Knowledge Representation for the Semantic Web

Winter Quarter 2011

Slides 3 – 01/11/2011

Pascal Hitzler
Kno.e.sis Center
Wright State University, Dayton, OH
http://www.knoesis.org/pascal/
Textbook (required)

Pascal Hitzler, Markus Krötzsch, Sebastian Rudolph

Foundations of Semantic Web Technologies

Chapman & Hall/CRC, 2010

Choice Magazine Outstanding Academic Title 2010 (one out of seven in Information & Computer Science)

http://www.semantic-web-book.org
Today: RDF syntax
Today’s Session: RDF

1. Motivation
2. Triples and Graphs
3. RDF syntaxes: Turtle and RDF/XML
4. Datatypes
5. n-ary relationships
6. Empty nodes
7. Lists
8. Class project
9. Class presentation
Two XML Problems

• How do you encode the piece of knowledge “The book FOST is published by CRC Press”

• `<book>
  `<title>FOST</title>
  `<publisher>CRC Press</publisher>
  `</book>

• `<publisher>
  `<name>CRC Press</name>
  `<book>`<title>FOST</title>`<book>
  `</publisher>

• etc.
Two XML Problems

- Merging trees is rather cumbersome and the result isn’t always clear.
  - <publisher>
    <name>CRC Press</name>
    <book><title>FOST</title></book>
  </publisher>

- <book>
  <title>Semantic Web</title>
  <publisher>Springer</publisher>
</book>
RDF idea

- Use (directed) graphs as data model
RDF

- “Resource Description Framework”

- W3C Recommendation 2004
  http://www.w3.org/RDF/

- RDF is a data model
  - originally for describing metadata for web pages, but has grown beyond that
  - structured information
  - universal, machine-readable data exchange format
  - main syntax uses XML for serialization
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Motivation</td>
</tr>
<tr>
<td>2</td>
<td>Triples and Graphs</td>
</tr>
<tr>
<td>3</td>
<td>RDF syntaxes: Turtle and RDF/XML</td>
</tr>
<tr>
<td>4</td>
<td>Datatypes</td>
</tr>
<tr>
<td>5</td>
<td>n-ary relationships</td>
</tr>
<tr>
<td>6</td>
<td>Empty nodes</td>
</tr>
<tr>
<td>7</td>
<td>Lists</td>
</tr>
<tr>
<td>8</td>
<td>Class project</td>
</tr>
<tr>
<td>9</td>
<td>Class presentation</td>
</tr>
</tbody>
</table>
RDF components

• URIs
  – for referencing resources

• Literals
  – data values

• Empty nodes
  – talking about something which doesn’t have a name (or the name of which isn’t known)
Literals

- for representing data values
- encoded as strings
- interpreted by means of datatypes
- literals without datatype are treated the same as strings
Graphs as sets of triples

- there are several possibilities for representing graphs

- we use: graph as list of (node-edge-node) triples
RDF triples

• An RDF triple consists of

Subject: http://semantic-web-book.org/uri

Predicate: http://example.org/publishedBy

Object: http://crcpress.com/uri

(borrowed from linguistics)

• allowed are:
  – In the subject: URIs and empty nodes
  – In the predicate: URIs (usually called properties)
  – In the object: URIs and empty nodes and literals

• Note that the graph can be reconstructed from the list of triples.
Table of contents : RDF

1. Motivation
2. Triples and Graphs
3. RDF syntaxes: Turtle and RDF/XML
4. Datatypes
5. n-ary relationships
6. Empty nodes
7. Lists
8. Class project
9. Class presentation
Turtle – Terse RDF Triple Language

- simple syntax for RDF
- triples are directly listed as such
  - URIs are in <angle brackets>
  - Literals are ”enclosed in quotes”
  - triples end with a full-stop .
  - whitespace (blanks, line feeds) is ignored

```turtle
<http://semantic-web-book.org/uri>
<http://semantic-web-book.org/uri>
  <http://example.org/title>
    "Foundations of Semantic Web Technologies" .
<http://crcpress.com/uri>
  <http://example.org/name> "CRC Press" .
```
• shortcuts for prefixes

@prefix ex: <http://example.org/> .
@prefix crc: <http://crcpress.com/> .

crc:uri  ex:name        "CRC Press" .
Turtle

- grouping of triples with the same subject
- grouping of triples with same subject and predicate
XML syntax for RDF

- Turtle is easy to read and write
- But XML is the basis for data transfer on the web
- There’s a lot of tool (and programming library) support for XML
- Hence, the main syntax for RDF is XML-based.
- Turtle is not a W3C recommendation
- The normative syntax for RDF is it’s XML syntax
XML syntax for RDF

- namespaces are used for disambiguating tags
- tags belonging to the RDF language come with a fixed namespace, usually abbreviated 'rdf'

```xml
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:ex ="http://example.org/">
    <rdf:Description rdf:about="http://semantic-web-book.org/uri">
        <ex:publishedBy>
            <rdf:Description rdf:about="http://crcpress.com/uri">
                </rdf:Description>
        </ex:publishedBy>
    </rdf:Description>
</rdf:RDF>
```
XML syntax for RDF

```
<rdf:Description rdf:about="http://semantic-web-book.org/uri">
  <ex:publishedBy>
    <rdf:Description rdf:about="http://crcpress.com/uri">
    </rdf:Description>
  </ex:publishedBy>
</rdf:Description>
```
XML syntax for RDF

- Untyped literals can be left as free text
- A subject can contain several property elements
- Object-descriptions can be used as subject-descriptions for further triples

```xml
<rdf:Description rdf:about="http://semantic-web-book.org/uri">
  <ex:title>Foundations of Semantic Web Technologies</ex:title>
  <ex:publishedBy>
    <rdf:Description rdf:about="http://crcpress.com/uri">
      <ex:name>CRC Press</ex:name>
    </rdf:Description>
  </ex:publishedBy>
</rdf:Description>
```
XML syntax for RDF

- Equivalent representation of literals using XML attributes
  - the attribute-name is then the property-URI

- Equivalent representation of objects by giving their URIs as value of a rdf:resource attribute within a property tag.

```
<rdf:Description rdf:about="http://semantic-web-book/uri"
    ex:title= "Foundations of Semantic Web Technologies">
  <ex:publishedBy rdf:resource="http://crcpress.com/uri" />
</rdf:Description>

<rdf:Description rdf:about="http://crcpress.com/uri"
    ex:Name="CRC Press" />
```
XML syntax for RDF

- The use of namespaces is essential since the use of the colon `:` in XML attributes is not allowed unless it is used with a namespace.

- Problem: namespaces cannot be used in values of XML attributes: `rdf:about="book:uri"` is wrong since `book` would be interpreted in the sense of a URI schema.

- Solution: use XML ENTITYs.

```xml
<?xml version="1.0" encoding="utf-8"?> <!DOCTYPE rdf:RDF[
]

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

  <rdf:Description rdf:about="&book;uri">
    <ex:title>Foundations of Semantic Web Technologies</ex:title>
  </rdf:Description>

</rdf:RDF>
```
• Use of the base namespace

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

  <rdf:Description rdf:about="uri">
    <ex:publishedBy rdf:resource="http://crcpress.com/uri" />
  </rdf:Description>

</rdf:RDF>
```
Table of contents : RDF

1. Motivation
2. Triples and Graphs
3. RDF syntaxes: Turtle and RDF/XML
4. Datatypes
5. n-ary relationships
6. Empty nodes
7. Lists
8. Class project
9. Class presentation
Datatypes in RDF

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
<br:tr rdf:resource="http://www.w3.org/TR/rdf-primer"
     rdf:about="http://www.w3.org/TR/rdf-primer">
  <http://example.org/title> "RDF Primer"^^xsd:string ;
  <http://example.org/publicationDate> "2004-02-10"^^xsd:date .

<rdf:Description rdf:about="http://www.w3.org/TR/rdf-primer">
  <ex:title rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    RDF Primer
  </ex:title>
  <ex:publicationDate
    rdf:datatype="http://www.w3.org/2001/XMLSchema#date">
    2004-02-10
  </ex:publicationDate>
</rdf:Description>
Datatypes

- usually use of XML Schema datatype
- Note that the same data value can have different representations:
  "3.14"^^xsd:decimal is the same as "+03.14"^^xsd:decimal
  but
  "3.14"^^xsd:string is not the same as "+03.14"^^xsd:string
- there is only one required datatype in RDF, called rdf:XMLLiteral
  - arbitrary (balanced) XML fragments
  - special syntax:

```
<rdf:Description rdf:about="http://semantic-web-book/uri">
  <ex:title rdf:parseType="Literal">
    Foundations of
    <br />
    <b>Semantic Web Technologies</b>
  </ex:title>
</rdf:Description>
```
Table of contents: RDF

1. Motivation
2. Triples and Graphs
3. RDF syntaxes: Turtle and RDF/XML
4. Datatypes
5. n-ary relationships
6. Empty nodes
7. Lists
8. Class project
9. Class presentation
What is wrong with these?

@prefix ex: <http://example.org/> .
ex:Chutney ex:hasIngredient "1lb green mango",
                 "1tsp. Cayenne pepper" .

@prefix ex: <http://example.org/> .
ex:Chutney ex:ingredient ex:greenMango; ex:amount "1lb";
ex:ingredient ex:CayennePepper; ex:amount "1tsp." .
It’s a ternary relationship!

@prefix ex: <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
ex:Chutney    ex:hasIngredient   ex:ingredient1 .
ex:ingredient1 rdf:value    ex:greenMango;
ex:amount      "1lb" .
Table of contents: RDF

1. Motivation
2. Triples and Graphs
3. RDF syntaxes: Turtle and RDF/XML
4. Datatypes
5. n-ary relationships
6. Empty nodes
7. Lists
8. Class project
9. Class presentation
It doesn’t need a name :)
Blank nodes syntax

```xml
<rdf:Description rdf:about="http://example.org/Chutney">
  <ex:hasIngredient rdf:nodeID="id1" />
</rdf:Description>

<rdf:Description rdf:nodeID="id1">
  <ex:ingredient rdf:resource="http://example.org/greenMango" />
  <ex:amount>1lb</ex:amount>
</rdf:Description>
```

**shortcut:**

```xml
<rdf:Description rdf:about="http://example.org/Chutney">
  <ex:hasIngredient rdf:parseType="Resource">
    <ex:ingredient rdf:resource="http://example.org/greenMango" />
    <ex:amount>1lb</ex:amount>
  </ex:hasIngredient>
</rdf:Description>
```
Blank nodes syntax

Turtle:

```turtle
@prefix ex: <http://example.org/> .
ex:Chutney  ex:hasIngredient  _:id1 .
_:id1  ex:ingredient  ex:greenMango;  ex:amount  "1lb" .
```
Blank nodes syntax

@prefix ex: <http://example.org/> .
ex:Chutney ex:hasIngredient _:id1 .
_:id1 ex:ingredient ex:greenMango; ex:amount "1lb" .

shortcut:

@prefix ex: <http://example.org/> .
ex:Chutney ex:hasIngredient
[ ex:ingredient ex:greenMango; ex:amount "1lb" ] .
Table of contents: RDF

1. Motivation
2. Triples and Graphs
3. RDF syntaxes: Turtle and RDF/XML
4. Datatypes
5. n-ary relationships
6. Empty nodes
7. Lists
8. Class project
9. Class presentation
Open lists (containers)

```
<rdf:Description rdf:about="http://semantic-web-book.org/uri">
  <ex:authors>
    <rdf:Seq>
    </rdf:Seq>
  </ex:authors>
</rdf:Description>
```
Types of containers

- “open”: new elements can be added.
- rdf:Seq – ordered list
- rdf:Bag – unordered set
- rdf:Alt – set of alternatives

Lists are actually hardly reflected in the formal semantics (more about this later)
Closed lists (collections)

```xml
<rdf:Description rdf:about="http://semantic-web-book/uri">
  <ex:authors rdf:parseType="Collection">
  </ex:authors>
</rdf:Description>
```

book:uri <http://example.org/authors>

Closed lists (collections)
<table>
<thead>
<tr>
<th></th>
<th>1. Motivation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2. Triples and Graphs</td>
</tr>
<tr>
<td></td>
<td>3. RDF syntaxes: Turtle and RDF/XML</td>
</tr>
<tr>
<td></td>
<td>4. Datatypes</td>
</tr>
<tr>
<td></td>
<td>5. n-ary relationships</td>
</tr>
<tr>
<td></td>
<td>6. Empty nodes</td>
</tr>
<tr>
<td></td>
<td>7. Lists</td>
</tr>
<tr>
<td></td>
<td><strong>8. Class project</strong></td>
</tr>
<tr>
<td></td>
<td>9. Class presentation</td>
</tr>
</tbody>
</table>
Class project – status

Domains:

- vehicles
- university
- stock exchange
- language
- computers
- butterflies
- games
- hostile human action
- social networks

- Be punctual!
- Send me readable input!
Class project: next step

- re-check your taxonomy for correctness!
- add ca. 10 instances to your taxonomy

Diagram:

```
Human
   /\  \
  Man  Woman
     /     / \
   pascal anne merula
```

- add ca. 8 subject-predicate-object triples to your taxonomy, reusing the instances you created, and inventing suitable predicates (RDF properties). Use Turtle syntax for these.
  ```turtle
  anne   motherOf   merula .
  ```
- introduce changes to your ontology in whatever way needed
- document briefly what you have done and why (in particular if you find bugs!
- send to me by Thursday noon
Table of contents : RDF

1. Motivation
2. Triples and Graphs
3. RDF syntaxes: Turtle and RDF/XML
4. Datatypes
5. n-ary relationships
6. Empty nodes
7. Lists
8. Class project
9. Class presentation
Class presentations – first topics

  http://www.w3.org/2009/sparql/docs/entailment/xmlspec.xml
- Jacopo Urbani, Spyros Kotoulas, Jason Maassen, Frank van Harmelen, Henri E. Bal: OWL Reasoning with WebPIE:
  Calculating the Closure of 100 Billion Triples. ESWC (1) 2010: 213-227
- Yuan Ren, Jeff Z. Pan, Yuting Zhao: Soundness Preserving Approximation for TBox Reasoning. AAAI 2010
- Franz Baader, Sebastian Brandt, Carsten Lutz: Pushing the EL Envelope. IJCAI 2005: 364-369
Thursday 13\textsuperscript{th} of January: RDFS Part I  
Tuesday 18\textsuperscript{th} of January: Exercise Session  
Thursday 25\textsuperscript{th} of January: RDF and RDFS Semantics  

Estimated breakdown of sessions:  
- Intro + XML: 2  
- RDF: 3  
- OWL and Logic: 6  
- SPARQL and Querying: 2  
- Class Presentations: 3  
- Exercise sessions: 3