

Open Information Systems 2019-2020

Lecture 2: RDF and RDF(S)

Christophe Debruyne

RDF Schema

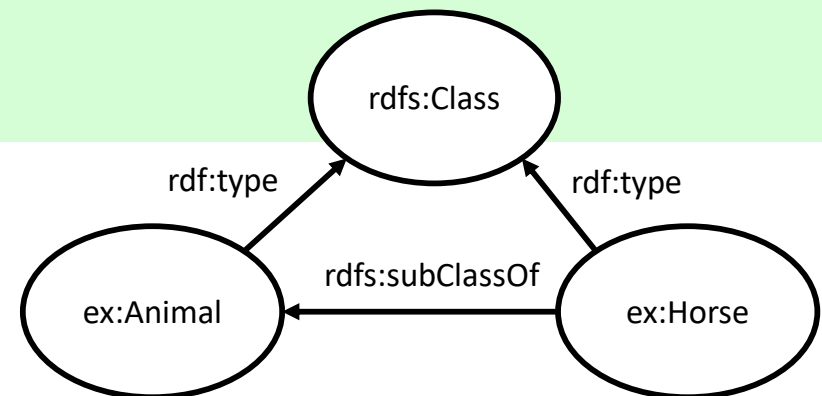


RDF Schema

- RDF(S) is a W3C Recommendation
- RDF(S) is an extension of RDF
- RDF(S) provides a framework to describe vocabularies
- RDF(S) describe resources with classes, properties and values

RDF(S) Example

```
@prefix ex:    <http://www.example.org/animals#> .  
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .  
  
ex:Animal a rdfs:Class .  
  
ex:Horse a rdfs:Class ;  
         rdfs:subClassOf ex:Animal .
```



RDF(S) Example

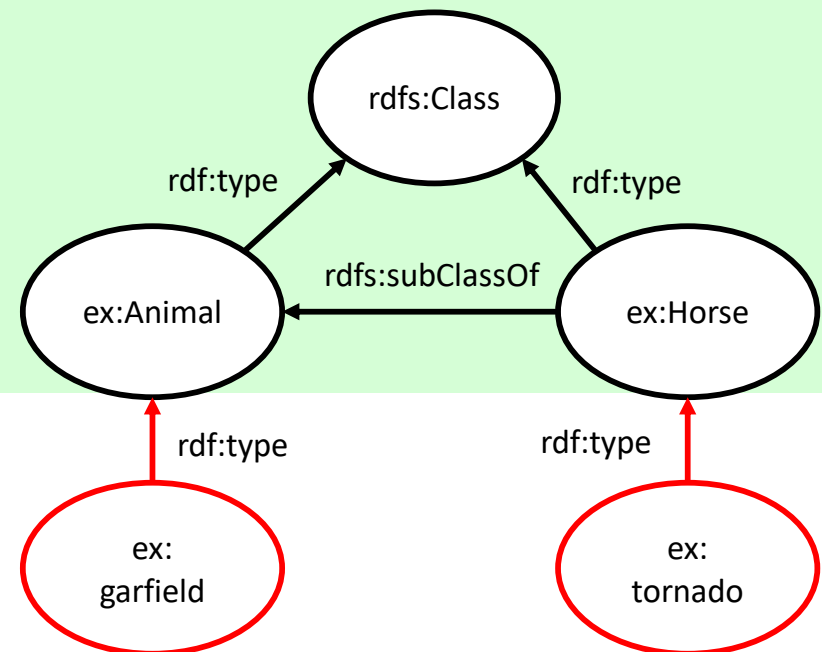
```
@prefix ex:    <http://www.example.org/animals#> .  
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
```

```
ex:Animal a rdfs:Class .
```

```
ex:Horse a rdfs:Class ;  
        rdfs:subClassOf ex:Animal .
```

```
ex:tornado a ex:Horse .
```

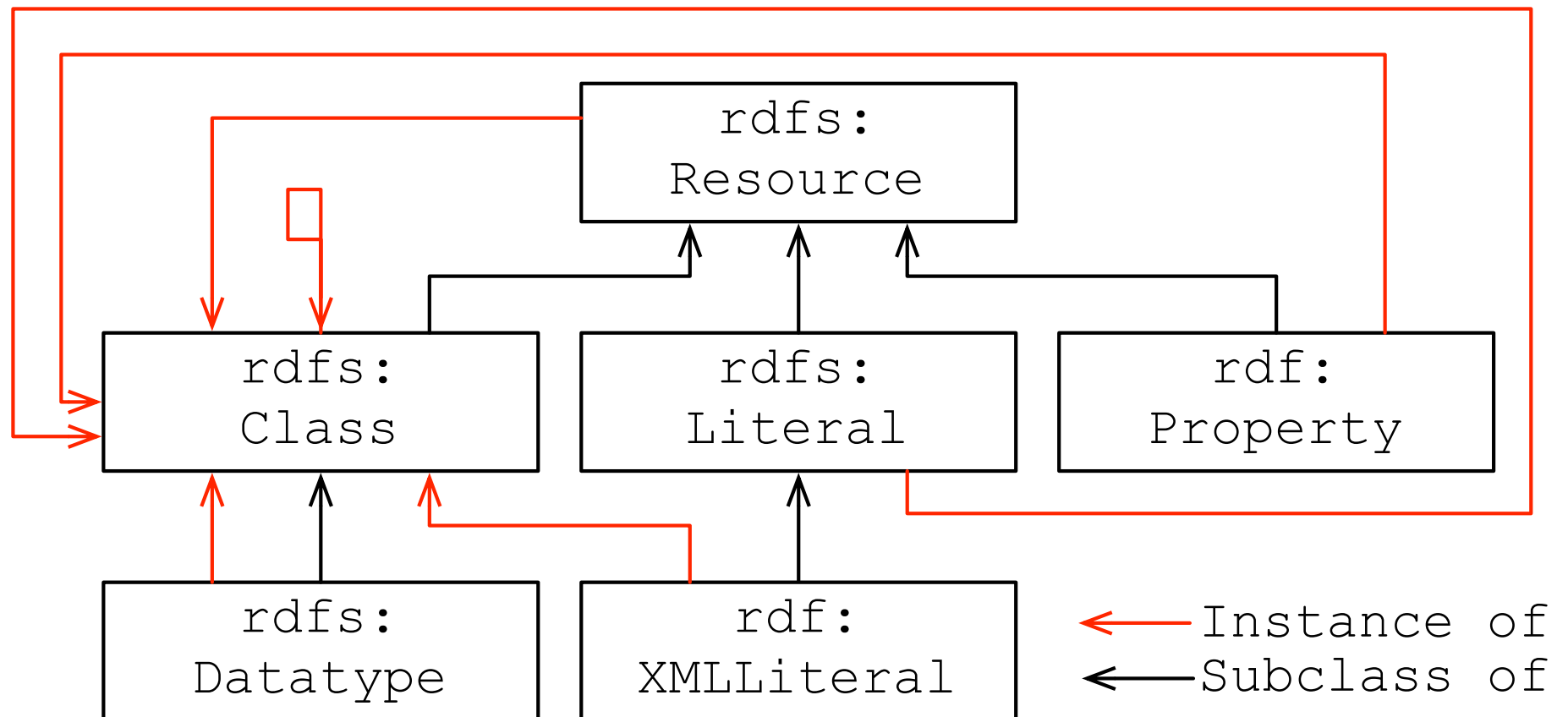
```
ex:garfield a ex:Animal .
```



RDF(S) Classes

rdfs:Resource	RDF(S) top element, all other classes are derived from this
rdfs:Class	The Class class
rdf:Property	Base class for properties
rdfs:Literal	The base class for literal values. Allows literal values such as strings and integers
rdfs:Datatype	The base class of data types
rdf:XMLLiteral	A data type for holding XML data

RDF(S) Class Organization



RDF(S) Properties

rdfs:subClassOf	Indicates the subject is a subclass of the object in a statement.
rdfs:subPropertyOf	The subject is a sub-property of the property.
rdfs:comment rdfs:label	Simple properties that take string literals as values. Labels refer to <i>human-readable</i> versions of a resource's <i>name</i> and a comment provides a human-readable <i>description</i> of a resource.
rdfs:domain	Used to state that any resource that has a given property is an instance of one or more classes.
rdfs:range	Used to state that the values of a property are instances of one or more classes.
rdfs:isDefinedBy	Points to the human readable definition of a class, usually a URL.

RDF(S) Classes and Properties

```
@prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix rdfs:   <http://www.w3.org/2000/01/rdf-schema#> .
```

```
<#Person>  a    rdfs:Class ;  
           rdfs:label  "Person"@en, "Person" ;  
           rdfs:comment "The Person Class" .
```

```
<#Teacher> a    rdfs:Class ;  
           rdfs:subClassOf <#Person> .
```

```
<#Course>  a    rdfs:Class .
```

```
<#hasTeacher> a    rdf:Property ;  
              rdfs:comment "The relation between courses and teachers." ;  
              rdfs:domain <#Course> ;  
              rdfs:range  <#Teacher> .
```

RDF(S) Classes and Properties

```
@prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix rdfs:   <http://www.w3.org/2000/01/rdf-schema#> .
```

```
<#Person>  a    rdfs:Class ;  
           rdfs:label  "Person"@en, "Person" ;  
           rdfs:comment "The Person Class" .
```

```
<#Teacher> a    rdfs:Class ;  
           rdfs:subClassOf <#Person> .
```

```
<#Course>  a    rdfs:Class .
```

```
<#hasTeacher> a    rdf:Property ;  
              rdfs:comment "The relation between courses and teachers." ;  
              rdfs:domain  <#Course> ;  
              rdfs:range   <#Teacher> .
```

RDF(S) Classes and Properties

```
@prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix rdfs:   <http://www.w3.org/2000/01/rdf-schema#> .
```

```
# A Class can be the subclass of more than one class
```

```
<#A> rdfs:subClassOf <#B>, <#C> .
```

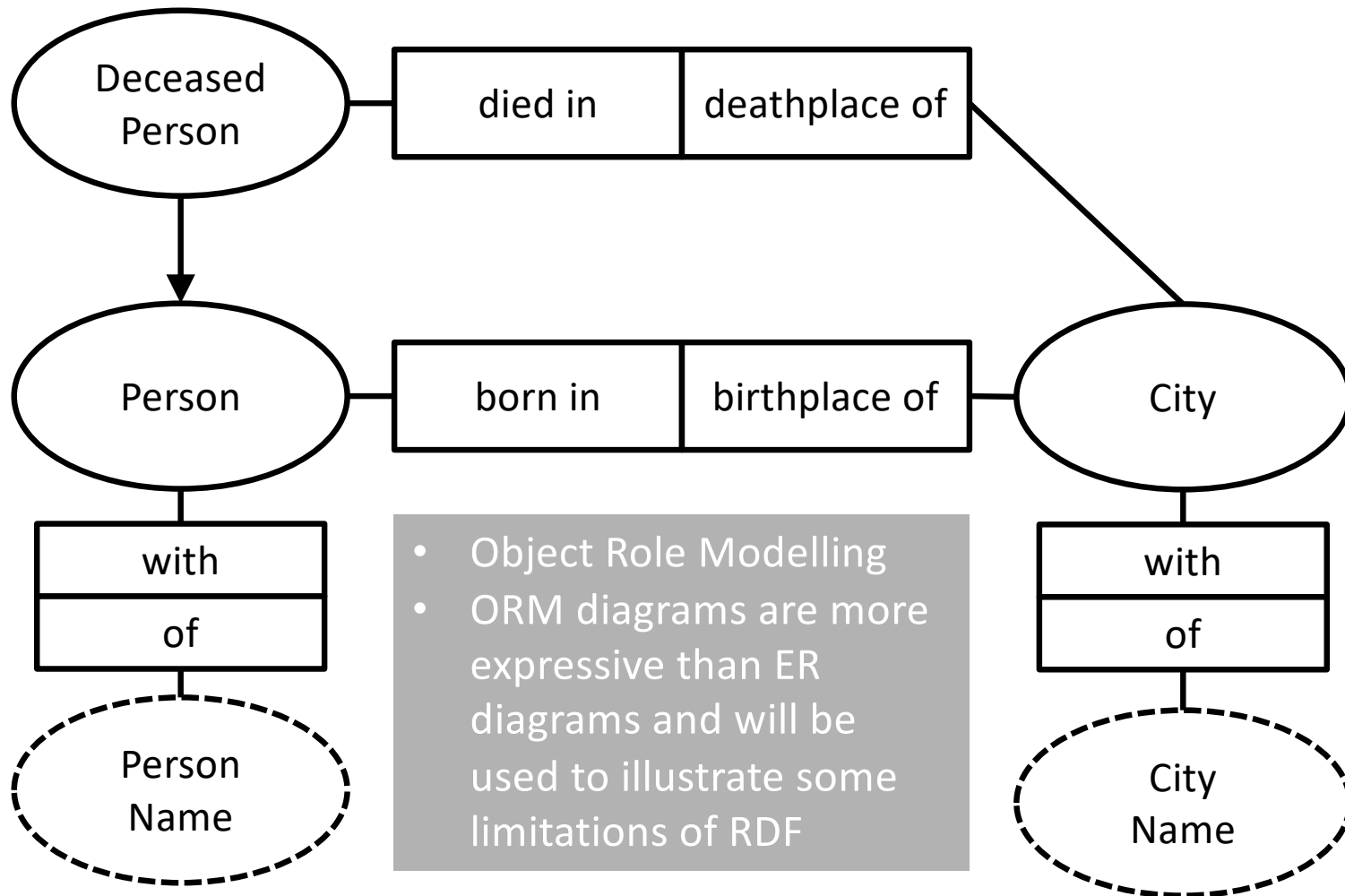
```
# When a property has more than one range (resp. domain) property, then the  
# resources denoted by the objects (resp. subjects) of triples with that property  
# are instances of all the classes stated by the range (resp. domain) properties.
```

```
<#name> rdfs:domain <#Person>, <#Cat> ;  
        rdfs:range  rdfs:Literal .
```

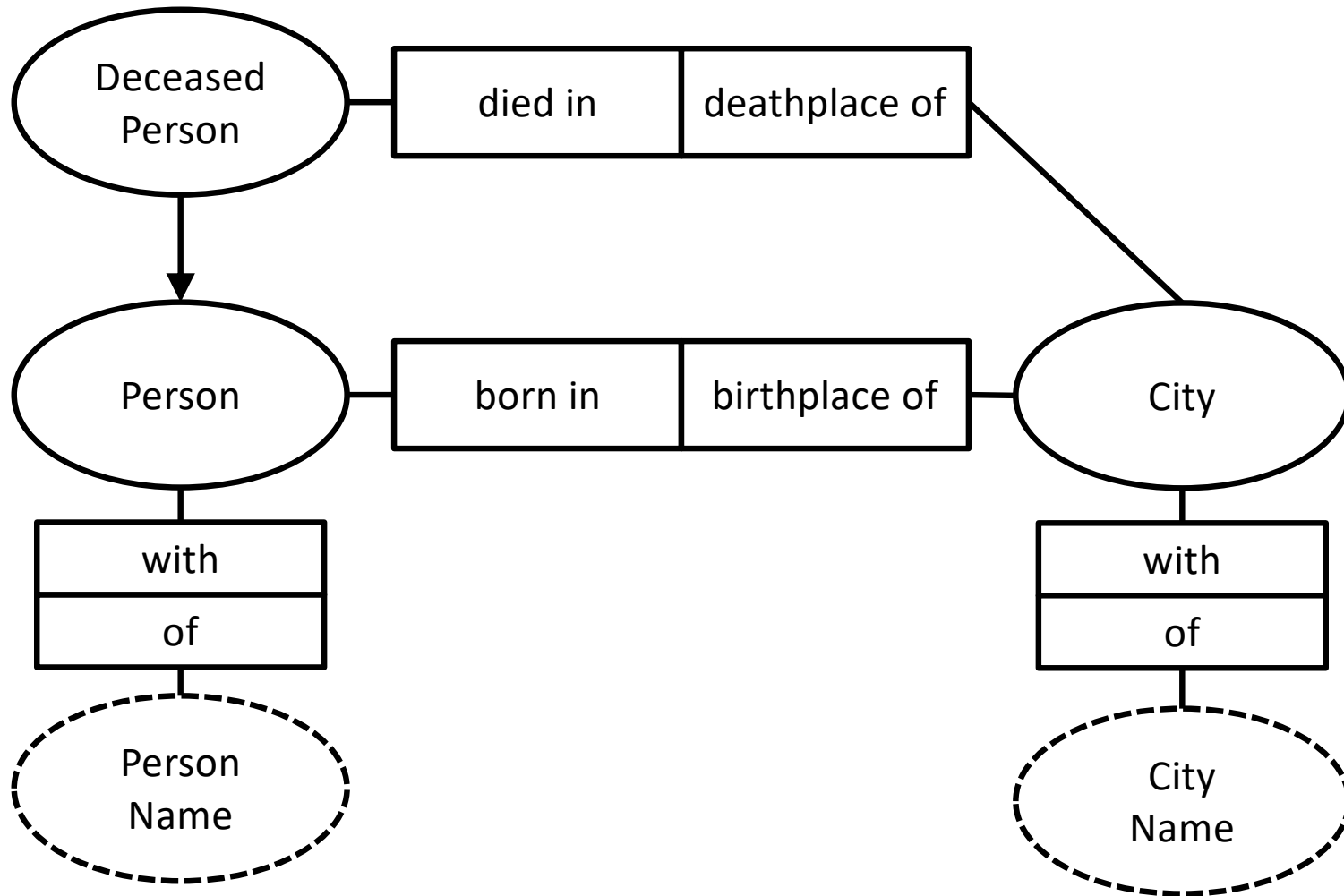
```
<#christophe> <#name> "Christophe" .
```

```
# Christophe is inferred to be both an instance of <#Person> and <#Cat>
```

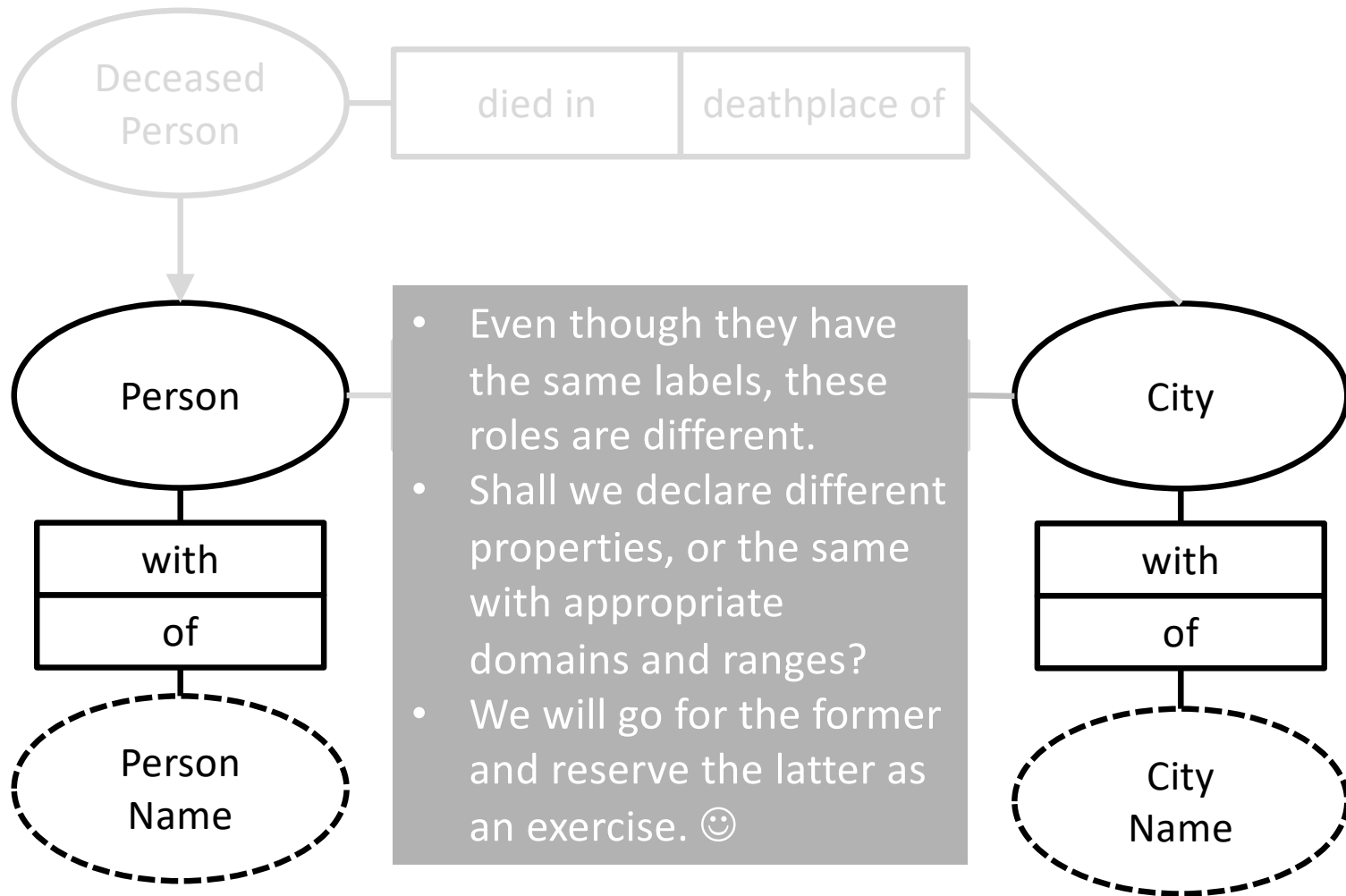
Small Exercise



Small Exercise



Small Exercise



```
@prefix ex:      <http://www.example.org/people.rdf#> .
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:    <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .
```

```
ex:Person      a          rdfs:Class ;
                rdfs:label "Person" .
```

```
ex:personName a          rdf:Property ;
                rdfs:domain ex:Person ;
                rdfs:label "person name" ;
                rdfs:range  rdfs:Literal .
```

```
# ...
```

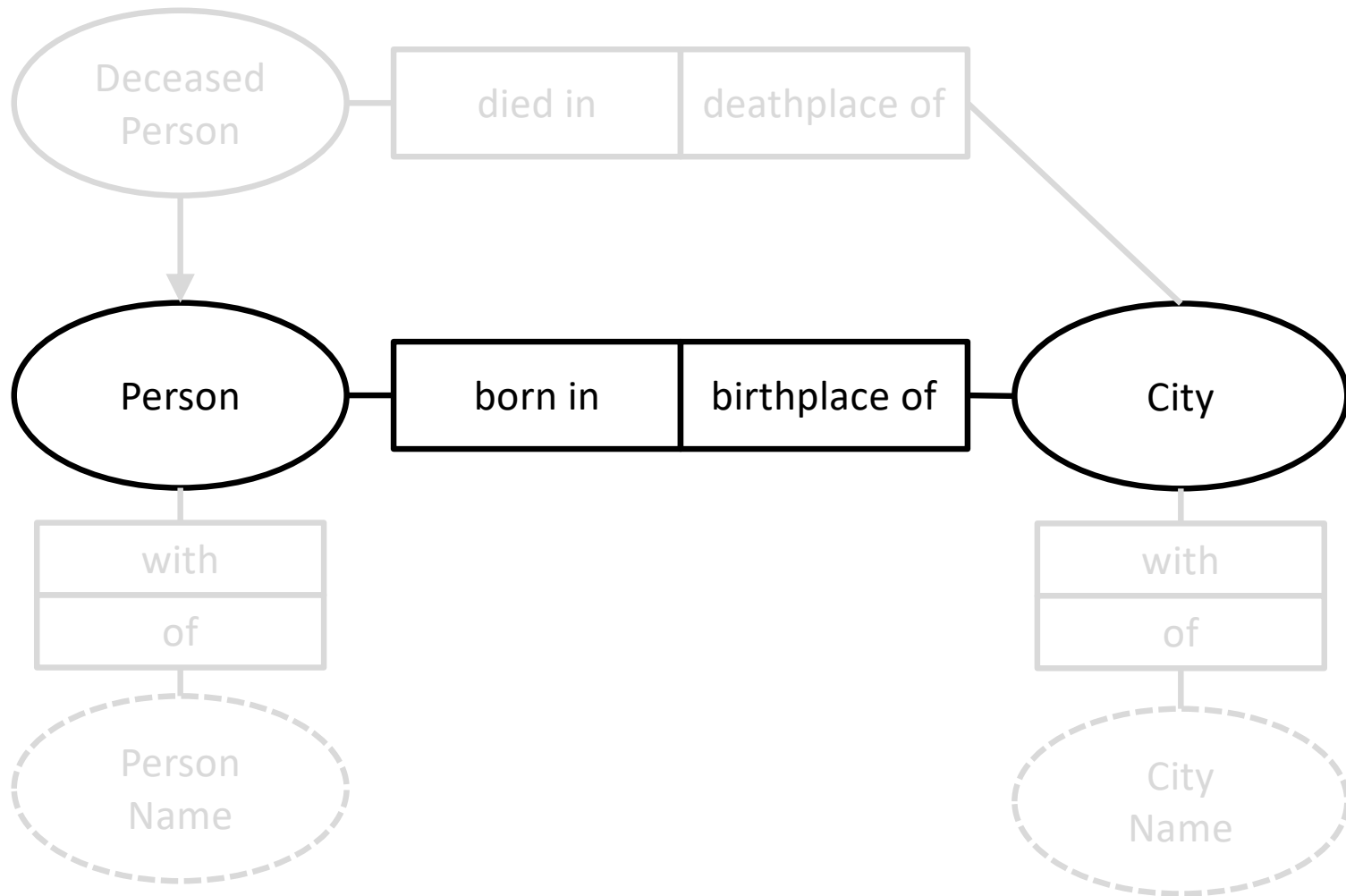
```
# ...
```

```
ex:City      a          rdfs:Class ;  
             rdfs:label  "City" .
```

```
ex:cityName  a          rdf:Property ;  
             rdfs:domain ex:City ;  
             rdfs:label  "city name" ;  
             rdfs:range  rdfs:Literal .
```

```
# ...
```


Small Exercise



```
# ...  
ex:bornIn    a          rdf:Property ;  
             rdfs:domain ex:Person ;  
             rdfs:label  "born in" ;  
             rdfs:range  ex:City .  
  
ex:birthplaceOf a      rdf:Property ;  
               rdfs:domain ex:City ;  
               rdfs:label  "birthplace of" ;  
               rdfs:range  ex:Person .  
# ...
```

Is something
wrong with this?

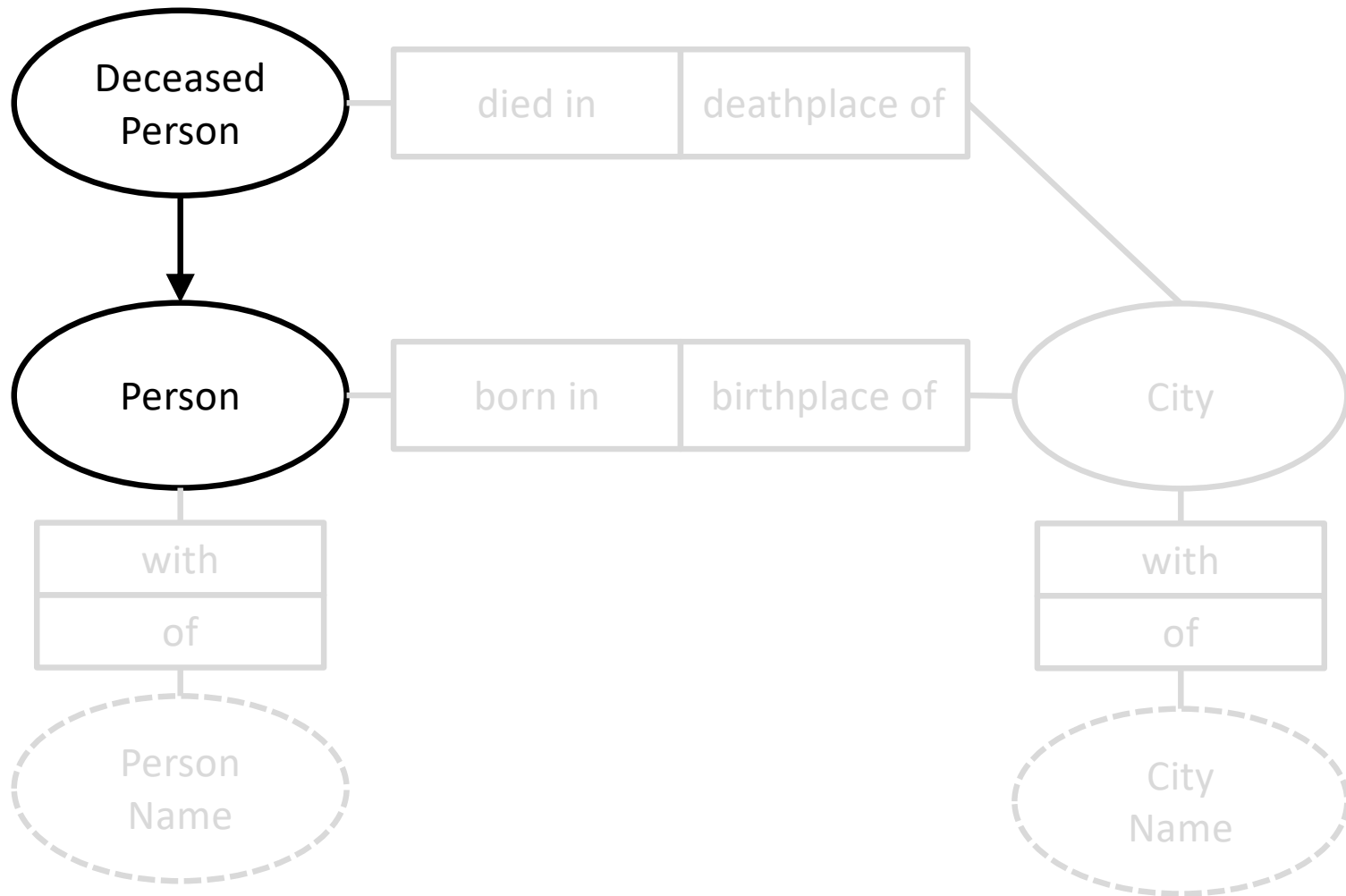
In our conceptual schema, *born in* was the role and *birthplace of* the co-role of a relation between *Person* and *City*. In other words, both roles were the inverse of each other. In the example above, the two properties are not related at all, which means:

- One can not infer that if <#Christophe> ex:bornIn <#Ghent>, then <#Ghent> ex:birthplaceOf <#Christophe>.
- If one were to state both <#Christophe> ex:bornIn <#Ghent> and <#Ghent> ex:birthplaceOf <#Christophe>, they are considered separate statements.

```
# ...  
ex:bornIn      a          rdf:Property ;  
                rdfs:domain ex:Person ;  
                rdfs:label  "born in" ;  
                rdfs:range  ex:City .  
  
ex:birthplaceOf a          rdf:Property ;  
                rdfs:label  "birthplace of" ;  
                owl:inverseof ex:bornIn .  
  
# ...
```

owl:inverseof
is part of the Web
Ontology Language
(OWL) but
commonly used
(and accepted)
when creating
RDF(S)

Small Exercise

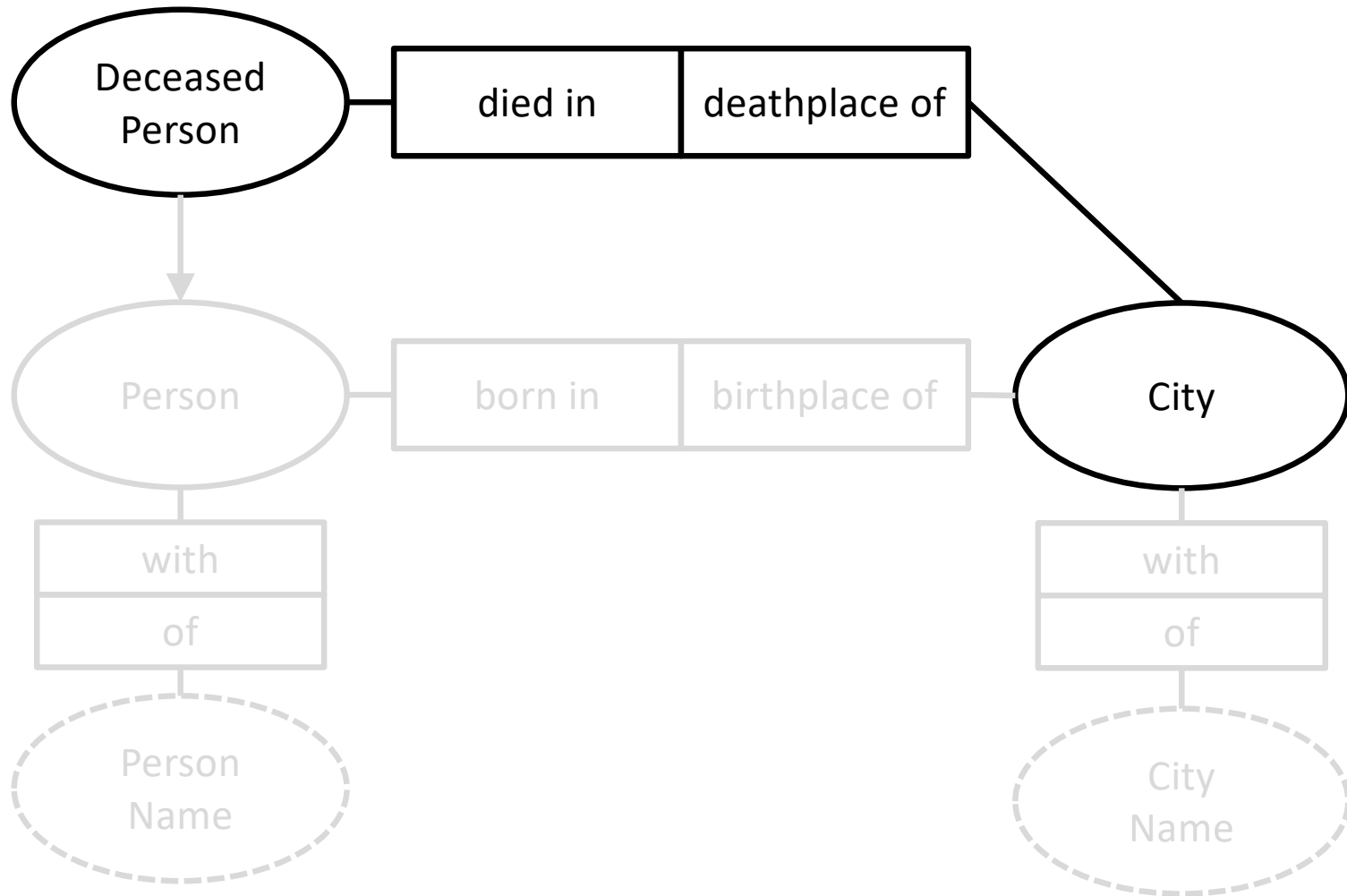


```
# ...
```

```
ex:DeceasedPerson    a          rdfs:Class ;  
                      rdfs:label  "Deceased Person" ;  
                      rdfs:subClassOf ex:Person .
```

```
# ...
```

Small Exercise

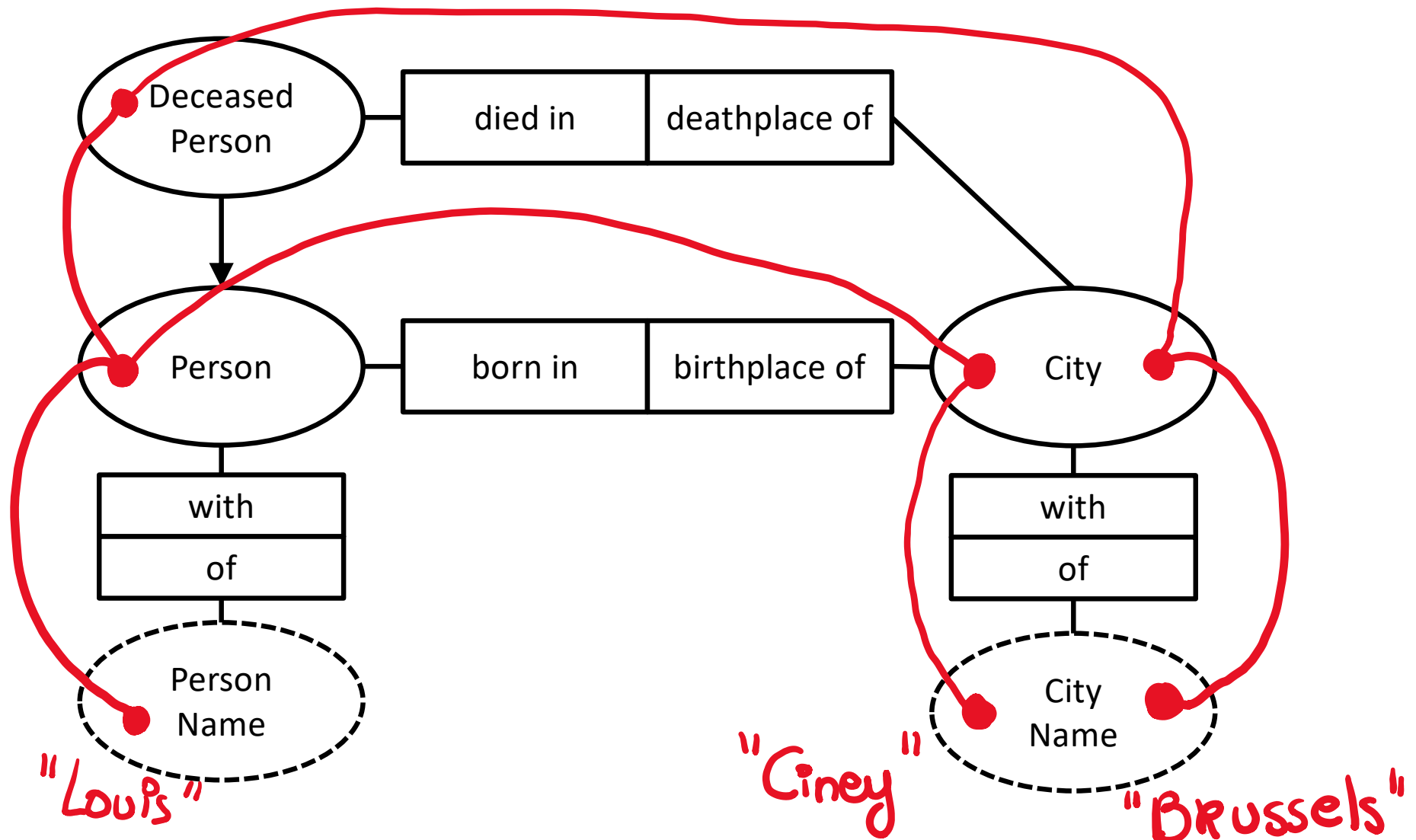


```
# ...
```

```
ex:diedIn    a          rdf:Property ;  
             rdfs:domain ex:DeceasedPerson ;  
             rdfs:label  "died in" ;  
             rdfs:range  ex:City .  
  
ex:deathplaceOf  a          rdf:Property ;  
                 rdfs:label  "deathplace of" ;  
                 owl:inverseof ex:diedIn .
```

```
# end of document
```

Instances




```
@prefix ont:    <http://www.example.org/people.rdf#> .
@prefix ex:     <http://www.example.org/instances.rdf#> .
@prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:   <http://www.w3.org/2000/01/rdf-schema#> .
```

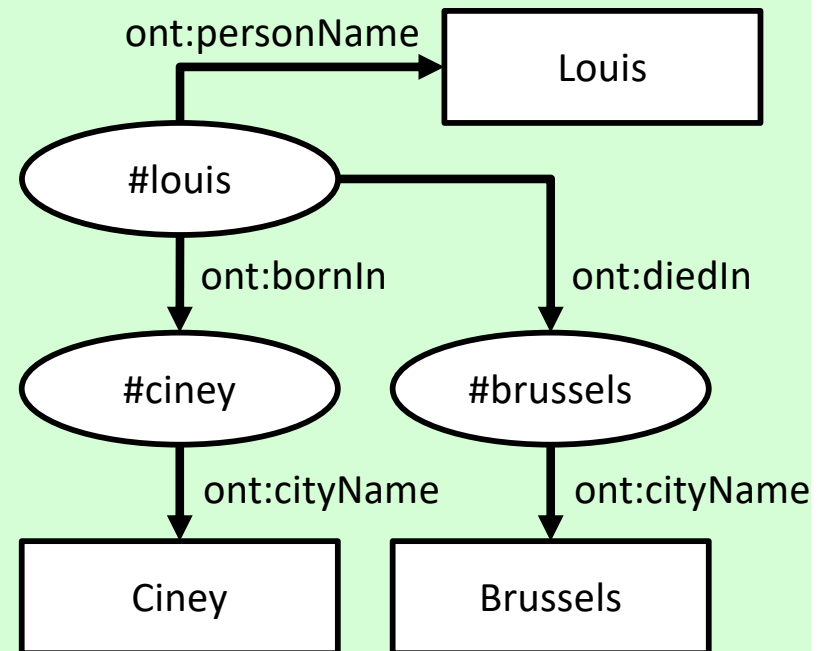
However ... RDF(S) Reasoning

An instance of a deceased person is also an instance of person.
Here, we will name the nodes.

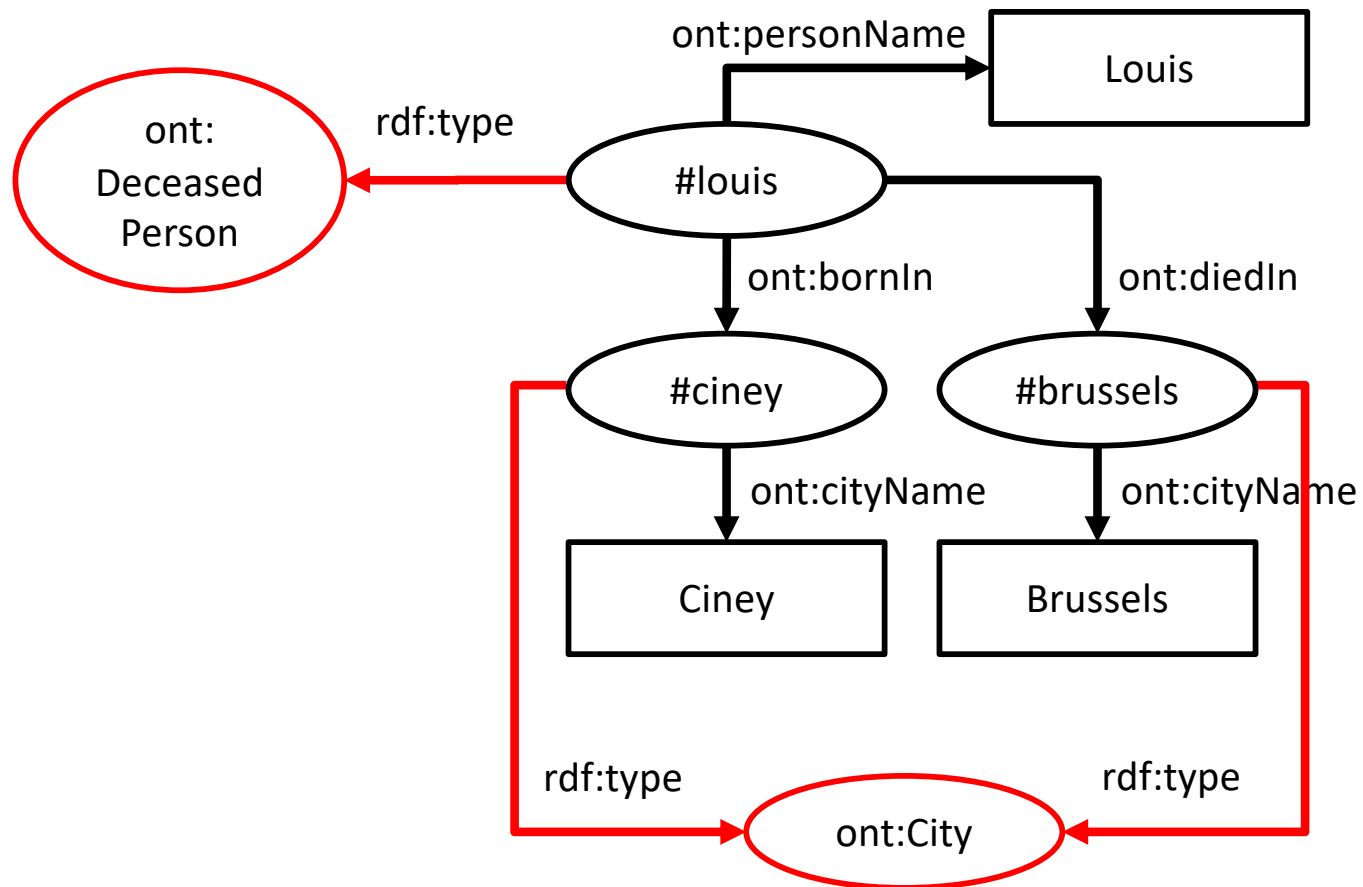
```
ex:louis a      ont:DeceasedPerson ;
          ont:bornIn ex:ciney ;
          ont:diedIn ex:brussels ;
          ont:personName "Louis" .

ex:brussels a      ont:City ;
             ont:cityName "Brussels" .

ex:ciney a      ont:City ;
           ont:cityName "Ciney" .
```



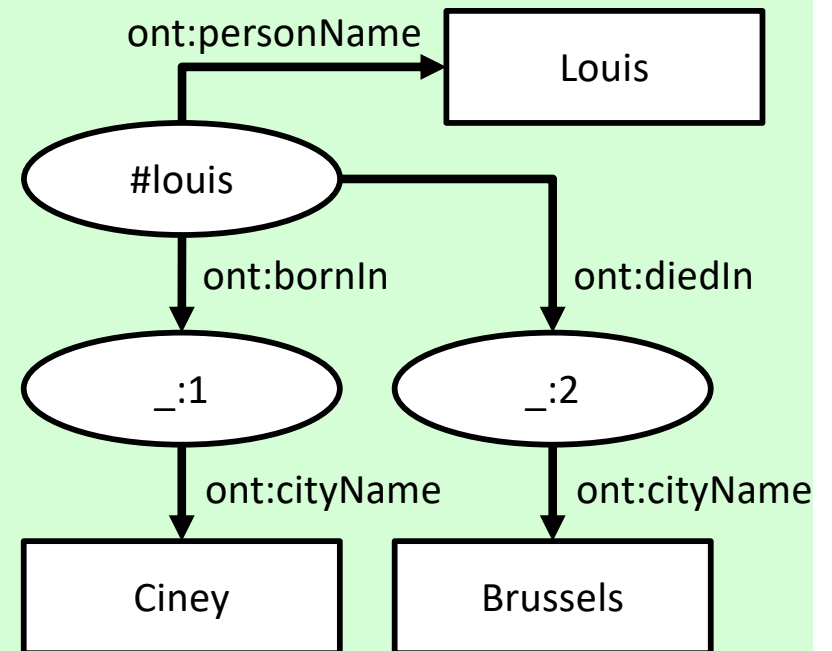
rdf:type relations omitted for brevity. How would they look like?



```
@prefix ont:    <http://www.example.org/people.rdf#> .
@prefix ex:     <http://www.example.org/instances2.rdf#> .
@prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:   <http://www.w3.org/2000/01/rdf-schema#> .
```

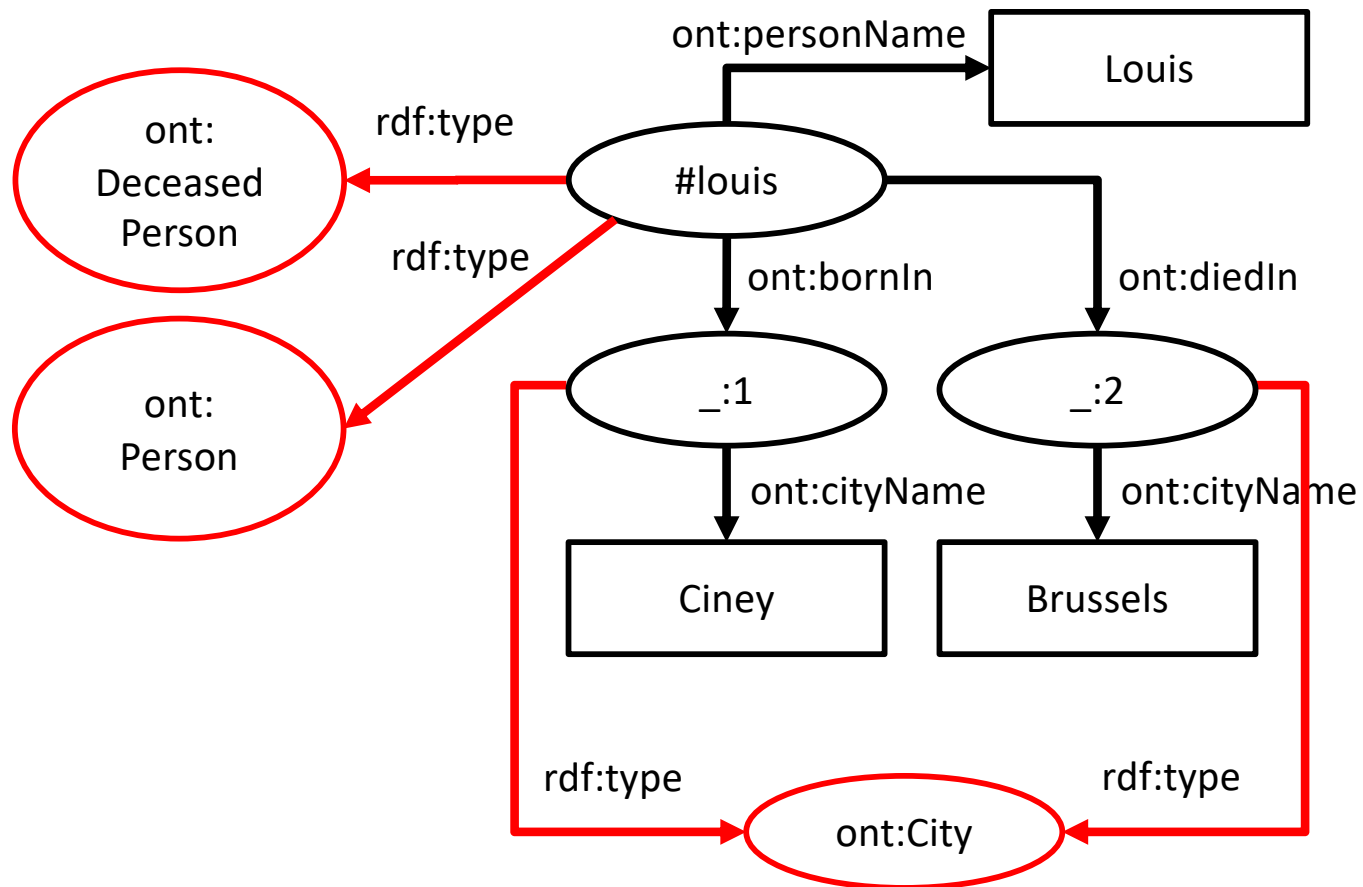
```
# We make louis an instance of both person and deceased person
# Instances of B are nested as blank nodes.
```

```
ex:louis
  a ont:Person, ont:DeceasedPerson ;
  ont:bornIn [ a ont:City ;
               ont:cityName "Ciney"
             ] ;
  ont:diedIn [ a ont:City ;
               ont:cityName "Brussels"
             ] ;
  ont:personName "Louis" .
```



Unless we provide blank node identifiers, we can make no assumptions about the blank node identifiers generated by software agents.

rdf:type relations omitted for brevity. How would they look like?



```

@prefix ont:    <http://www.example.org/people.rdf#> .
@prefix ex:     <http://www.example.org/instances3.rdf#> .
@prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:   <http://www.w3.org/2000/01/rdf-schema#> .

```

```

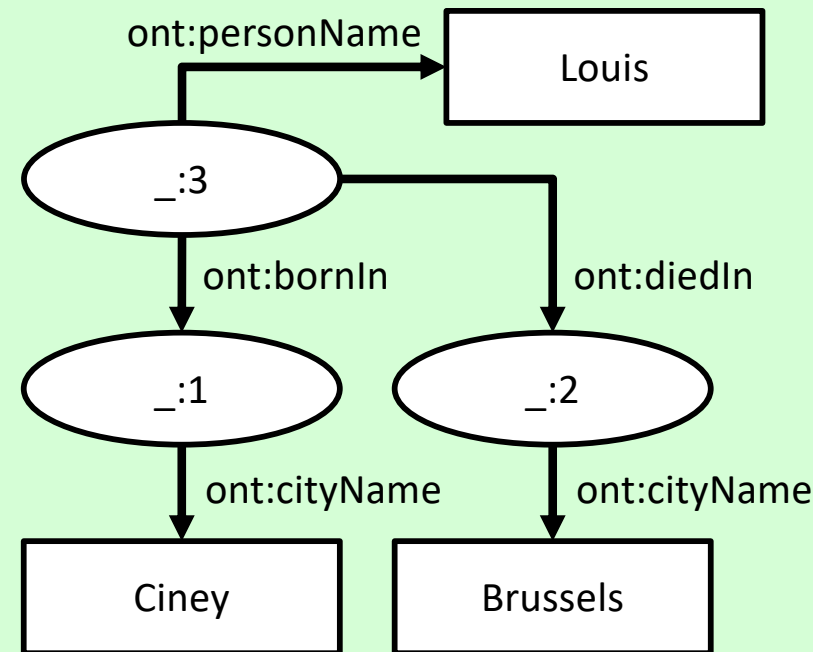
# We make louis an instance of both person and deceased person
# All instances as blank nodes.

```

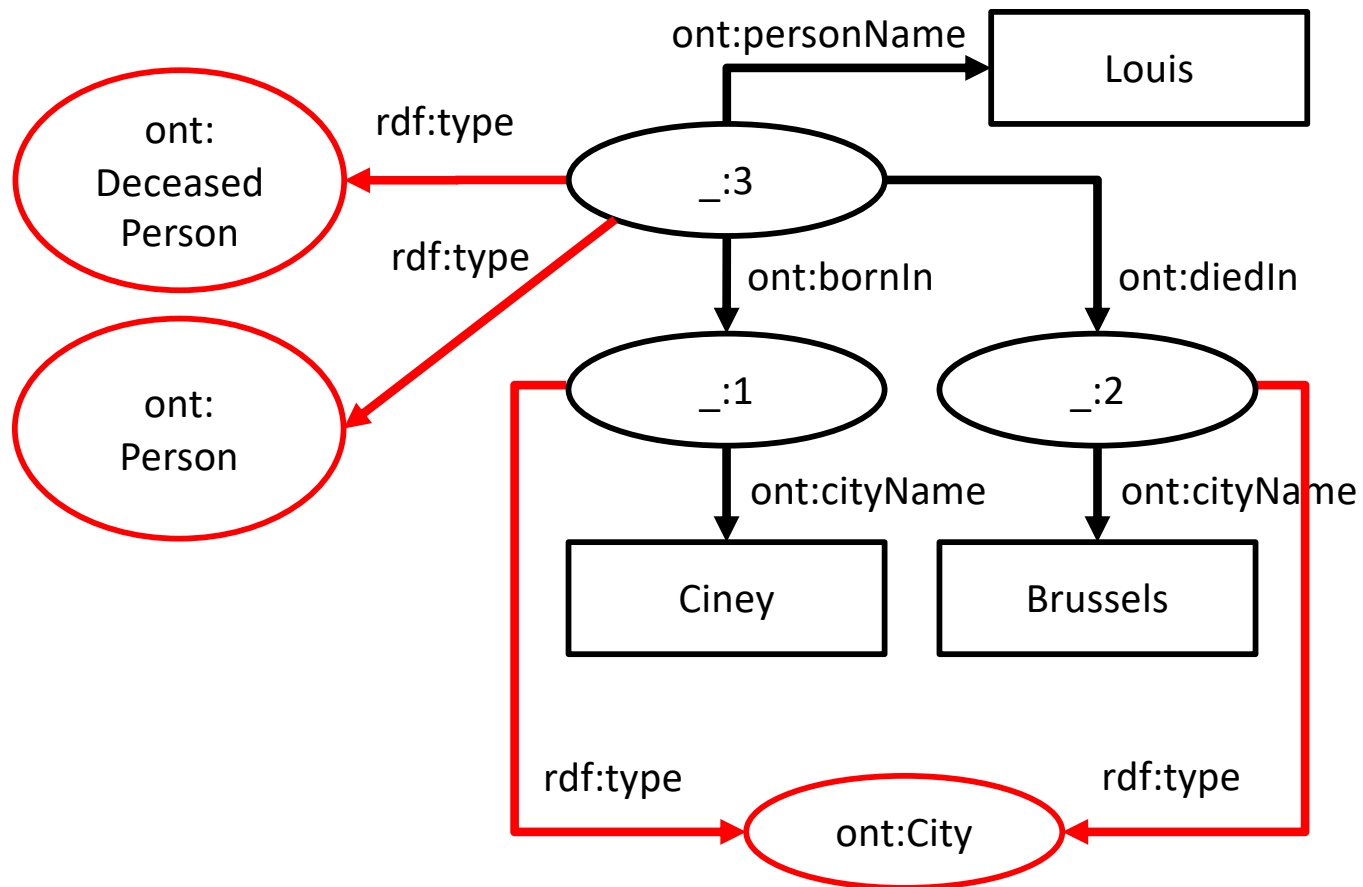
```

[
  a ont:Person, ont:DeceasedPerson ;
  ont:bornIn [ a ont:City ;
               ont:cityName "Ciney"
             ] ;
  ont:diedIn [ a ont:City ;
               ont:cityName "Brussels"
             ] ;
  ont:personName "Louis"
]

```



rdf:type relations omitted for brevity. How would they look like?



On final(ish) note on blank nodes

RDF Document 1

```
_:bn1 rdfs:label "Hello" .  
  
_:bn1 rdfs:label "World" .
```

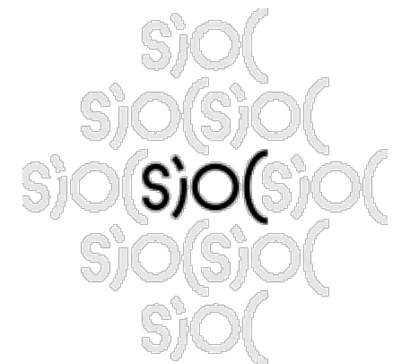
RDF Document 2

```
_:bn1 rdfs:label "Hello World".
```

Self-declared blank node identifiers within the same document refer to the same "thing", but **the same self-declared blank node identifiers in different documents do not.**

Some well known RDF(S) vocabularies

- Reuse or extend existing terms from well known vocabularies:
 - Dublin Core (DC)
Describing documents
 - Friend-of-a-friend (FOAF)
Describing relations between people
 - Simple Knowledge Organization Systems (SKOS)
Describing thesauri
 - Semantically Interlinked Online Communities (SIOC)
Describing Fora and Posts



References

- XML Schema
 - <https://www.w3.org/XML/Schema>
- RDF, RDF/XML, TURTLE
 - <https://www.w3.org/TR/rdf-primer/>
 - <https://www.w3.org/TR/rdf-syntax-grammar/>
 - <https://www.w3.org/TR/turtle/>
- RDF Schema
 - <https://www.w3.org/TR/rdf-schema/>
- OWL 2.0
 - <https://www.w3.org/TR/owl2-overview/>