



- RDF (1990) < XML (1996)
- RDF (1990) pour représenter des méta données des pages Web (auteurs, copyright)
- RDF(2004) pour le Web sémantique

http://exemple.org/entraîne

http://www.ffr.fr/  
entraîneur/RemiLaurent

http://lorraine.ffr.fr/RCLunevillois

http://exemple.org/prénom

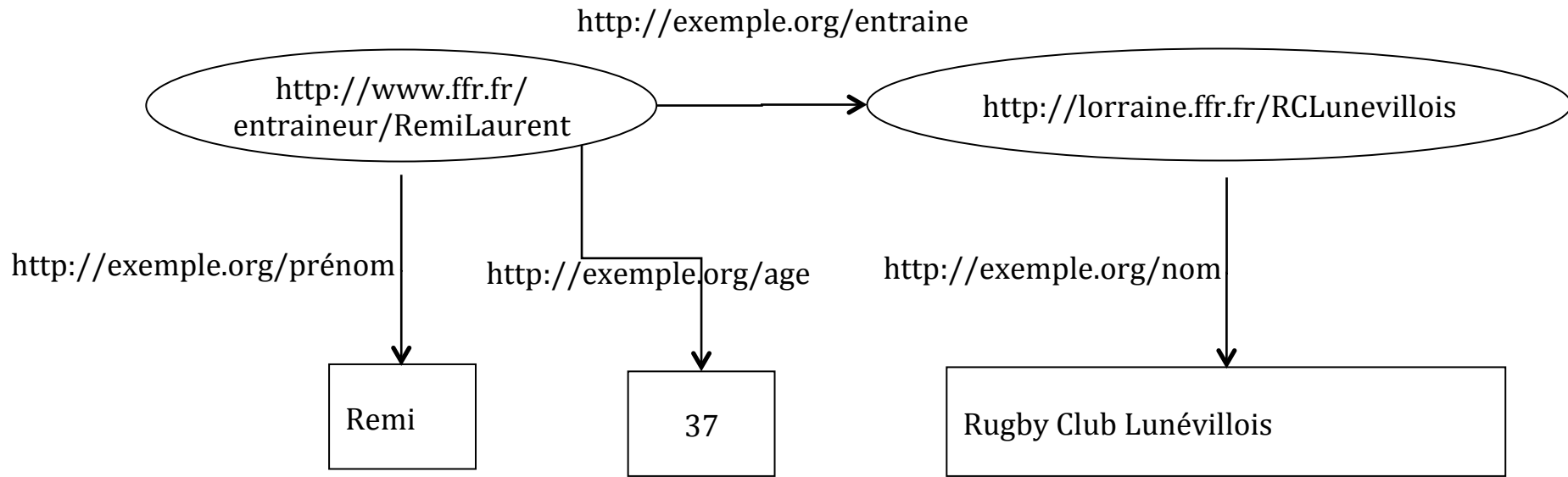
http://exemple.org/age

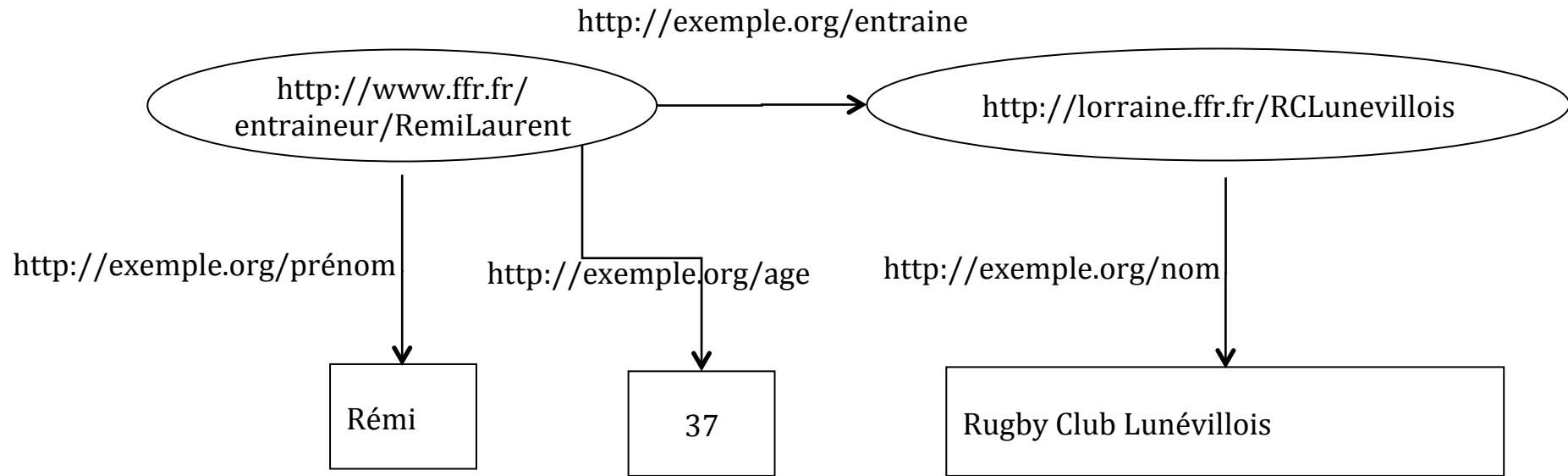
http://exemple.org/nom

Remi

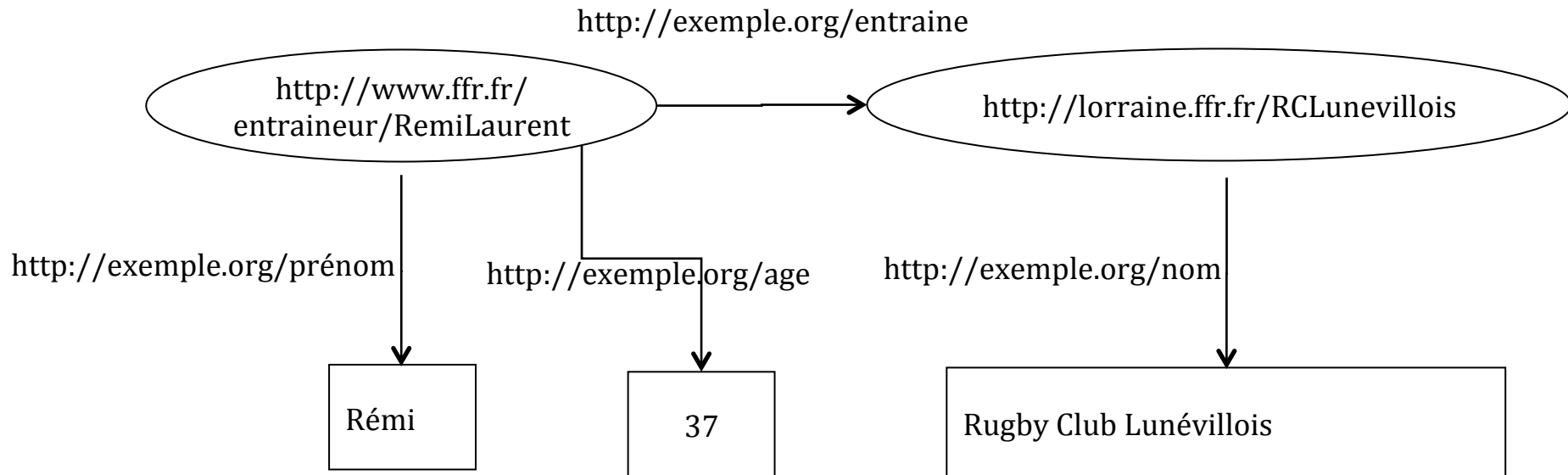
37

Rugby Club Lunévillois





<`http://www.ffr.fr/entraîneur/RemiLaurent`> <`http://exemple.org/entraîne`> <`http://lorraine.ffr.fr/RCLunevillois`> .  
<`http://lorraine.ffr.fr/RCLunevillois`> <`http://exemple.org/nom`> "Rugby Club Lunévillois" .



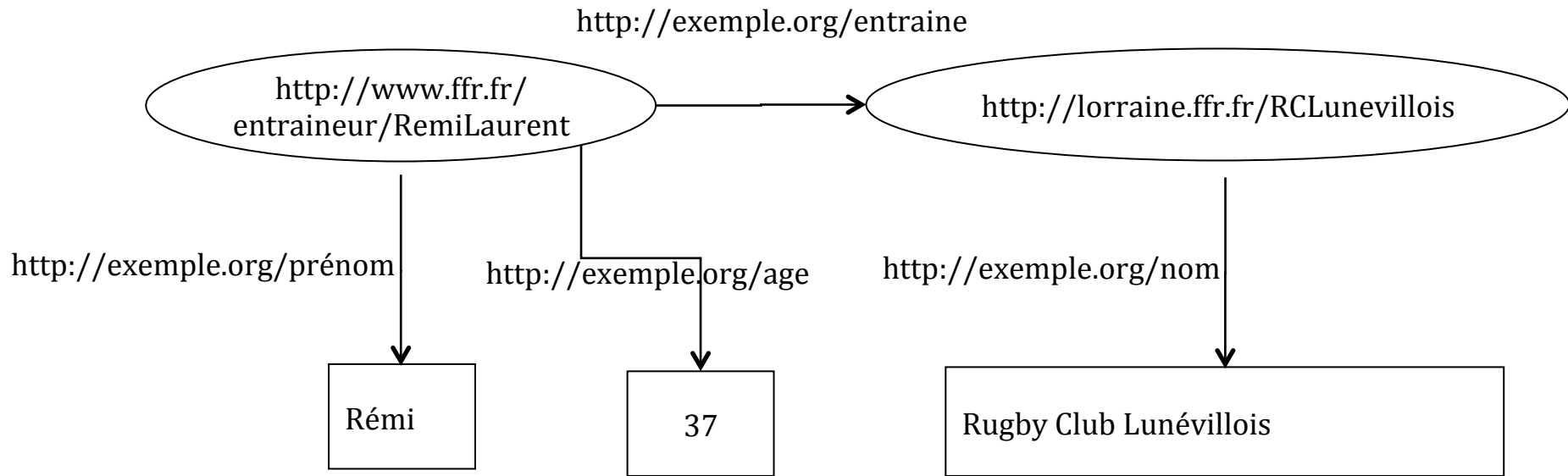
**@prefix fede : <http://lorraine.ffr.fr/> .**

**@prefix ex : <http://exemple.org/> .**

**@prefix entr : <http://www.ffr.fr/entraîneur/> .**

fede: RCLunevillois      ex: nom      "Rugby Club Lunévillois".

entr :RemiLaurent      ex: entraîne      fede: RCLunevillois .  
 entr :RemiLaurent      ex : age      "37" .



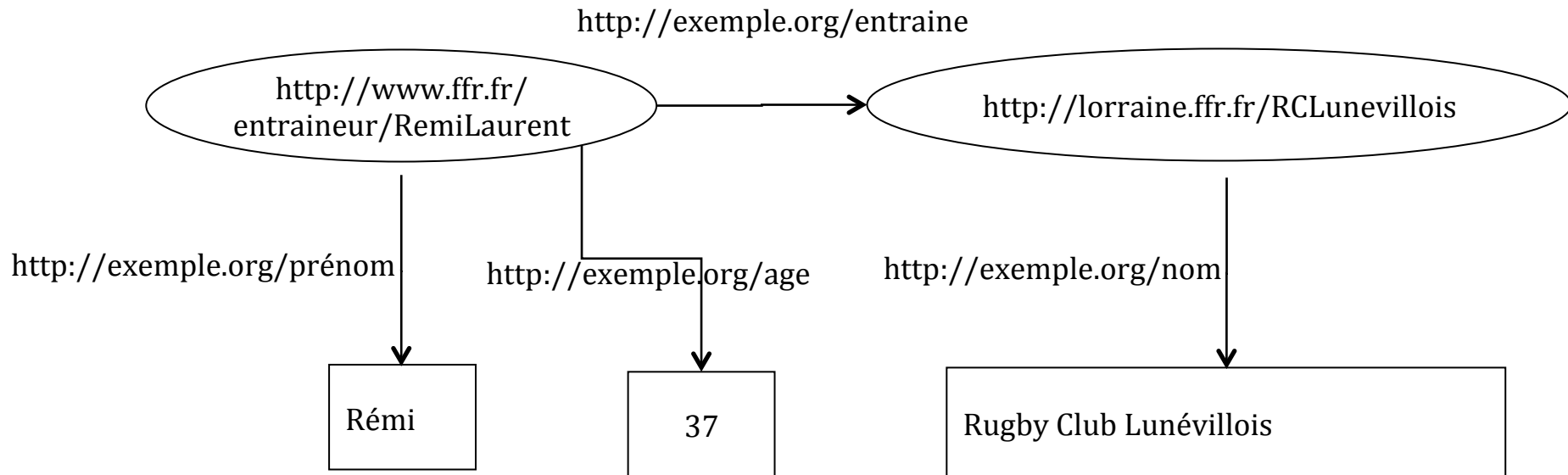
@prefix fede : <<http://lorraine.ffr.fr/>> .

@prefix ex : <http://example.org/> .

@prefix entr : <<http://www.ffr.fr/entraîneur/>> .

fede: RCLunevillois      ex: nom      "Rugby Club Lunévillois".

```
entr :RemiLaurent      ex: entraine   fede: RCLunevillois ;
                        ex : age      "37" .
```



@prefix fede : <http://lorraine.ffr.fr/> .

@prefix ex : <http://exemple.org/> .

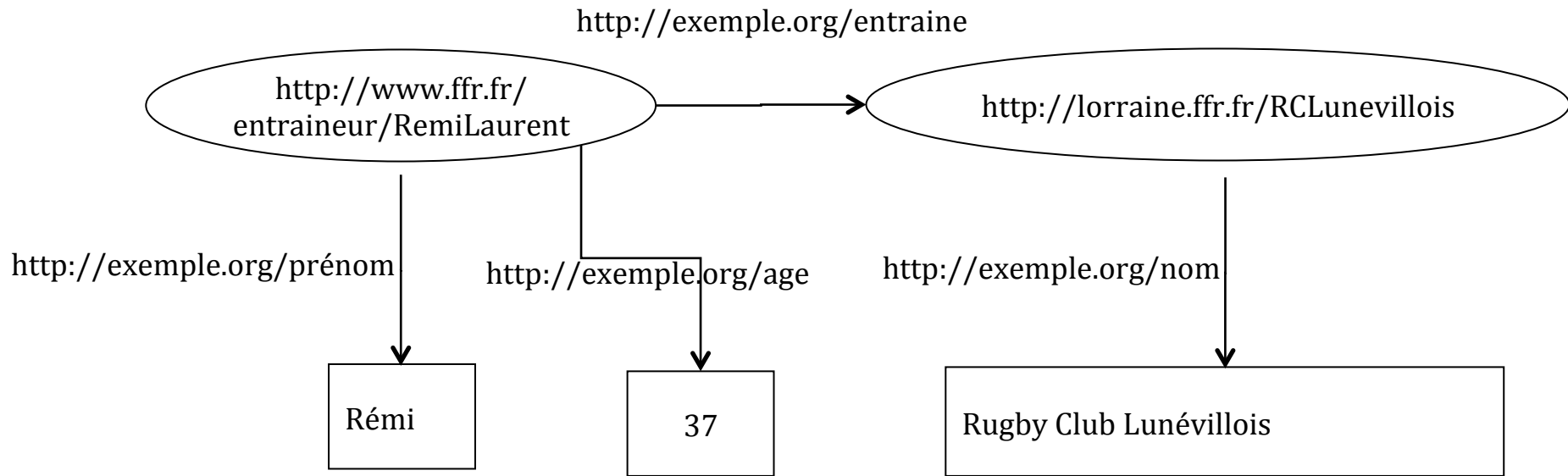
@prefix entr : <http://www.ffr.fr/entraîneur/> .

fede: RCLunevillois      ex: nom      "Rugby Club Lunévillois".

entr :RemiLaurent      ex: entraîne      fede: RCLunevillois ;  
ex : age      "37" .

fede: RCLunevillois      ex: nom "Rugby Club Lunévillois" .

fede: RCLunevillois      ex: nom "RC Lunevillois" .



@prefix fede : <http://lorraine.ffr.fr/> .

@prefix ex : <http://exemple.org/> .

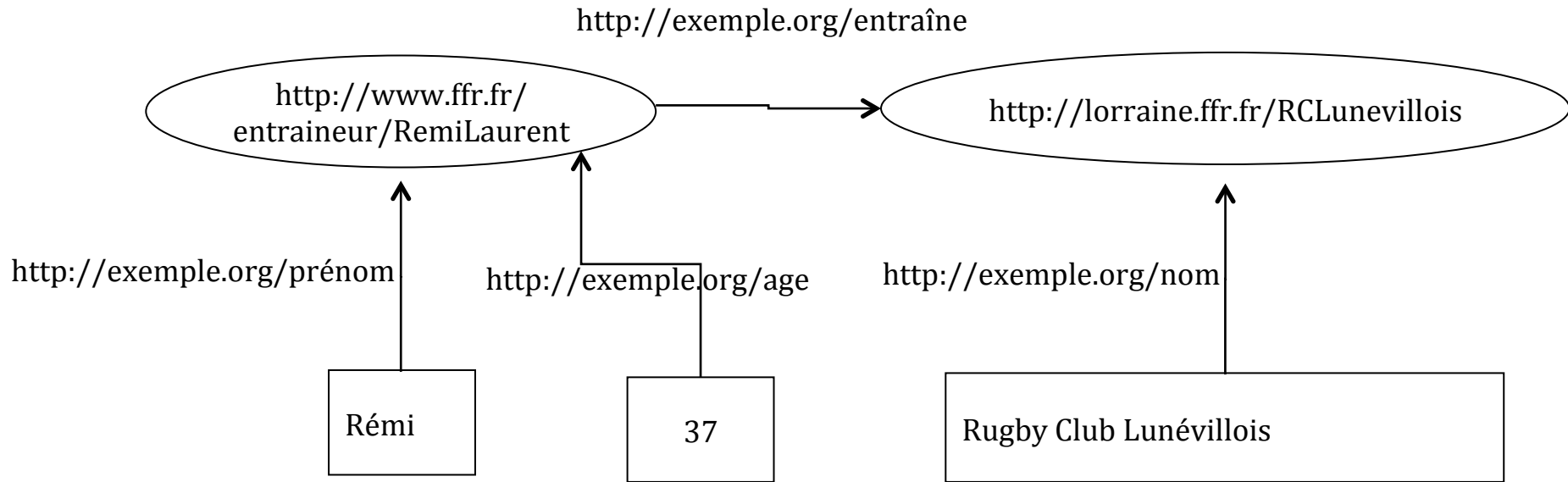
@prefix entr : <http://www.ffr.fr/entraîneur/> .

fede: RCLunevillois      ex: nom      "Rugby Club Lunévillois".

entr :RemiLaurent      ex: entraîne      fede: RCLunevillois ;  
ex : age      "37" .

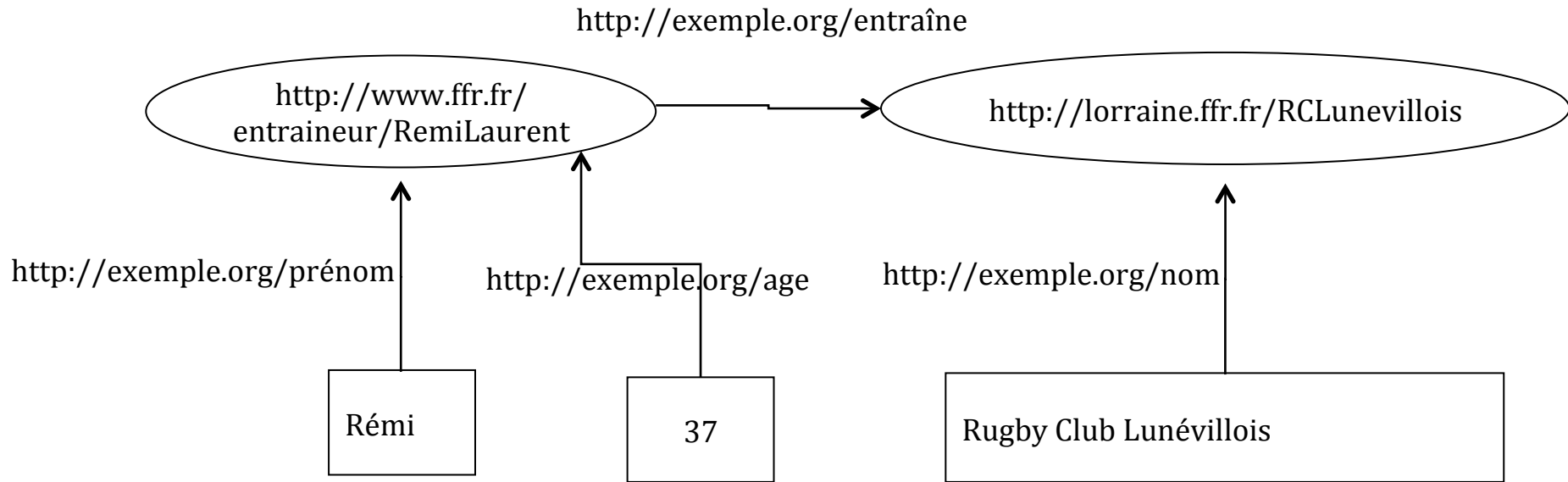
fede: RCLunevillois      ex: nom "Rugby Club Lunévillois",  
"RC Lunévillois" .





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

  <rdf:Description rdf:about="http://www.ffr.fr/entraîneur/RemiLaurent">
    <ex:entraîne>
      <rdf:Description rdf:about="http://lorraine.ffr.fr/RCLunevillois">
      </ex:entraîne>
    </rdf:Description>
  </rdf:RDF>
```



```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://exemple.org/">
```

```
  <rdf:Description rdf:about="http://www.ffr.fr/entraîneur/RemiLaurent">
```

```
    <ex:prenom>Rémi</ex:prenom>
```

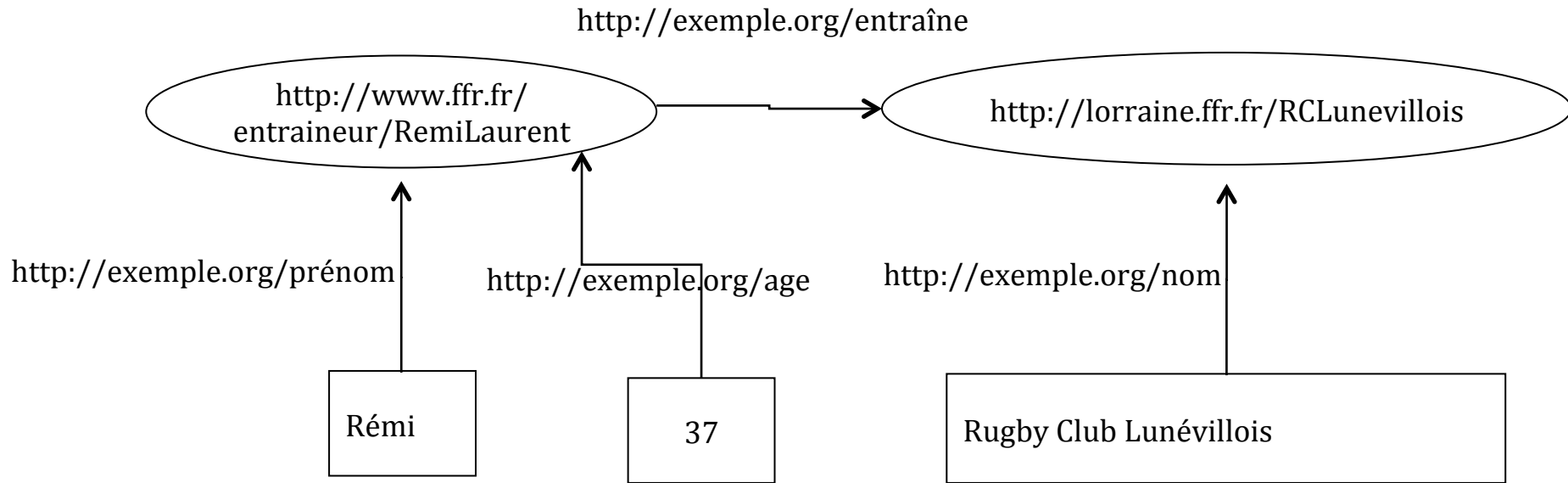
```
    <ex:entraîne>
```

```
      <rdf:Description rdf:about="http://lorraine.ffr.fr/RCLunevillois">
```

```
    </ex:entraîne>
```

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

```
</rdf:RDF>
```



```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://exemple.org/">
```

```
<rdf:Description rdf:about="http://www.ffr.fr/entraîneur/RemiLaurent"
```

```
  ex:prenom="Remi">
```

```
    <ex:entraîne rdf:resource="http://lorraine.ffr.fr/RCLunevillois">
```

```
  </ex:entraîneLEquipe>
```

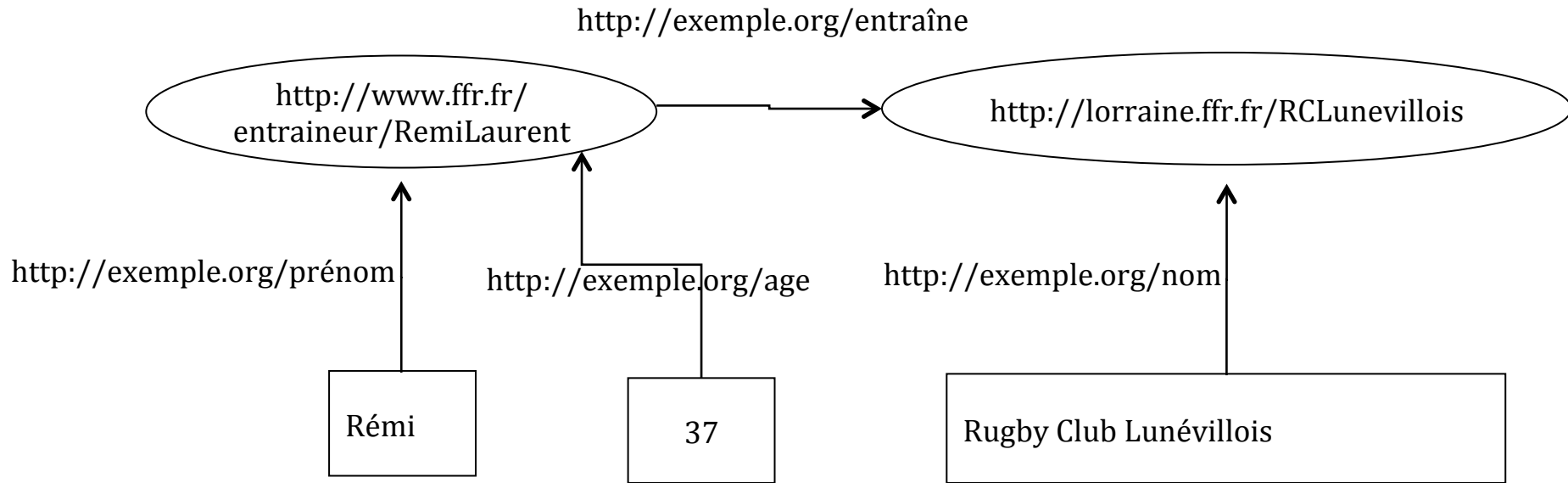
```
</rdf:Description>
```

```
<rdf:Description rdf:about="http://lorraine.ffr.fr/RCLunevillois"
```

```
  ex:name="RC Vandoeuvre">
```

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

```
</rdf:RDF>
```



```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://exemple.org/"
  xmlns:entr="http://www.ffr.fr/entraîneur/">
```

```
<rdf:Description rdf:about="entr;RemiLaurent"
  ex:prenom="Remi">
    <ex:entraîne rdf:resource="http://lorraine.ffr.fr/RCLunevillois">
    </ex :entraîneLEquipe>
</rdf:Description>
<rdf:Description rdf:about="http://lorraine.ffr.fr/RCLunevillois"
  ex:name="RC Vandoeuvre">
</rdf:Description>
f:RDF>
```

@prefix fede: <http://lorraine.ffr.fr/> .  
@prefix ex: <http://exemple.org/> .  
@prefix entr: <http://www.ffr.fr/entraîneur/> .  
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

entr: RemiLaurent      ex: entraine      fede: RCLunevillois ;  
ex: age      "37"^^xsd:integer ;  
ex: prénom      "Rémi"^^xsd:string ;  
ex: ddn      "1956-02-10"^^xsd:date .

@prefix fede: <http://lorraine.ffr.fr/> .  
@prefix ex: <http://exemple.org/> .  
@prefix entr: <http://www.ffr.fr/entraîneur/> .  
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

entr: RemiLaurent      ex: entraine      fede: RCLunevillois ;  
ex: age      "37"^^xsd:integer ;  
ex: prénom      "Rémi"^^xsd:string ;  
ex: ddn      "1956-02-10"^^xsd:date .

```
<rdf:Description rdf:about="http://www.ffr.fr/entraîneur/RemiLaurent">
  <ex:entraine>
    <rdf :Description rdf:about="http://lorraine.ffr.fr/RCLunevillois">
  </ex:entraine>
  <ex:prénom rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    Rémi
  </ex:prénom >
  <ex:ddn rdf:datatype="http://www.w3.org/2001/XMLSchema#date">
    1956-02-10
  </ex:ddn>
</rdf:Description>
```

entr: RemiLaurent

ex: prénom

"Rémi"@fr , "Remigio"@it .

```
<ex:prénom xml:lang="fr" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
```

Rémi

```
</ex:prénom >
```

```
<ex:prénom xml:lang="it" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
```

Remigio

```
</ex:prénom >
```

```
@prefix ex:          <http://exemple.org/> .
ex:Chutney           ex:contientIngrédient    "300g de mangue verte";
                     ex:contientIngrédient    "1càc de piment de Cayenne"
```



```
@prefix ex:      <http://exemple.org/> .
ex:Chutney      ex:contientIngrédient  ex:mangueVerte;
                ex:quantité            "300g";
                ex:contientIngrédient  ex:pimentDeCayenne;
                ex:quantité            "1càc".
```

@prefix ex:	<http://exemple.org/>	.
ex:Chutney	ex:contientIngrédient	ex:mangueVerte;
	ex:contientIngrédient	ex:pimentDeCayenne.
ex:mangueVerte	ex:quantité	"300g";
ex:pimentDeCayenne	ex:quantité	"1càc".

@prefix ex:	<http://exemple.org/> .	
ex:Chutney	ex:contientIngrédient	ex:mangueVerte;
	ex:contientIngrédient	ex:pimentDeCayenne.
ex:mangueVerte	ex:quantité	"300g".
ex:pimentDeCayenne	ex:quantité	"1càc".
ex:SaladeDeMangue	ex:contientIngrédient	ex:mangueVerte;
	ex:contientIngrédient	ex:banane.
ex:mangueVerte	ex:quantité	"100g".
ex:banane	ex:quantité	"100g".

```
@prefix ex:      <http://exemple.org/> .
ex:Chutney       ex:contientIngrédient  ex:ingrédient1;
                 ex:contientIngrédient  ex:ingrédient2.
ex:ingrédient1   ex:ingrédient          ex:mangueVerte;
                 ex:quantité            "300g".
ex:ingrédient2   ex:ingrédient          ex:pimentDeCayenne;
                 ex:quantité            "1càc".
```

### Exemple en RDF/XML

```
<rdf:Description rdf:about="http://exemple.org/Chetnay">
  <ex:contientIngrédient rdf:nodeID="ID1" />
</rdf:Description>
<rdf:Description rdf:nodeID="ID1">
  <ex:ingrédient rdf:resource="http://exemple.org/mangueVerte"/>
  <ex:quantité>300g<ex:quantité/>
</rdf:Description>
```

### Equivalent en Turtle

```
@prefix ex:      <http://exemple.org/> .
ex:Chutney      ex:contientIngrédient _:id1.
_:id1           ex:ingrédient          ex:mangueVerte;
                ex:quantité            "300g".
```

### Exemple en RDF/XML

```
<rdf:Description rdf:about="http://exemple.org/Chetnay">
  <ex:contientIngrédient rdf:nodeID="ID1"/>
</rdf:Description>
<rdf:Description rdf:nodeID="ID1">
  <ex:ingrédient rdf:resource="http://exemple.org/mangueVerte"/>
  <ex:quantité>300g</ex:quantité>
</rdf:Description>
```

### Equivalent sans ID

```
<rdf:Description rdf:about="http://exemple.org/Chetnay">
  <ex:contientIngrédient rdf:parseType="Resource">
    <ex:ingrédient rdf:resource="http://exemple.org/mangueVerte"/>
    <ex:quantité>300g</ex:quantité>
  </ex:contientIngrédient>
</rdf:Description>
```



*"Le détective suppose que le professeur a tué l'appartieur"*

ex:détective

ex:suppose "Le professeur a tué l'appariteur".



*"Le détective suppose que le professeur a tué l'appartieur"*

ex:détective

ex:suppose "Le professeur a tué l'appariteur".

ex:détective

ex:suppose ex:leProfesseurATuéLAppariteur.

*"Le détective suppose que le professeur a tué l'appariteur"*

ex:détective	ex:suppose	"Le professeur a tué l'appariteur".
ex:détective	ex:suppose	ex:leProfesseurATuéLAppariteur.
ex:professeur	ex:aTué	ex:appariteur.

*"Le détective suppose que le professeur a tué l'appariteur"*

ex:détective	ex:suppose	"Le professeur a tué l'appariteur".
ex:détective	ex:suppose	ex:leProfesseurATuéLAppariteur.
ex:professeur	ex:aTué	ex:appariteur.

?????

*"Le détective suppose que le professeur a tué l'appartieur"*

ex:détective	ex:suppose	ex:theoriel
ex:theoriel	rdf:subject	ex:professeur;
	rdf:predicat	ex:aTué;
	rdf:object	ex:appartieur.

*"Le détective suppose que le professeur a tué l'appartieur"*

ex:détective	ex:suppose	ex:theoriel
ex:theoriel	rdf:subject	ex:professeur;
	rdf:predicat	ex:aTué;
	rdf:object	ex:appartieur.

Attention !



# RDF / RDFS

## *instance (RDF)*

```
ex:adrien    rdf:type          ex:associate_professeur .
```

## *hiérarchie de classe*

```
ex:associate_professeur rdf:type rdfs:Class .  
ex:associate_professeur rdfs:subClassOf ex:faculty .
```

## *hiérarchie de propriétés*

## *domaine et co-domaine d'une propriété*

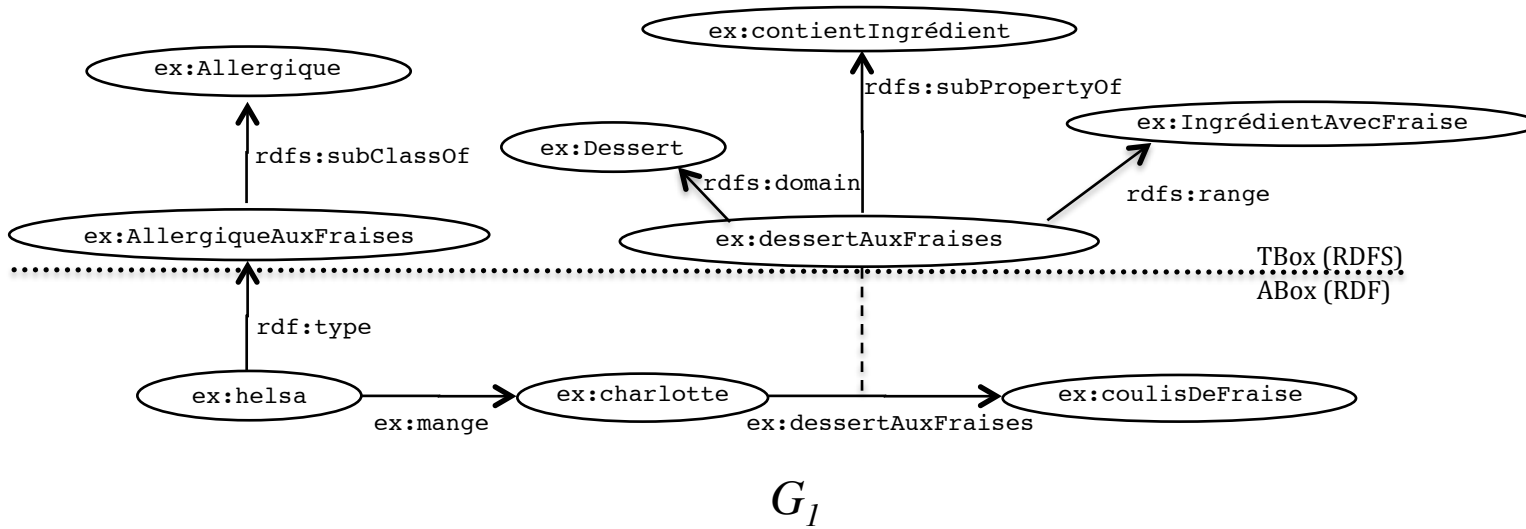
...

ex:estMariéA	rdfs:domain	ex:Personne.
ex:estMariéA	rdfs:range	ex:Personne.
ex:ferme	rdf:type	ex:Bâtiment.

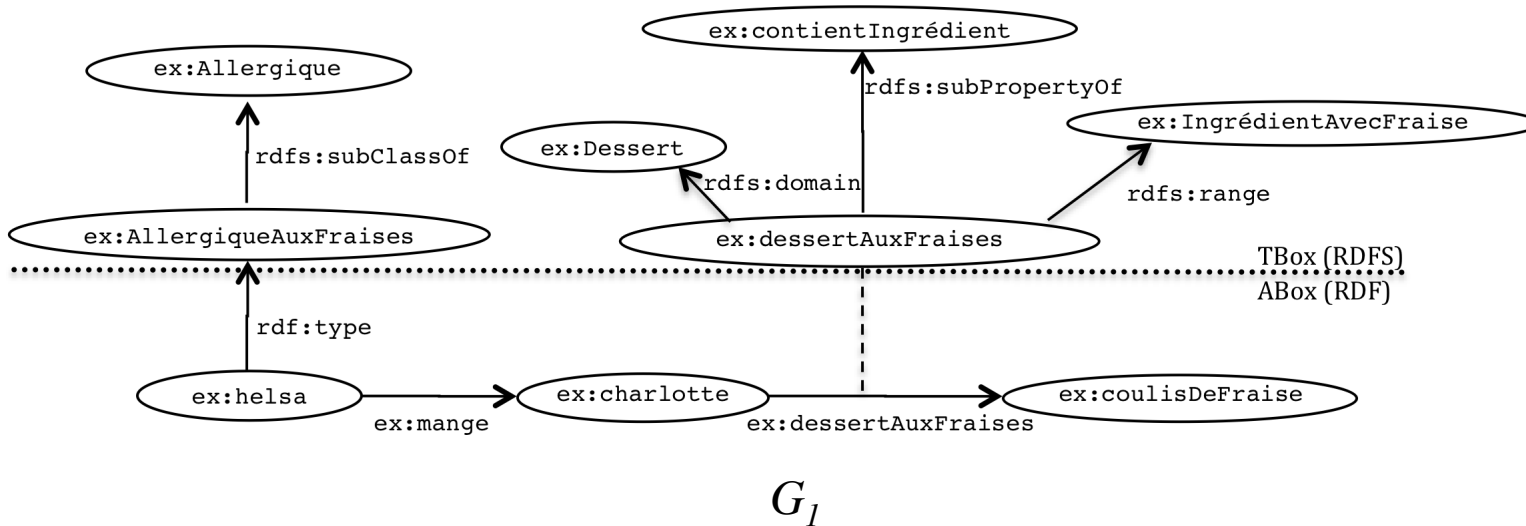
est-ce qu'on peut ajouter

ex:josé	ex:estMariéA	ex:ferme.	?
---------	--------------	-----------	---

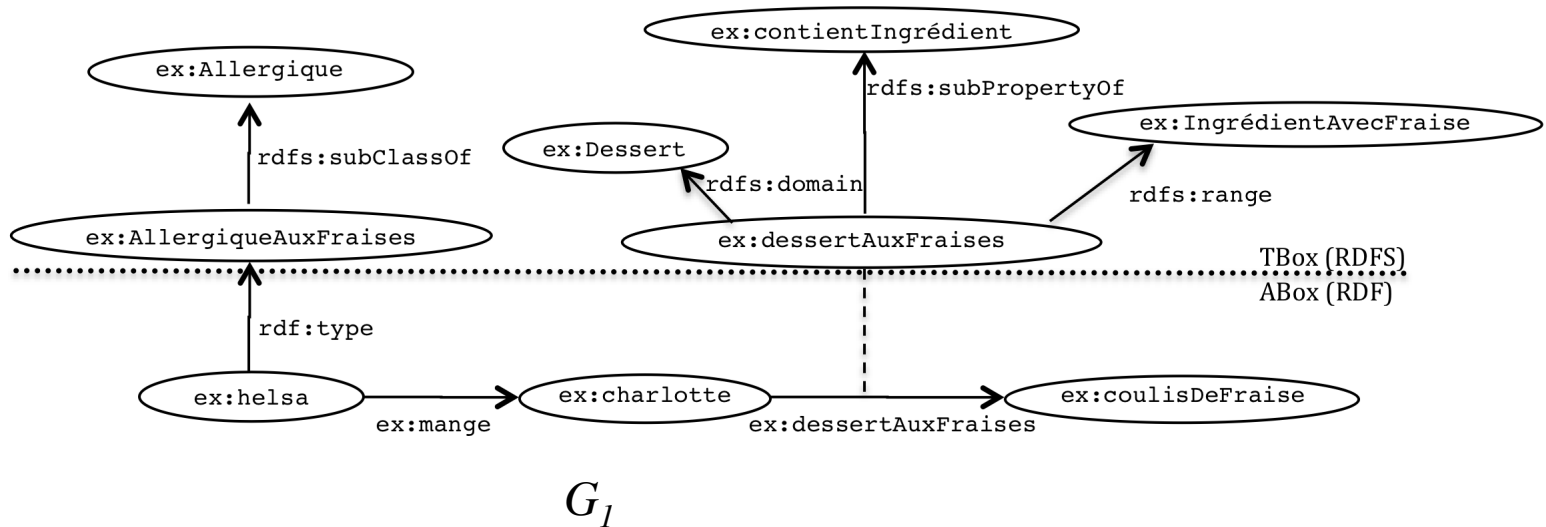




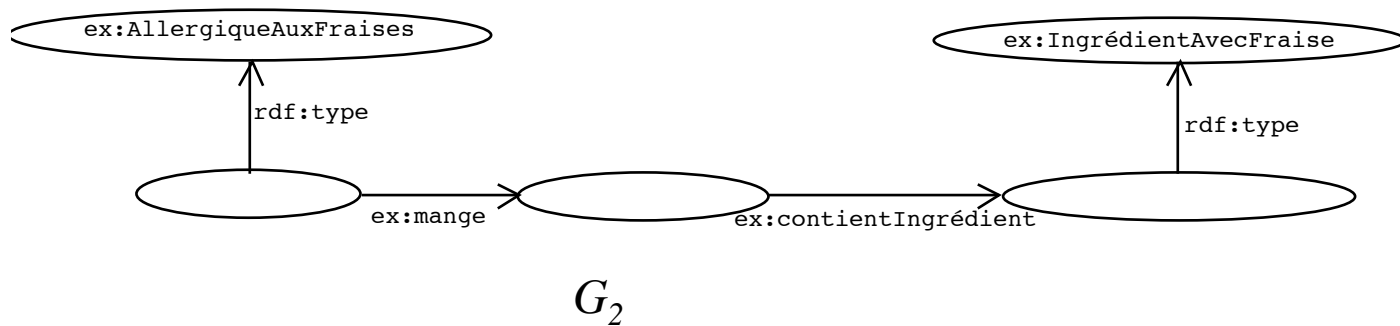
et en pratique ? => voyons Protégé



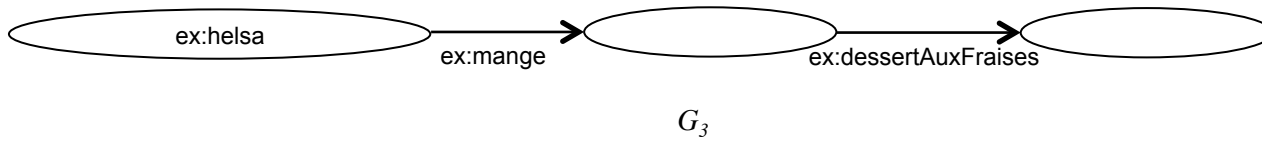
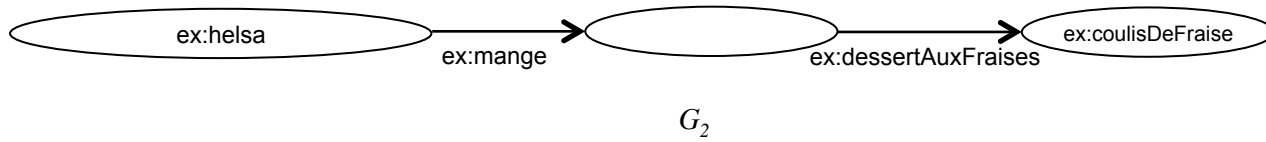
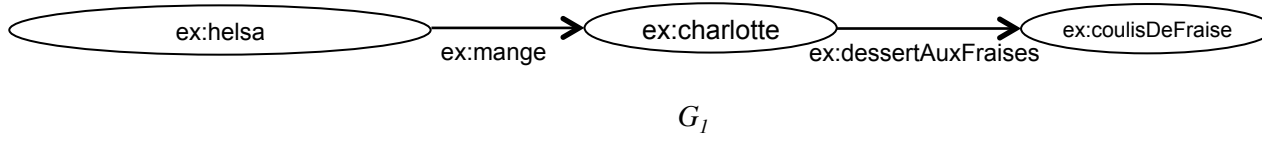
une requête similaire à celle du début :  
*Quelqu'un qui est allergique à la fraise va t-il manger un plat contenant des ingrédients à base de fraise ?*



Quelqu'un qui est allergique à la fraise va t-il manger un plat contenant des ingrédients à base de fraise ?



$$G_1 \models G_2 \quad ?$$



$$G_1 \models G_2, \quad G_2 \models G_3, \quad G_1 \models G_3$$



© Solent News and Photos

$\top$
$\perp$
$C \sqcap D$
$C \sqcup D$
$\neg C$
$\forall R.C$
$\exists R.C$
$\geq n R$
$\leq n R$
$= n R$
$\geq n R.C$
$\leq n R.C$
$= n R.C$
$R \subseteq S$
$R = S$
$u_1 \doteq u_2$
$u_1 \neq u_2$
$I$

owl:Thing  
 owl:Nothing  
 owl:disjointWith\*  
 owl:equivalentClass  
 owl:intersectionOf  
 owl:unionOf\*  
 owl:complementOf\*  
 owl:allValuesFrom  
 owl:someValuesFrom  
 owl:hasValue  
 owl:cardinality\*  
 owl:maxCardinality\*  
 owl:minCardinality\*  
 owl:ObjectProperty  
 owl:DatatypeProperty  
 owl:equivalentProperty

owl:imports

owl:sameAs owl:differentFrom owl:AllDifferent owl:distinctMembers

owl:Thing owl:Nothing

owl:disjointWith\* owl:equivalentClass owl:intersectionOf owl:unionOf\*  
owl:complementOf\*

owl:allValuesFrom owl:someValuesFrom owl:hasValue owl:cardinality\*  
owl:maxCardinality\* owl:minCardinality\*

owl:ObjectProperty owl:DatatypeProperty owl:equivalentProperty

owl:TransitiveProperty owl:SymmetricProperty owl:FunctionalProperty  
owl:InverseFunctionalProperty owl:inverseOf

# écrire en OWL

Patient

Traitement

Antibiotique

aPourTraitement

Antibiotique  $\sqsubseteq$  Traitement

SousAntibiotique  $\equiv$  Patient  $\sqcap \exists$ aPourTraitement.Antibiotique

Patient (philippe)

Antibiotique (penicillin285)

aPourTraitement (philippe, penicillin285)

heureusement il y a Protégé !



# raisonner avec OWL, Protégé et Hermit

Patient

Traitement

Antibiotique

aPourTraitement

Antibiotique  $\sqsubseteq$  Traitement

Toux  $\sqsubseteq$  Symptome

SousAntibiotique  $\equiv$  Patient  $\sqcap \exists$ aPourTraitement.Antibiotique

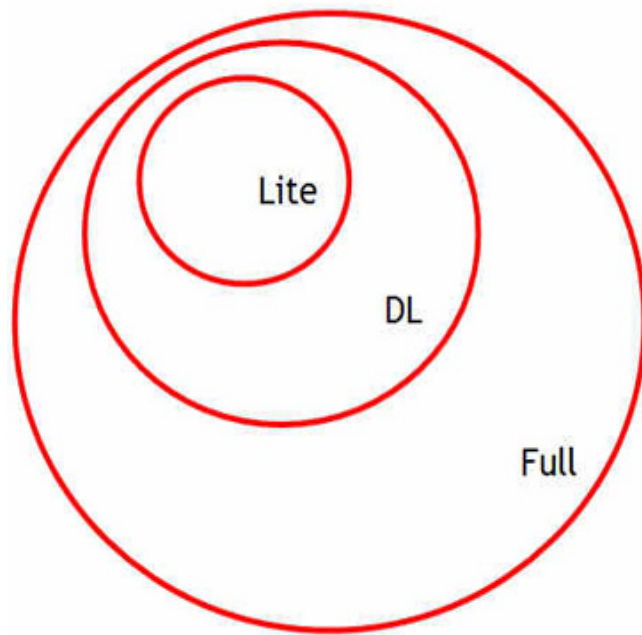
Malade  $\equiv \exists$ aPourSymptome.Symptome

Enrhumé  $\equiv \exists$ aPourSymptome.Toux

Patient (philippe)

Antibiotique (penicillin285)

aPourTraitement (philippe, penicillin285)



# les espèces de OWL 1

- OWL Full
  - indécidable
  - pas d'outil ou de raisonneur
  - notamment à cause du "metamodelling"
  - utilisé pour la représentation de connaissance quand celle-ci n'a pas besoin de mécanisme de raisonnement (*e.g.*, consistency checking)
- OWL DL
  - décidable
  - fragment de la logique des prédicats du premier ordre
  - le plus utilisé
- OWL Lite
  - sensé être plus simple
  - mais il ne l'est pas d'un point de vue computationnel

# les profiles de OWL2

- OWL 2 DL

en  $\mathcal{EL}^{++}$ , pas de  $\forall$ , ni de  $\sqcap$  mais tps polynomial  
prévu pour classifciation de grandes ontologies

- OWL 2 QL

prévu pour permettre des requêtes conjonctives

- OWL 2 RL

prévu pour utiliser des mécanismes de raisonnement à base de règles

- OWL 2 Full

OWL 2 DL + RDFS

Quelle est la LD associée à OWL ?

- $\mathcal{ALC}$

$C, D ::= A \mid \top \mid \perp \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C \mid \neg C$

$\top$  ?

$\perp$  ?

disjonction ?

domain ?

range ?

Quelle est la LD associée à OWL ?

- $\mathcal{ALC}$

$$C, D ::= A \mid \top \mid \perp \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C \mid \neg C$$

- OWL DL utilise  $\mathcal{SHOIN}(\mathcal{D})$
- OWL 2 DL utilise  $\mathcal{SROIQ}(\mathcal{D})$

Quelle est la LD associée à OWL ?

- $\mathcal{ALC}$

$$C, D ::= A \mid \top \mid \perp \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C \mid \neg C$$

- OWL DL utilise  $\mathcal{SHOIN}(\mathcal{D})$ 
  - $\mathcal{S}$  =  $\mathcal{ALC}$  plus role transitif
  - $\mathcal{H}$  = hiérarchie de role
  - $\mathcal{O}$  = nominal i.e.,  $\{a\}$
  - $\mathcal{I}$  = role inverse
  - $\mathcal{N}$  = restriction sur les cardinalités
  - $(\mathcal{D})$  = *types de données*

Quelle est la LD associée à OWL ?

- $\mathcal{ALC}$

$$C, D ::= A \mid \top \mid \perp \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C \mid \neg C$$

- OWL 2 DL utilise  $\mathcal{SROIQ}(\mathcal{D})$

- $\mathcal{S}$  =  $\mathcal{ALC}$  plus role transitif
- ~~$\mathcal{H}$  = hiérarchie de rôle~~  $\mathcal{R}$  = inclusion de rôles généralisée
- $\mathcal{O}$  = nominal i.e.,  $\{a\}$
- $\mathcal{I}$  = rôle inverse
- ~~$\mathcal{N}$  = restriction sur les cardinalités~~  $\mathcal{Q}$  = restriction sur les cardinalités qualifiées
- $(\mathcal{D})$  = types de données