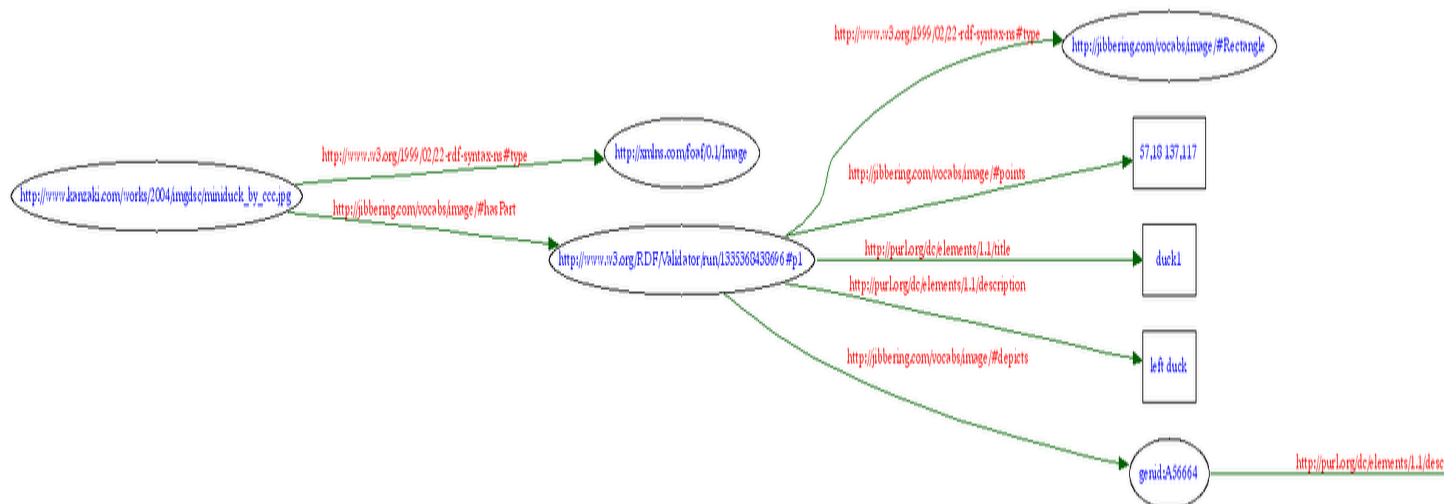


Lab 1 - Introduction to the Semantic Web

1. Extracting RDF metadata

```
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:foaf="http://xmlns.com/foaf/0.1/"
xmlns:image="http://jibbering.com/vocabs/image/#"
xmlns:an="http://www.w3.org/2000/10/annotation-ns#"
>
<foaf:Image rdf:about="http://www.kanzaki.com/works/2004/imgdsc/
miniduck_by_ccc.jpg">
  <image:hasPart>
    <image:Rectangle rdf:ID="p1">
      <image:points>57,18 137,117</image:points>
      <dc:title>duck1</dc:title>
      <dc:description>left duck</dc:description>
      <image:depicts rdf:parseType="Resource">
        <dc:description>the duck is called bob</dc:description>
      </image:depicts>
    </image:Rectangle>
  </image:hasPart>
</foaf:Image>
</
rdf:RDF>
```



2. Common Vocabularies

1. What is each vocabulary for?

- a. SKOS - thesauri, classification schemes, subject heading lists, taxonomies, folksonomies, and other types of controlled vocabulary
 - b. DCMI - from simple data description to , to combining metadata vocabularies of different metadata standards (linked data cloud, semantic web implementation)
 - c. FOAF - for describing people, groups, documents and links
 - d. DOAP - (link in wiki is dead) describe semantic information associated with open-source software projects
2. What application use them?
- a. SKOS - Library of Congress Subject Headings, UN FAO project AGROVOC
 - b. DCMI - Open Source Metadata Framework (used by GNOME, KDE help browsers), PBCore.
 - c. FOAF - Profiles on LiveJournal, My Opera, WordPress, Identi.ca or FriendFeed.
 - d. DOAP - Mozilla Foundation project page

3. FOAF

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:admin="http://webns.net/mvcb/">
<foaf:PersonalProfileDocument rdf:about="">
  <foaf:maker rdf:resource="#me"/>
  <foaf:primaryTopic rdf:resource="#me"/>
  <admin:generatorAgent rdf:resource="http://www.ldodds.com/foaf/foaf-a-matic"/>
  <admin:errorReportsTo rdf:resource="mailto:leigh@ldodds.com"/>
</foaf:PersonalProfileDocument>
<foaf:Person rdf:ID="me">
<foaf:name>Konrad Malawski</foaf:name>
<foaf:title>Mr</foaf:title>
<foaf:givenname>Konrad</foaf:givenname>
<foaf:family_name>Malawski</foaf:family_name>
<foaf:nick>ktoso</foaf:nick>
<foaf:mbox_sha1sum>fb669964925db13fad4433e41b2e7ed25c275fd</foaf:mbox_sha1sum>
<foaf:homepage rdf:resource="http://blog.project13.pl"/>
<foaf:phone rdf:resource="tel:555555"/>
<foaf:workplaceHomepage rdf:resource="http://softwaremill.pl"/>
<foaf:workInfoHomepage rdf:resource="Software design and development"/>
>
<foaf:schoolHomepage rdf:resource="http://www.agh.edu.pl"/>
<foaf:knows>
<foaf:Person>
<foaf:name>Kolegal</foaf:name>
<foaf:mbox_sha1sum>76ca316c5dbb81081d63cfceecbf53611cdc6160</foaf:mbox_sha1sum>
<rdfs:seeAlso rdf:resource="kolega.site.example.com"/></foaf:Person></foaf:knows>
<foaf:knows>
```

```
<foaf:Person>
<foaf:name>Kolega2</foaf:name>
<foaf:mbox_shalsum>e00932d9b0c7f528fcac5461957ddbaed8c4acea</
foaf:mbox_shalsum>
<rdfs:seeAlso rdf:resource="kolega2.site.example.com"/></
foaf:Person></foaf:knows></foaf:Person>
</rdf:RDF>
```

<http://up.project13.pl/files/ktoso.foaf>

4. Ontology reasoning

Unable to complete task due to OwlSight webpage not working.

5. Semantic Web tools

RDFizers: which in your opinion are the most important? What an RDFizer would you add (converting from what format)?

We found JIRA -> RDF conversion really intriguing, JIRA issues contain loads of information which could thanks to RDF be easily leveraged by other tooling.

A converter from GPX (<http://www.topografix.com/gpx.asp>), an GPS+POI data information exchange format to RDF could be quite interesting - preparing ground for semantic geographic search engines possibly?

Semantic Web browsers: How many of them are up and running?

The given website lists 12 such browsers. But razorbase and objectviewer.daml.org seem to be offline.

Semantic Web Search Engines: How easy is it to get started with each of them?

Very easy. Queries can be typed in human readable format - simply words.

Applications of the Linked Data, including mashups and everyday tools: Which ones do you find useful? What other applications can you think of?

sameAs.org seems to have a very appealing use case of allowing you to find information about some resource on different websites, allowing easy "cross site but same search topic" searches.

One could apply the same mechanism to searching for music or movies, based on tags (which are embedded in MP3 files for example).

6. Control Questions

How one can add semantic annotation to a web page?

Link to RDF document in head section or embedd semantic annotation using RDFa.

Explain the Semantic Web Stack of technologies.

It illustrates architecture of Semantic Web.

XML provides an elemental syntax for content structure within documents using Unicode charset (UTF-8). RDF is simple language to express data models: objects (resources) and thier relationship. RDF Schema is vocabulary describing classes and properties. OWL extends vocabulary with things such as relations between classes or properties characteristics. SPARQL is query language for semantic data.

What is the main syntax for RDF? What are its advantages over other syntaxes?

RDF contains this classes:

- `rdf:XMLLiteral` - the class of XML literal values
- `rdf:Property` - the class of properties
- `rdf:Statement` - the class of RDF statements
- `rdf:Alt`, `rdf:Bag`, `rdf:Seq` - containers of alternatives, unordered containers, and ordered containers (`rdfs:Container` is a super-class of the three)
- `rdf:List` - the class of RDF Lists
- `rdf:nil` - an instance of `rdf:List` representing the empty list

and this properties:

- `rdf:type` - an instance of `rdf:Property` used to state that a resource is an instance of a class
- `rdf:first` - the first item in the subject RDF list
- `rdf:rest` - the rest of the subject RDF list after `rdf:first`
- `rdf:value` - idiomatic property used for structured values
- `rdf:subject` - the subject of the subject RDF statement
- `rdf:predicate` - the predicate of the subject RDF statement
- `rdf:object` - the object of the subject RDF statement

XML syntax (so RDF too) is very descriptive and it could by read even without browsers. XML also is very simple so everyone can use it.

What is the role of the ontologies in the Semantic Web?

Ontologies are used for representing knowlage concepts and relations between these contepts.

What are the 4 principles of Linked Data?

1. Use URIs to identify things.
2. Use HTTP URIs so that these things can be referred to and looked up ("dereferenced") by people and user agents.
3. Provide useful information about the thing when its URI is dereferenced, using standard formats such as RDF/XML.
4. Include links to other, related URIs in the exposed data to improve discovery of other related information on the Web.

Lab 2 - eXtensible Markup Language (XML)

1. Warm-up: Well-formedness of the XML

```
<a><b/><a>
```

```
a
|-- b
```

```
<a><b>foo</b></a><a>bar</a>
```

```
a
|-- b -- foo
a -- bar
```

```
<a><b>foo</b><b>bar</a>
```

^ invalid, b not closed

```
<a><b><c>foo</b>bar</c></a>
```

^ invalid, closing unexpected element

```
<a/><b>foo</b><b>bar<b>
```

^ invalid, b element (2 times) not closed

```
<a><b><c>foo</c>baz<c>bar</c></b></a>
```

^ invalid

```
<a x="1" X="2"><b>foo</b></a>
```

```
a x = 1
|-- b -- foo
```

```
<a x="1"><b x="2">foo</b></a>
```

```
a x=1
|-- b x=2 -- foo
```

```
<a x="1" y="2" x="3"><b>foo</b></a>
```

^ attribute redefinition

```
<a><b x="3"><c y="1">foo</c><c>bar</c></b></a>
```

```
a
|-- b x = 3
   |-- c y = 1 -- foo
   |-- c -- bar
```

2. Creating XML document

```
<book isbn="978-83-246-0956-7">
  <author>Eric A. Mayer</author>
  <title>CSS: The Definitive Guide</title>
  <year>2007</year>
  <publisher>O'Reilly</publisher>
</book>
```

3. DTD

```
<!DOCTYPE BOOK [

  <!ELEMENT BOOK (AUTHOR,TITLE,YEAR,PUBLISHER)>
  <!ELEMENT AUTHOR (#PCDATA)>
  <!ELEMENT TITLE (#PCDATA)>
  <!ELEMENT YEAR (#PCDATA)>
  <!ELEMENT PUBLISHER (#PCDATA)>

  <!ATTLIST BOOK ISBN CDATA #REQUIRED>

]>
```

4. XSD

```
<schema
  xmlns='http://www.w3.org/2000/10/XMLSchema'
  targetNamespace='http://www.w3.org/namespace/'
  xmlns:t='http://www.w3.org/namespace/'>

  <element name='BOOK'>
    <complexType>
      <sequence>
        <element ref='t:AUTHOR'/>
        <element ref='t:TITLE'/>
        <element ref='t:YEAR'/>
        <element ref='t:PUBLISHER'/>
      </sequence>
      <attribute name='ISBN' type='string' use='required'/>
    </complexType>
```

```
</element>  
  
<element name='AUTHOR' type='xs:string' />  
  
<element name='TITLE' type='xs:string' />  
  
<element name='YEAR' type='xs:string' />  
  
<element name='PUBLISHER' type='xs:string'>  
</schema>
```