

Modellierung nachhaltiger Systeme und Semantic Web

RDF Basics

**Vorlesung im Modul 10-202-2330 im Master
und Lehramt Informatik sowie
im Modul 10-202-2309 im Master Informatik**

Prof. Dr. Hans-Gert Gräbe

<http://informatik.uni-leipzig.de/~graebe>

Beschreibungen und Interpretationen

- Information als interpretierte Daten?
 - Messwerte als Daten?
- Sprache ist voraussetzungsreich. Ein Beispiel:
 - Am 8.11. wurde an der Station Leipzig-Flughafen um 17 Uhr eine Temperatur von 16°C gemessen.
 - Am `8.11.` wurde an der *Station* `Leipzig-Flughafen` um `17 Uhr` eine `Temperatur` von `16°C` *gemessen*.
 - Dinge und deren Bezeichnungen.

RDF Basics (1)

Beispiel: Beschreibung unserer Lehrveranstaltungen

```
@prefix od: <http://od.fmi.uni-leipzig.de/model/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix odr: <http://od.fmi.uni-leipzig.de/rooms/> .
@prefix odp: <http://od.fmi.uni-leipzig.de/personal/> .

<http://od.fmi.uni-leipzig.de/w20/BIS.IADW.1>
  a od:Vorlesung ;
  od:beginsAt "11:15" ;
  od:dayOfWeek "donnerstags" ;
  od:endsAt "12:45" ;
  od:locatedAt odr:Hs_19 ;
  od:servedBy odp:Graebe_HansGert ;
  rdfs:label "Vorlesung ..." .
```

- Identifier und Literale. Namensräume.

RDF Basics (2)

Auflösung in Drei-Wort-Sätze

Subjekt Prädikat Objekt .

w20:BIS.IADW.1	a	od:Vorlesung .
w20:BIS.IADW.1	od:beginsAt	"11:15" .
w20:BIS.IADW.1	od:dayOfWeek	"donnerstags" .
w20:BIS.IADW.1	od:endsAt	"12:45" .
w20:BIS.IADW.1	od:locatedAt	odr:Hs_19 .
w20:BIS.IADW.1	od:servedBy	odp:Graebe_HansGert .
w20:BIS.IADW.1	rdfs:label	"Vorlesung ..." .

Mehr dazu siehe <http://od.fmi.uni-leipzig.de/>

Interpretationen im Internet der Dinge

- Industrie 4.0 und das Internet der Dinge (IoT – Internet of Things)
- Fiktion: Im Internet gibt es keine Dinge, sondern nur *Repräsentationen* von Dingen, genau wie Repräsentationen von Personen.
- Beschreibungen als relationale Komplexe zwischen Repräsentationen von realen Dingen oder auch nur Bedeutungskomplexen.
- Auch diesen Dingen und Bedeutungskomplexen müssen „digitale Identitäten“ als textuelle Repräsentationen zugeordnet werden, um über sie im digitalen Universum Sätze formulieren zu können.

RDF Basics (3)

Konzeptionelle „Zutaten“:

- UTF-8 als **einheitliche Zeichenbasis** für URIs und Literale.
 - Best Practise: URIs nur aus ASCII-Zeichen, keine Umlaute oder Ähnliches
- URI als „digitale Identitäten“ von Ressourcen, *zeigen* auf Ressourcen
 - Wie digitale Identitäten von Personen sind dies **textuelle Repräsentationen der „Dinge“** in den im Internet kursierenden Textfragmenten.
- Für Computer sind URIs einfach Zeichenketten, für Menschen ist es hilfreich, wenn die URI bereits eine Ahnung von der Semantik des Gezeigten vermittelt.
 - Best Practise: „sprechende Namen“ als URIs

RDF Basics (4)

- **RDF – Resource Description Framework**

- Konzept zum Aufschreiben von Geschichten über „die Welt“ als *Mengen* von Drei-Wort-Sätzen
 $\langle \text{Subjekt} \rangle \langle \text{Prädikat} \rangle \langle \text{Objekt} \rangle .$
- Subjekt und Prädikat müssen URIs sein, als Objekt kann eine URI oder ein Literal (Typ `rdfs:Literal`) stehen. Literale können Typ- und Sprachmarkierungen tragen.
- Es gibt verschiedene Notationen für dieselbe Menge von RDF-Sätzen (Turtle, rdf/xml, json, ntriples) und Werkzeuge, diese Notationen ineinander umzuwandeln.
 - Redland RDF libraries <http://librdf.org/>
- Mustersuche als mächtiges Konzept der Analyse derartiger Mengen. SPARQL als Anfragesprache.

RDF Basics (5)

Beispiel einer Anfrage an den SPARQL Endpunkt

<http://od.fmi.uni-leipzig.de:8892/sparql>

Einfache Anfrage nach allen LV mit URI-Präfix SWS.IADW

PREFIX od: <http://od.fmi.uni-leipzig.de/model/>

SELECT distinct ?l ?name ?d ?b ?r

from <http://od.fmi.uni-leipzig.de/w20/>

WHERE {

 ?l a od:LV .

 ?l rdfs:label ?name .

 ?l od:beginsAt ?b .

 ?l od:dayOfWeek ?d .

 ?l od:locatedAt ?r .

filter regex(?l,'BIS.IADW') .

}

RDF Basics (6)

- **Turtle-Notation** – fasst alle Sätze zum selben Subjekt zusammen. Menge von Prädikat-Objekt-Paaren kann als Menge von Schlüssel-Wert-Paaren (key – value) betrachtet werden, welche dieses Subjekt beschreiben.
 - Aber: ein Schlüssel kann mehrere Werte haben!
 - Besonders verbreitete von Menschen lesbare Notation.
 - Subjekt-zentrierte Sicht, die spezifische Sichtweisen auf „MEINE Welt“ – wie früher thematisiert – gut bedient.
 - Computer arbeiten lieber mit Tripel-Mengen.
- Interpretiert man Subjekte und Objekte als Knoten und Prädikate als Kanten eines Graphen, so beschreibt eine Menge von RDF-Sätzen einen **RDF-Graphen** (und umgekehrt).
 - Ein Bild sagt oft mehr als tausend Worte.

RDF – Sätze und Muster

Sätze folgen dabei Mustern:

1) Turtle: Sammle alle Sätze mit demselben Subjekt → Eigenschaften eines Individualsubjekts als Schlüssel-Wert-Paare.

- Schlüssel-Wert = Eigenschaft und Ausprägung

2) Sammle alle Sätze mit demselben Prädikat

A od:beginsAt B

- od:beginsAt nicht nur eine URI (*Syntax*), sondern auch ein zweistelliges Prädikat mit einer gewissen *Semantik*, die in allen Sätzen mit diesem Prädikat als *Instanziierungen des Prädikats* transportiert wird.

3) Andere Muster sind möglich, SPARQL als genereller Standard einer Anfragesprache für Mustersuche.

Siehe die Datei *Queries.txt* im Ordner des OLAT-Kurses.

RDF – Beschreibungen von Beschreibungen

- **Selbstähnlichkeit des Konzepts:** Auch Beschreibungen von Beschreibungen können als RDF-Sätze formuliert werden. Insbesondere kann man RDF verwenden, um RDF zu beschreiben.
 - Eine URI, die in einem Satz als Prädikat auftritt, kann in einem anderen Satz als Subjekt oder Objekt auftreten.
Beispiel:
 `od:beginsAt rdfs:domain od:LV .`
 `od:beginsAt rdfs:range rdfs:Literal .`
- Damit können auch *Begriffe und Konzepte* RDF-basiert beschrieben werden. → **Universalien**
 - Was sind Universalien? Ideen aus Platons Ideenhimmel oder institutionalisierte Konventionen, d.h. „Fiktionen“ im Sinne der Vorlesung?

RDF – Grundlegende Beschränkungen

- Mengensemantik, Reihenfolge der Sätze spielt keine Rolle
 - Dies ist in anderen Ansätzen, etwa dem XML-basierten TEI der Text Encoding Initiative anders, das in den Digital Humanities eine zentrale Rolle spielt.
- Problem der Kontextualisierung. In welchem raumzeitlichen Kontext ist der Satz gültig? Hier gibt es mehrere Ansätze:
 - Aufbohren zu Quadrupeln mit vierter Komponente als URI auf die Provenienz.
 - Wird der Satz als Instanziierung eines zweistelligen Prädikats interpretiert, so ergibt sich der Kontext oft implizit aus dem Prädikat.
 - Oft ergibt sich der Kontext allgemeiner aus dem Namensraum des Prädikats und steht damit als (explizites oder implizites) Modell für eine ganze Klasse von Begrifflichkeiten. Damit verlagert sich die Frage aber nur in die Beschreibung des Modells und damit eine Abstraktionsebene nach oben.

RDF – Zentrale Konzepte

Zusammenfassung der zentralen Konzepte

- Zentrale Idee: Speichere textuelle Beschreibungen auf uniforme Weise als Tripel und verwende Standardkonzepte und -werkzeuge zur Verwaltung dieser Daten
- *Ressourcen*: URI, HTTP access
 - URI = Unique Resource Identifier
 - Damit kann auf einen weltweit verteilten Datenbestand auf uniforme Weise über ein gängiges Protokoll zugegriffen werden.
- *Resource Descriptions*: Gib auf Anfrage an die HTTP-Adresse ein nützliches Stück Information im RDF-Format zurück, das mit anderen solchen Informationseinheiten zu neuen RDF-Sätzen kombiniert werden kann.
- Betreibe *RDF Triple Stores* als Teil einer weltweiten verteilten Datenspeicher-infrastruktur, etwa <http://od.fmi.uni-leipzig.de/>
- (Verteilte) Anfragesprache SPARQL
 - Stelle *SPARQL Endpunkte* auf RDF Triple Stores bereit

RDF – Sprachformen und Praxen

Verfahrenswissen → Verfahrensweisen

- Korrespondenz zwischen der Kohärenz der Sprachform und der Kohärenz der Praxen
- Die Etablierung kohärenter Praxen als Verfahrensweisen erlaubt es, Prädikate zu substantivieren. „Zum Beginn der Schulstunde werden die Pausenbrote weggepackt“. Erst nach einer solchen Transposition des Prädikats an die Subjekt-Position werden normative Sätze möglich.
 - Parallelen zum Konzertbeispiel in der ersten Vorlesung.
- Grundweisheiten der Informatik:
 1. Eine Funktion kann erst aufgerufen werden, nachdem sie definiert wurde.
 2. Eine Funktion, die definiert wird, aber nicht aufgerufen, deutet auf einen Designfehler hin.
- *Verfahrenswissen* ist die Beschreibungsform, *Verfahrensweise* die Ausführungsform.