



Vorlesung

Dr. Harald Sack

Hasso-Plattner-Institut für Softwaresystemtechnik

Universität Potsdam

Wintersemester 2008/09



<http://sw0809.blogspot.com/>

Blog zur Vorlesung: <http://sw0809.blogspot.com/>

Semantic Web - Vorlesungsinhalt

2

1. Einführung
2. Die Sprachen des Semantic Web
3. **Wissensrepräsentation**
4. Web of Trust
5. Ontology Engineering
6. Semantic Web Anwendungen

1

2

3

4

5

6

7

8.01.2009 – Vorlesung Nr. 8

9

10

11

12

13

3. Wissensrepräsentationen

3.0 Motivation

3.1 Ontologien in der Philosophie

3.2 Ontologien in der Informatik

3.3 Ontologie Beschreibungssprachen

3.4 Ontologietypen

3.5 Wiederholung Aussagenlogik und Prädikatenlogik

3.6 Semantik von RDF(S)

3.7 Beschreibungslogiken und Web Ontology Language OWL

3.8 Regeln mit SWRL / RIF

3.9 Logikbasierte Systeme

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)

4

3.6 Semantik von RDF(S)

3.6.1 Warum eine Semantik für RDF(S)?

3.6.2 Modelltheoretische Semantik für RDF(S)

3.6.3 Einfache Interpretationen

3.6.4 RDF Interpretationen

3.6.5 RDFS Interpretationen

3.6.6 RDF(S) Folgerungen

3.6.7 Semantische Grenzen von RDF(S)



Warum ist eine formale Semantik für RDF(S) notwendig?

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.1 Warum eine Semantik?

6

Warum eine Semantik für RDF(S)?

- RDF(S) Spezifikation beinhaltet keine **formale Definition** der RDF(S)-Semantik
- Tools-Hersteller beklagten Inkompatibilitäten
 - insbesondere bei Triple-Stores, z.B.
 - gleiche Anfrage an verschiedene Triple-Stores liefert unterschiedliche Ergebnisse
 - Grund:
unterschiedliche Interpretation von RDF-Dokumenten bzw. RDF-Anfragen
- Daher:
Definition einer formalen Semantik notwendig

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.1 Warum eine Semantik?



7

Welche Voraussetzungen benötigen wir?

- Mathematische Logik dient der Formalisierung des korrekten Ziehens von Schlussfolgerungen

- Notwendige Voraussetzung:
 - Menge von Aussagen über die Schlussfolgerungen gezogen werden können (=Sätze S)

 - Schlussfolgerungsrelation (entailment relation) $\models \subseteq 2^S \times S$
 - um Schlussfolgerungen zu ziehen, wie z.B. $\{s_1, s_2, s_3\} \models s$

 - Logik $L = (S, \models)$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)

8

3.6 Semantik von RDF(S)

3.6.1 Warum eine Semantik für RDF(S)?

3.6.2 Modelltheoretische Semantik für RDF(S)

3.6.3 Einfache Interpretationen

3.6.4 RDF Interpretationen

3.6.5 RDFS Interpretationen

3.6.6 RDF(S) Folgerungen

3.6.7 Semantische Grenzen von RDF(S)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.2 Modelltheoretische Semantik

9

Modelltheoretische Semantik

- Grundidee:
Aussagen einer Logik mit Interpretationen ins Verhältnis setzen

- Interpretation (Δ^I, I)
 - Δ^I ... Domain of Discourse, $\Delta^I \neq \emptyset$
 - Interpretationsfunktion I
 - $I : A \rightarrow A^I \subseteq \Delta^I$, A ... atomares Konzept
 - $I : R \rightarrow R^I \subseteq \Delta^I \times \Delta^I$, R ... atomare Relation

- Definition von Kriterien zur Entscheidung, ob eine konkrete Interpretation I einen Satz $s \in S$ „erfüllt“ (Modellrelation)
 - I ist Modell von s , $I \models s$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.2 Modelltheoretische Semantik



10

Modelltheoretische Semantik

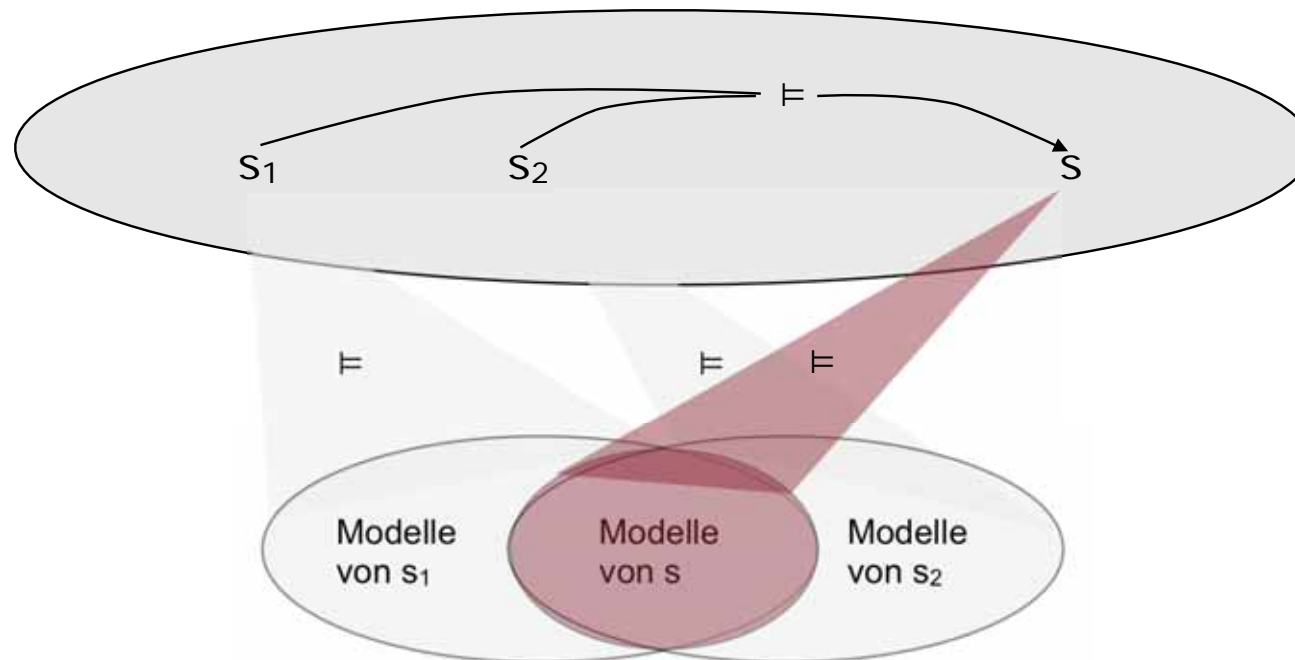
- Definition der Schlussfolgerungsrelation \models :
 - Ein Satz $s \in S$ folgt aus einer Menge von Sätzen $S \subseteq S$ (d.h. $S \models s$) genau dann, wenn jede Interpretation I , die jeden Satz $s' \in S$ erfüllt (also $I \models s'$, für alle $s' \in S$), auch ein Modell von s ist (also $I \models s$)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.2 Modelltheoretische Semantik

11

Modelltheoretische Semantik



3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.2 Modelltheoretische Semantik



12

Modelltheoretische Semantik für RDF(S)

- Was sind die Sätze (Aussagen) in RDF(S)?
 - jedes Tripel (s,p,o) ist ein Satz
 - Tripel werden beschrieben mit Grundvokabular V
 - URIs, bnodes und Literale
- $(s,p,o) \in (\text{URI} \cup \text{bnode}) \times \text{URI} \times (\text{URI} \cup \text{bnode} \cup \text{Literal})$
- Ein (RDF-)Graph ist eine endliche Menge von Tripeln
- Jeder (RDF-)Graph ist ein Satz

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.2 Modelltheoretische Semantik



13

Modelltheoretische Semantik für RDF(S)

- Schlussfolgerungsrelation \models
 - \models gibt an, wann ein RDF(S)-Graph G' aus einem RDF(S)-Graphen G folgt,
 - d.h. $G \models G'$
- Zur Definition einer modelltheoretischen Semantik für RDF(S) definieren wir eine Menge von **Interpretationen** und legen fest, wann eine Interpretation **Modell** eines Graphen ist

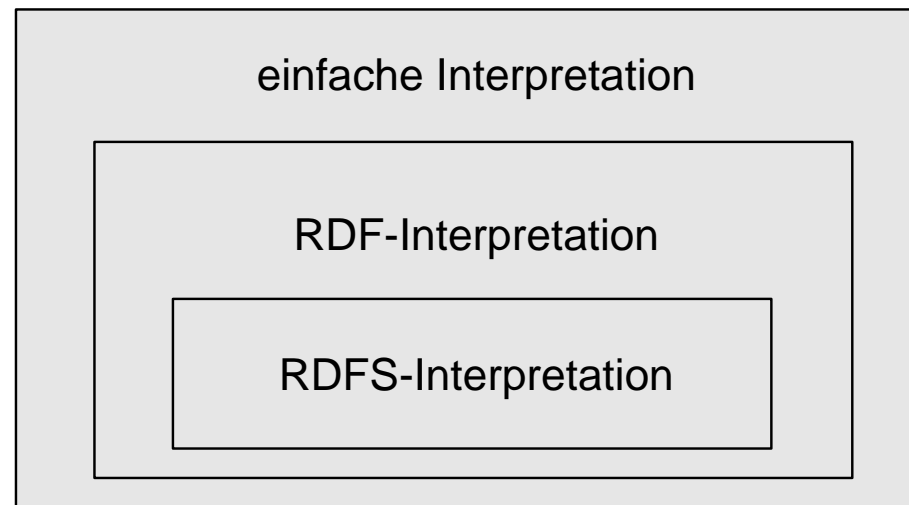
3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.2 Modelltheoretische Semantik

14

Modelltheoretische Semantik für RDF

- schrittweises Vorgehen bei der Definition



- Ziel: formal korrekte Abbildung der Intuition hinter RDF(S)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)

15

3.6 Semantik von RDF(S)

3.6.1 Warum eine Semantik für RDF(S)?

3.6.2 Modelltheoretische Semantik für RDF(S)

3.6.3 Einfache Interpretationen

3.6.4 RDF Interpretationen

3.6.5 RDFS Interpretationen

3.6.6 RDF(S) Folgerungen

3.6.7 Semantische Grenzen von RDF(S)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

16

Eine **einfache Interpretation I** für ein Vokabular V besteht aus

- **IR**, einer nichtleeren Menge von Ressourcen, auch Domäne oder (Diskurs-)Universum von I ,
- **IP**, der Menge der Properties von I ,
- **I_{EXT}**, einer Funktion, die jedem Property eine Menge von Paaren aus IR zuordnet, d.h. $I_{EXT}: IP \rightarrow 2^{IR \times IR}$, dabei nennt man $I_{EXT}(p)$ auch die Extension des Property p ,
- **I_S**, einer Funktion, die URIs aus V in die Vereinigung der Mengen IR und IP abbildet, d.h. $I_S: V \rightarrow IR \cup IP$,
- **I_L**, einer Funktion von (getypten) Literalen aus V in die Menge IR der Ressourcen, d.h. $I_L: V \rightarrow IR$ und
- **LV** $\subseteq IR$, die Menge der Literalwerte, die (mindestens) alle ungetypten Literale aus V enthält

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

17

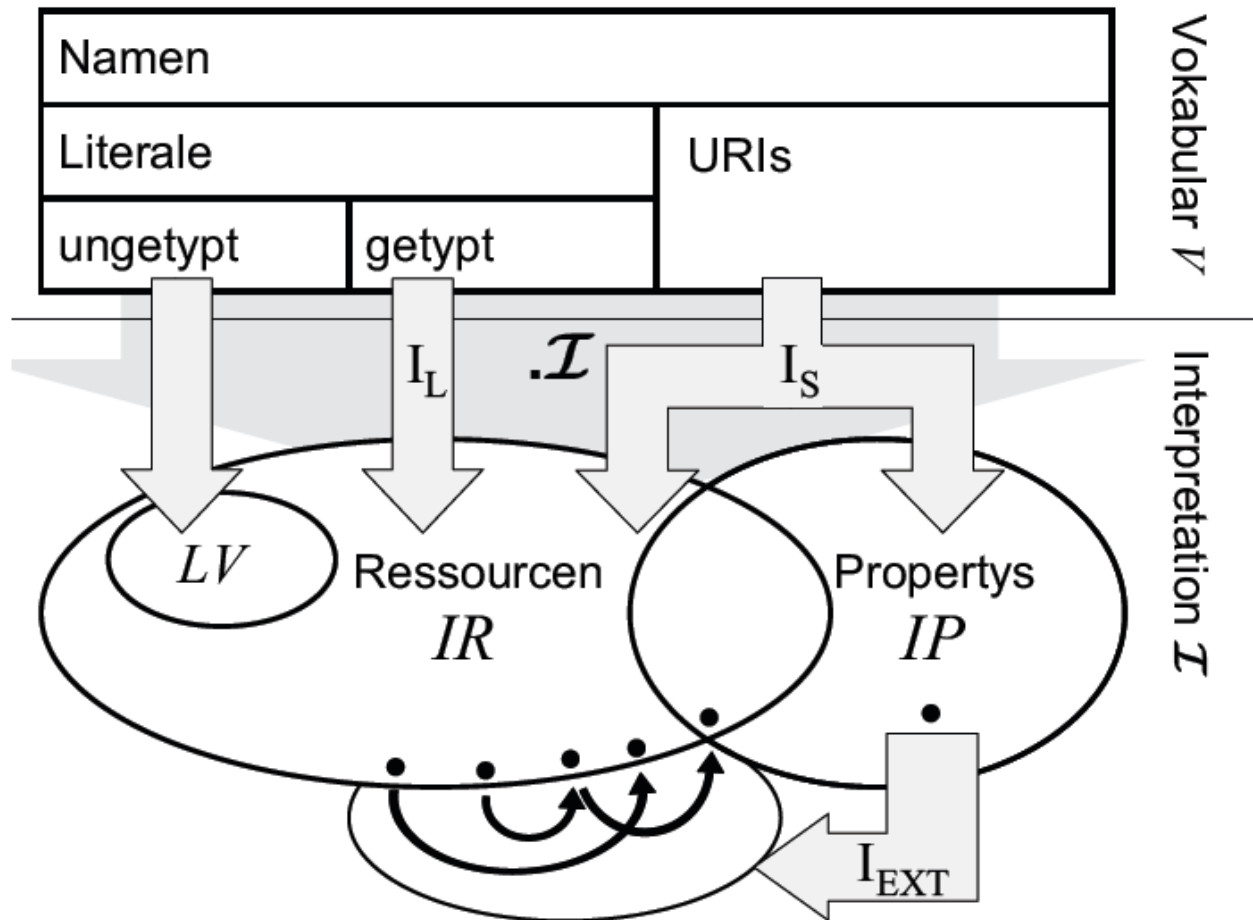
- Wir definieren eine einfache Interpretationsfunktion I' , die alle im Vokabular V enthaltenen Literale und URIs auf Ressourcen und Properties abbildet.
 - jedes ungetypte Literal "a" wird auf a abgebildet: $(\text{"a"})^{I'} = a$
 - jedes ungetypte Literal mit Sprachangabe "a"@t wird auf das Paar $\langle a, t \rangle$ abgebildet: $(\text{"a@t"})^{I'} = \langle a, t \rangle$
 - jedes getypte Literal l wird auf $I_L(l)$ abgebildet: $l^{I'} = I_L(l)$
 - jede URI u wird auf $I_S(u)$ abgebildet: $u^{I'} = I_S(u)$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

18

Einfache Interpretation (schematisch)

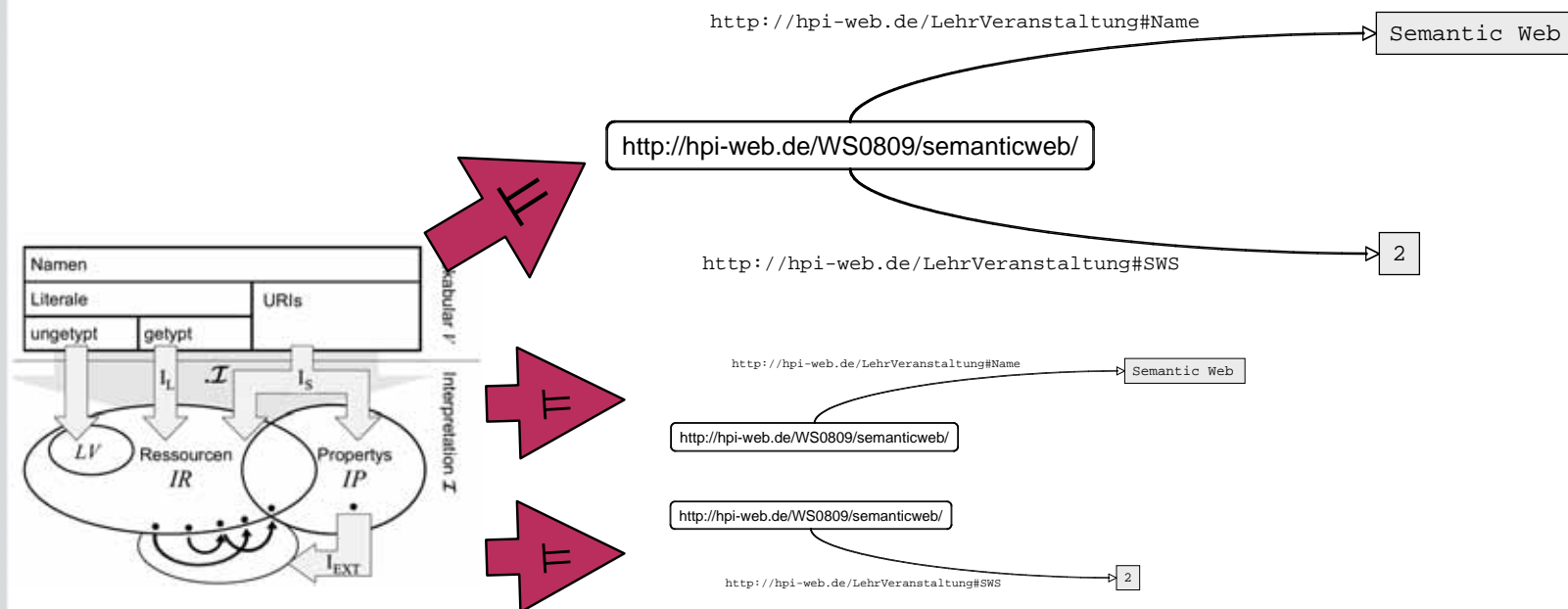


3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

19

Wann ist eine Interpretation Modell eines Graphen?



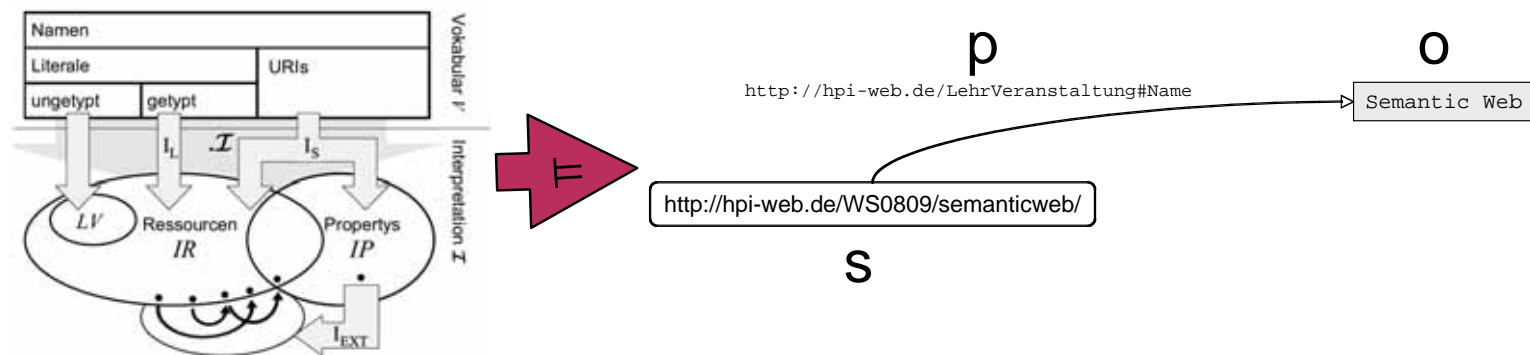
...wenn die Interpretation ein Modell für **jedes** Tripel des Graphen ist

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

20

Wann ist eine Interpretation Modell eines Tripels?



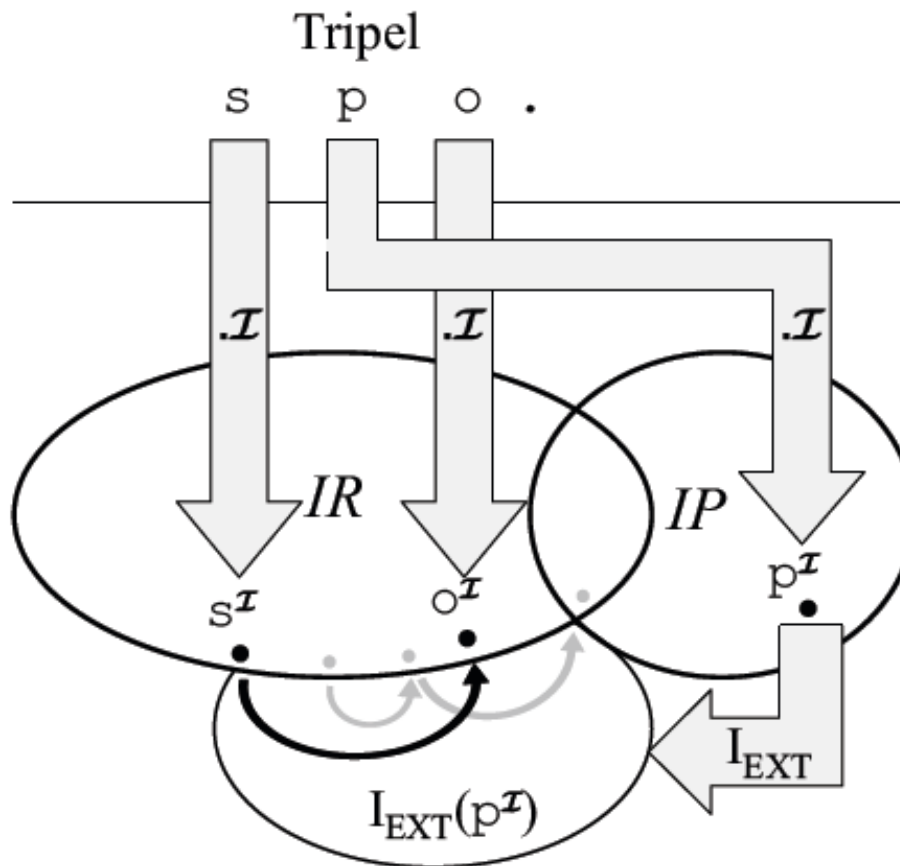
- genau dann, wenn $s, p, o \in V$ und $\langle s', o' \rangle \in I_{EXT}(p')$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

21

Einfache Interpretation eines Tripels (Schematisch)



Die Interpretation \cdot^I ordnet dem Graph G einen Wahrheitswert zu

$G^I = \text{wahr}$ gdw. $T^I = \text{wahr}$ für alle $T \in G$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

22

Berücksichtigung von Blank Nodes

- sei A Funktion, die alle blank nodes auf Elemente von IR abbildet
- für eine Interpretation I , sei $I+A$ wie I , wobei zusätzlich für jeden blank node b gilt
 - $b^{I+A} = A(b)$
- eine Interpretation I ist jetzt Modell eines RDF-Graphen G , wenn es ein A gibt, so dass alle Tripel bezüglich $I+A$ wahr werden

- **Fazit:**

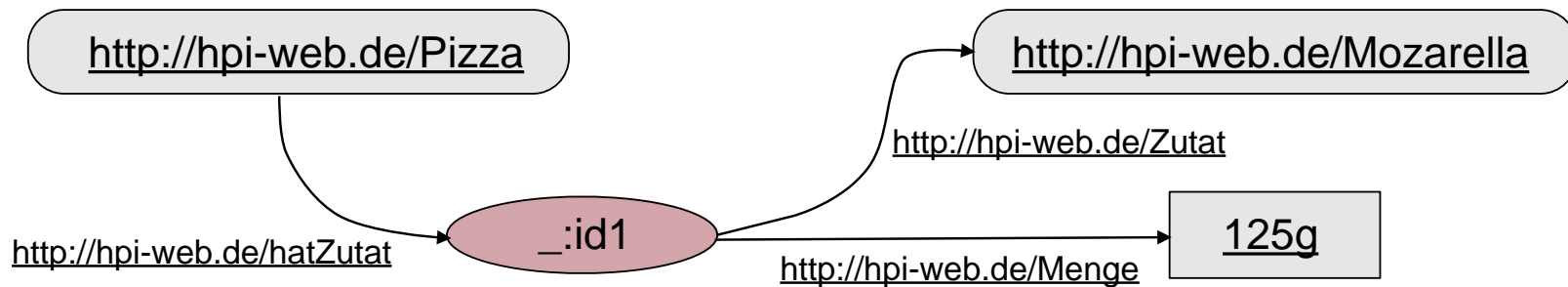
Ein Graph G_2 *folgt einfach* aus einem Graph G_1 , wenn jede einfache Interpretation, die Modell von G_1 ist auch Modell von G_2 ist.

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

23

Beispiel



$IR = \{\chi, \upsilon, \tau, \nu, \varepsilon, \iota, 125g\}$

$IP = \{\tau, \nu, \iota\}$

$LV = \{125g\}$

$I_{EXT} = \tau \rightarrow \{\langle \chi, \varepsilon \rangle\}$

$\nu \rightarrow \{\langle \varepsilon, \upsilon \rangle\}$

$\iota \rightarrow \{\langle \varepsilon, 125g \rangle\}$

$I_S = \text{hpi:Pizza} \rightarrow \chi$

$\text{hpi:Mozarella} \rightarrow \upsilon$

$\text{hpi:hatZutat} \rightarrow \tau$

$\text{hpi:Zutat} \rightarrow \nu$

$\text{hpi:Menge} \rightarrow \iota$

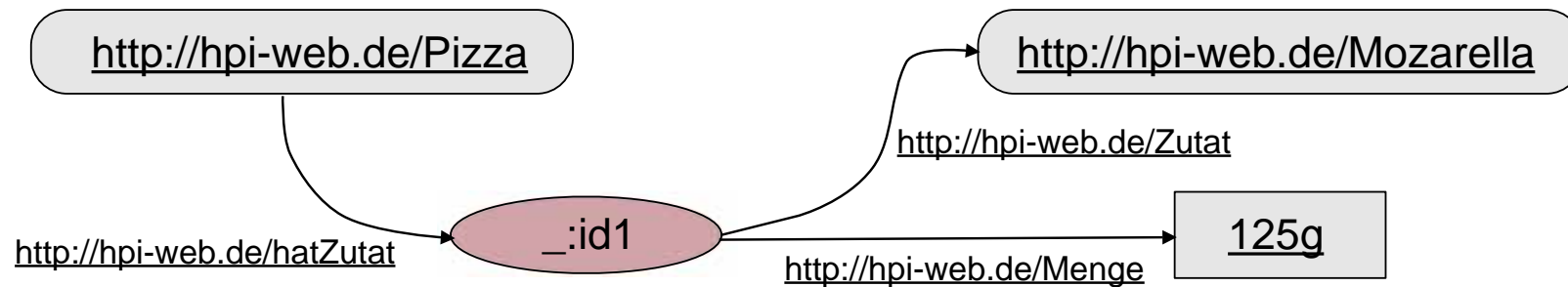
$I_L = \text{leer, da keine getypten Literale}$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.3 einfache Interpretation

24

Beispiel



Wählt man A: `_:id1` $\rightarrow \epsilon$ dann ergibt sich

$$\langle \text{hpi:Pizza}^{I+A}, _:\text{id1}^{I+A} \rangle = \langle \chi, \epsilon \rangle \in I_{\text{EXT}}(\tau) = I_{\text{EXT}}(\text{hpi:hatZutat}^{I+A})$$

$$\langle _:\text{id1}^{I+A}, \text{hpi:Mozarella}^{I+A} \rangle = \langle \epsilon, \nu \rangle \in I_{\text{EXT}}(\nu) = I_{\text{EXT}}(\text{hpi:Zutat}^{I+A})$$

$$\langle _:\text{id1}^{I+A}, \text{"125g"}^{I+A} \rangle = \langle \epsilon, 125\text{g} \rangle \in I_{\text{EXT}}(\iota) = I_{\text{EXT}}(\text{hpi:Menge}^{I+A})$$

Also wird auch der beschriebene Graph als Ganzes wahr.
I ist ein Modell des Graphen (bzgl. der einfachen Interpretation)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)

25

3.6 Semantik von RDF(S)

3.6.1 Warum eine Semantik für RDF(S)?

3.6.2 Modelltheoretische Semantik für RDF(S)

3.6.3 Einfache Interpretationen

3.6.4 RDF Interpretationen

3.6.5 RDFS Interpretationen

3.6.6 RDF(S) Folgerungen

3.6.7 Semantische Grenzen von RDF(S)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.4 RDF Interpretation

26

RDF-Interpretationen

- Einfache Interpretationen behandeln alle URIs gleich
- Zur korrekten Behandlung des **RDF-Vokabulars** müssen zusätzliche Anforderungen an die Menge der zulässigen Interpretationen gestellt werden
- RDF Vokabular V_{RDF} :

```
rdf:type rdf:Property rdf:XMLLiteral rdf:nil  
rdf:List rdf:Statement rdf:subject rdf:predicate rdf:object  
rdf:first rdf:rest red:Seq rdf:Bag rdf:Alt  
rfd:_1 rfd:_2 ...
```

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.4 RDF Interpretation

27

Semantik des RDF Vokabulars

- `rdf:type`
 - weist einer URI einen Typ zu
 - Klassenzugehörigkeit der durch den URI bezeichneten Ressource
- `rdf:Property`
 - bezeichnet einen bestimmten Typ von Ressource
 - charakterisiert alle URIs, die in Tripeln als Prädikat (Property) vorkommen
- `rdf:XMLLiteral`
 - vordefinierter Datentyp (XML-Fragment)
 - unterscheidet wohlgeformte / nicht-wohlgeformte Literale

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.4 RDF Interpretation

28

- Eine **RDF-Interpretation** für ein Vokabular V ist nun eine einfache Interpretation für das Vokabular $V \cup V_{\text{RDF}}$, die zusätzlich folgende Bedingungen erfüllt:

(1) $x \in IP$ genau dann, wenn $\langle x, \text{rdf:Property}' \rangle \in I_{\text{EXT}}(\text{rdf:type}')$

- x ist eine Property genau dann, wenn es mit der durch `rdf:Property` bezeichneten Ressource über die `rdf:type`-Property verbunden ist
- (dies führt automatisch dazu, dass für jede RDF-Interpretation $IP \subseteq IR$ gilt).

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.4 RDF Interpretation

29

(2) wenn $"s" \wedge \wedge \text{rdf:XMLLiteral}$ in V enthalten und s ein **wohlgeformtes XML-Literal** ist, dann

- $I_L("s" \wedge \wedge \text{rdf:XMLLiteral})$ ist der XML-Wert von s
- $I_L("s" \wedge \wedge \text{rdf:XMLLiteral}) \in LV$
- $\langle I_L("s" \wedge \wedge \text{rdf:XMLLiteral}), \text{rdf:XMLLiteral} \rangle \in I_{\text{EXT}}(\text{rdf:type})$

(3) wenn $"s" \wedge \wedge \text{rdf:XMLLiteral}$ in V enthalten und s **kein wohlgeformtes XML-Literal** ist, dann

- $I_L("s" \wedge \wedge \text{rdf:XMLLiteral}) \notin LV$
- $\langle I_L("s" \wedge \wedge \text{rdf:XMLLiteral}), \text{rdf:XMLLiteral} \rangle \notin I_{\text{EXT}}(\text{rdf:type})$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.4 RDF Interpretation

30

- Zusätzliche Forderung für RDF-Interpretation:
sämtliche folgenden „axiomatischen“ Tripel sind wahr

rdf:type	rdf:type	rdf:Property .
rdf:subject	rdf:type	rdf:Property .
rdf:predicate	rdf:type	rdf:Property .
rdf:object	rdf:type	rdf:Property .
rdf:first	rdf:type	rdf:Property .
rdf:rest	rdf:type	rdf:Property .
rdf:value	rdf:type	rdf:Property .
rdf:_1	rdf:type	rdf:Property .
rdf:_2	rdf:type	rdf:Property .
...
rdf:nil	rdf:type	rdf:List .

- ein Graph G_2 **RDF-folgt** aus einem Graph G_1 ,
wenn jede RDF-Interpretation, die Modell von G_1 ist, auch Modell
von G_2 ist

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)

31

3.6 Semantik von RDF(S)

3.6.1 Warum eine Semantik für RDF(S)?

3.6.2 Modelltheoretische Semantik für RDF(S)

3.6.3 Einfache Interpretationen

3.6.4 RDF Interpretationen

3.6.5 RDFS Interpretationen

3.6.6 RDF(S) Folgerungen

3.6.7 Semantische Grenzen von RDF(S)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.5 RDFS Interpretation

32

RDFS-Interpretationen

- Zur korrekten Behandlung des **RDFS-Vokabulars** müssen zusätzliche Anforderungen an die Menge der zulässigen RDF-Interpretationen gestellt werden
- RDF Vokabular V_{RDFS} :

`rdfs:domain rdfs:range rdfs:Resource rdfs:Literal rdfs:Datatype
rdfs:Class rdfs:subClassOf rdfs:subPropertyOf rdfs:member
rdfs:Container rdfs:ContainerMembershipProperty
rdfs:comment rdfs:seeAlso rdfs:isDefinedBy rdfs:label`

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.5 RDFS Interpretation

33

- Zur einfacheren Darstellung:
 - Klassenextensionsfunktion $I_{\text{CEXT}}: IR \rightarrow 2^{IR}$
 - $I_{\text{CEXT}}(y)$ enthalte genau diejenigen Elemente x , für die $\langle x, y \rangle \in I_{\text{EXT}}(\text{rdf:type})$
 - $IC = I_{\text{CEXT}}(\text{rdfs:Class})$
IC ist Extension der speziellen URI `rdfs:Class`

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.5 RDFS Interpretation

34

- Eine **RDFS-Interpretation** für ein Vokabular V ist nun eine RDF-Interpretation für das Vokabular $V \cup V_{\text{RDFS}}$, die zusätzlich folgende Bedingungen erfüllt:

(1) $IR = I_{\text{EXT}}(\text{rdfs:Resource})$
jede Ressource ist vom Typ `rdfs:Resource`

(2) $LV = I_{\text{EXT}}(\text{rdfs:Literal})$
jedes ungetypte und jedes wohlgeformte getypte Literal ist vom Typ `rdfs:Literal`

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.5 RDFS Interpretation

35

- Eine **RDFS-Interpretation** für ein Vokabular V ist nun eine RDF-Interpretation für das Vokabular $V \cup V_{\text{RDFS}}$, die zusätzlich folgende Bedingungen erfüllt:

(3) Wenn $\langle x, y \rangle \in I_{\text{EXT}}(\text{rdfs:domain})$ und $\langle u, v \rangle \in I_{\text{EXT}}(x)$,
dann ist $u \in I_{\text{CEXT}}(y)$

(4) Wenn $\langle x, y \rangle \in I_{\text{EXT}}(\text{rdfs:range})$ und $\langle u, v \rangle \in I_{\text{EXT}}(x)$,
dann ist $v \in I_{\text{CEXT}}(y)$

Ist x und y durch Property $\text{rdfs:domain}/\text{rdfs:range}$
verbunden und verbindet das Property x die Ressourcen u
und v , dann ist u/v vom Typ y

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.5 RDFS Interpretation

36

- Eine **RDFS-Interpretation** für ein Vokabular V ist nun eine RDF-Interpretation für das Vokabular $V \cup V_{\text{RDFS}}$, die zusätzlich folgende Bedingungen erfüllt:

- (5) $I_{\text{EXT}}(\text{rdfs:subPropertyOf})$ ist reflexiv und transitiv auf IP
- (6) Wenn $\langle x, y \rangle \in I_{\text{EXT}}(\text{rdfs:subPropertyOf})$,
dann $x, y \in IP$ und $I_{\text{EXT}}(x) \subseteq I_{\text{EXT}}(y)$
- (7) Wenn $x \in IC$,
dann $\langle x, \text{rdfs:Resource} \rangle \in I_{\text{EXT}}(\text{rdfs:subClassOf})$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.5 RDFS Interpretation

37

- Eine **RDFS-Interpretation** für ein Vokabular V ist nun eine RDF-Interpretation für das Vokabular $V \cup V_{\text{RDFS}}$, die zusätzlich folgende Bedingungen erfüllt:

(8) Wenn $\langle x, y \rangle \in I_{\text{EXT}}(\text{rdfs:subClassOf}')$,
dann $x, y \in IC$ und $I_{\text{CEXT}}(x) \subseteq I_{\text{CEXT}}(y)$

(9) $I_{\text{EXT}}(\text{rdfs:subClassOf}')$ ist reflexiv und transitiv
auf IC

(10) Wenn $x \in I_{\text{CEXT}}(\text{rdfs:ContainerMembershipProperty}')$,
dann $\langle x, \text{rdfs:member}' \rangle \in I_{\text{EXT}}(\text{rdfs:subPropertyOf}')$

(11) Wenn $x \in I_{\text{CEXT}}(\text{rdfs:Datatype}')$,
dann $\langle x, \text{rdfs:Literal}' \rangle \in I_{\text{EXT}}(\text{rdfs:subClassOf}')$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.5 RDFS Interpretation

38

Dazu kommen noch zahlreiche axiomatische Tripel

```

rdfs:type          rdfs:domain      rdfs:Resource .
rdfs:domain       rdfs:domain      rdf:Property .
rdfs:range        rdfs:domain      rdf:Property .
rdfs:subPropertyOf rdfs:domain      rdf:Property .
rdfs:subClassOf   rdfs:domain      rdfs:Class .
rdf:subject       rdfs:domain      rdf:Statement .
rdf:predicate     rdfs:domain      rdf:Statement .
rdf:object        rdfs:domain      rdf:Statement .
rdfs:member       rdfs:domain      rdfs:Resource .
rdf:first         rdfs:domain      rdf:List .
rdf:rest          rdfs:domain      rdf:List .
rdfs:seeAlso      rdfs:domain      rdfs:Resource .
rdfs:isDefinedBy  rdfs:domain      rdfs:Resource .
rdfs:comment      rdfs:domain      rdfs:Resource .
rdfs:label        rdfs:domain      rdfs:Resource .
rdf:value         rdfs:domain      rdfs:Resource .

rdfs:ContainerMembershipProperty
rdfs:subClassOf   rdfs:Property .
rdfs:subClassOf   rdfs:Container .
rdfs:subClassOf   rdfs:Container .
rdfs:isDefinedBy  rdfs:subPropertyOf rdfs:seeAlso .

rdf:XMLLiteral   rdf:type          rdfs:Datatype .
rdf:XMLLiteral   rdfs:subClassOf  rdfs:Literal .
rdfs:Datatype    rdfs:subClassOf  rdfs:Class .

rdf:_1           rdf:type          rdfs:ContainerMembershipProperty .
rdf:_1           rdfs:domain      rdfs:Resource .
rdf:_1           rdfs:range       rdfs:Resource .
rdf:_2           rdf:type          rdfs:ContainerMembershipProperty .
rdf:_2           rdfs:domain      rdfs:Resource .
rdf:_2           rdfs:range       rdfs:Resource .
...

rdf:type          rdfs:range      rdfs:Class .
rdfs:domain       rdfs:range      rdfs:Class .
rdfs:range        rdfs:range      rdfs:Class .
rdfs:subPropertyOf rdfs:range      rdf:Property .
rdfs:subClassOf   rdfs:range      rdfs:Class .
rdf:subject       rdfs:range      rdfs:Resource .
rdf:predicate     rdfs:range      rdfs:Resource .
rdf:object        rdfs:range      rdfs:Resource .
rdfs:member       rdfs:range      rdfs:Resource .
rdf:first         rdfs:range      rdfs:Resource .
rdf:rest          rdfs:range      rdf:List .
rdfs:seeAlso      rdfs:range      rdfs:Resource .
rdfs:isDefinedBy  rdfs:range      rdfs:Resource .
rdfs:comment      rdfs:range      rdfs:Literal .
rdfs:label        rdfs:range      rdfs:Literal .
rdf:value         rdfs:range      rdfs:Resource .

```

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.5 RDFS Interpretation

39

- ein Graph G_2 **RDFS-folgt** aus einem Graph G_1 , wenn jede RDFS-Interpretation, die Modell von G_1 ist, auch Modell von G_2 ist

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)

40

3.6 Semantik von RDF(S)

3.6.1 Warum eine Semantik für RDF(S)?

3.6.2 Modelltheoretische Semantik für RDF(S)

3.6.3 Einfache Interpretationen

3.6.4 RDF Interpretationen

3.6.5 RDFS Interpretationen

3.6.6 RDF(S) Folgerungen

3.6.7 Semantische Grenzen von RDF(S)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

41

Syntaktisches Schlussfolgern mit Ableitungsregeln

- Modelltheoretische Semantik beschreibt das Verhalten einer Logik bzgl. korrekter Schlussfolgerungen, ist aber für direkte algorithmische Verwendung wenig geeignet
- Um mit Hilfe der Modelltheoretischen Semantik zu zeigen, dass $G_1 \models G_2$, müssten ALLE (RDFS)-Interpretationen betrachtet werden
- Daher versucht man **Verfahren** zu entwickeln, **die die Gültigkeit von Schlussfolgerungen syntaktisch entscheiden können**
(Verfahren arbeiten nur auf den Sätzen der Logik, ohne auf die Interpretation zurückzugreifen)
- Beweis der Korrektheit notwendig (!), d.h. operationale Semantik (=Resultate des algorithmischen Verfahrens) stimmt mit Modelltheoretischer Semantik überein

Vorlesung Semantic Web, Dr. Harald Sack, Hasso-Plattner-Institut, Universität Potsdam

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

42

Syntaktisches Schlussfolgern mit Ableitungsregeln

- Allgemeine Form von Ableitungsregeln (Deduktionsregeln):

$$\frac{S_1 \dots S_n}{S}$$

- Sind Sätze s_1, \dots, s_n in der Menge der bekannten gültigen Aussagen enthalten, dann kann auch der Satz s dieser Menge hinzugefügt werden
- Die Gesamtheit aller für eine Logik gegebenen Ableitungsregeln nennt man **Deduktionskalkül**

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen



43

Allgemeine Notation für RDF(S)-Ableitungsregeln

- a und b stehen für beliebige URIs, die in einem Tripel an der Stelle des **Prädikats** stehen können
- $_:n$ steht für die ID eines beliebigen **Blank Nodes**
- u und v stehen für beliebige URIs oder IDs von Blank Nodes, die in einem Tripel an der Stelle des **Subjekts** stehen können
- l steht für ein beliebiges **Literal**
- x und y stehen für beliebige URIs oder IDs von Blank Nodes, die in einem Tripel an der Stelle des **Objekts** stehen können

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

44

Ableitungsregeln für einfache Folgerung

- alle URIs werden gleich behandelt

$$\frac{u \ a \ x \ .}{u \ a \ _ : n \ .} \quad \text{se1}$$

$$\frac{u \ a \ x \ .}{_ : n \ a \ x \ .} \quad \text{se2}$$

- **Satz:**
Ein Graph G_2 **folgt einfach** aus einem Graph G_1 , wenn G_1 mit Hilfe der Regeln se1 und se2 zu einem Graphen G_1' ergänzt werden kann, so dass G_2 in G_1' enthalten ist.

3. Wissensrepräsentationen

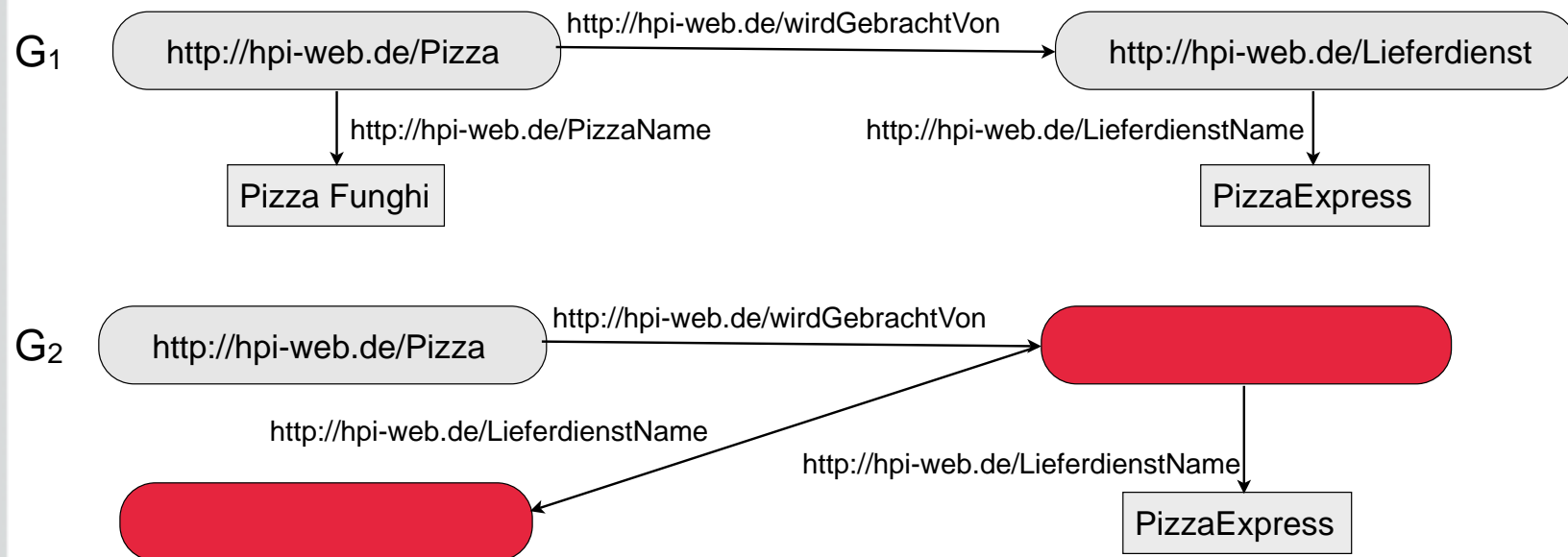
3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

45

- **Satz:**

Ein Graph G_2 folgt einfach aus einem Graph G_1 , wenn G_1 mit Hilfe der Regeln $se1$ und $se2$ zu einem Graphen G_1' ergänzt werden kann, so dass G_2 in G_1' enthalten ist.

- **Beispiel**



3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

46

Ableitungsregeln für RDF-Folgerung

$\frac{}{u \ a \ x .}$	rdfax	jedes axiomatische Tripel „u a x .“ kann immer abgeleitet werden
$\frac{u \ a \ l .}{u \ a \ _ : n .}$	lg	Literale dürfen durch nicht anderweitig gebundene bnodes ersetzt werden
$\frac{u \ a \ y .}{a \ \text{rdf:type} \ \text{rdf:Property}}$	rdf1	für jedes Tripelprädikat kann abgeleitet werden dass es eine Entität aus der Klasse der Properties ist
$\frac{u \ a \ l .}{_ : n \ \text{rdf:type} \ \text{rdf:XMLLiteral}}$	rdf2	wobei $_ : n$ dem wohlgeformten XML-Literal l durch lg zugewiesen wurde

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

47

Ableitungsregeln für RDF-Folgerung

- **Satz:**
Ein Graph G_2 **RDF-folgt** aus einem Graph G_1 genau dann, wenn es einen Graphen G_1' gibt, der aus G_1 mit Hilfe der Regeln rdfax, lg, rdf1 und rdf2 hergeleitet werden kann, aus dem G_2 einfach folgt.

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

48

Ableitungsregeln für RDFS-Folgerung

$$\frac{}{u \text{ a } x .}$$
 rdfsax jedes axiomatische Tripel „u a x .“ kann immer abgeleitet werden

$$\frac{u \text{ a } _ : n .}{u \text{ a } l .}$$
 gl wobei $_ : n$ ein Blank Node ist, der durch vormalige Anwendung von lg entstanden ist

$$\frac{u \text{ a } l .}{_ : n \text{ rdf: type rdfs: Literal}}$$
 rdfs1 wobei l ein ungetyptes Literal darstellt, und $_ : n$ einen durch Anwendung von lg dem Literal l zugewiesenen Blank Node

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

49

Ableitungsregeln für RDFS-Folgerung

- Property Einschränkungen

$$\frac{a \text{ rdfs:domain } x . \quad u \text{ a } y .}{u \text{ rdf:type } x .} \quad \text{rdfs2}$$

$$\frac{a \text{ rdfs:range } x . \quad u \text{ a } y .}{y \text{ rdf:type } x .} \quad \text{rdfs3}$$

- Alles ist eine Ressource

$$\frac{u \text{ a } x .}{u \text{ rdf:type rdfs:Resource .}} \quad \text{rdfs4a}$$

$$\frac{u \text{ a } x .}{x \text{ rdf:type rdfs:Resource .}} \quad \text{rdfs4b}$$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

50

Ableitungsregeln für RDFS-Folgerung

- Sub-Properties

$$\frac{u \text{ rdfs:subPropertyOf } v . \quad v \text{ rdfs:subPropertyOf } x .}{u \text{ rdfs:subPropertyOf } x .} \quad \text{rdfs5}$$

$$\frac{u \text{ rdf:type } \text{rdf:Property} .}{u \text{ rdfs:subPropertyOf } u .} \quad \text{rdfs6}$$

$$\frac{a \text{ rdfs:subPropertyOf } b . \quad u \text{ a } y .}{u \text{ b } y .} \quad \text{rdfs7}$$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

51

Ableitungsregeln für RDFS-Folgerung

- Sub-Klassen

$$\frac{u \text{ rdf:type rdfs:Class .}}{u \text{ rdfs:subClassOf rdfs:Resource .}} \quad \text{rdfs8}$$

$$\frac{u \text{ rdfs:subClassOf } x . \quad v \text{ rdf:type } u .}{v \text{ rdf:type } x .} \quad \text{rdfs9}$$

$$\frac{u \text{ rdf:type rdfs:Class .}}{u \text{ rdfs:subClassOf } u .} \quad \text{rdfs10}$$

$$\frac{u \text{ rdfs:subClassOf } v . \quad v \text{ rdfs:subClassOf } x .}{u \text{ rdfs:subClassOf } x .} \quad \text{rdfs11}$$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

52

Ableitungsregeln für RDFS-Folgerung

- Container

$$\frac{u \text{ rdf:type rdfs:ContainerMembershipProperty .}}{u \text{ rdfs:subPropertyOf rdfs:member .}} \quad \text{rdfs12}$$

- Literale

$$\frac{u \text{ rdf:type rdfs:Datatype .}}{u \text{ rdfs:subClassOf rdfs:Literal .}} \quad \text{rdfs13}$$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen



53

RDFS-Schlussfolgerungen und Inkonsistenzen

- Aus einem gegebenen inkonsistenten Graphen G kann jeder beliebige Graph gefolgert werden
- Inkonsistenz: es gibt keine Interpretation I , für die $G^I = \text{wahr}$
- Allerdings gibt es in RDFS nur eingeschränkte Möglichkeiten zur Erzeugung von Inkonsistenzen

- Bsp. „XML-Clash“:

```
hpi:hatSmiley rdfs:range rdf:Literal .
```

```
hpi:böseBemerkung hpi:hatSmiley „>:->“ ^^ XMLLiteral .
```

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

54

RDFS-Schlussfolgerungen

- **Satz:**

Ein Graph G_2 **RDFS-folgt** aus einem Graph G_1 genau dann, wenn es einen Graphen G_1' gibt, der aus G_1 mit Hilfe der Regeln rdfax, lg, rdf1, rdf2, rdfs1 - rdfs13 und rdfsax hergeleitet werden kann, so dass

- (1) G_2 aus G_1' einfach folgt, oder
- (2) G_1' einen XML-Clash enthält (inkonsistent ist)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen



55

Zusätzliche Regeln für externe Datentypen

- Externe Datentypen können in RDFS als `rdfs:datatype` charakterisiert werden
- „Funktionsweise“ der externen Datentypen lässt sich nicht mit RDFS-Graphen vollständig charakterisieren
- Zusätzliche Ableitungsregeln für allgemeine Zusammenhänge externer Datentypen

$$\frac{d \text{ rdf:type rdfs:Datatype . } \cup a \text{ "s" } ^d .}{_ :n \text{ rdf:type d.}}$$

rdfD1

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.6 RDF(S) Folgerungen

56

Zusätzliche Regeln für externe Datentypen

- Wertebereiche bestimmter Datentypen können sich überlappen, z.B. "15" $\wedge \wedge$ xsd:double und "15" $\wedge \wedge$ xsd:Integer
- Bezeichne s mit Datentyp d denselben Wert wie t mit Datentyp e , dann

$$\frac{\begin{array}{l} d \text{ rdf:type rdfs:Datatype .} \\ e \text{ rdf:type rdfs:Datatype .} \\ u \text{ a "s" } \wedge \wedge d . \end{array}}{u \text{ a "t" } \wedge \wedge e .} \quad \text{rdfD2}$$

- Liegt der Wertebereich des Datentyps d im Wertebereich des Datentyps e

$$\frac{}{d \text{ rdfs:subClassOf } e .} \quad \text{rdfDax}$$

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)

57

3.6 Semantik von RDF(S)

3.6.1 Warum eine Semantik für RDF(S)?

3.6.2 Modelltheoretische Semantik für RDF(S)

3.6.3 Einfache Interpretationen

3.6.4 RDF Interpretationen

3.6.5 RDFS Interpretationen

3.6.6 RDF(S) Folgerungen

3.6.7 Semantische Grenzen von RDF(S)

3. Wissensrepräsentationen

3.6 Semantik von RDF(S)/3.6.7 semantische Grenzen von RDF(S)



58

Intensionale vs. Extensionale Semantik

- Angegebene Semantik („Standard-Semantik“, intentionale Semantik) ist nicht die einzig „sinnvolle“ Semantik für RDF(S)
- Andere Semantiken können strengere Anforderungen an die Interpretationen stellen (extensionale Semantik)
- Aber: Ableitungsregeln der intensionalen Semantik lassen sich implementationstechnisch einfacher umsetzen

- Problem: RDF(S) enthält keine Möglichkeit der Negation
 - `hpi:harald rdf:type hpi:Nichtraucher .`
`hpi:harald rdf:type hpi:Raucher .`

--> führt nicht automatisch zum Widerspruch....

1

2

3

4

5

6

7

8.01.2009 – Vorlesung Nr. 8

9

10

11

12

13

3. Wissensrepräsentationen

3.0 Motivation

3.1 Ontologien in der Philosophie

3.2 Ontologien in der Informatik

3.3 Ontologie Beschreibungssprachen

3.4 Ontologietypen

3.5 Wiederholung Aussagenlogik und Prädikatenlogik

3.6 Semantik von RDF(S)

3.7 Beschreibungslogiken und Web Ontology Language OWL

3.8 Regeln mit SWRL / RIF

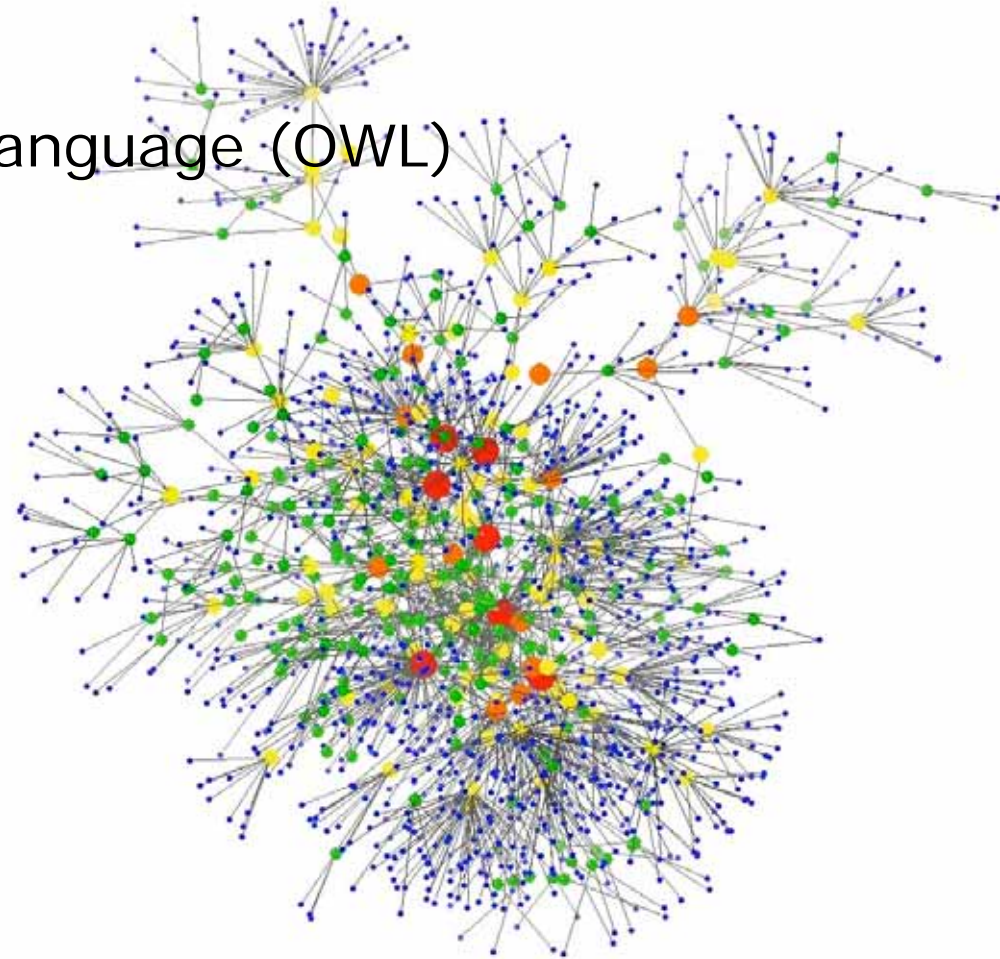
3.9 Logikbasierte Systeme

3. Wissensrepräsentationen

3.6 Semantik von RDF/RDFS

50

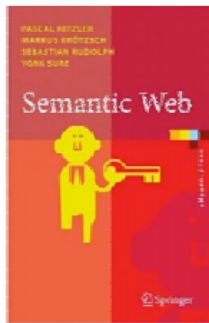
Nächste Vorlesung:
Web Ontology Language (OWL)



3. Wissensrepräsentationen

51

Literatur



» P. Hitzler, M. Krötzsch, S. Rudolph, Y. Sure
[Semantic Web Grundlagen](#), Springer, 2008.

3. Wissensrepräsentationen

52

Literatur



- Blog
<http://sw0809.blogspot.com/>
- Materialien-Webseite
http://www.hpi.uni-potsdam.de/meinel/teaching/semantic_web_ws08090.html



- bibsonomy - Bookmarks
<http://www.bibsonomy.org/user/lysander07/sw0809-08>