

RDFS

Michael Wollowski

RDF Statements

- A **Statement** consists of a *subject*, *predicate* and *object*

- Example:

The author of <http://www.w3schools.com/RDF> is Jan Egil Refsnes

- Subject: <http://www.w3schools.com/RDF>
- Predicate: author
- Object: Jan Egil Refsnes

RDF Statements

- In technical terms, we have a **Resource**, a **Property**, and a **Property value** corresponding to subject, predicate and object of a Statement.
- Example:
The author of <http://www.w3schools.com/RDF> is Jan Egil Refsnes.
- Resource: <http://www.w3schools.com/RDF>
- Property: author
- Property value: Jan Egil Refsnes

Resources

- We can think of a resource as an object, a “thing” we want to talk about
 - E.g. authors, books, publishers, places, people, hotels
- Every resource has a **URI**, a Universal Resource Identifier
- A URI can be
 - a URL (Web address) or
 - some other kind of unique identifier

Properties

- Properties are a special kind of resources
- They describe relations between resources
 - E.g. “written by”, “age”, “title”, etc.
- Properties are also identified by URIs
- Advantages of using URIs:
 - A global, worldwide, unique naming scheme
 - Reduces the homonym problem of distributed data representation

RDF Example

- XML based

```
<?xml version="1.0"?>
```

```
<rdf:RDF >
```

```
<rdf:Description rdf:about="http://www.recshop.fake/cd/Empire Burlesque">
```

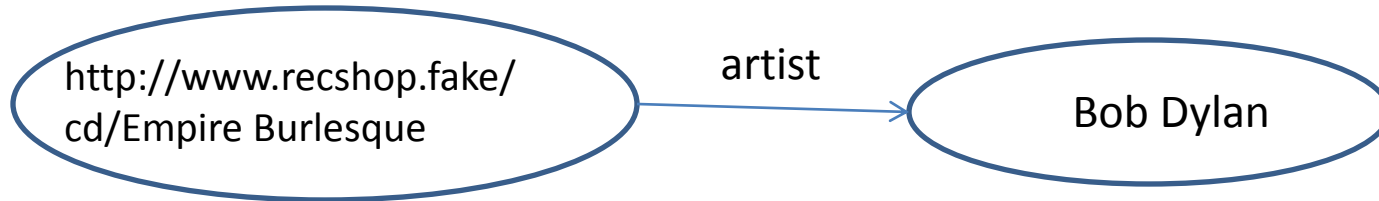
```
  <cd:artist>Bob Dylan</cd:artist>
```

```
  <cd:country>USA</cd:country>
```

```
  ...
```

```
</rdf:RDF>
```

Semantic Net



- A directed graph with labeled nodes and arcs
 - **from** the resource (the **subject** of the statement)
 - **to** the value (the **object** of the statement)
- Known in AI as a *semantic net*
- The value of a statement may be a resource
 - It may be linked to other resources

Example Evaluated

- How to avoid the Tower of Babel?
 - All services use same namespace
 - May as well agree on the same syntax for XML
 - In other words, develop a single standard
 - At least, develop a very small set of standards
- Alternatively, facilitate matching of namespace elements.

RDFS

- RDFS stands for RDF Schema
- It enables the definition of classes and class hierarchies
- Additionally, we can define property hierarchies

RDFS Example

```
<rdfs:Class rdf:ID="recordingMedium" />
```

```
<rdfs:Class rdf:ID="cd">
```

```
  <rdfs:subClassOf rdf:resource="#recordingMedium"/>
```

```
</rdfs:Class>
```

```
<rdfs:Property rdf:ID="artist"/>
```

```
<rdfs:Property rdf:ID="starvingArtist">
```

```
  <rdfs:subPropertyOf rdf:resource="#artist"/>
```

```
</rdfs:Class>
```

RDFS Example

- See slides

OWL

- More sophisticated relationships
- Disjointness of classes
 - Sometimes we wish to say that classes are disjoint (e.g. **male** and **female**)
- Boolean combinations of classes
 - Sometimes we wish to build new classes by combining other classes using union, intersection, and complement
 - E.g. **person** is the disjoint union of the classes **male** and **female**

OWL

- Cardinality restrictions
 - E.g. a person has exactly two parents, a course is taught by at least one lecturer
- Special characteristics of properties
 - Transitive property (like “greater than”)
 - Unique property (like “is mother of”)
 - A property is the inverse of another property (like “eats” and “is eaten by”)

Owl Example

- See slides