

---

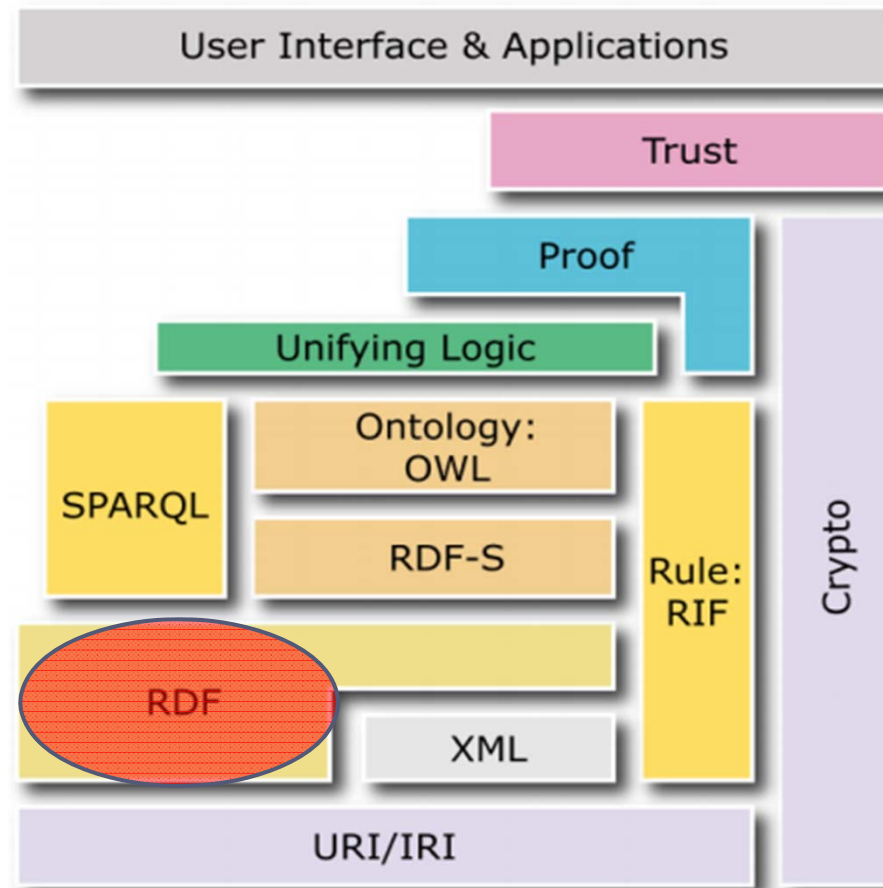
# Introduction au Web Sémantique & XML

Master IPS 2021/2022



# Web sémantique

## Approche en couches





# Plan

---

## **Chapitre 3** **RDF & Syntaxes RDF**

I – Introduction

II – Modèle de Données RDF

**III – Sérialisation des graphes RDF**

**Syntaxe – RDF/XML**

Syntaxe – Turtle



## III - Sérialisation des graphes RDF

---

- ▶ Les déclarations RDF
  - ▶ reposent sur un modèle de multi-graphe orienté étiqueté,
  - ▶ Il a l'avantage d'avoir la flexibilité et l'extensibilité nécessaires pour un monde de données ouvert
  - ▶ Il est visuellement très compréhensible et intuitif.
- ▶ Le modèle des graphes RDF est un **modèle abstrait** qu'il faut le doter d'une **syntaxe concrète** afin qu'il puisse être représenté sur machine.
- ▶ Un certain nombre de **langages (formats de sérialisation)** différents existent et permettent l'écriture des graphes RDF.
  - ▶ ces différents langages qui décrivent le même graphique, mènent exactement aux mêmes triples. Elles sont donc logiquement équivalentes.



## III - Sérialisation des graphes RDF

---

- ▶ Parmi les langages d'écriture d'un graphe RDF on verra :
  - ▶ **RDF/XML** (XML syntaxes pour RDF).
  - ▶ **Turtle** (et sa famille **N-Triples**, **TriG** and **N-Quads**);
- ▶ Dans ce qui suit, on s'intéressera plus particulièrement **RDF/XML** et **Turtle**



## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

---



- ▶ Le **graphe RDF** comporte des **n uds** et des **arcs orient s et  tiquet s** qui relient des paires de n uds.
- ▶ Chaque paire de n uds reli s par un arc repr sente un **triples RDF**.
- ▶ Chaque triplet contenant un **n ud sujet**, un **pr dicat** et un **n ud objet**.
- ▶ Les n uds sont des **IRI**, des **litt raux** ou des **n uds vides**.
- ▶ Les n uds vides peuvent recevoir un identifiant local au document appel  **identifiant de n ud vide**.
- ▶ Les pr dicats sont des **IRI** et peuvent  tre interpr t s comme une **relation** entre deux n uds ou comme **d finissant une valeur d'attribut** (n ud objet) **pour un n ud sujet**
- ▶ Le n ud objet, qui repr sente la valeur de la propri t  lorsque elle est appliqu e au sujet, peut  tre une **IRI**, un **litt ral** ou un **n ud vide**.

### Syntaxe RDF/XML



# III - Séri­alisation des graphes RDF

## Syntaxe - RDF/XML

---

### Syntaxe RDF/XML

- ▶ Pour coder le graphe en XML, les nœuds et les prédicats doivent être représentés en termes XML (**noms d'élément**, **noms d'attribut**, **contenu de l'élément** et **valeurs d'attribut**).
- ▶ RDF/XML utilise des **noms qualifiés XML** (XML **QNames**) définis dans des espaces de noms XML pour représenter les IRI.
- ▶ Tous les QNames ont un espace de nom qui est un IRI et un nom local court.
- ▶ Les QNames peuvent avoir un préfixe court ou être liée à l'espace de nom par défaut et n'en avoir aucun préfixe (mais ont toujours un nom d'espace de nom)
- ▶ L'identité IRI représentée par le QName est déterminée en ajoutant la partie nom local du QName après la partie du nom d'espace nom



## III - Séri­alisation des graphes RDF

### Syntaxe - RDF/XML

---

- ▶ Un graphe peut être considéré comme une collection de chemins qui recouvre la totalité du graphe. Ils sont sous forme :

nœud, arc prédicat, nœud, arc prédicat, ... , nœud

- ▶ Un graphe RDF/XML se transforme en **séquences d'éléments à l'intérieur d'éléments** en alternant entre des éléments pour des nœuds et d'autre élément pour des arcs de prédicats.
- ▶ Le **nœud** au début de la séquence devient l'élément le plus à l'extérieur, l'arc de prédicat suivant se transforme en un élément enfant, etc.
- ▶ Un document RDF/XML commencent toujours par des nœuds.

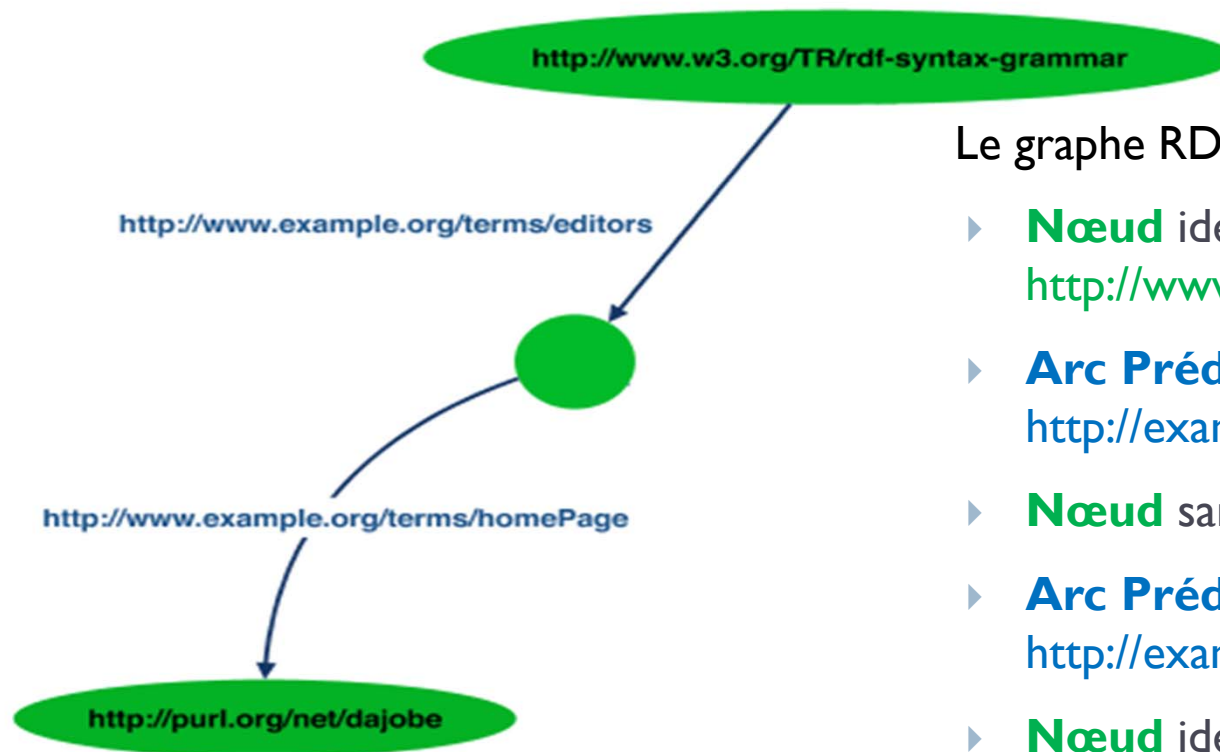




# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML

- Un graphe peut  tre consid r  comme une collection de chemins :  
**n ud**, **arc pr dicat**, **n ud**, **arc pr dicat**, ... , **n ud**



Le graphe RDF suivant est constitu  de :

- N ud** identifier par l'IRI :  
<http://www.w3.org/TR/rdf-syntax-grammar>
- Arc Pr dicat** identifi  par l'IRI :  
<http://example.org/terms/editor>
- N ud** sans IRI
- Arc Pr dicat** identifi  par l'IRI :  
<http://example.org/terms/homePage>
- N ud** identifier par l'IRI :  
<http://purl.org/net/dajobe/>



## III - Séri­a­li­sa­tion des graphes RDF

### Syntaxe - RDF/XML

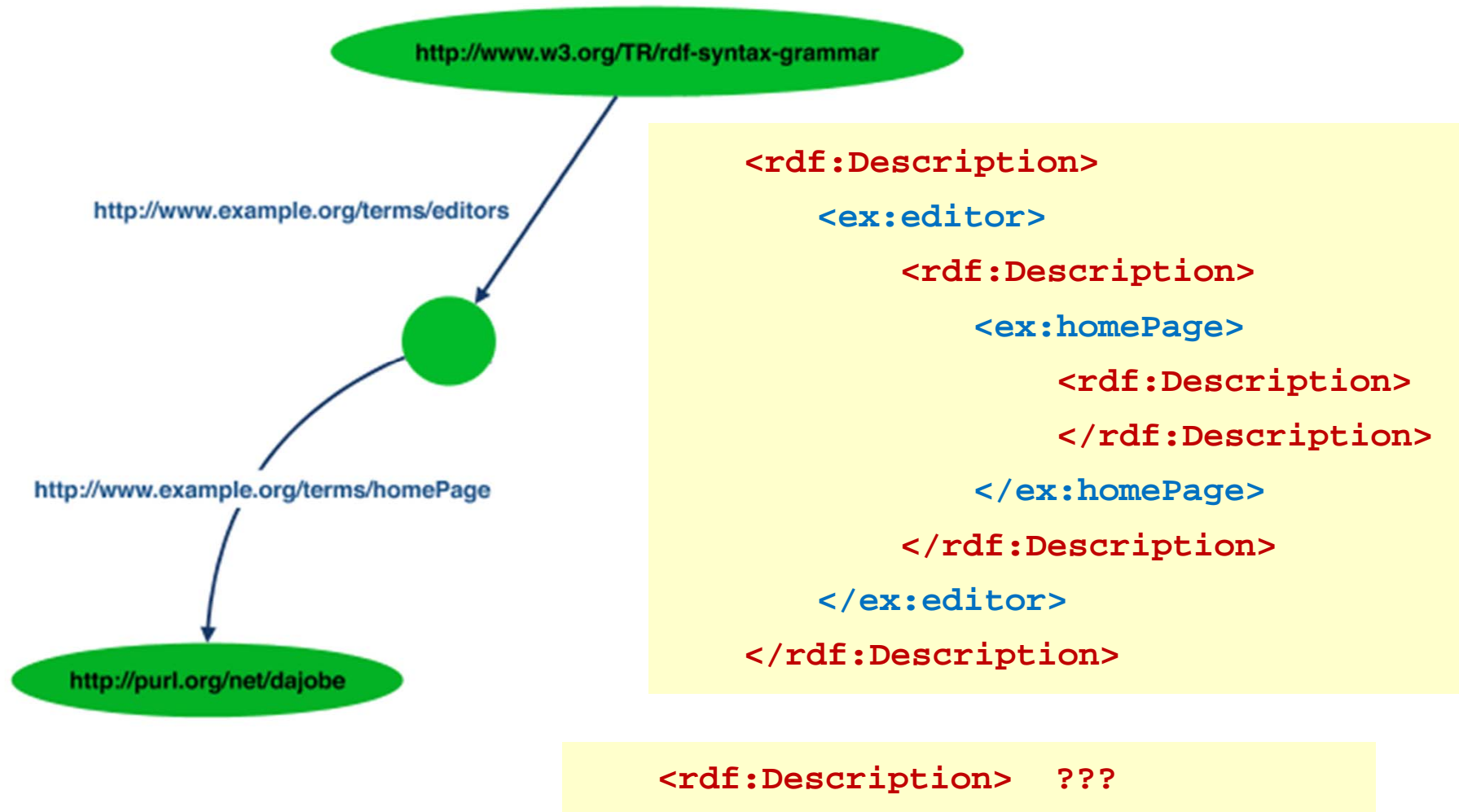
---

- ▶ En RDF/XML, un graphe RDF/XML se transforme en **séquences d'éléments à l'intérieur d'éléments** en alternant entre **des éléments pour des nœuds** et **des éléments pour des arcs de prédicats**.
- ▶ la séquence de **nœuds** et **arcs de prédicats** correspond à l'utilisation de **deux types d'éléments XML** :
  - ▶ **Éléments nœud** (*node elements*) représenté par l'élément **`rdf:Description`**
  - ▶ **Éléments propriété** (*property elements*) qui sont le contenu des **éléments nœuds**



# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML





## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

#### L 'attribut `rdf:about`

- ▶ Lorsque les  l ments **n ud** sont des **IRI**, on utilise l'attribut `rdf:about` de l' l ments `rdf:Description` pour associer cette **IRI**   l' l ment en question
- ▶ Mais Lorsque il s'agit d'un **n ud vide**, les  l ments `rdf:Description` reste vide (sans aucun attribut `rdf:about`)

```
...  
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar" >  
  <ex:editor>  
    <rdf:Description >  
      <ex:homePage>  
        <rdf:Description rdf:about="http://purl.org/net/dajobe/" >  
      ...  
    ...  
  ...  
</ex:editor>  
</rdf:Description>  
</rdf:Description>  
...
```

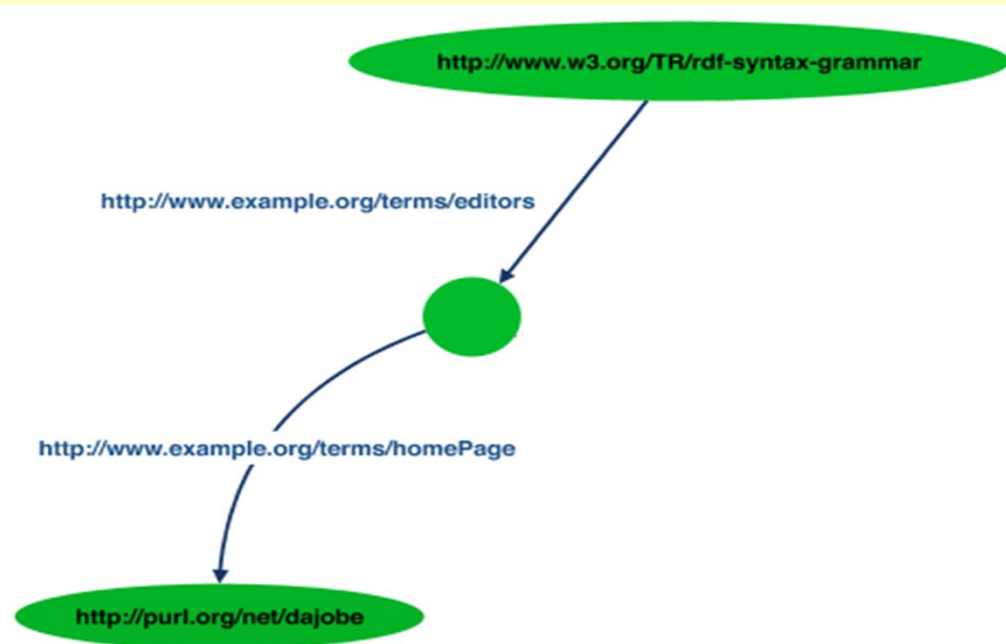




# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML

```
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">  
  <ex:editor>  
    <rdf:Description >  
      <ex:homePage>  
        <rdf:Description rdf:about="http://purl.org/net/dajobe/" >  
        </rdf:Description>  
      </ex:homePage>  
    </rdf:Description>  
  </ex:editor>  
</rdf:Description>
```



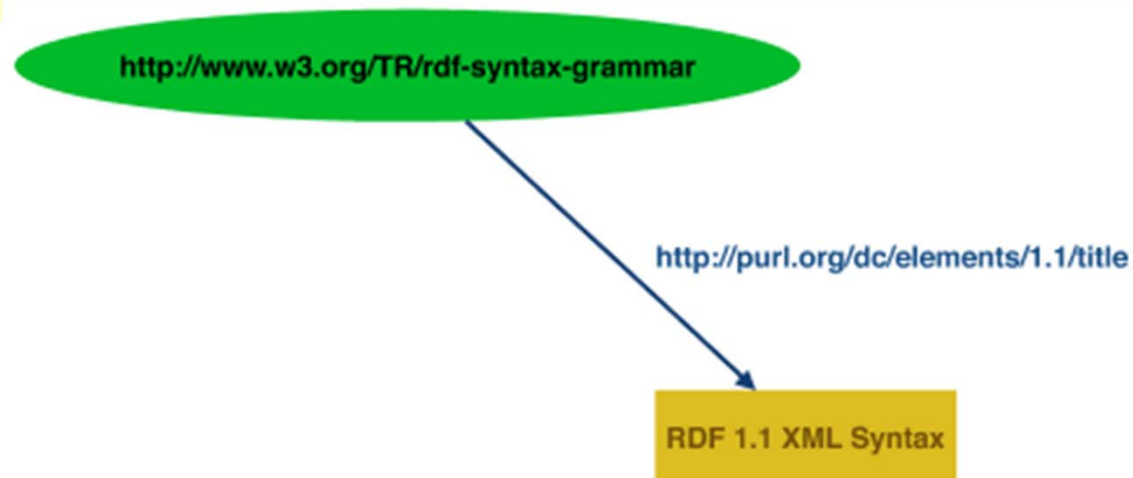


## III - Sérialisation des graphes RDF

### Syntaxe - RDF/XML

- ▶ Lorsque l'élément nœud (L'objet) est un **littéral**, il constitue le **contenue** de l'**éléments propriété**

```
...  
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">  
  <dc:title>RDF 1.1 XML Syntax</dc:title>  
</rdf:Description>  
...
```

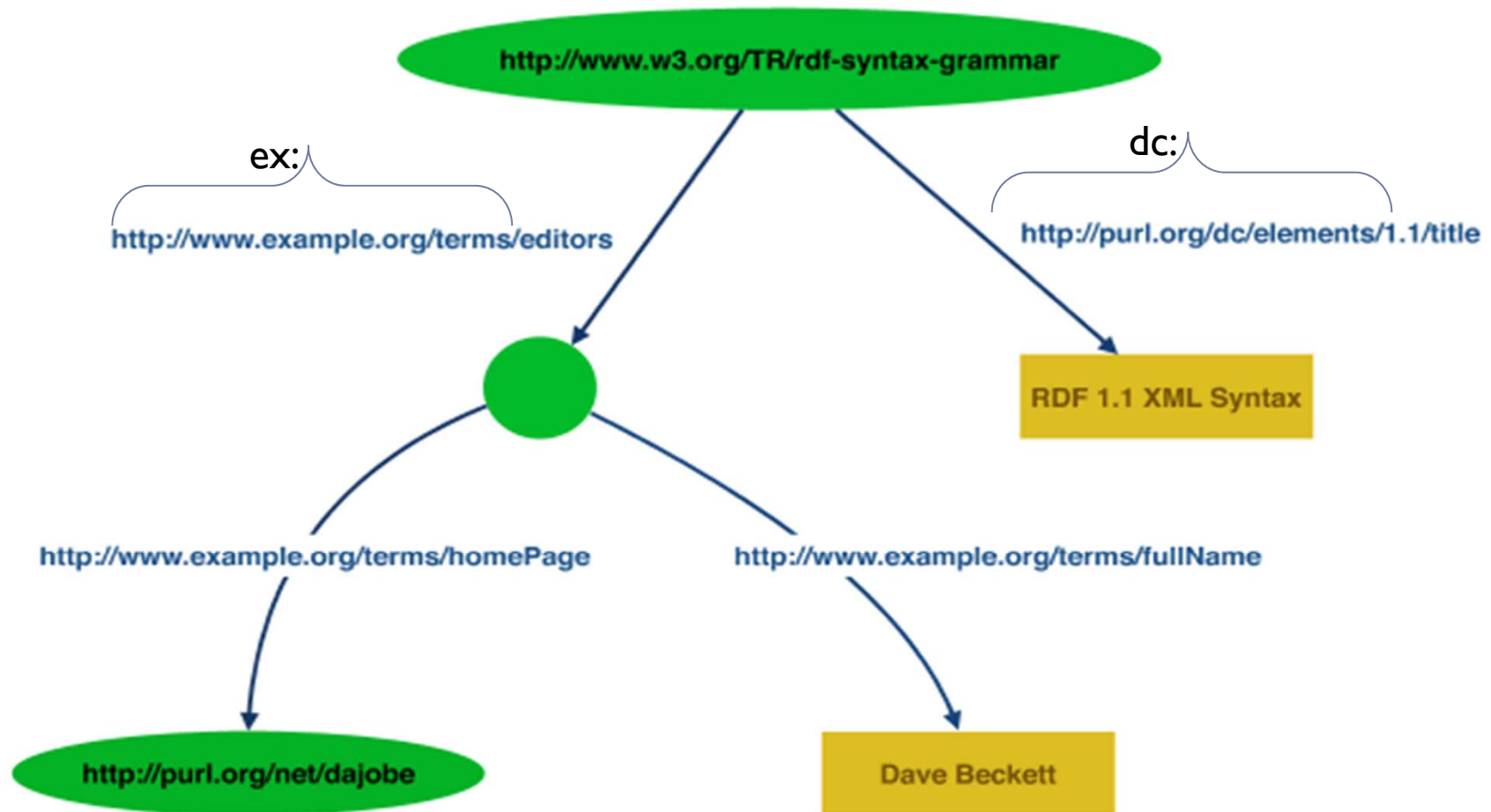




# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML

Pour le graphe RDF suivant :



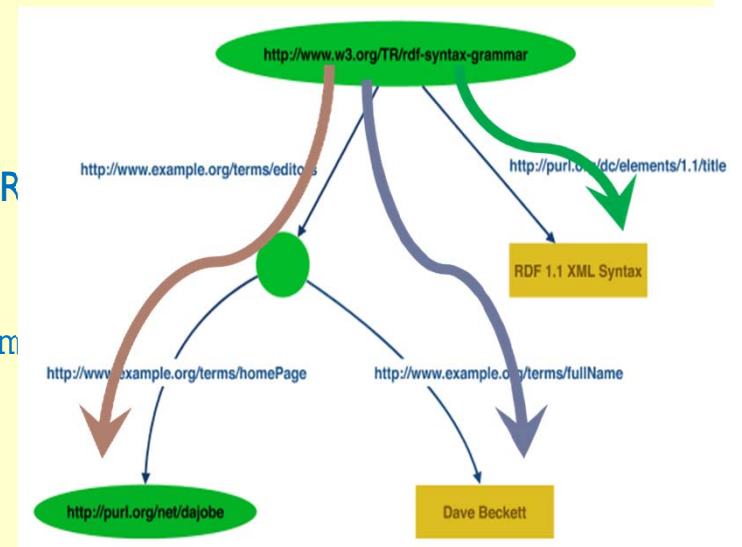
...

```
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar" >
  <ex:editor>
    <rdf:Description>
      <ex:homePage>
        <rdf:Description rdf:about="http://purl.org/net/dajobe/" >
          </rdf:Description>
        </ex:homePage>
      </rdf:Description>
    </ex:editor>
  </rdf:Description>
```

```
<rdf:Description rdf:about="http://www.w3.org/TR/
  <ex:editor>
    <rdf:Description>
      <ex:fullName>Dave Beckett</ex:fullName>
    </rdf:Description>
  </ex:editor>
</rdf:Description>
```

```
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar" >
  <dc:title>RDF 1.1 XML Syntax</dc:title>
</rdf:Description>
```

...







## III - Séri­alisation des graphes RDF

### Syntaxe - RDF/XML

- ▶ Plusieurs abréviations peuvent être utilisées pour réduire le code RDF/XML représentant un graphe RDF
- ▶ **Plusieurs éléments propriété (Multiple Property Elements)**  
lorsqu'un élément nœud a plusieurs éléments propriété, le code RDF/XML peut être abrégé en utilisant **plusieurs éléments propriétés enfants à l'intérieur de l'élément nœud** décrivant le nœud sujet.

```
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar" >  
    ...  
</rdf:Description>  
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar" >  
    ...  
</rdf:Description>  
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar" >  
    ...  
</rdf:Description>
```



...

```
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
```

```
  <ex:editor>
```

```
    <rdf:Description>
```

```
      <ex:homePage>
```

```
        <rdf:Description rdf:about="http://purl.org/net/dajobe/">
```

```
        </rdf:Description>
```

```
      </ex:homePage>
```

```
    </rdf:Description>
```

```
  </ex:editor>
```

```
</rdf:Description>
```

```
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
```

```
  <ex:editor>
```

```
    <rdf:Description>
```

```
      <ex:fullName>Dave Beckett</ex:fullName>
```

```
    </rdf:Description>
```

```
  </ex:editor>
```

```
</rdf:Description>
```

```
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
```

```
  <dc:title>RDF 1.1 XML Syntax</dc:title>
```

```
</rdf:Description>
```

...



## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

- le code RDF/XML du graphe RDF pr c dent peut  tre  crit sous forme

```
...  
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">  
  <ex:editor>  
    <rdf:Description>  
      <ex:homePage>  
        <rdf:Description rdf:about="http://purl.org/net/dajobe/">  
          </rdf:Description>  
      </ex:homePage>  
      <ex:fullName>Dave Beckett</ex:fullName>  
    </rdf:Description>  
  </ex:editor>  
  <dc:title>RDF 1.1 XML Syntax</dc:title>  
</rdf:Description>  
...
```



## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

- ▶ Plusieurs  l ments propri t  (Multiple Property Elements)

- ▶  l ments propri t  vide (Empty Property Elements)

Lorsqu'un arc de pr dicat dans un graphe RDF pointe vers un n ud objet qui n'a plus d'arc de pr dicat ( l ment n ud vide `<rdf:Description rdf:about = "..."></rdf:Description>`), on utilise l'IRI du n ud objet en tant que valeur d'un attribut XML **`rdf:resource`** sur l' l ment de propri t .

...

```
<ex:homePage>
```

```
  <rdf:Description rdf:about="http://purl.org/net/dajobe/">
```

```
  </rdf:Description>
```

```
</ex:homePage>
```

...



...

```
<ex:homePage rdf:resource="http://purl.org/net/dajobe">
```

```
</ex:homePage>
```

...



## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

---

- le code RDF/XML du graphe RDF r sultant :

```
...  
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">  
  <ex:editor>  
    <rdf:Description>  
      <ex:homePage>  
        <rdf:Description rdf:about="http://purl.org/net/dajobe/">  
          </rdf:Description>  
        </ex:homePage>  
      <ex:fullName>Dave Beckett</ex:fullName>  
    </rdf:Description>  
  </ex:editor>  
  <dc:title>RDF 1.1 XML Syntax</dc:title>  
</rdf:Description>  
...
```



## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

- le code RDF/XML du graphe RDF pr c dent peut  tre  crit sous forme

```
...  
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">  
  <ex:editor>  
    <rdf:Description>  
      <ex:homePage rdf:resource="http://purl.org/net/dajobel"/>  
      <ex:fullName>Dave Beckett</ex:fullName>  
    </rdf:Description>  
  </ex:editor>  
  <dc:title>RDF 1.1 XML Syntax</dc:title>  
</rdf:Description>  
...
```



## III - Sérialisation des graphes RDF

### Syntaxe - RDF/XML

- ▶ Plusieurs éléments propriété (Multiple Property Elements)
- ▶ Éléments propriété vide (Empty Property Elements)
- ▶ **Attributs propriété (Property Attributes)**

Lorsque le contenu d'un élément propriété est un **littéral de type chaîne de caractère**, il est possible de l'utiliser comme attribut XML sur l'élément nœud le contenant.

```
...  
  <rdf:Description>  
    <ex:homePage rdf:resource="http://purl.org/net/dajobe/" />  
    <ex:fullName>Dave Beckett</ex:fullName>  
  </rdf:Description>  
...
```



```
...  
  <rdf:Description ex:fullName="Dave Beckett">  
    <ex:homePage rdf:resource="http://purl.org/net/dajobe/" />  
  </rdf:Description>  
...
```





## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

- le code RDF/XML du graphe RDF pr c dent peut  tre  crit sous forme

```
...  
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">  
  <ex:editor>  
    <rdf:Description ex:fullName="Dave Beckett" >  
      <ex:homePage rdf:resource="http://purl.org/net/dajobe/" />  
    </rdf:Description>  
  </ex:editor>  
  <dc:title>RDF 1.1 XML Syntax</dc:title>  
</rdf:Description>  
...
```





## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

---

- ▶ le code complet RDF/XML du graphe RDF :
  - ▶ Pour cr er un document RDF/XML complet, la s rialisation du graphe RDF en RDF/XML est g n ralement contenue dans un  l ment **rdf:RDF** qui l' l ment racine de document XML.
  - ▶ L' l ment **rdf:RDF** est  galement utilis  pour d clarer les espaces de noms XML utilis s, bien que cela ne soit pas obligatoire.
  - ▶ Lorsque **rdf:RDF** ne contient qu'un seul  l ment n ud de niveau sup rieur, la balise **rdf:RDF** peut  tre omis bien que tout espace de nom XML doive toujours  tre d clar .
  - ▶ La sp cification XML permet  galement une d claration du prologue XML sp cifiant la version XML et le codage du contenu XML.



# III - S rialisation des graphes RDF

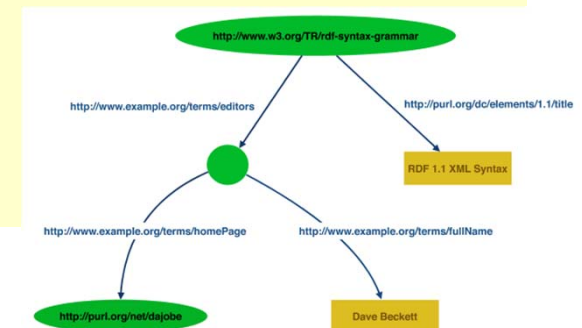
## Syntaxe - RDF/XML

- le code complet RDF/XML du graphe RDF :

```
<?xml version="1.0" encoding="utf-8"?>

<rdf:RDF      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
              xmlns:dc="http://purl.org/dc/elements/1.1/"
              xmlns:ex="http://example.org/stuff/1.0/">

  <rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar"
                  dc:title="RDF1.1 XML Syntax">
    <ex:editor>
      <rdf:Description ex:fullName="Dave Beckett">
        <ex:homePage rdf:resource="http://purl.org/net/dajobe/" />
      </rdf:Description>
    </ex:editor>
  </rdf:Description>
</rdf:RDF>
```





# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML - Validation



### Validation Service

<https://www.w3.org/RDF/Validator/>

[Skip Navigation](#) [Home](#)  
[Documentation](#)  
[Feedback](#)

### Check and Visualize your RDF documents

[olde servlet](#)

Enter a URI or paste an RDF/XML document into the text field above. A 3-tuple (triple) representation of the corresponding data mode

Check by Direct Input

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description rdf:about="http://www.w3.org/">
    <dc:title>World Wide Web Consortium</dc:title>
  </rdf:Description>
</rdf:RDF>
```

**Display Result Options:**  
Triples and/or Graph:   
Graph format:

Paste an RDF/XML document into the following text field to have it checked. More options are available in the [Extended interface](#).



# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML - Validation

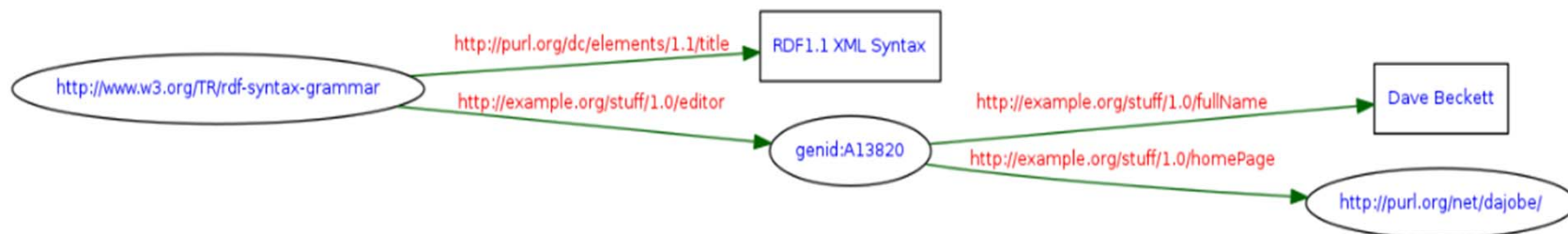
### Triples of the Data Model

Number	Subject	Predicate	Object
1	<a href="http://www.w3.org/TR/rdf-syntax-grammar">http://www.w3.org/TR/rdf-syntax-grammar</a>	<a href="http://purl.org/dc/elements/1.1/title">http://purl.org/dc/elements/1.1/title</a>	"RDF1.1 XML Syntax"
2	genid:A13820	<a href="http://example.org/stuff/1.0/fullName">http://example.org/stuff/1.0/fullName</a>	"Dave Beckett"
3	<a href="http://www.w3.org/TR/rdf-syntax-grammar">http://www.w3.org/TR/rdf-syntax-grammar</a>	<a href="http://example.org/stuff/1.0/editor">http://example.org/stuff/1.0/editor</a>	genid:A13820
4	genid:A13820	<a href="http://example.org/stuff/1.0/homePage">http://example.org/stuff/1.0/homePage</a>	<a href="http://purl.org/net/dajobe/">http://purl.org/net/dajobe/</a>

### The original RDF/XML document

```
1: <?xml version="1.0"?>
2: <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3:   xmlns:dc="http://purl.org/dc/elements/1.1/"
4:   xmlns:ex="http://example.org/stuff/1.0/">
5:
6:   <rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar"
7:     dc:title="RDF1.1 XML Syntax">
8:     <ex:editor>
9:       <rdf:Description ex:fullName="Dave Beckett">
10:        <ex:homePage rdf:resource="http://purl.org/net/dajobe/" />
11:      </rdf:Description>
12:    </ex:editor>
13:  </rdf:Description>
14: </rdf:RDF>
```

### Graph of the data model



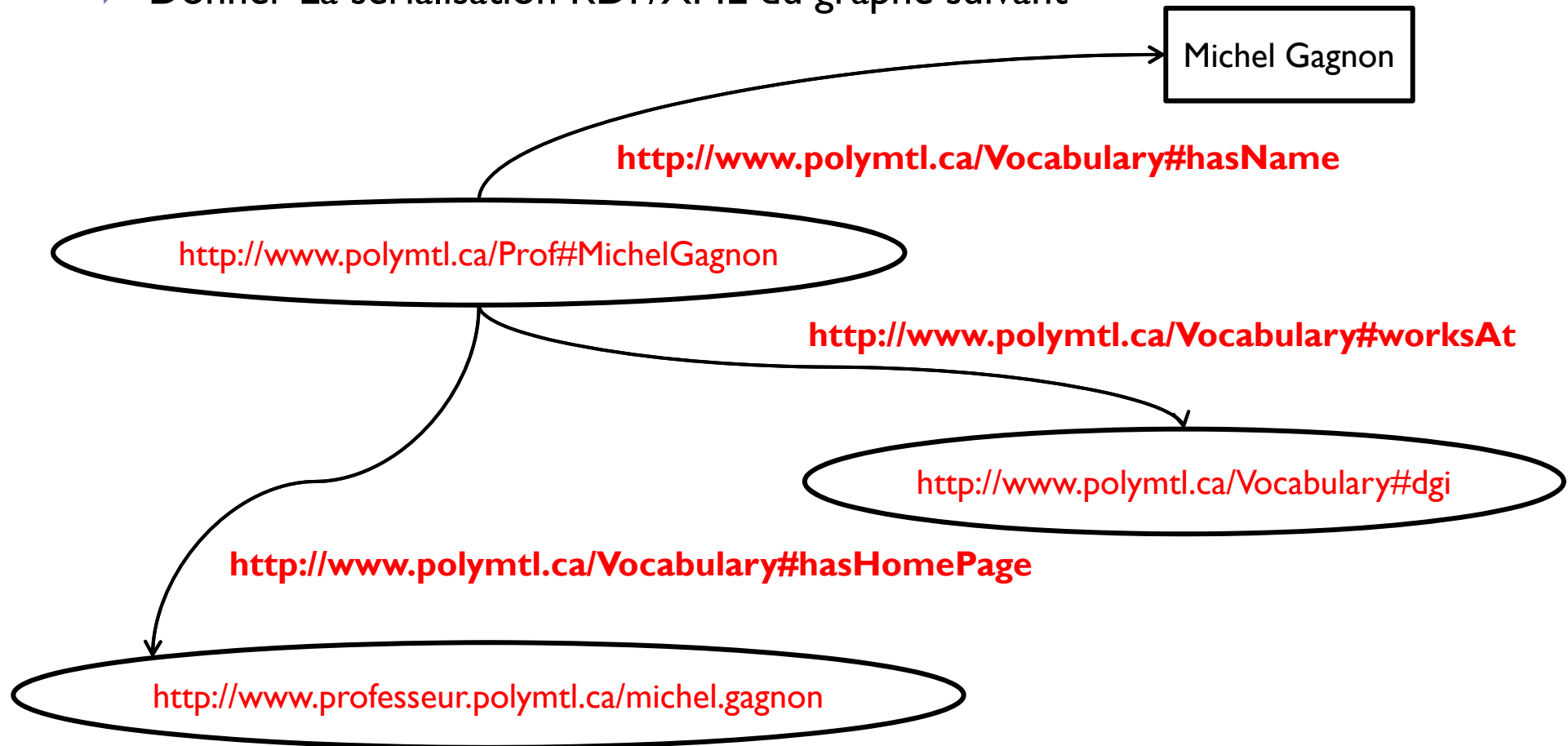


# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML

### Exercice 4:

- Donner La sérialisation RDF/XML du graphe suivant





## Exercise 4

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <rdf:RDF
3     xmlns:local="http://www.polymtl.ca/Vocabulary#"
4     xmlns:prof="http://www.polymtl.ca/Profs#"
5     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
6
7 <rdf:Description rdf:about="http://www.polymtl.ca/Profs#MichelGagnon">
8     <local:worksAt rdf:resource="http://www.polymtl.ca/Vocabulary#dgi"/>
9 </rdf:Description>
10
11 <rdf:Description rdf:about="http://www.polymtl.ca/Profs#MichelGagnon">
12     <local:hasName>Michel Gagnon</local:hasName>
13 </rdf:Description>
14
15 <rdf:Description rdf:about="http://www.polymtl.ca/Profs#MichelGagnon">
16     <local:hasHomePage rdf:resource="http://www.professeurs.polymtl.ca/michel.gagnon"/>
17 </rdf:Description>
18 </rdf:RDF>
```





## Exercise 4

---

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <rdf:RDF
3     xmlns:local="http://www.polymtl.ca/Vocabulary#"
4     xmlns:prof="http://www.polymtl.ca/Profs#"
5     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
6
7 <rdf:Description rdf:about="prof:MichelGagnon">
8     <local:worksAt rdf:resource="local:dgi"/>
9 </rdf:Description>
10
11 <rdf:Description rdf:about="prof:MichelGagnon">
12     <local:hasName>Michel Gagnon</local:hasName>
13 </rdf:Description>
14
15 <rdf:Description rdf:about="prof:MichelGagnon">
16     <local:hasHomePage rdf:resource="http://www.professeurs.polymtl.ca/michel.gagnon"/>
17 </rdf:Description>
18 </rdf:RDF>
```



## Exercise 4

---

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <rdf:RDF
3   xmlns:local="http://www.polymtl.ca/Vocabulary#"
4   xmlns:prof="http://www.polymtl.ca/Profs#"
5   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
6
7   <rdf:Description rdf:about="prof:MichelGagnon">
8     <local:worksAt rdf:resource="local:dgi"/>
9     <local:hasName>Michel Gagnon</local:hasName>
10    <local:hasHomePage rdf:resource="http://www.professeurs.polymtl.ca/michel.gagnon"/>
11  </rdf:Description>
12 </rdf:RDF>
```





## Exercise 4

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <rdf:RDF
3     xmlns:local="http://www.polymtl.ca/Vocabulary#"
4     xmlns:prof="http://www.polymtl.ca/Profs#"
5     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
6
7     <rdf:Description rdf:about="prof:MichelGagnon" local:hasName="Michel Gagnon">
8         <local:worksAt rdf:resource="local:dgi"/>
9         <local:hasHomePage rdf:resource="http://www.professeurs.polymtl.ca/michel.gagnon"/>
10    </rdf:Description>
11 </rdf:RDF>
```

# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML – Langues



**Langues:** **xml:lang**

- ▶ RDF/XML permet l'utilisation de l'attribut **xml:lang** pour l'identification de la langue du contenu.
- ▶ L'attribut **xml:lang** peut être utilisé sur tout **élément nœud** ou **élément propriété** pour indiquer que le contenu inclus est dans la langue donnée.

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">

  <rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
    <dc:title>RDF 1.1 XML Syntax</dc:title>
    <dc:title xml:lang="en">RDF 1.1 XML Syntax</dc:title>
    <dc:title xml:lang="en-US">RDF 1.1 XML Syntax</dc:title>
  </rdf:Description>
</rdf:RDF>
```

# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML – littéraux XML



**littéraux XML :** `rdf:parseType="Literal"`

- ▶ RDF permet aux **littéraux XML** d'être définie en tant que **nœud objet** d'un **prédicat**.
- ▶ Ils sont écrites en tant que contenu d'un élément propriété à l'aide de l'attribut `rdf:parseType="Literal"`.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://example.org/stuff/1.0/">

  <rdf:Description rdf:about="chap3">
    <ex:code rdf:parseType="Literal">
      <p>contenu <b>contenu</b> contenu</p>
    </ex:code>
  </rdf:Description>
</rdf:RDF>
```



# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML – Littéraux typés

### Littéraux typés : **rdf:datatype**

- ▶ RDF permet aux littéraux typés d'être donnés en tant que **nœud objet** d'un **prédicat**.
- ▶ Ils sont écrites en tant que contenu d'un élément propriété (pas d'attribut de propriété) à l'aide de l'attribut supplémentaire **rdf:datatype="datatypeURI"** de l'élément de propriété.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://example.org/stuff/1.0/">

  <rdf:Description rdf:about="http://example.org/item01">
    <ex:size rdf:datatype="http://www.w3.org/2001/XMLSchema#int">
      123
    </ex:size>
  </rdf:Description>
</rdf:RDF>
```



# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML – Nœuds Vides

### Identification des nœuds vides: **rdf:nodeID**

- ▶ Un identifiant peut être attribuer au nœud vide pour l'identifier dans un document RDF/XML
  - ▶ Un **identifiant de nœud vide** est utilisé :
    - ▶ sur un élément nœud pour remplacer **rdf:about="IRI"** ou
    - ▶ sur un élément propriété pour remplacer **rdf:resource="IRI"**
- Par **rdf:nodeID="identifiant de nœud vierge"** dans les deux cas.

### Exemple:



# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML – Nœuds Vides

Identification des nœuds vides: **rdf:nodeID**

### Exemple:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:ex="http://example.org/stuff/1.0/">

  <rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar"
    dc:title="RDF 1.1 XML Syntax">
    <ex:editor rdf:nodeID="abc"/>
  </rdf:Description>
  <rdf:Description rdf:nodeID="abc" ex:fullName="Dave Beckett">
    <ex:homePage rdf:resource="http://purl.org/net/dajobe/" />
  </rdf:Description>
</rdf:RDF>
```



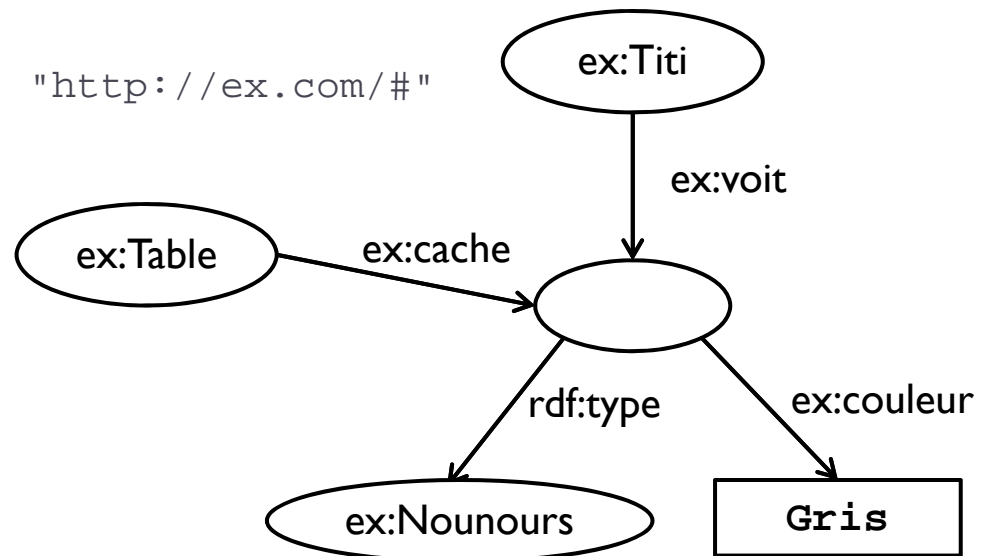
# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML – N uds Vides

### ► Exercice : 05

Le pr fixe `ex` remplace l'IRI

"http://ex.com/#"



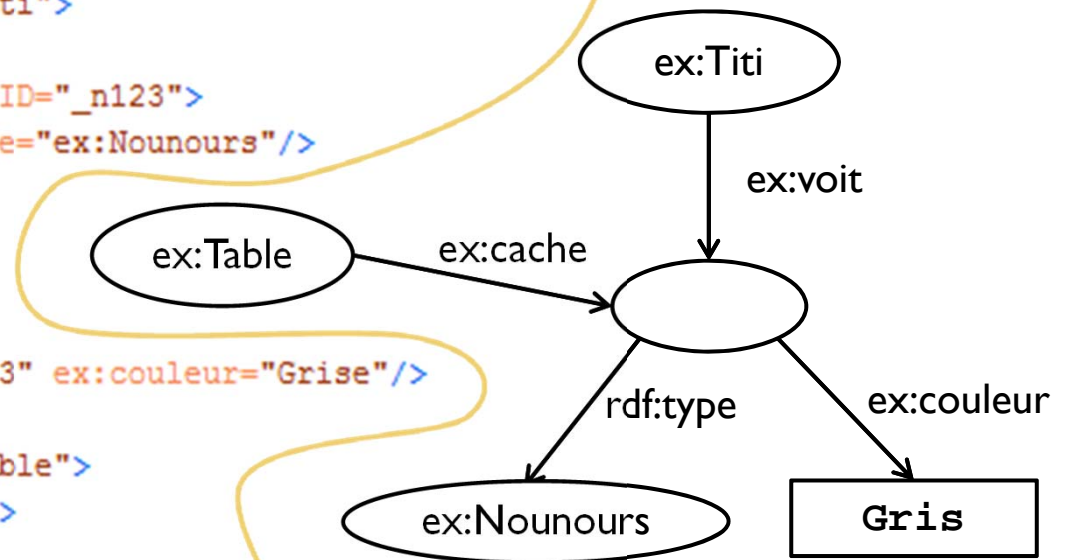


# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML – N uds Vides



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:ex="http://ex.com/#">
4
5   <rdf:Description rdf:about="ex:Titi">
6     <ex:voit>
7       <rdf:Description rdf:nodeID="_n123">
8         <rdf:type rdf:resource="ex:Nounours"/>
9       </rdf:Description>
10    </ex:voit>
11  </rdf:Description>
12
13  <rdf:Description rdf:nodeID="_n123" ex:couleur="Grise"/>
14
15  <rdf:Description rdf:about="ex:Table">
16    <ex:cache rdf:nodeID="_n123"/>
17  </rdf:Description>
18
19 </rdf:RDF>
```







## III - Sérialisation des graphes RDF

---

### Conteneurs et Collections (RDF/XML)



## III - S rialisation des graphes RDF

### Syntaxe - RDF/XML

---

#### Conteneurs

- ▶ Un **conteneur** est une sorte de ressource (groupe ouvert) qui contient d'autres ressources avec  ventuellement des doublons,
- ▶ Avec RDF on peut sp cifier qu'une ressource est un **conteneur** et indiquer les relations entre cette ressource et les entit s qu'elle contient.
- ▶ RDF propose trois classes de conteneur :
  - ▶ **rdf:Bag**: les membres n'ont aucun ordre entre eux.
  - ▶ **rdf:Seq**: suppose l'existence d'un ordre entre les membres.
  - ▶ **rdf:Alt**: pr sente des alternatives parmi lesquelles on s'attend   ce qu'une seule soit s lectionn e.



# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML

---

- ▶ Le conteneur est **relié à chacun des ses membres** par une relation `rdf:_n`, où `n` est un entier.
- ▶ RDF n'interdit pas d'avoir deux membres avec la **même valeur** `n`, même avec le conteneur `rdf:Seq`
- ▶ RDF n'interdit pas qu'il y ait des sauts dans la numérotation, on peut décrire un conteneur avec les relations `rdf:_1`, `rdf:_2` et `rdf:_4`.
- ▶ RDF fournit **une abréviation** en utilisant la relation `rdf:li` pour chaque membre, au lieu de la relation spécifique `rdf:_n`.



# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML

### Exemple :

Supposons que les trois professeurs Java-Prof, CPP-Prof et WS-prof sont parmi les membres de d partement d'informatique.

```
1 <?xml version="1.0"?>
2 <rdf:RDF xmlns:exp="http://example.org/"
3   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
4   <rdf:Seq rdf:about="exp:Computer-Science-department">
5     <rdf:_1 rdf:resource="exp:Java-prof"/>
6     <rdf:_5 rdf:resource="exp:IA-prof"/>
7     <rdf:_3 rdf:resource="exp:WS-prof"/>
8   </rdf:Seq>
9 </rdf:RDF>
```





# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML

### Exemple :

Supposons que les trois professeurs Java-Prof, CPP-Prof et WS-prof sont parmi les membres de d partement d'informatique.

```
1 <?xml version="1.0"?>
2 <rdf:RDF xmlns:exp="http://example.org/"
3   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
4   <rdf:Seq rdf:about="exp:Computer-Science-department">
5     <rdf:li rdf:resource="exp:Java-prof"/>
6     <rdf:li rdf:resource="exp:IA-prof"/>
7     <rdf:li rdf:resource="exp:WS-prof"/>
8   </rdf:Seq>
9 </rdf:RDF>
```





# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML

---

- ▶ Un des inconvénients des conteneurs est qu'il n'est pas possible de les considérer comme des ensembles fermés.
- ▶ Ainsi, si dans une description on spécifie l'existence de trois éléments dans un conteneur, on n'a jamais de garantie qu'il ne contient que ces trois éléments.
- ▶ Rien n'empêche de spécifier d'autres éléments de ce conteneur dans une autre description.
- ▶ Pour pallier cette lacune, RDF permet de définir des **collections**, qui sont en quelque sorte des listes.



# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML

---

### Collections

- La liste des membres d'une collection d bute avec une ressource de type `rdf:List`
- On construit une liste de mani re r cursive en utilisant le pr dicat `rdf:first` pour indiquer le premier  l ment de la liste, et le pr dicat `rdf:rest` pour indiquer le reste de la liste, qui est lui-m me une liste.
- La liste vide est repr sent e par une ressource sp ciale pr d finie dont l'IRI est `rdf:nil`.

### Exemple

- Si, par exemple, on sait que les trois professeurs cit s dans le mod le pr c dent sont les seuls du d partement, on peut utiliser une liste au lieu d'un conteneur :



# III - Sérialisation des graphes RDF

## Syntaxe - RDF/XML

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <rdf:RDF xmlns:exp="http://example.org/"
3     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
4
5     <rdf:Description rdf:about="exp:Computer-Science-department">
6         <exp:hasProf>
7             <rdf:List>
8                 <rdf:first rdf:resource="exp:Java-prof"/>
9                 <rdf:rest>
10                    <rdf:List>
11                        <rdf:first rdf:resource="exp:IA-prof"/>
12                        <rdf:rest>
13                            <rdf:List>
14                                <rdf:first rdf:resource="exp:WS-prof"/>
15                                <rdf:rest rdf:resource="rdf:nil"/>
16                            </rdf:List>
17                        </rdf:rest>
18                    </rdf:List>
19                </rdf:rest>
20            </rdf:List>
21        </exp:hasProf>
22    </rdf:Description>
23 </rdf:RDF>
```

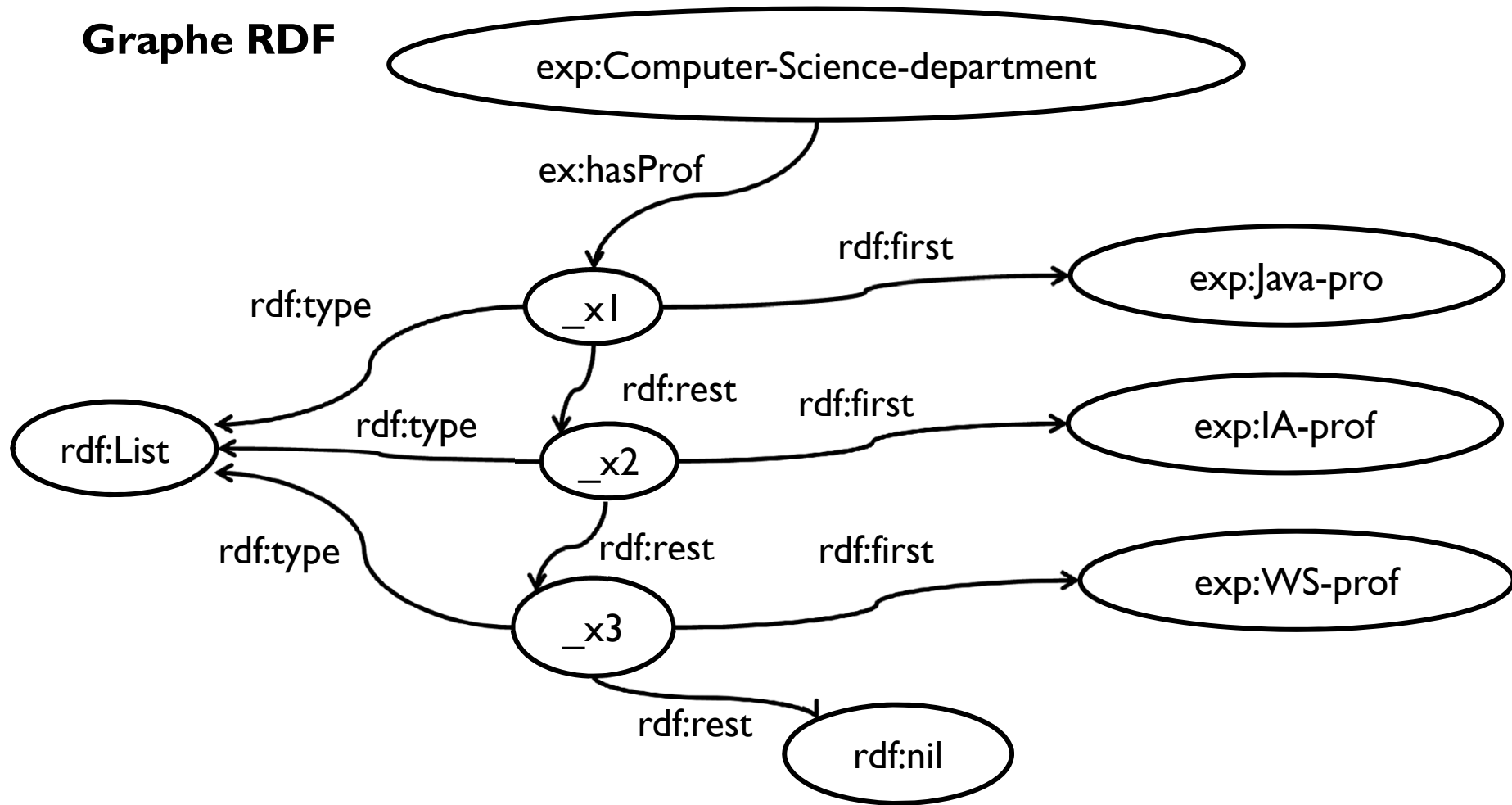




# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML

### Graphe RDF





# III - S rialisation des graphes RDF

## Syntaxe - RDF/XML

### Collections

- Comme cette syntaxe est tr s lourde, il existe donc une abr viation, mais qui ne pourra  tre utilis e que si la liste est l'objet d'un triplet,

```
1  <?xml version="1.0"?>
2  <rdf:RDF      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3      xmlns:ex="http://example.org/"
4
5      <rdf:Description rdf:about="exp:Computer-Science-department">
6          <ex:hasProf rdf:parseType="Collection">
7              <rdf:Description rdf:about="exp:Java-pro"/>
8              <rdf:Description rdf:about="exp:IA-prof"/>
9              <rdf:Description rdf:about="exp:WS-prof"/>
10         </ex:hasProf>
11     </rdf:Description>
12 </rdf:RDF>
```

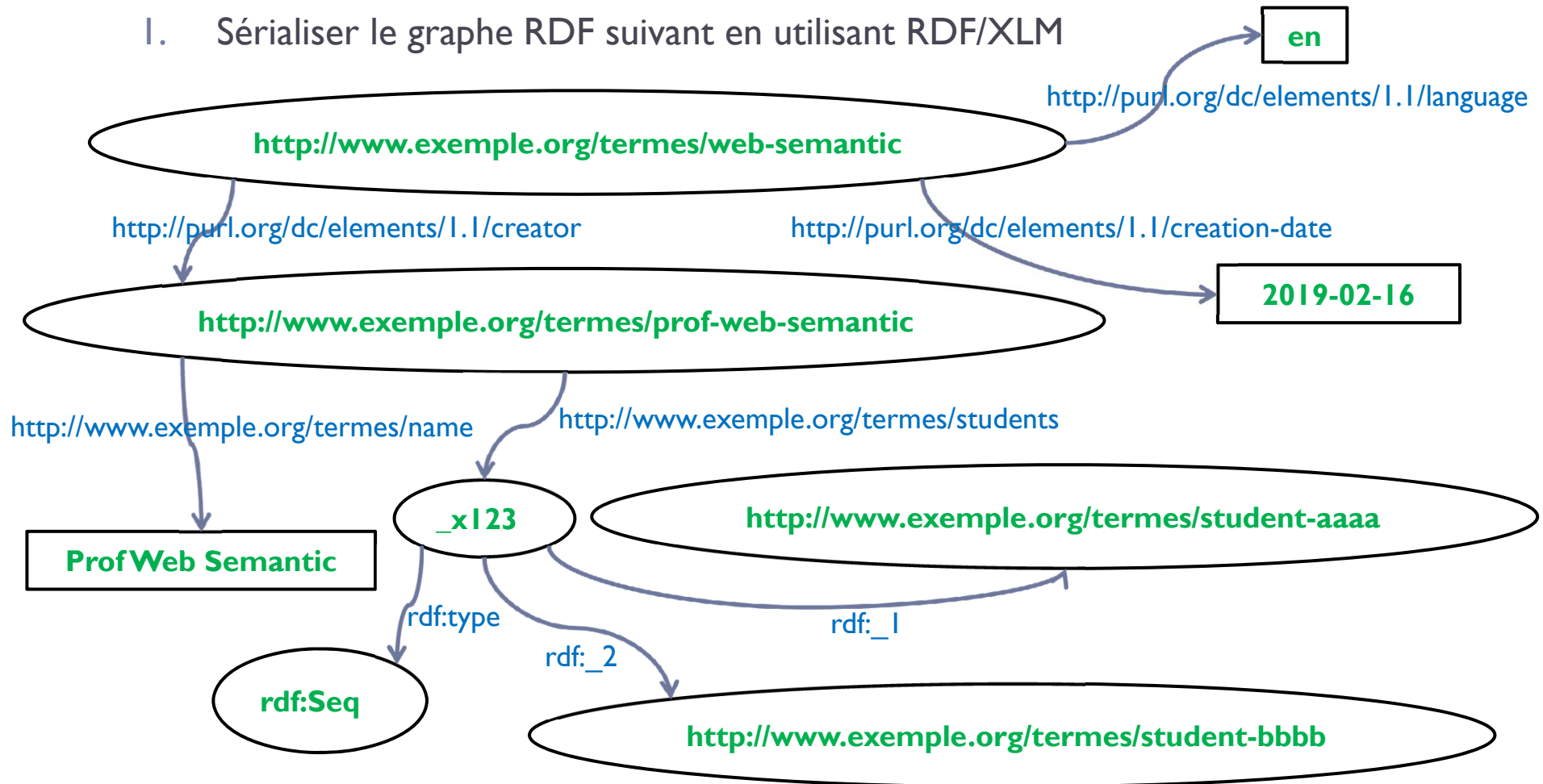


## II - Modèle de Données RDF

### Exercices

#### ► Exercice 6

1. Sérialiser le graphe RDF suivant en utilisant RDF/XML





## Plan

---

# **Chapitre 2** **RDF & Syntaxes RDF**

I – Introduction

II – Modèle de Données RDF

**III – Sérialisation des graphes RDF**

Syntaxe – RDF/XML

**Syntaxe – Turtle**