -, \*, /.

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre
WHERE {
    ?livre      ex:titre      ?titre
    FILTER ( REGEX(?title, "^Handbook of") )
}
```

`isURI(a)`, `isBLANK(a)`, `isLITERAL(a)`, `LANG(a)`, `REGEX(a,b)`, etc.

## Formater les résultats

```
PREFIX ex:<http://exemple.org/>
SELECT ?personne ?email ?telephone.
WHERE {
    ?personne      ex:email    ?email.
    ?personne      ex:tel      ?telephone
}
```

personne	email	telephone
philippe	<a href="mailto:philippe@laposte.net">philippe@laposte.net</a>	"06 12 34 56 78"
philippe	<a href="mailto:philippe@laposte.net">philippe@laposte.net</a>	"03 83 12 34 56"
philippe	<a href="mailto:philippe@gmail.com">philippe@gmail.com</a>	"06 12 34 56 78"
philippe	<a href="mailto:philippe@gmail.com">philippe@gmail.com</a>	"03 83 12 34 56"



## Formater les résultats

```
PREFIX ex:<http://exemple.org/>
CONSTRUCT {
    ?personne    ex:email    ?email.
    ?personne    ex:tel      ?telephone
}
WHERE {
    ?personne    ex:email    ?email.
    ?personne    ex:tel      ?telephone
}
```

retourne un document RDF

```
@prefix    ex:<http://example.org/>
ex:philippe    ex:email    "philippe@laposte.net"
ex:philippe    ex:email    "philippe@gmail.com"
ex:philippe    ex:tel      "06 12 34 56 78"
ex:philippe    ex:tel      "03 83 12 34 56"
```

## Formater les résultats

```
PREFIX ex:<http://exemple.org/>
ASK {
    ?personne      ex:email  ?email.
    ?personne      ex:tel    ?telephone
}
```

retourne true or false

```
PREFIX ex:<http://exemple.org/>
SELECT ?livre, ?prix
WHERE { ?livre ex:prix ? prix.}
ORDER BY ?price
LIMIT 5 OFFSET 25
```

# la négation (de 3 façons différentes)

- la négation par l'échec

```
PREFIX ex:<http://example.org>
SELECT * WHERE {
    ?x A ex:Book
    OPTIONAL {?x ex:topic ?y .
               FILTER (?y = ex:ComputerScience)}
    FILTER(! BOUND(?y))
}
```

- avec MINUS

```
PREFIX ex:<http://example.org>
SELECT * WHERE {
    ?x A ex:Book }
MINUS {?x ex:topic ex:ComputerScience}
```

- avec FILTER (NOT EXISTS

```
PREFIX ex:<http://example.org>
SELECT * WHERE {
    ?x A ex:Book
    FILTER (NOT EXISTS { ?x ex:topic ex:ComputerScience } )
}
```

# SPARQL et les fct° d'aggrégation

- COUNT, AVG

```
PREFIX ex:<http://example.org>
SELECT (COUNT(distinct ?book) AS ?total) WHERE {
    ?book A ex:Book.
    ?book ex:topic ex:ComputerScience
}
```

- MIN, MAX

```
PREFIX ex:<http://example.org>
SELECT ?title (MIN(?price) AS ?min) WHERE {
    ?x A ex:Book.
    ?book ex:price ?price
} GROUP BY ?titre
```

- avec HAVING

```
PREFIX ex:<http://example.org>
SELECT ?title (MIN(?price) AS ?min) WHERE {
    ?x A ex:Book.
    ?book ex:price ?price
} GROUP BY ?titre
HAVING (?min < 50)
```

# SPARQL et les graphs (1/2)

- Il est possible d'associer les données d'un triple store à différents graphes, çàd à différents sous ensemble de données

```
PREFIX ex:<http://example.org>
SELECT * WHERE {
    ?x A ex:Book
    GRAPH <http://www.seuil.fr/livre> {?x ex:titre ?y}
    GRAPH ?g {?x ex:author ?y}
}
```

- on oblige les titres à venir d'un graphe précis
- on demande à connaître les graphes d'origine des auteurs associés au livres trouvés.

# SPARQL et les graphs (2/2)

- Il est possible de préciser les graphes auxquels s'applique une requête

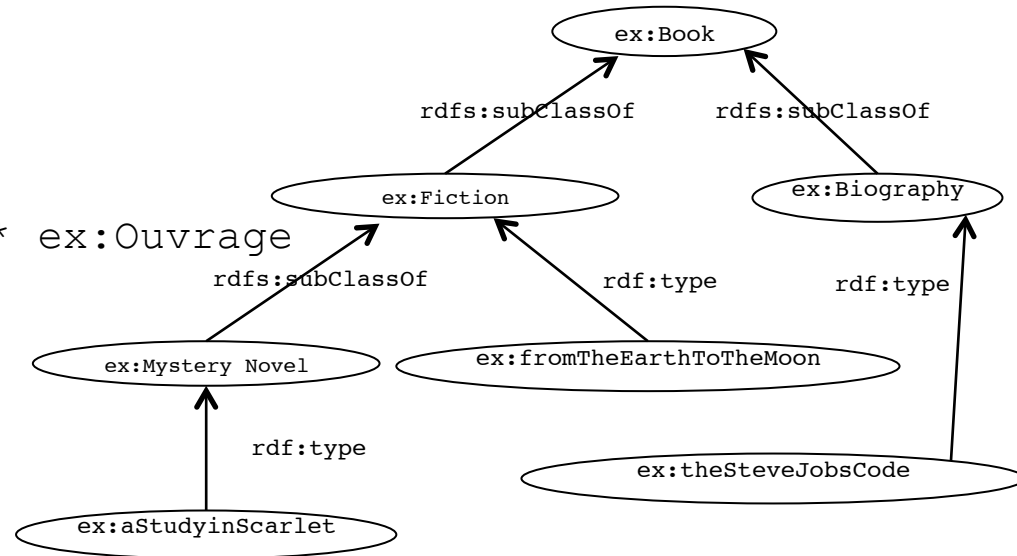
```
PREFIX ex:<http://example.org>
FROM <http://www.seuil.fr/livre>
FROM <http://www.flamarion.fr/livre>
SELECT * WHERE {
    ?x A ex:Book.
    ?x ex:titre ?y
}
```

- on oblige les titres à venir d'un graphe précis
- on demande à connaître les graphes d'origine des auteurs associés au livres trouvés.

# SPARQL et les chemins

- les chemins permettent de retrouver des ressources reliés indirectement (via un chemin)

```
PREFIX ex:<http://example.org>
SELECT * WHERE {
  ?x rdf:type/rdfs:subClassOf* ex:Ouvrage
}
```

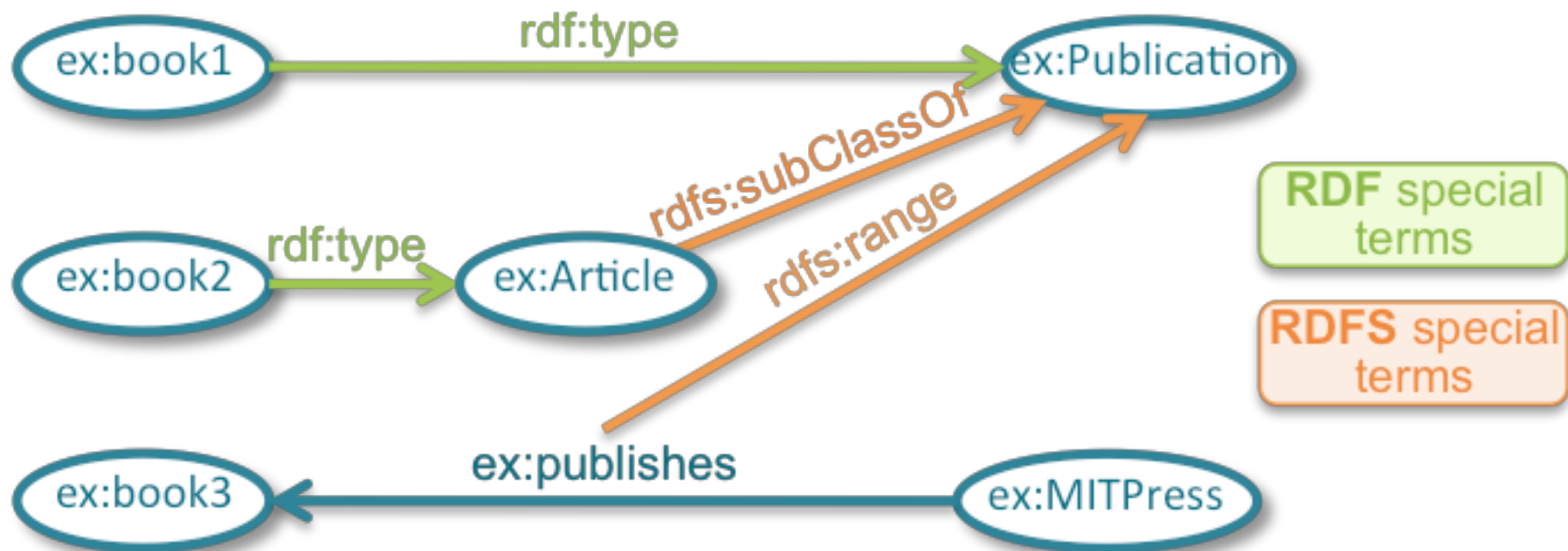


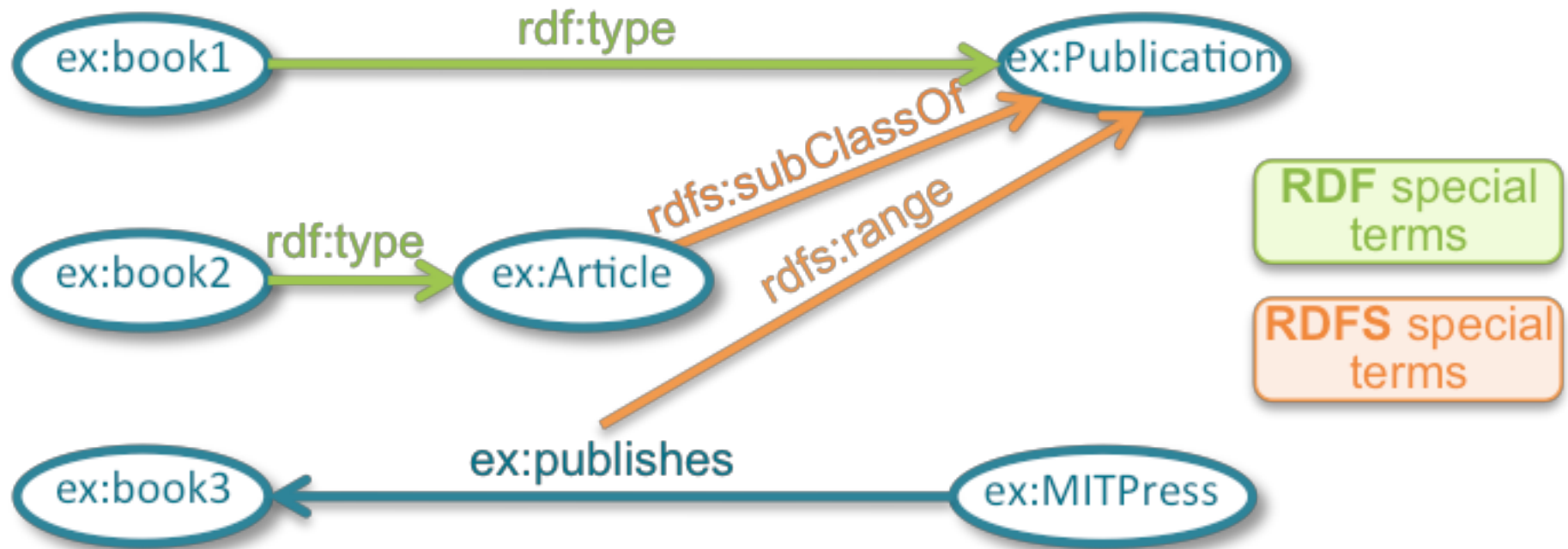
```
PREFIX ex:<http://example.org>
SELECT * WHERE {
  ?x ex:author|ex:writer/ex:coauthorWith*/ex:name "Einstein"
}
```

# Plusieurs SPARQL endpoint ?

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
PREFIX efo: [...]
SELECT distinct ?dbXrefProt ?pathwayname ?moleculeLabel ?expressionValue ?propertyValue
WHERE {
#Get differentially expressed genes (and proteins) where factor is asthma
  ?value atlasterms:pValue ?pvalue .
  ?value atlasterms:hasFactorValue ?factor .
  ?value rdfs:label ?expressionValue .
  ?value atlasterms:isMeasurementOf ?probe .
  ?probe atlasterms:dbXref ?dbXrefProt .
  ?dbXrefProt a atlasterms:UniprotDatabaseReference .
  ?factor atlasterms:propertyType ?propertyType .
  ?factor atlasterms:propertyValue ?propertyValue .
  ?factor rdf:type efo:EFO_0000249 .
#Compunds target them
SERVICE <http://www.ebi.ac.uk/rdf/services/chembl/sparql> {
  ?act a cco:Activity ;
  cco:hasMolecule ?molecule ;
  cco:hasAssay ?assay .
  ?molecule rdfs:label ?moleculeLabel .
  ?assay cco:hasTarget ?target .
  ?target cco:hasTargetComponent ?targetcmpt .
  ?targetcmpt cco:targetCmptXref ?dbXrefProt .
  ?targetcmpt cco:taxonomy <http://identifiers.org/taxonomy/9606> .
  ?dbXrefProt a cco:UniprotRef .}
SERVICE <http://www.ebi.ac.uk/rdf/services/reactome/sparql> {
  ?protein rdf:type biopax3:Protein .
  ?protein biopax3:memberPhysicalEntity
  [biopax3:entityReference ?dbXrefProt] .
  ?pathway biopax3:displayName ?pathwayname .
  ?pathway biopax3:pathwayComponent ?reaction .
  ?reaction ?rel ?protein } }
```







Si je fais une requête pour retrouver toutes les publications, est ce que mon système me les retourne bien tous ?