

RDF und RDF's - Kommentare

Cristina Vertan

Anmerkungen zu RDF und RDF's -1-

- URI's sind URL's sehr ähnlich aber nicht identisch. Ein URI bezeichnet eine Ressource die nicht unbedingt im WWW ist
- Die Semantik von "Vocabulary" und "Schema" ist dieselbe. Für RDF wird nur Vocabulary benutzt.
- QName. Präfix :local name
 - Das Präfix ist einem "Parent folder" ähnlich. In RDF ist das Präfix ein URI.
 - Wenn mehrere URI's ein gemeinsames Präfix haben, bedeutet das nicht dass es zwischen die URI's irgendwelche Beziehungen gibt.

Anmerkungen zu RDF und RDF's -2-

- Man muß sich gut überlegen ob ein Objekt eines Statements nicht Subjekt für ein weiteres Statement sein kann. Literals können nicht Subjekt sein !
- URI's identifizieren eine Ressource einmalig ABER: unterschiedliche URI's können dieselbe Ressource identifizieren. Es ergibt sich dann die Nötwendigkeit eine Beziehung zwischen diese URI's zu bilden (Ontologien).
- Was die Menschen von der URI bedeutung verstehen \neq was die Maschinen verstehen

Anmerkungen zu RDF und RDF's -3-

- Leere Knoten (Blank-nodes) werden meistens um eine N-are Relation in eine binäre Relation zu transformieren.
- Mehrere leere Knoten können diesen Namen haben
- Typed Literals: www.w3.org/2001/XMLSchema#type. Im Vergleich zu Programmiersprachen, kann man nicht prüfen ob die Assoziation: Literal \leftrightarrow Data-Typ korrekt ist.

Typed Literals und XML Entitäten

```
<?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:about="http://www.example.org/index.html">
    <exterms:creation-date rdf:datatype="&xsd:date">1999-08-16
  </exterms:creation-date>
</rdf:Description>
</rdf:RDF>
```

Abkürzungen für URI-Refs

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
```

RDF-Containers

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s="http://example.org/students/vocab#">
  <rdf:Description rdf:about="http://example.org/courses/6.001">
    <s:students>
      <rdf:Bag>
        <rdf:li rdf:resource="http://example.org/students/John"/>
        <rdf:li rdf:resource="http://example.org/students/Marie"/>
        ....
      </rdf:Bag>
    </s:students>
  </rdf:Description>
</rdf:RDF>
```

RDF-Reification

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
```

RDF:Value

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
```

RDF Schema vs. OOP

- Merkmale sind unabhängig von Definitionen von Klassen
- Eine RDF Schema kann ein Merkmal (Property) beschreiben ohne eine bestimmte Klasse zu spezifizieren .
- In RDF Schema kann ein “Property” keine lokale Werte haben (abhängig von Klassen von Ressourcen)

Grenzen des RDF-Schemas

- Kardinalität-Bedingungen für Merkmalen
- Transitivität ist für Merkmale nicht unterstützt
- Man kann nicht spezifizieren dass ein Merkmal nur für eine Klasse gilt.
- Man kann nicht spezifizieren dass ein Merkmal unterschiedliche Werte hat, den Klassen entsprechend, wo es angewandt wird.
- Man kann nicht Operationen zwischen Klassen definieren..

Praktische Aufgabe

- RDF Beschreibung der Pro-Seminar Web Seite
 - Identifizierung der Merkmale und wichtigen Informationen
 - Graf-Darstellung
 - Definition eines Vocabularys
 - RDF-Annotierung (man kann auch Dublin Core tags benutzen)
 - Definition eines RDF-Schemas
- Man soll nicht versuchen jede Satz sondern nur die wichtigen Informationen als RDF-Graf darzustellen