RDF Schema

Maciej Janik

Semantic Web
2009-07-01
RDF Schemas

- Describe rules for using RDF properties
- Are expressed in RDF
- Extends original RDF vocabulary
- Are not to be confused with XML Schemas

- RDFS defines a basic set of classes and properties, together with their **semantics** (interpretation) and **logic**
RDFS Vocabulary

- RDFS namespace
  - http://www.w3.org/2000/01/rdf-schema#

- rdfs:Resource
- rdfs:Class
- rdfs:Literal
- rdfs:Datatype
- rdf:XMLLiteral
- rdf:Property

- rdfs:range
- rdfs:domain
- rdf:type
- rdfs:subClassOf
- rdfs:subPropertyOf

- rdfs:label
- rdfs:comment

- rdfs:Container
  - rdf:Bag
  - rdf:Seq
  - rdf:Alt
- rdfs:ContainerMembershipProperty
  - rdf:_1, rdf_2 ...
- rdfs:member
  - rdf:List
  - rdf:first
  - rdf:rest
  - rdf:nil

- rdf:Statement
  - rdf:subject
  - rdf:predicate
  - rdf:object

- rdfs:seeAlso
- rdfs:isDefinedBy
- rdf:value
RDF Classes

- Are groups of Web resources
- Have URLs to identify them
- The special class “rdfs:Literal” consists of all possible RDF string values
Property-centric classes

- In typical OO classes, each class specifies completely what properties it has and what their types are.
- In RDF classes, each property specifies what classes of subjects and objects it relates.
- Therefore, new properties can be added to a class without modifying the class.
Specifying classes

- To specify a class, create an RDF resource of type rdfs:Class

```xml
<rdfs:Class id="MyClass">
  <rdfs:label>My Class</rdfs:label>
  <rdfs:comment>Sample demonstration Class</rdfs:comment>
</rdfs:Class>
```
Specify properties

To specify a property, create an RDF resource of type rdfs:Property

```xml
<rdfs:Property id="myProperty">
  <rdfs:comment>Sample demo property</rdfs:comment>
  <rdfs:domain resource="#MyClass"/>
  <rdfs:range resource="#Literal"/>
</rdfs:Property>
```
Schema URIs

- Ordinary XML namespace URIs are just to guarantee uniqueness: there is no assumption that the URI refers to anything useful (or even refers at all)
- URIs for namespaces used in RDF, though, should refer to an RDF schema document
Properties (1)

- "rdf:type" relates any resource to its class

- "rdfs:subClassOf" relates a subclass to its superclass (multiple inheritance is OK)

  baby rdfs:subClassOf child

- "rdfs:subPropertyOf" relates a subproperty to its superproperty

  mother rdfs:subPropertyOf parent
Properties (2)

- “rdfs:seeAlso” relates a resource to another resource explaining it (use a subproperty to specify the nature of the explanation)

- “rdfs:isDefinedBy” is a subproperty of “rdfs:seeAlso” and relates a resource to its definition, typically an RDF schema
“rdfs:domain” specifies the domain of a property (the classes of its subjects); if unknown, anything can be a subject.

“rdfs:range” specifies the range of a property (the single class of its objects); if unknown, anything can be an object.

Domain and range effectively restrict classes that can be in specific relationship.
Properties (4)

- "rdf:subject" is the property relating a reified statement to its subject (resource)

- "rdf:predicate" is the property relating a reified statement to its predicate (property)

- "rdf:object" is the property relating a reified statement to its object (value)
Properties (5)

- “rdfs:label” specifies a human-readable name for this Class, Property, or whatever

- “rdfs:comment” specifies human-readable documentation
  - Multiple values are useful for specifying multiple languages
“rdfs:Resource” is the class of all resources

“rdfs:Literal” is the class of all strings

“rdfs:Class” is the class of all classes

“rdfs:Property” is the class of all properties

“rdf:Statement” is the class of all asserted RDF statements
“rdfs:Container” is the superclass of all container classes

“rdf:Bag”, “rdf:Seq”, “rdf:Alt” are the classes of Bags (multi-set), Seqs (sequence), and Alts (alternative)

Any other class that is a subclass of “rdfs:Container” can be used in RDF syntax in place of a standard container
ex:Seminar rdfs:subClassOf ex:Meeting
ex:Proseminar rdfs:subClassOf ex:Seminar
ex:Lecture rdfs:subClassOf ex:Meeting
ex:Seminar rdfs:subClassOf ex:Meeting
ex:Proseminar rdfs:subClassOf ex:Seminar
ex:Lecture rdfs:subClassOf ex:Meeting
ex:Professor rdfs:subClassOf ex:Employee
All classes are instances of **rdfs:Class**.
All classes are sub-classes of `rdfs:Resource`
Such duality in RDF definition can be useful further in meta-modeling.
Orange arrows (connections) belong to RDFS standard and are always included in model.
Inference (2) - without property names

Orange connections belong to RDFS standard and are always included (or added) in model
Inference (3) – transitivity of `rdfs:subclassOf`

```
ex:Meeting 
  ├── rdfs:subclassOf 
  │   ├── ex:Seminar 
  │   └── ex:Lecture 
  └── rdfs:subclassOf 
      └── ex:ProSeminar
```
Instances and classes

ex:Ethics \text{rdf:type} ex:Lecture  \quad ex:SOA \text{rdf:type} ex:Seminar
ex:Kant \text{rdf:type} ex:Professor

Terminological Knowledge

ex:Employee \text{rdfs:subclassOf} ex:Professor

ex:Meeting \text{rdfs:subclassOf} ex:Lecture
ex:Meeting \text{rdfs:subclassOf} ex:Seminar

ex:Professor \text{rdf:type} ex:Employee
ex:Professor \text{rdf:type} ex:Employee

ex:Lecture \text{rdf:type} ex:ProSeminar
ex:Seminar \text{rdf:type} ex:ProSeminar

ex:Ethics \text{rdf:type} ex:Lecture
ex:SOA \text{rdf:type} ex:Seminar

Assertional Knowledge
Inference (4) - Ploymorphism

ex:Ethics rdf:type ex:Lecture
ex:SOA rdf:type ex:Seminar
ex:Kant rdf:type ex:Professor
Multiple inheritance

ex:Employee

rdfs:subclassOf

ex:Professor

rdf:type

ex:Kant

rdfs:subclassOf

ex:Independent_Researcher

rdfs:subclassOf

ex:Genius

rdf:type
ex:Professor and ex:University_Teacher define in fact the same class
Identity

- `ex:Employee` is a subclass of `ex:Professor`.
- `ex:Professor` is a subclass of `ex:Independent_Researcher`.
- `ex:Independent_Researcher` is a subclass of `ex:University_Teacher`.
- `ex:Kant` is of type `ex:Employee`.
RDF Schema in XML

```xml
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xml:base="http://example.org"
         xmlns:ex="http://example.org/">
  <rdfs:Class rdf:about="Proseminar">
    <rdfs:subClassOf rdf:resource="Seminar"/>
    <rdfs:subClassOf rdf:resource="Meeting"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="Lecture">
    <rdfs:subClassOf>Meeting</rdfs:subClassOf>
    <rdfs:label>Lecture at the University</rdfs:label>
    <rdfs:comment>Comment goes here …</rdfs:comment>
  </rdfs:Class>

  <ex:Lecture rdf:ID="Ethics"/>
  <ex:Seminar rdf:ID="SOA">
    <ex:taughtBy>Staab</ex:taughtBy>
    <ex:taughtBy>Grimm</ex:taughtBy>
  </ex:Seminar>
</rdf:RDF>
```
Properties

ex:taughtBy
  rdf:type rdfs:Property
  rdfs:domain ex:Lecture
  rdfs:range ex:Professor

ex:Ethics ex:taughtBy ex:Kant

ex:Ethics rdf:type ex:Lecture
Properties

ex:taughtBy
  rdf:type rdfs:Property
  rdfs:domain ex:Lecture
  rdfs:range ex:Professor

ex:Ethics ex:taughtBy ex:Kant

ex:Ethics rdf:type ex:Lecture

ex:taughtBy
  rdfs:domain ex:Book

ex:Ethics rdf:type ex:Book
Properties

This is NOT object-oriented

uni:taughtBy
  rdfs:domain ex2:Lecture
  rdfs:range ex1:Professor
  rdf:type rdfs:Property

ex1:Professor rdf:type rdfs:Class.

This is NOT object-oriented

ex2:Lecture rdf:type rdfs:Class.
Sub-Properties

\[
\text{ex:worksAt} \\
\text{rdfs:domain ex:Person} \\
\text{rdfs:range ex:Institution} \\
\text{rdf:type rdfs:Property}
\]

\[
\text{ex:teachesAt} \\
\text{rdfs:domain ex:Professor} \\
\text{rdfs:range ex:Institution} \\
\text{rdf:type rdfs:Property}
\]

\[
\text{ex:teachesAt rdfs:subPropertyOf ex:worksAt}
\]
Sub-Properties

ex:worksAt
  rdfs:domain ex:Person
  rdfs:range ex:Institution
  rdf:type rdfs:Property

ex:teachesAt
  rdfs:domain ex:Professor
  rdfs:range ex:Institution
  rdf:type rdfs:Property

ex:teachesAt rdfs:subPropertyOf ex:worksAt

ex:Kant ex:teachesAt ex:University

ex:Kant ex:worksAt ex:University

ex:Kant rdf:type ex:Person

ex:Kant rdf:type ex:Professor

Kant must be BOTH Professor and Person
RDF Common Vocabulary

Commonly used vocabularies on the web
Widely used and accepted RDF Schema

- Dublin Core
  - Metadata initiative – annotation of resources

- RSS 1.0
  - RDF Site Summary 1.0

- FOAF
  - Friend of a friend – social networks

- SWRC
  - Semantic Web for Research Communities
Dublin Core

- Originally
  - A set of fifteen (15) basic properties for describing generalized Web resources

- Now
  - *Dublin Core Metadata Initiative*: open organization engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models
  - Core set of metadata terms (DCMI Metadata Terms)
    - Core vocabulary size over 100 *common agreed* terms
  - Multiple working groups / communities dedicated to specific issues
1. “Title”: the name given to the resource
2. “Creator”: the person or organization primarily responsible for the resource
3. “Subject”: what the resource is about
4. “Description”: a description of the content
5. “Publisher”: the person or organization responsible for making the resource available
6. “Contributor”: someone who has provided content to the resource other than the creator
7. “Date”: date of creation or publication
8. “Type”: type of resource, such as home page, technical report, novel, photograph...
9. “Format”: data format of the resource
10.“Identifier”: URL, ISBN number, …
11.“Source”: another resource that this resource is derived from
12.“Language”: the language of the content
13.“Relation”: another resource and its relationship to this one
14.“Coverage”: the portion of time or space described by this resource (atlases, histories, etc.)
15.“Rights”: the intellectual property rights adhering to this resource, or a pointer to them
RSS 1.0

- **RSS**
  - Really Simple Syndication protocol → RDF Site Summary
- **RSS 1.0** is a lightweight multipurpose extensible metadata description and syndication format.
- **Syndication**
  - making data available online for retrieval and further transmission, aggregation, or online publication.
- **RSS**
  - Almost anything that can be broken down into discrete items can be syndicated via RSS: news, recent changes page of a wiki, blog entries …
  - News aggregators are popular in the weblogging community
  - Many weblogs and sited make content available in RSS.
RSS 1.0

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns="http://purl.org/rss/1.0/">
  <channel rdf:about="http://www.xml.com/xml/news.rss">
    <title>XML.com</title>
    <link>http://xml.com/pub</link>
    <description>XML.com description</description>
    <image rdf:resource="http://xml.com/universal/images/xml_tiny.gif" />
    <items>
      <rdf:Seq>
        <rdf:li resource="http://..." />
      </rdf:Seq>
    </items>
  </channel>
  <image rdf:about="http://xml.com/universal/images/xml_tiny.gif">
    <title>XML.com</title>
    <link>http://www.xml.com</link>
    <url>http://xml.com/universal/images/xml_tiny.gif</url>
  </image>
    <title>Putting RDF to Work</title>
    <description>Description goes here ...</description>
  </item>
  <item rdf:about="http://...">
    ...
  </item>
</rdf:RDF>
FOAF

- Friend-Of-A-Friend
- machine-readable **ontology** describing **persons**, their **activities** and their **relations** to other people and objects
- interlinks people and explicitly defines social relationships (friend), building a global social network

- facilitates the creation of the Semantic Web equivalent of the **archetypal personal homepage**
  - “My name is Leigh, this is a picture of me, I'm interested in XML, and here are some links to my friends.”
FOAF vocabulary


**Personal info**

**Online Accounts/IM**
- **OnlineAccount**, **OnlineChatAccount**, **OnlineEcommerceAccount**, **OnlineGamingAccount**, **socialAccount**, **accountServiceHomepage**, **accountName**, **cqChalID**, **msrChatID**, **amChatID**, **ebberID**, **yahooChatID**

**Projects and Groups**
- **Project**, **Organization**, **Group**, **member**, **membershipClass**, **fundedBy**, **theme**

**Documents and Images**
- **Document**, **Image**, **PersonaProfileDocument**, **topic (page)**, **primaryTopic**, **tipjar**, **sha1**, **made (maker)**, **thumbna1**, **logo**
<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/">

  <foaf:Person rdf:ID="me">
    <foaf:name>Maciej Janik</foaf:name>
    <foaf:title>Dr</foaf:title>
    <foaf:givenname>Maciej</foaf:givenname>
    <foaf:family_name>Janik</foaf:family_name>
    <foaf:family_name>Janik</foaf:family_name>
    <foaf:mbox_sha1sum>809d...a3f</foaf:mbox_sha1sum>
    <foaf:homepage rdf:resource="http://isweb.uni-koblenz.de/~janik" />
    <foaf:depiction rdf:resource="http://.../MaciejJanik.jpg" />
    <foaf:phone rdf:resource="tel:00492612872718" />
    <foaf:schoolHomepage rdf:resource="http://www.agh.edu.pl" />
    <foaf:knows>
      <foaf:Person>
        <foaf:name>Steffen Staab</foaf:name>
        <foaf:mbox_sha1sum>ae8f...f90</foaf:mbox_sha1sum>
      </foaf:Person>
    </foaf:knows>
  </foaf:Person>
</rdf:RDF>
SWRC

- Semantic Web for Research Communities
- ontology for modeling entities of research communities such as persons, organizations, publications (bibliographic metadata) and their relationships

Main concepts of the SWRC ontology
In this work, control strategies for combining two potentially powerful buffer management techniques in object bases were devised and evaluated: (1) buffer pool segmentation with segment-specific replacement criteria, and (2) dual buffering consisting of copying objects from pages into object buffers. Two dimensions exist for exerting control on the buffer pool: (1) the \{it copying\} time which determines at what time objects are copied from their memory-resident home page, and (2) the \{it relocation\} time which determines when a (copied) object is to be transferred back to its home page. Along both dimensions, it is possible to differentiate between an \{it eager\} and a \{it lazy\} strategy. The extensive experimental results indicate that lazy object copying combined with an eager relocation strategy is almost always superior and significantly outperforms page-based buffering in most applications.
Extended vocabulary of RDF
- Class and property (relationship) inheritance
- Inference schemes
  - Classes
  - Properties
  - Transitivity of some properties (subClass, subPropertyOf)
- Division to schema (definitions) and instances (factual knowledge)
- Domain and range restrictions (with its implications
- Many widely used vocabularies
- Still not expressive enough …

→ OWL (Web Ontology Language)