



RDF/XML Source Declaration

Fabien Gandon, Virginie Bottollier, Olivier Corby, Priscille Durville

► **To cite this version:**

Fabien Gandon, Virginie Bottollier, Olivier Corby, Priscille Durville. RDF/XML Source Declaration. 2007. hal-01154394

HAL Id: hal-01154394

<https://hal.inria.fr/hal-01154394>

Submitted on 22 May 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

RDF/XML Source Declaration

W3C Member Submission 5 September 2007

This version:

<http://www.w3.org/Submission/2007/SUBM-rdfsource-20071018/>

Latest version:

<http://www.w3.org/Submission/rdfsource/>

Last update:

Date: 2007/09/05 17:24:08

Revision:

Revision: 1.5

Editor:

[Fabien Gandon](#) - INRIA

Contributors:

[Virginie Bottollier](#) - INRIA

[Olivier Corby](#) - INRIA

[Priscille Durville](#) - INRIA

Copyright © 2004-2007 by INRIA.

This document is available under the [W3C Document License](#). See the [W3C Intellectual Rights Notice and Legal Disclaimers](#) for additional information.



Regarding underlying technology, this proposal relies on W3C's [RDF](#) technology.

Abstract

When querying or reasoning on metadata from the semantic web, the source of this metadata can be of great importance. While the SPARQL query language provides a keyword to match patterns against named graphs, the RDF data model focuses on expressing triples. In many cases it is interesting to augment these RDF triples with the notion of a source for each triple (or set of triples), typically an IRI specifying their real or virtual origin. This document proposes an RDF/XML syntax extension providing an attribute to specify the source of triples in an RDF/XML representation.

Status of this document

NOTE: *This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications can be found in the [W3C technical reports index](#) at <http://www.w3.org/TR/>.*

This specification is an evolving document. Authors welcome suggestions on this document. Please send comments to [Fabien Gandon](#). This document may be updated or added to based on implementation experience, but no commitment is made by the authors regarding future updates.

By publishing this document, W3C acknowledges that INRIA has made a formal Submission request to W3C for discussion. Publication of this document by W3C indicates no endorsement of its content by W3C, nor that W3C has, is, or will be allocating any resources to the issues addressed by it. This document is not the product of a chartered W3C group, but is published as potential input to the [W3C Process](#). A [W3C Team Comment](#) has been published in conjunction with this Member Submission. Publication of acknowledged Member Submissions at the W3C site is one of the benefits of [W3C Membership](#). Please consult the requirements associated with Member Submissions of [section 3.3 of the W3C Patent Policy](#). Please consult the complete [list of acknowledged W3C Member Submissions](#).

Table of contents

- [1. Introduction](#)
- [2. Namespaces](#)
- [3. Source declaration attribute in RDF/XML](#)
 - [3.1. SPARQL queries on the dataset](#)
 - [3.2. Syntax extension: the source attribute](#)
 - [3.3. Examples of source declaration](#)
- [4. Some particular cases](#)
 - [4.1. Blank nodes and sources](#)
 - [4.2. Containers](#)
 - [4.3. Collections](#)
 - [4.4. Structured values](#)
 - [4.5. Reification](#)
 - [4.6. Describing sources](#)
 - [4.7. RDFS cases](#)
 - [4.8. OWL cases](#)
- [5. Tests and implementation](#)
- [6. Discussion](#)
- [7. Acknowledgments](#)
- [8. References](#)

1. Introduction

When querying or reasoning on metadata from the semantic web, the source of this metadata can be of great importance. The Resource Description Framework (RDF) is a general-purpose language for representing data and metadata on the web and it has an XML syntax called RDF/XML. The formal grammar for the syntax is annotated with actions generating triples of the RDF graph.

In SPARQL when querying a collection of graphs, the GRAPH keyword is used to match patterns against named graphs. However the RDF data model focuses on expressing triples with a subject, predicate and object and neither it nor its RDF/XML syntax provide a mechanism to specify the source of each triple. A typical means would be an XML syntax to associate to the triples encoded in RDF/XML an IRI specifying their origin.

This document proposes an extension of the syntax (a single attribute) to specify for these triples represented in RDF/XML the source they should be attached to.

2. Namespaces

Prefixes and associated namespaces used in this document:

```

rdf http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs http://www.w3.org/2000/01/rdf-schema
cos http://www.inria.fr/acacia/corese#
dc http://purl.org/dc/elements/1.1/
foaf http://xmlns.com/foaf/0.1/

```

3. Source declaration attribute in RDF/XML

In this section we recall how the source is used in SPARQL and then proceed with proposing an RDF/XML syntax extension to specify a source IRI for a set of triples.

3.1. SPARQL queries on the dataset

In [SPARQL](#) when querying a collection of graphs, the GRAPH keyword is used to match patterns against named graphs. GRAPH can provide an IRI to select one graph or use a variable which will range over the IRI of all the named graphs in the query's RDF dataset. The query below matches two graph patterns against each of the named graphs in the dataset and form solutions which have the `?srcname` and `?srctitle` variables bound to IRIs of the graph being matched.

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?srcname ?name ?srctitle ?title
WHERE
{
  GRAPH ?srctitle
  {
    ?doc dc:title ?title .
    ?doc dc:creator ?author
  }
  GRAPH ?srcname { ?author foaf:name ?name }
}
```

Unfortunately the syntax of a [SPARQL](#) source has no equivalent in terms of the RDF syntax. We propose here a possible mechanism to standardize the declaration of sources in an RDF graph serialized in [RDF/XML](#).

3.2. Syntax extension: the source attribute

Using the [Corese SPARQL engine](#), we implemented and tested an extension of the [RDF/XML](#) syntax: an attribute `cos:graph` may be inserted in an [RDF/XML](#) document to specify a source IRI. The value of this attribute is interpreted as an IRI Reference.

The source IRI of a triple is:

1. the source IRI specified by a `cos:graph` attribute on the XML element encoding this triple, if one exists, otherwise
2. the source IRI of the element's parent element (obtained following recursively the same rules), otherwise
3. the base IRI of the document.

The base IRI of a document entity or an external entity is determined by RFC 2396 rules, namely, that the base IRI is the IRI used to retrieve the document entity or external entity. In other words, if no source is specified, the URL of the RDF/XML document is used as a default source.

The scope of a source declaration extends from the beginning of the start-tag in which it appears to the end of the corresponding end-tag, excluding the scope of any inner source declarations. Such a source declaration applies to all elements and attributes within its scope. In the case of an empty tag, the scope is the tag itself.

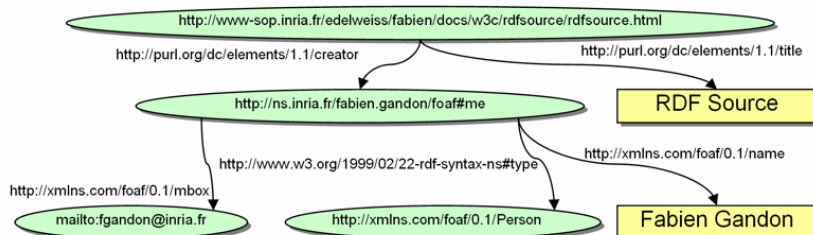
Only one source can be declared as attribute of a single element.

Thus the `cos:graph` attribute can be used on any node element or property element to indicate that the included content is from a given source IRI. The most specific in-scope source present (if any) is applied.

We allow explicitly null sources: the `cos:graph=""` form indicates the absence of a source identifier so the associated source will explicitly be null and the base IRI of the document won't even be considered.

3.3. Examples of source declaration

Let us consider the following RDF graph stating that a resource has a title ("RDF Source") and a creator and that this creator is of type Person and has a name ("Fabien Gandon") and a mailbox ("mailto:fgandon@inria.fr"):



A possible representation of this graph in RDF/XML is:

```
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
  <rdf:Description rdf:about="http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsourcel/rdfsourcel.html">
    <dc:title>RDF Source</dc:title>
    <dc:creator>
      <foaf:Person rdf:about="http://ns.inria.fr/fabien.gandon/foaf#me">
        <foaf:name>Fabien Gandon</foaf:name>
        <foaf:mbox rdf:resource="mailto:fgandon@inria.fr"/>
      </foaf:Person>
    </dc:creator>
  </rdf:Description>
</rdf:RDF>
```

The corresponding triples obtained from parsing this RDF/XML are:

```
<http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsourcel/rdfsourcel.html> dc:title "RDF Source"
<http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsourcel/rdfsourcel.html> dc:creator <http://ns.inria.fr/fabien.gandon/foaf#me>
<http://ns.inria.fr/fabien.gandon/foaf#me> rdf:type foaf:Person
<http://ns.inria.fr/fabien.gandon/foaf#me> foaf:name "Fabien Gandon"
<http://ns.inria.fr/fabien.gandon/foaf#me> foaf:mbox <mailto:fgandon@inria.fr>f:mbox rdf:resource="mailto:fgandon@inria.fr"/>
```

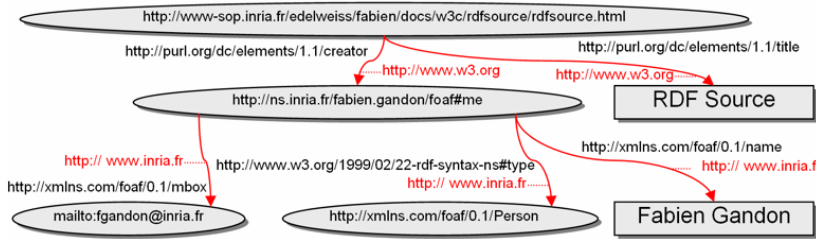
The RDF/XML syntax extension proposed here turns triples into quadruples with the forth term being the IRI of the source of the triple. For instance consider the previous example augmented with two occurrences of the `cos:graph` attribute as follows. It results in having all the triples about the person in the source `http://www.inria.fr` including the type declaration as a `foaf:Person`:

```
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  cos:graph="http://www.w3.org">
  <rdf:Description rdf:about="http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsourcel/rdfsourcel.html">
    <dc:title>RDF Source</dc:title>
    <dc:creator>
      <foaf:Person rdf:about="http://ns.inria.fr/fabien.gandon/foaf#me"
        cos:graph="http://www.inria.fr" >
        <foaf:name>Fabien Gandon</foaf:name>
        <foaf:mbox rdf:resource="mailto:fgandon@inria.fr"/>
      </foaf:Person>
    </dc:creator>
  </rdf:Description>
</rdf:RDF>
```

Quadruples resulting from the parsing of this file would be:

```
<http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html> dc:title "RDF Source" <- http://www.w3.org
<http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html> dc:creator <http://ns.inria.fr/fabien.gandon/foaf#me> <- http://www.w3.org
<http://ns.inria.fr/fabien.gandon/foaf#me> rdf:type foaf:Person <- http://www.inria.fr
<http://ns.inria.fr/fabien.gandon/foaf#me> foaf:name "Fabien Gandon" <- http://www.inria.fr
<http://ns.inria.fr/fabien.gandon/foaf#me> foaf:mbox <mailto:fgandon@inria.fr> <- http://www.inria.fr
```

A possible graphical representation would be to decorate the arcs with their source:



4. Some particular cases

Generally speaking this sections shows that it is dangerous to change sources around blank nodes: since a blank node can only belong to one source, changing sources on properties of a blank node will result in splitting the blank node into several blank nodes, one for each source.

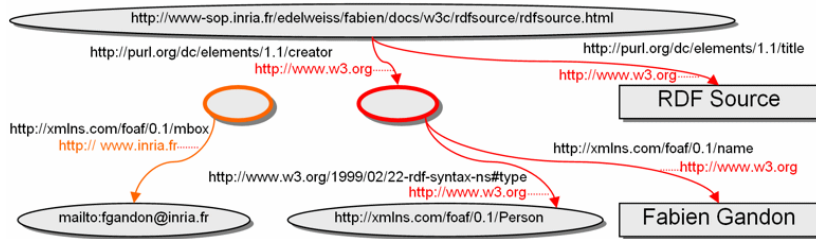
4.1. Blank nodes and sources

Blank nodes in descriptions with sources declaration can lead to surprising results. They should be used with care. In particular a blank node referenced in two (or more) different sources is interpreted as two (or more) different blank nodes, one for each source.

For instance the following RDF/XML leads to the creation of two blank nodes representing the person:

```
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  cos:graph="http://www.w3.org">
  <rdf:Description rdf:about="http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html">
    <dc:title>RDF Source</dc:title>
    <dc:creator>
      <foaf:Person>
        <foaf:name>Fabien Gandon</foaf:name>
        <foaf:mbox rdf:resource="mailto:fgandon@inria.fr" cos:graph="http://www.inria.fr" />
      </foaf:Person>
    </dc:creator>
  </rdf:Description>
</rdf:RDF>
```

A graphical representation of the resulting graph could be:



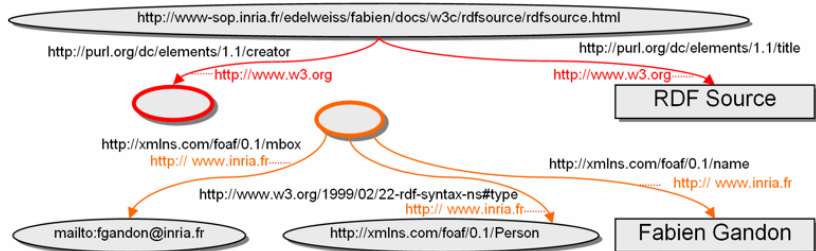
Quadruples generated for this example are:

```
<http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html> dc:title "RDF Source" <- http://www.w3.org
<http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html> dc:creator _:a <- http://www.w3.org
_:a rdf:type foaf:Person <- http://www.w3.org
_:a foaf:name "Fabien Gandon" <- http://www.w3.org
_:b foaf:mbox <mailto:fgandon@inria.fr> <- http://www.inria.fr
```

As a second example the following RDF/XML also leads to the creation of two blank nodes representing the person but with a different attribution of the properties:

```
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  cos:graph="http://www.w3.org">
  <rdf:Description rdf:about="http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html">
    <dc:title>RDF Source</dc:title>
    <dc:creator>
      <foaf:Person cos:graph="http://www.inria.fr">
        <foaf:name>Fabien Gandon</foaf:name>
        <foaf:mbox rdf:resource="mailto:fgandon@inria.fr" />
      </foaf:Person>
    </dc:creator>
  </rdf:Description>
</rdf:RDF>
```

A graphical representation of the resulting graph could be:



Quadruples generated for this example are:

```
<http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html> dc:title "RDF Source" <- http://www.w3.org
<http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html> dc:creator _:a <- http://www.w3.org
_:b rdf:type foaf:Person <- http://www.inria.fr
_:b foaf:name "Fabien Gandon" <- http://www.inria.fr
_:b foaf:mbox <mailto:fgandon@inria.fr> <- http://www.inria.fr</dc:title>
```

In cases where in one file we have two blank nodes with the same node ID and the same source there should be only one resulting blank node. Following RDF specifications, if the same node ID and the same source are used for a blank node in different files, it results in the creation of different blank nodes. Finally if the source is not effectively changed (e.g. in the code below several `cos:graph` are used on properties of a blank node but with the same value) then only one blank node is created in the source.

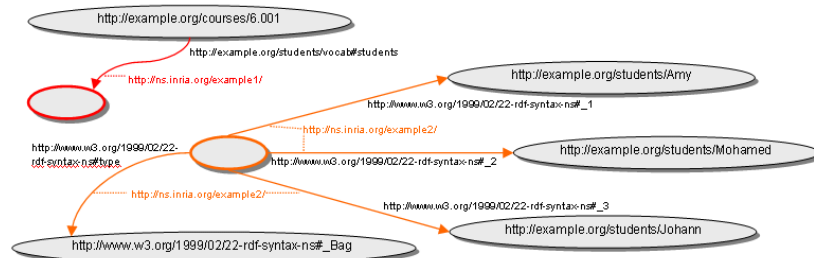
```
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  cos:graph="http://www.w3.org" >
  <rdf:Description rdf:about="http://www-sop.inria.fr/edelweiss/fabien/docs/w3c/rdfsource/rdfsource.html">
  <dc:title>RDF Source</dc:title>
  <dc:creator>
  <foaf:Person>
  <foaf:name>Fabien Gandon</foaf:name>
  <foaf:mbox rdf:resource="mailto:fgandon@inria.fr" cos:graph=" http://www.w3.org" />
  </foaf:Person>
  </dc:creator>
  </rdf:Description>
  </rdf:RDF>
```

4.2. Containers

Containers make use of explicit blank nodes. Changing the source on the blank node of a container or on one of its `rdf:li` properties will result in the creation of several blank nodes. In the example below, the bag used as a value of the property `s:student` and the bag used for the enumeration of `rdf:li` will be two different bags:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  xmlns:s="http://example.org/students/vocab#"
  cos:graph="http://ns.inria.org/example1/" >
  <rdf:Description rdf:about="http://example.org/courses/6.001">
  <s:students>
  <rdf:Bag cos:graph="http://ns.inria.org/example2/" >
  <rdf:li rdf:resource="http://example.org/students/Amy"/>
  <rdf:li rdf:resource="http://example.org/students/Mohamed"/>
  <rdf:li rdf:resource="http://example.org/students/Johann"/>
  </rdf:Bag>
  </s:students>
  </rdf:Description>
  </rdf:RDF>
```

The resulting graph is:



Quadruples generated for this example are:

```
<http://example.org/courses/6.001 s:students _:a <- http://ns.inria.org/example1/
_:b rdf:type rdf:Bag <- http://ns.inria.org/example2/
_:b rdf:_1 <http://example.org/students/Amy> <- http://ns.inria.org/example2/
_:b rdf:_2 <http://example.org/students/Mohamed> <- http://ns.inria.org/example2/
_:b rdf:_3 <http://example.org/students/Johann> <- http://ns.inria.org/example2/
```

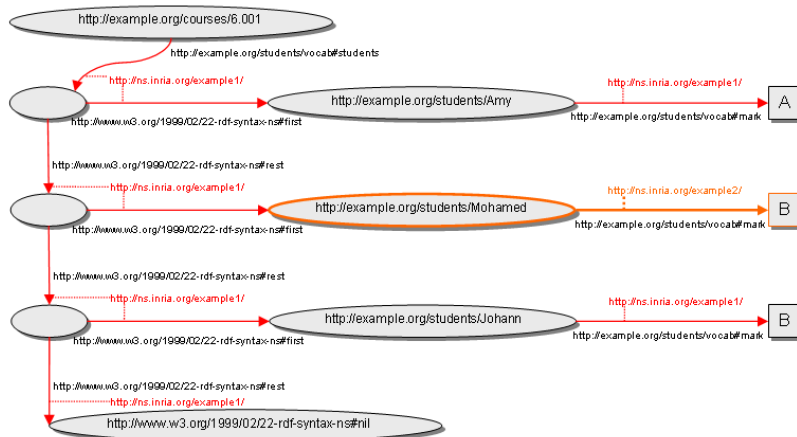
The example above uses `rdf:Bag` but the same is true when using other constructs with blank nodes such as `rdf:Alt` or `rdf:Seq`.

4.3. Collections

Collections make use of implicit blank nodes. Changing the source on the descriptions members of a collection does not cause any problem as long as these descriptions are not blank nodes themselves:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  xmlns:s="http://example.org/students/vocab#"
  cos:graph="http://ns.inria.org/example1/" >
  <rdf:Description rdf:about="http://example.org/courses/6.001">
  <s:students rdf:parseType="Collection" >
  <rdf:Description rdf:about="http://example.org/students/Amy">
  <s:mark>A</s:mark>
  </rdf:Description>
  <rdf:Description rdf:about="http://example.org/students/Mohamed">
  cos:graph="http://ns.inria.org/example2/" >
  <s:mark>B</s:mark>
  </rdf:Description>
  <rdf:Description rdf:about="http://example.org/students/Johann">
  <s:mark>B</s:mark>
  </rdf:Description>
  </s:students>
  </rdf:Description>
  </rdf:RDF>
```

The resulting graph is:



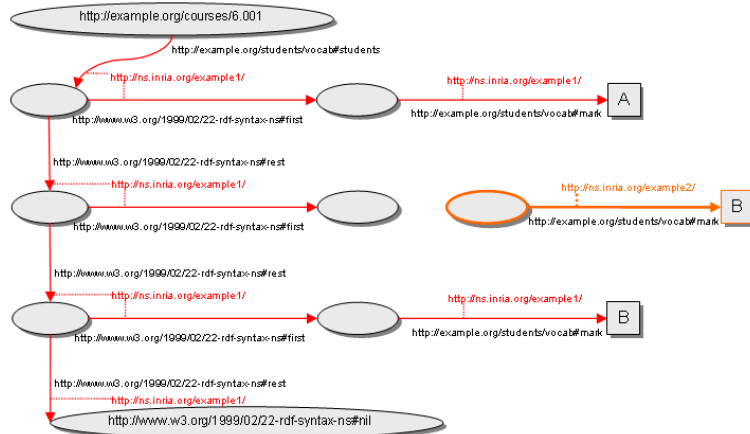
Quadruples generated for this example are:

```
<http://example.org/courses/6.001> s:students _:a <- http://ns.inria.org/example1/
_:a rdf:first <http://example.org/students/Amy> <- http://ns.inria.org/example1/
_:a rdf:rest _:b <- http://ns.inria.org/example1/
<http://example.org/students/Amy> s:mark "A" <- http://ns.inria.org/example1/
_:b rdf:first <http://example.org/students/Mohamed> <- http://ns.inria.org/example1/
_:b rdf:rest _:c <- http://ns.inria.org/example1/
<http://example.org/students/Mohamed> s:mark "B" <- http://ns.inria.org/example2/
_:c rdf:_3 <http://example.org/students/Johann> <- http://ns.inria.org/example1/
_:c rdf:rest rdf:nil <- http://ns.inria.org/example1/
<http://example.org/students/Johann> s:mark "B" <- http://ns.inria.org/example1/
```

Now if we have a blank node:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s="http://example.org/students/vocab#"
  cos:graph="http://ns.inria.org/example1/" >
  <rdf:Description rdf:about="http://example.org/courses/6.001">
    <s:students rdf:parseType="Collection">
      <rdf:Description>
        <s:mark>A</s:mark>
      </rdf:Description>
      <rdf:Description cos:graph="http://ns.inria.org/example2/">
        <s:mark>B</s:mark>
      </rdf:Description>
      <rdf:Description>
        <s:mark>B</s:mark>
      </rdf:Description>
    </s:students>
  </rdf:Description>
</rdf:RDF>
```

The resulting graph is:



Quadruples generated for this example are:

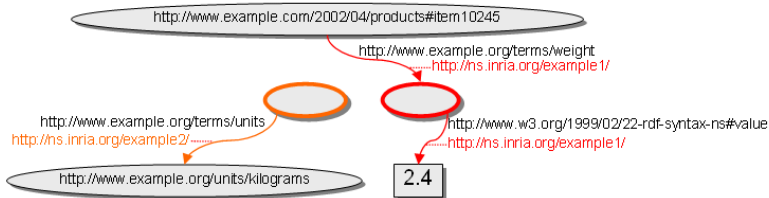
```
<http://example.org/courses/6.001> s:students _:a <- http://ns.inria.org/example1/
_:a rdf:first :d <- http://ns.inria.org/example1/
_:a rdf:rest _:b <- http://ns.inria.org/example1/
_:d s:mark "A" <- http://ns.inria.org/example1/
_:b rdf:first _:e <- http://ns.inria.org/example1/
_:b rdf:rest _:c <- http://ns.inria.org/example1/
_:e s:mark "B" <- http://ns.inria.org/example2/
_:c rdf:first :g <- http://ns.inria.org/example1/
_:c rdf:rest rdf:nil <- http://ns.inria.org/example1/
_:g s:mark "B" <- http://ns.inria.org/example1/
```

4.4. Structured values

Structured values usually make use of implicit blank nodes generated by `rdf:parseType`. One should be careful in changing the source associated to a property of a value. In the following example the link between the value and its units is lost:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:extermns="http://www.example.org/terms/"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  cos:graph="http://ns.inria.org/example1/" >
  <rdf:Description rdf:about="http://www.example.com/2002/04/products#item10245">
    <extermns:weight rdf:parseType="Resource">
      <rdf:value rdf:datatype="&xsd:decimal">2.4</rdf:value>
      <extermns:units rdf:resource="http://www.example.org/units/kilograms"
        cos:graph="http://ns.inria.org/example2/" />
    </extermns:weight>
  </rdf:Description>
</rdf:RDF>
```

The resulting graph is:



Quadruples generated for this example are:

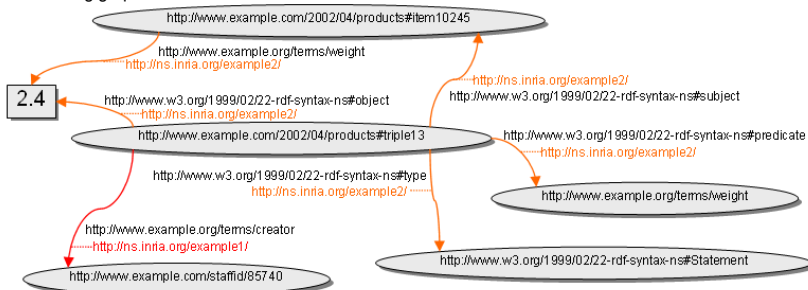
```
<http://www.example.com/2002/04/products#item10245> exterm:weight _:a <- http://ns.inria.org/example1/
_:a rdf:value "2.4"^^xsd:decimal <- http://ns.inria.org/example1/
_:b rdf:units <http://www.example.org/units/kilograms> <- http://ns.inria.org/example2/
```

4.5. Reification

Triples generated by the reification of a triple belong to the same source as the original triple:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:exterm="http://www.example.com/terms/"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  xml:base="http://www.example.com/2002/04/products"
  cos:graph="http://ns.inria.org/example1/">
  <rdf:Description rdf:ID="item10245">
    <exterm:weight rdf:ID="triple13" cos:graph="http://ns.inria.org/example2/">2.4</exterm:weight>
  </rdf:Description>
  <rdf:Description rdf:about="#triple13">
    <exterm:creator rdf:resource="http://www.example.com/staffid/85740"/>
  </rdf:Description>
</rdf:RDF>
```

The resulting graph is:



Quadruples generated for this example are:

```
<http://www.example.com/2002/04/products#item10245> exterm:weight "2.4" <- http://ns.inria.org/example2/
<http://www.example.com/2002/04/products#triple13> rdf:object "2.4" <- http://ns.inria.org/example2/
<http://www.example.com/2002/04/products#triple13> rdf:type rdf:Statement <- http://ns.inria.org/example2/
<http://www.example.com/2002/04/products#triple13> rdf:predicate exterm:weight <- http://ns.inria.org/example2/
<http://www.example.com/2002/04/products#triple13> rdf:subject <http://www.example.com/2002/04/products#item10245> <- http://ns.inria.org/example2/
<http://www.example.com/2002/04/products#triple13> exterm:creator <http://www.example.com/staffid/85740> <- http://ns.inria.org/example1/
```

4.6. Describing sources

The IRI of a source can itself be the subject or object of RDF triple:

```
<rdf:RDF xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  cos:graph="http://ns.inria.fr">
  <foaf:Person rdf:about="http://ns.inria.fr/fabien.gandon">
    <foaf:name>Fabien Gandon</foaf:name>
    <foaf:mbox rdf:resource="mailto:Fabien.Gandon@sophia.inria.fr"/>
  </foaf:Person>
  <foaf:Organization rdf:about="http://ns.inria.fr">
    <foaf:name>INRIA</foaf:name>
    <foaf:member rdf:resource="http://ns.inria.fr/fabien.gandon"/>
  </foaf:Organization>
</rdf:RDF>
```

Syntactically it is possible to use the same IRI for a property and a source; it is considered a bad practice and if a source is a described resource in OWL the OWL semantics would enforce that its IRI is not used to identify a property.

Note: one can even declare a class (here `cos:Graph`) that corresponds to all resources that are used as sources e.g.:

```
<rdfs:Class rdf:about="http://www.inria.fr/acacia/corese#Graph">
  <rdfs:isDefinedBy rdf:resource="http://www.inria.fr/acacia/corese#" />
  <rdfs:label xml:lang="en">Source</rdfs:label>
  <rdfs:comment xml:lang="en">The class of the RDF sources.</rdfs:comment>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2000/01/rdf-schema#Resource" />
</rdfs:Class>
```

With a rule like `IF GRAPH ?src { ?x ?p ?y } THEN ?src rdf:type cos:Graph`

4.7. RDFS cases

Sources can be used in RDFS schemas to specify the source of the definitions. In the following example the property definition is associated to a different source.

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns="http://www.w3.org/2000/01/rdf-schema#"
  cos:graph="http://ns.inria.fr/2006/05/research_onto.rdfs"
  xml:base="http://ns.inria.fr/2006/05/research_onto.rdfs">
  <Class rdf:ID="Man">
    <subClassOf rdf:resource="#Person"/>
    <subClassOf rdf:resource="#Male"/>
    <label xml:lang="en">man</label>
    <comment xml:lang="en">an adult male person</comment>
  </Class>
  <rdf:Property rdf:about="http://xmlns.com/foaf/0.1/name"
    cos:graph="http://xmlns.com/foaf/0.1/"
    rdfs:label="name" rdfs:comment="A name for some thing.">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty" />
  </rdf:Property>
</rdf:RDF>
```



```

<rdfs:domain rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
<rdfs:range rdf:resource="http://www.w3.org/2000/01/rdf-schema#Literal"/>
<rdfs:isDefinedBy rdf:resource="http://xmlns.com/foaf/0.1/" />
<rdfs:subPropertyOf rdf:resource="http://www.w3.org/2000/01/rdf-schema#label"/>
</rdf:Property>
</rdf:RDF>

```

4.8. OWL cases

Sources can be used in OWL schemas in particular to link the source of a schema and its namespace:

```

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns="http://www.w3.org/2000/01/rdf-schema#"
  cos:graph="http://ns.inria.fr/2006/05/research_onto.rdfs"
  xml:base="http://ns.inria.fr/2006/05/research_onto.rdfs">
  <owl:Ontology rdf:about="http://ns.inria.fr/2006/05/research_onto.rdfs" >
    <rdfs:label>research ontology</rdfs:label>
    <rdfs:comment>defines primitives to describe research activities</rdfs:comment>
    <rdfs:comment cos:graph="http://www.w3.org">RDFS_VALID</rdfs:comment>
    <owl:versionInfo>1.3</owl:versionInfo>
  </owl:Ontology>
</rdf:RDF>

```

Once again, one should be careful with blank nodes when annotating schemas, be they explicit or implicit. This is particularly true for some primitives in OWL such as restrictions, unions, intersections.

Example causing a problem: the restriction blank node will generate two different blank nodes, one in each source.

```

<Class rdf:ID="Human" cos:graph="http://ns.inria.fr/2006/05/research_onto.rdfs">
  <subClassOf>
    <owl:Restriction cos:graph="http://ns.inria.fr/2004/02/research_onto.rdfs">
      <owl:onProperty rdf:resource="#hasMother" />
      <owl:allValuesFrom rdf:resource="#Woman" />
    </owl:Restriction>
  </subClassOf>
</Class>

```

Example causing no problem: sources are changed on named nodes.

```

<owl:Class rdf:ID="Professor" cos:graph="http://ns.inria.fr/2006/05/research_onto.rdfs">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Researcher" cos:graph="http://ns.inria.fr/2004/02/research_onto.rdfs" />
    <owl:Class rdf:about="#Lecturer" />
  </owl:intersectionOf>
</owl:Class>

```

5. Tests and implementation

This specification was tested with a modified version of the [SPARQL engine CORESE](#).

Test description	RDF/XML source	SPARQL query	SPARQL XML Binding
Simple source example	test01.rdf	test01.sparql	test01.xml
Blank node example with source on property	test02.rdf	test02.sparql	test02.xml
Blank node example with source on the blank node	test03.rdf	test03.sparql	test03.xml
Blank node example with same sources	test04.rdf	test04.sparql	test04.xml
Bag example with the blank node problem	test05.rdf	test05.sparql	test05.xml
Collection with no blank node problem	test06.rdf	test06.sparql	test06.xml
Collection with a blank node problem	test07.rdf	test07.sparql	test07.xml
Structured value with a blank node problem	test08.rdf	test08.sparql	test08.xml
Reification example	test09.rdf	test09.sparql	test09.xml
Description of source	test10.rdf	test10.sparql	test10.xml
RDFS ontology example	test11.rdf	test11.sparql	test11.xml
OWL ontology example	test12.rdf	test12.sparql	test12.xml

6. Discussion

This specification was driven by use cases from several of our projects. However there is one case we left out of the scope: the case where one wants to attach several sources to a triple. We did not find a good syntax for this case and we don't know what it would imply in terms of SPARQL querying.

7. Acknowledgements

We would like to thank all the members of our research team Edelweiss for fruitful discussions and INRIA for funding our standardization activities.

8. References

- [CORESE], COnceptual REsource Search Engine, INRIA, August 2007, <http://www-sop.inria.fr/edelweiss/wiki/wakka.php?wiki=Corese>.
- [RDF], Resource Description Framework (RDF): Concepts and Abstract Syntax, Graham Klyne, Jeremy J. Carroll, Brian McBride, W3C Recommendation 10 February 2004, <http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>.
- [RDF/XML], RDF/XML Syntax Specification (Revised), Dave Beckett, Brian McBride, W3C Recommendation 10 February 2004, <http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/>.
- [SPARQL], SPARQL Query Language for RDF, Eric Prud'hommeaux, Andy Seaborne, W3C Candidate Recommendation 14 June 2007, <http://www.w3.org/TR/2007/CR-rdf-sparql-query-20070614/>.