

Linguaggi per il Web

Introduction to RDF

Riccardo Rosati

Corso di Laurea in Ingegneria Informatica e Automatica

Corso di Laurea in Ingegneria dei Sistemi Informatici

Sapienza Università di Roma

a.a. 2011/2012

What is the Semantic Web?

- “The Semantic Web is a Web of actionable information—information derived from data through a **semantic theory** for interpreting the symbols.”
- “The semantic theory provides an account of ‘meaning’ in which the logical connection of terms establishes **interoperability** between systems”

(Shadbolt, Hall, Berners-Lee, The Semantic Web revisited, IEEE Intelligent Systems, May 2006)

The Semantic Web: why?

- search on the Web: problems...
- ...due to the way in which information is stored on the Web
- **Problem 1:** web documents do not distinguish between information content and presentation (“solved” by XML)
- **Problem 2:** different web documents may represent in different ways semantically related pieces of information
- this leads to hard problems for “intelligent” information search on the Web

Separating content and presentation

Problem 1: web documents do not distinguish between information content and presentation

- problem due to the HTML language
- problem “solved” by current technology
 - stylesheets (HTML, XML)
 - XML
- stylesheets allow for separating formatting attributes from the information presented

Separating content and presentation

- XML: eXtensible Mark-up Language
- XML documents are written through a user-defined set of tags
- tags are used to express the “semantics” of the various pieces of information

XML: example

HTML:

```
<H1>Seminari di Ingegneria del Software</H1>
  <UL>
    <LI>Teacher: Giuseppe De Giacomo
    <LI>Room: 7
    <LI>Prerequisites: none
  </UL>
```

XML:

```
<course>
  <title>Seminari di Ingegneria del Software
</title>
  <teacher>Giuseppe De Giacomo</teacher>
  <room>1AI, 1I</room>
  <prereq>none</prereq>
</course>
```

Limitations of XML

XML does not solve all the problems:

- legacy HTML documents
- different XML documents may express information with the **same meaning** using **different tags**

The need for a “Semantic” Web

Problem 2: different web documents may represent in different ways semantically related pieces of information

- different XML documents do not share the “semantics” of information
- idea: annotate (mark-up) pieces of information to express the “meaning” of such a piece of information
- the meaning of such tags is shared!
⇒ **shared semantics**

The Semantic Web initiative

viewpoint:

the Web = a web of data

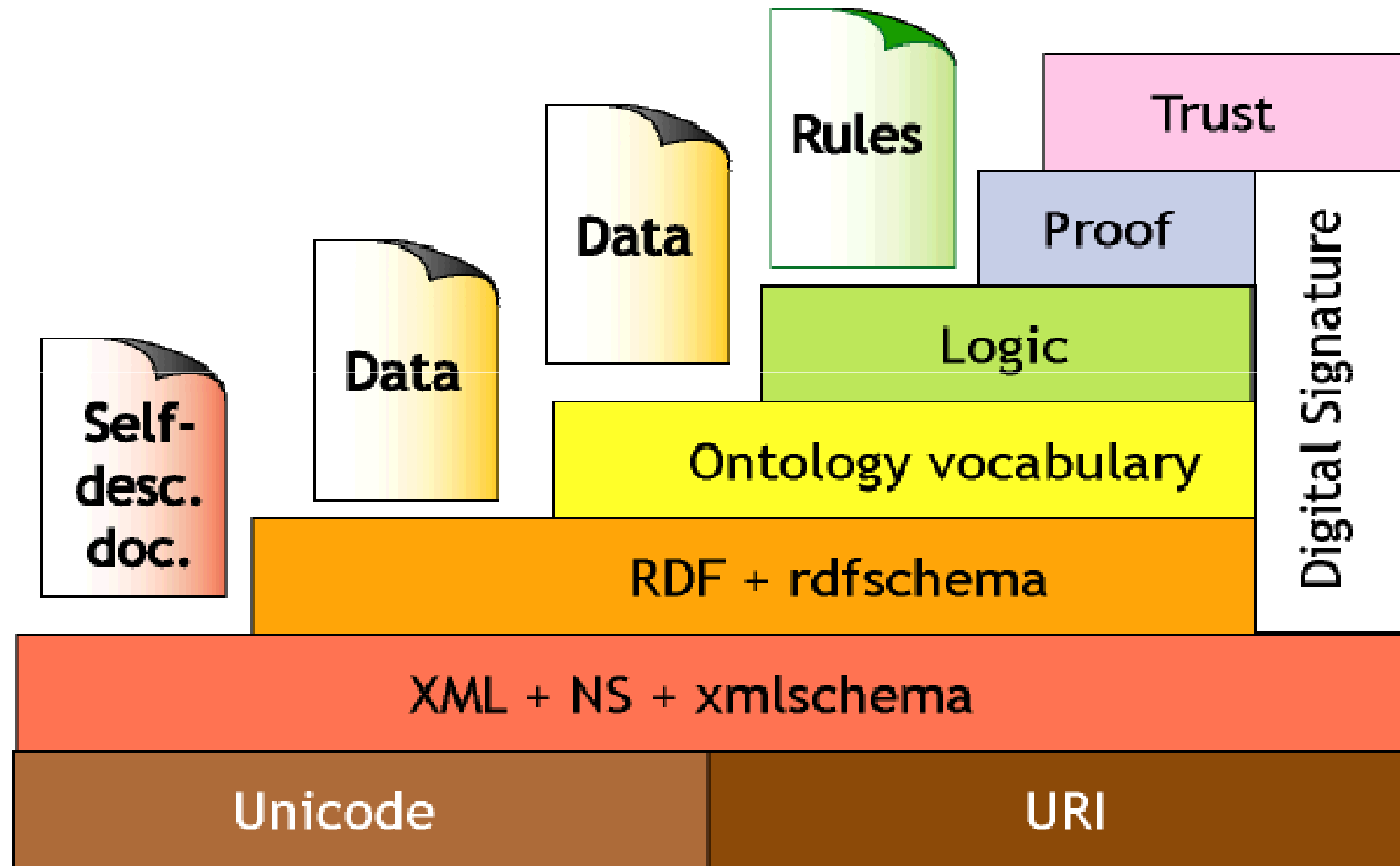
goal:

to provide a common framework to share data on the Web across application boundaries

main ideas:

- ontology
- standards
- “layers”

The Semantic Web Tower



The Semantic Web Layers

- XML layer
- RDF + RDFS layer
- Ontology layer
- Proof-rule layer
- Trust layer

The XML layer

- XML (eXtensible Markup Language)
 - user-definable and domain-specific markup
- URI (Uniform Resource Identifier)
 - universal naming for Web resources
 - same URI = same resource
 - URIs are the “ground terms” of the SW
- W3C standards

XML: example

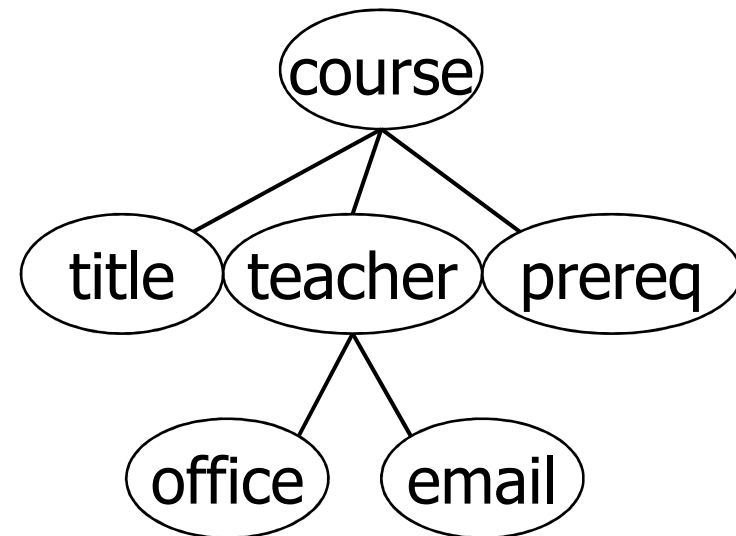
```
<course date="2007">  
  <title>Seminari di Ingegneria del Software  
  </title>  
  <teacher>  
    <name>Giuseppe De Giacomo</name>  
    <email>degiacomo@dis.uniroma1.it</email>  
  </teacher>  
  <prereq>none</prereq>  
</course>
```

XML

- XML: document = labelled tree
- node = label + attributes/values + contents

```
<course date="...">  
  <title>...</title>  
  <teacher>  
    <office>...</office>  
    <email>...</email>  
  </teacher>  
  <room>...</room>  
  <prereq>...</prereq>  
</course>
```

=



XML

- XML Schema = grammar for describing legal trees and datatypes
- can we use XML to represent semantics?

XML and semantics

<Predator>

...

</Predator>

- Predator: a medium-altitude, long-endurance unmanned aerial vehicle system.
- Predator : one that victimizes, plunders, or destroys, especially for one's own gain.
- Predator : an organism that lives by preying on other organisms.
- ...

Limitations for semantic markup

- XML makes no commitment on:
 1. Domain-specific ontological vocabulary
 2. Ontological modeling primitives
- Requires pre-arranged agreement on 1 and 2
- Only feasible for closed collaboration:
 - agents in a small & stable community
 - pages on a small & stable intranet
- Not suited for sharing Web-resources

RDF

- RDF is a data model
 - the model is domain-neutral, application-neutral and ready for internationalization
 - the model can be viewed as directed, labeled graphs or as an object-oriented model (object/attribute/value)
- RDF data model is an abstract, conceptual layer independent of XML
 - consequently, XML is a transfer syntax for RDF, not a component of RDF
 - RDF data might never occur in XML form

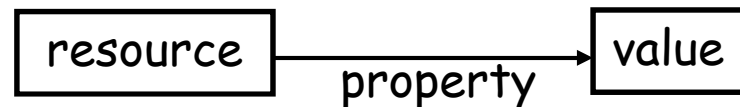
RDF model

RDF model = set of RDF triples

triple = expression (statement)

(subject, predicate, object)

- subject = resource
- predicate = property (of the resource)
- object = value (of the property)



RDF triples

example: “the document at
<http://www.w3c.org/TR/REC-rdf-syntax>
has the author Ora Lassila”

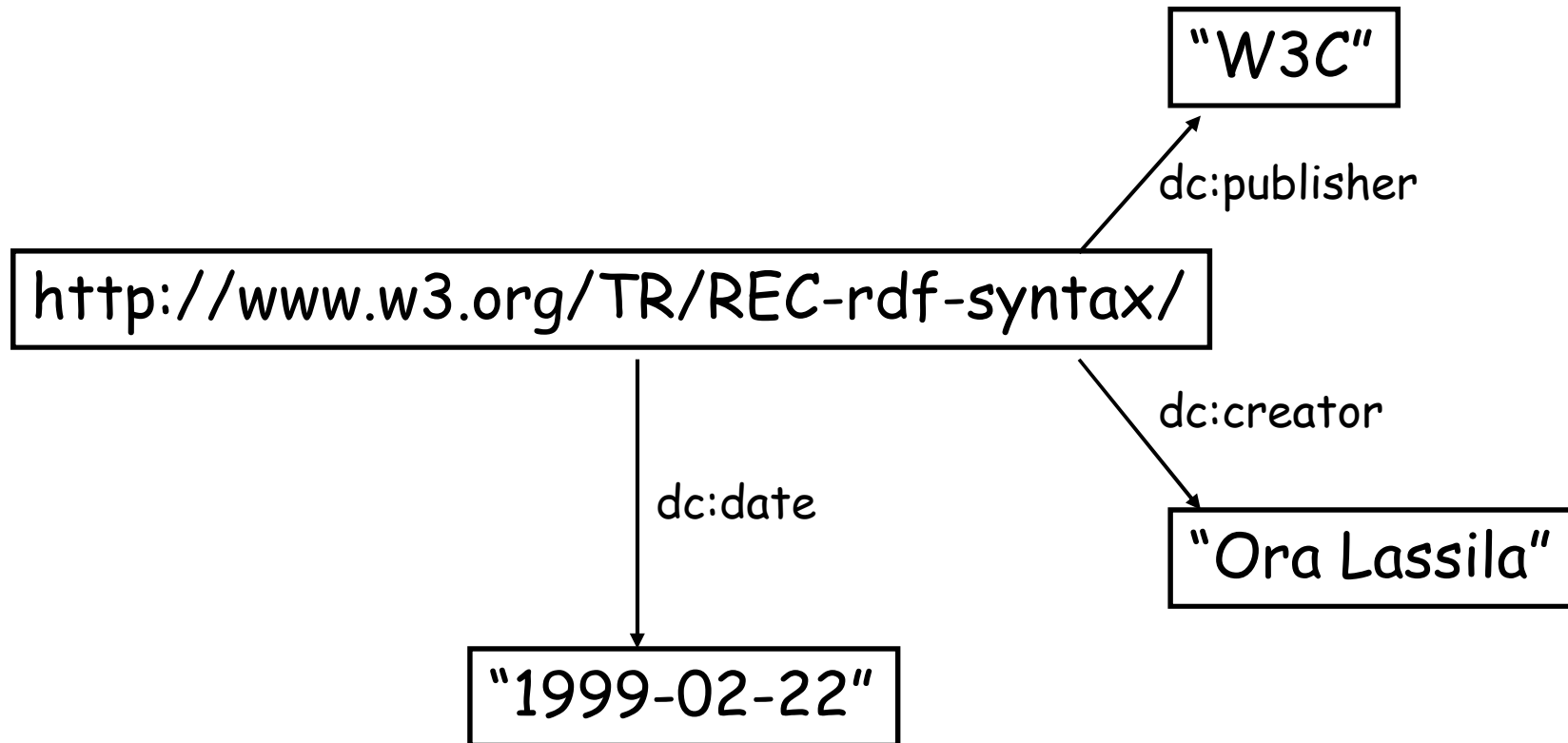
triple:

<http://www.w3c.org/TR/REC-rdf-syntax> author “OraLassila”



⇒ RDF model = **graph**

RDF graph: example



Node and edge labels in RDF graphs

node and edge labels:

- **URI**
- **literal** (string)
- **blank node** (anonymous label)

but:

- a literal can only appear in object positions
- a blank node can only appear in subject or object positions

Complex values

- values of properties need not be simple strings
- the value of a property can also be a graph node (corresponding to a resource)
 - arbitrarily complex tree and graph structures are possible
 - syntactically, values can be embedded (i.e., lexically in-line) or referenced (linked)

Blank nodes

- **blank node** (bnode) = RDF graph node with “anonymous label” (i.e., not associated with an URI)
- example:
- "John has a friend born the 21st of April"

```
ex:John    foaf:knows    _:p1
_:p1      foaf:birthdate  04-21
```

- **_:p1** is the blank node (b-node)
- bnodes can be used both as subjects and objects

Higher-order statements

- One can make RDF statements about other RDF statements
 - example: “Ralph believes that the web contains one billion documents”
- Higher-order statements
 - allow us to express beliefs (and other modalities)
 - are important for trust models, digital signatures, etc.
 - also: metadata about metadata
 - are represented by modeling RDF in RDF itself

⇒ reification

Reification

Reification in RDF = using an RDF statement as the subject (or object) of another RDF statement

Examples of statement that need reification to be expressed in RDF:

- “the New York Times claims that Joe is the author of book ABC”
- “the statement “The technical report on RDF was written by Ora Lassila” was written by the Library of Congress”

Reification

- RDF provides a built-in predicate vocabulary for reification:
 - **rdf:subject**
 - **rdf:predicate**
 - **rdf:object**
 - **rdf:statement**
- Using this vocabulary (i.e., these URIs from the rdf: namespace) it is possible to represent a triple through a blank node

Reification: example

- the statement “The technical report on RDF was written by Ora Lassila” can be represented by the following four triples:

```
_:x rdf:predicate dc:creator.  
_:x rdf:subject http://www.w3.org/TR/REC-rdf-syntax.  
_:x rdf:object "Ora Lassila".  
_:x rdf:type rdf:statement.
```

- The blank node `_:x` is the **reification** of the statement (it is an anonymous URI that represents the whole triple)
- Now, we can express properties about the triple represented by the blank node `_:x`.
- For instance, we can express the fact that the triple `_:x` was created by the Library of Congress through the following triple: `_:x dc:creator "Library of Congress"`.

Reification in RDF: example

The statement

“the creator of the statement “the creator of the document on RDF syntax is Ora Lassila” is the Library of Congress”

is therefore represented by the following five triples:

```
_:x rdf:predicate dc:creator .
_:x rdf:subject "http://www.w3.org/TR/RED-rdf-syntax" .
_:x rdf:object "Ora Lassila" .
_:x rdf:type "rdf:statement" .
_:x dc:Creator "Library of Congress" .
```

RDF syntaxes

RDF model = edge-labeled graph = set of triples

- graphical notation (graph)
- (informal) triple-based notation
e.g., (**subject, predicate, object**)
- formal syntaxes:
 - N3 notation
 - Turtle notation
 - concrete (serialized) syntax: RDF/XML syntax

RDF syntaxes

- N3 notation:

`subject predicate object .`

- Turtle notation: example:

`@prefix`

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

`:mary rdf:type <http://www.ex.org/Gardener>.`

`:mary :worksFor :ElJardinHaus.`

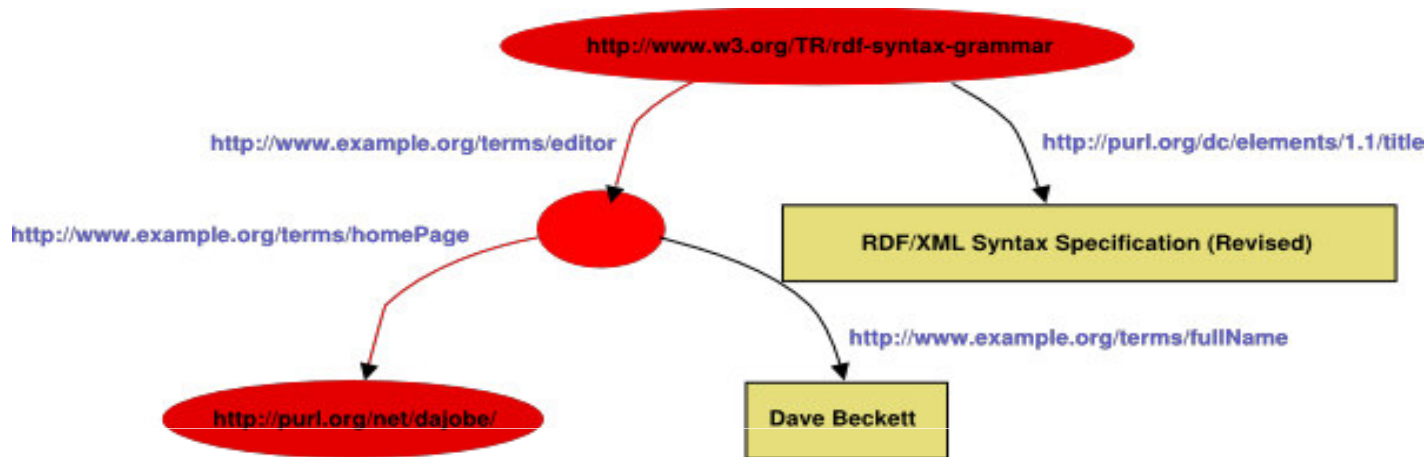
`:mary :name "Dalileh Jones"@en.`

`_:john :worksFor :ElJardinHas.`

`_:john :idNumber "54321"^^xsd:integer.`

- concrete (serialized) syntax: RDF/XML syntax

RDF/XML syntax: Example



```
<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/XML syntax: Example (2)

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">]>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:exterms="http://www.example.com/terms/">
  <rdf:Description rdf:ID="item10245">
    <exterms:model rdf:datatype="&xsd:string">Overnighter</exterms:model>
    <exterms:sleeps rdf:datatype="&xsd:integer">2</exterms:sleeps>
    <exterms:weight rdf:datatype="&xsd:decimal">2.4</exterms:weight>
    <exterms:packedSize rdf:datatype="&xsd:integer">784</exterms:packedSize>
  </rdf:Description>
</rdf:RDF>
```