Lehrveranstaltung

Semantic Web Technologien

WS 2009/10

HTWG Konstanz

Formale Semantik in RDF(S)

Formale Semantik in RDF(S)

- Warum benötigen wir eine formale Semantik?
  - In den Dokumenten der W3C Recommendation zu RDF(S) ist keine formale Definition der hinter RDF(S) liegenden Semantik enthalten
  - Dies führte zu diversen Problemen, da die Konzepte von unterschiedlichen Tool-Herstellern unterschiedlich interpretiert wurden
    - Vor allem bei Triple-Stores
      - Gleiche Anfrage an zwei unterschiedliche Store-Systeme kann unterschiedliche Resultate liefern obwohl die Systeme die gleichen Daten enthalten.

=> Wir benötigen eine formale Semantik

Formale Semantik in RDF(S)

- Voraussetzungen
  - Mit mathematischer Logik können wir das korrekte ziehen von Schlüssen formalisieren
  - Was brauchen wir?
    - Menge von Sätzen über die wir Schlüsse ziehen wollen Ω
    - Schlussfolgerungsrelation
      - Um Schlüsse zu ziehen (z.B. : $\{s_1,s_2,s_3\} \models s$)
    - Logik $L = (\Omega \models)$

Formale Semantik in RDF(S)

Modelltheoretische Semantik

- Grundidee:
  - Aussagen einer Logik mit Interpretationen ins Verhältnis setzen
- Interpretationen $(\Delta^I, {}^I)$
  - Δ^I Domain of Disclosure
    - $\Delta^I \neq \emptyset$
  - Interpretationsfunktion I
    - $I: A \rightarrow A^I \quad \subseteq \Delta^I$ A: Atomares Konzept
    - $I: R \rightarrow R^I \quad \subseteq \Delta^I \times \Delta^I$ R: Atomare Relation
- Kriterien, die die Entscheidung ermöglichen, ob eine konkrete Interpretation I einen Satz $s \in \Omega$ "erfüllt" (Modellrelation)
  - I ist ein Modell von s: $I \models s$

Formale Semantik in RDF(S)

Modelltheoretische Semantik

- Definition der Relation $\models$

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

Formale Semantik in RDF(S)

Modelltheoretische Semantik für RDF(S)

- In RDF(S) werden die Sätze durch (s,p,o)-Tripel dargestellt
  - Jedes Tripel ist ein Satz
- Tripel werden mit dem Grundvokabular V beschrieben
  - URIs, B-Nodes und Literale

- $(s,p,o) \in (URI \cup B\text{-}Node) \times URI \times (URI \cup B\text{-}Node \cup Literal)$
- Ein RDF-Graph ist eine endliche Menge von Tripeln
- Auch jeder RDF Graph ist ein Satz

Formale Semantik in RDF(S)

Modelltheoretische Semantik für RDF(S)

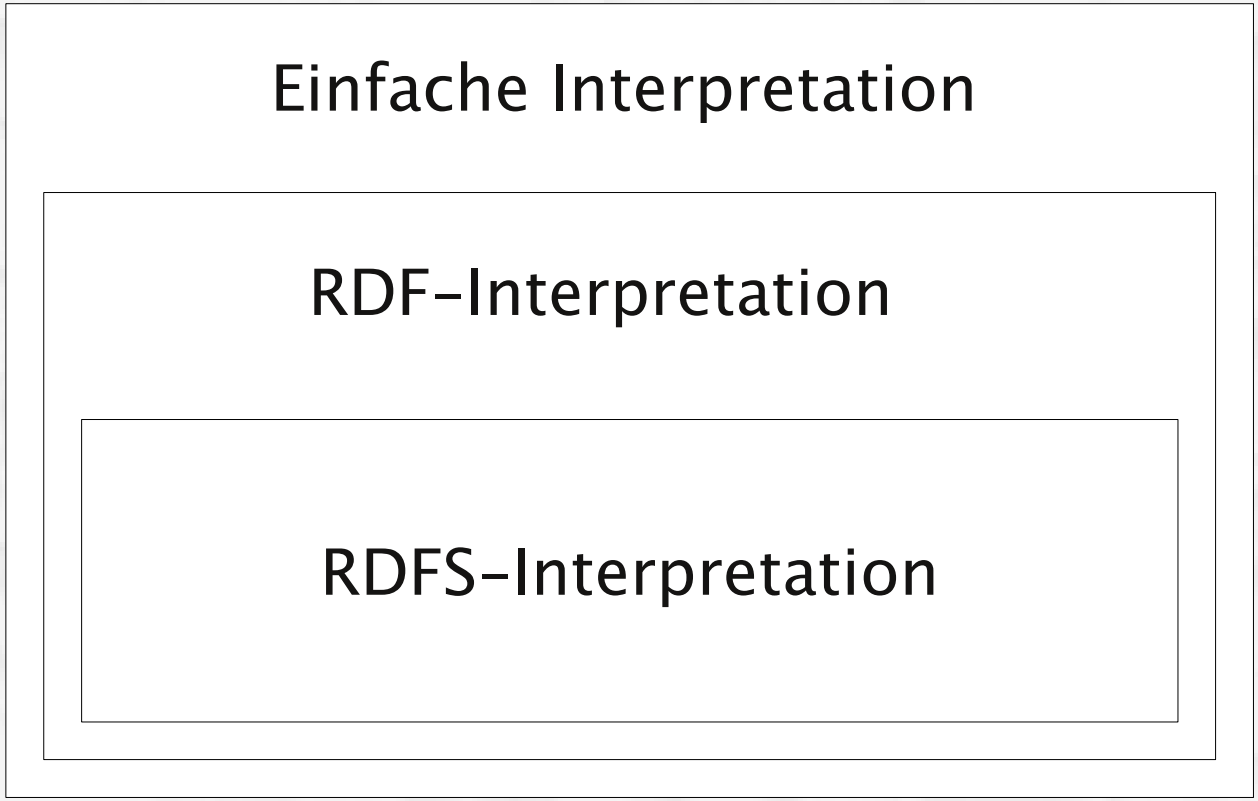
- Schlussfolgerungsrelation ⊨
  - ⊨ Gibt an, wann ein RDF(S)-Graph G' aus einem RDF(S)-Graph G folgt
  - d.h. G ⊨ G'

- Um eine modelltheoretische Semantik für RDF(S) definieren, definieren wir eine Reihe von **Interpretationen** und legen fest, wann eine Interpretation **Modell** eines Graphen ist

Formale Semantik in RDF(S)

Modelltheoretische Semantik für RDF(S)

- Schrittweises Vorgehen bei der Definition



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

Formale Semantik in RDF(S)

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,
- $\mathbf{I_{EXT}}$, 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,
- $\mathbf{I_S}$, einer Funktion, die URIs aus V in die Vereinigung der Mengen IR und IP abbildet, d.h. $I_S: V \rightarrow IR \cup IP$,
- $\mathbf{I_L}$, 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

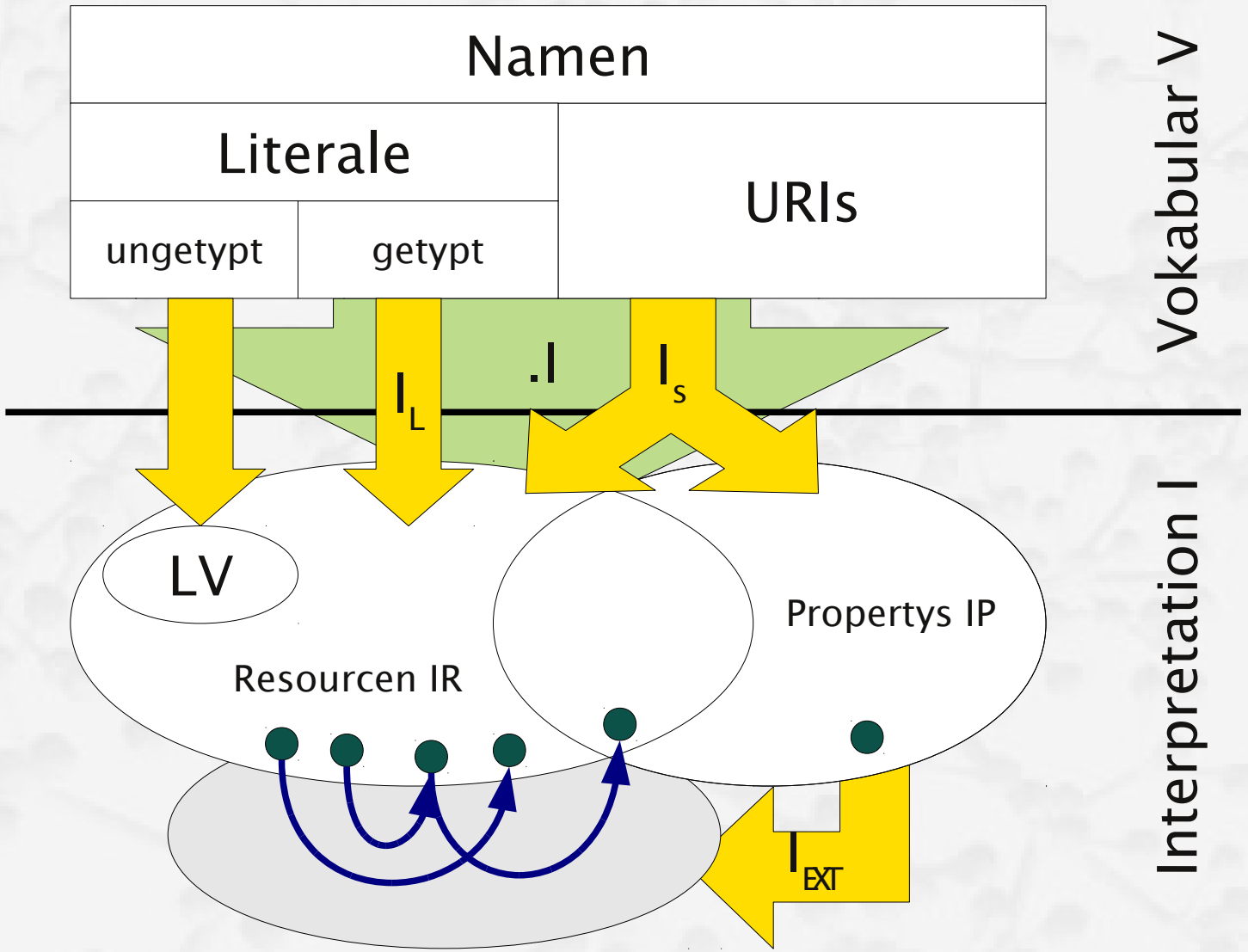
Formale Semantik in RDF(S)

- Wir definieren eine einfache Interpretationsfunktion $.^I$, die alle im Vokabular V enthaltenen Literale und URIs auf Resourcen und Properties abbildet.

- jedes ungetypte Literal “a“ wird auf a abgebildet: (“a“)^I=a
- jedes ungetypte Literal mit Sprachangabe “a“@t wird auf das Paar $\langle$ a,t $\rangle$ abgebildet: (“a@t“)^I= $\langle$ a,t $\rangle$
- jedes getypte Literal l wird auf $I_L(l)$ abgebildet:l^I= $I_L(l)$

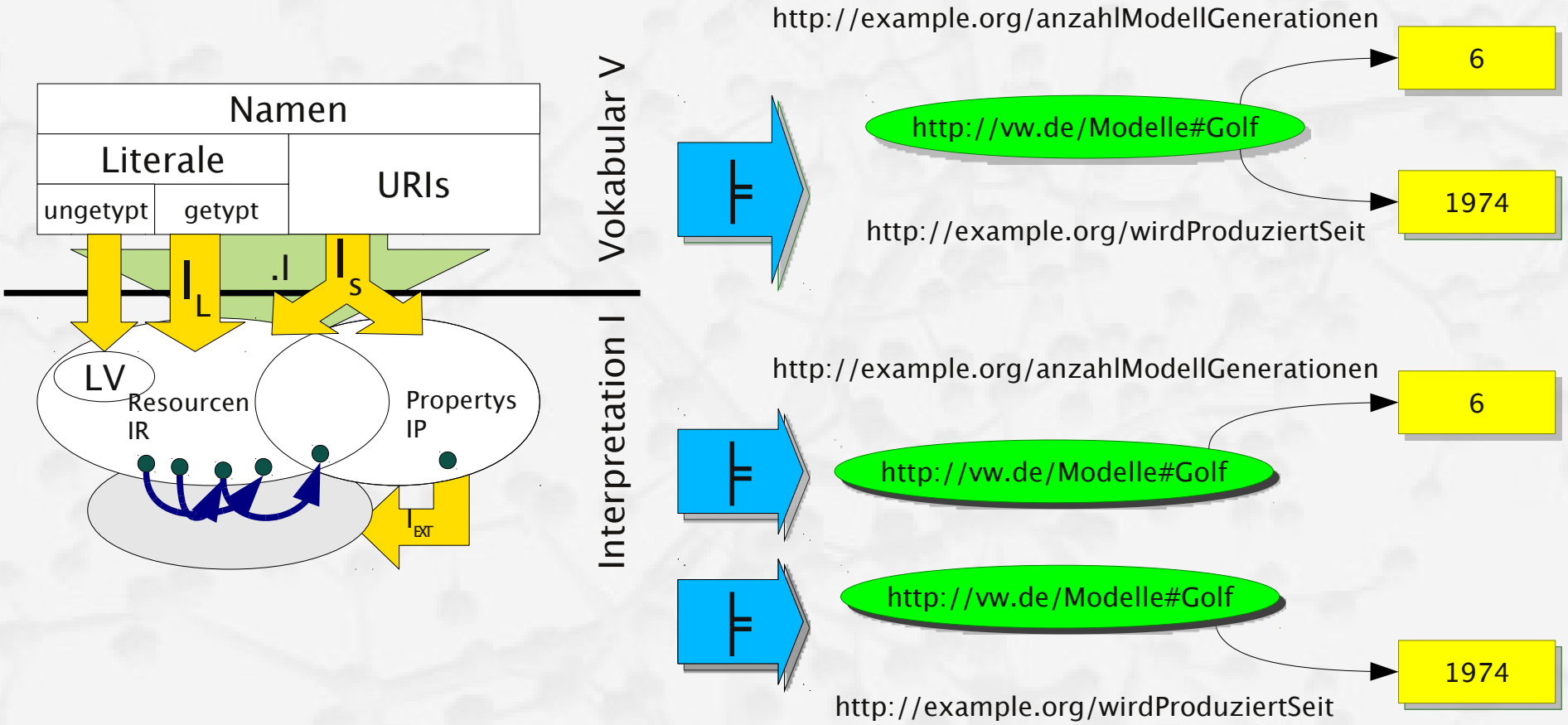
- jede URI u wird auf $I_S(u)$ abgebildet:u^I=$I_S(u)$

Formale Semantik in RDF(S)



Formale Semantik in RDF(S)

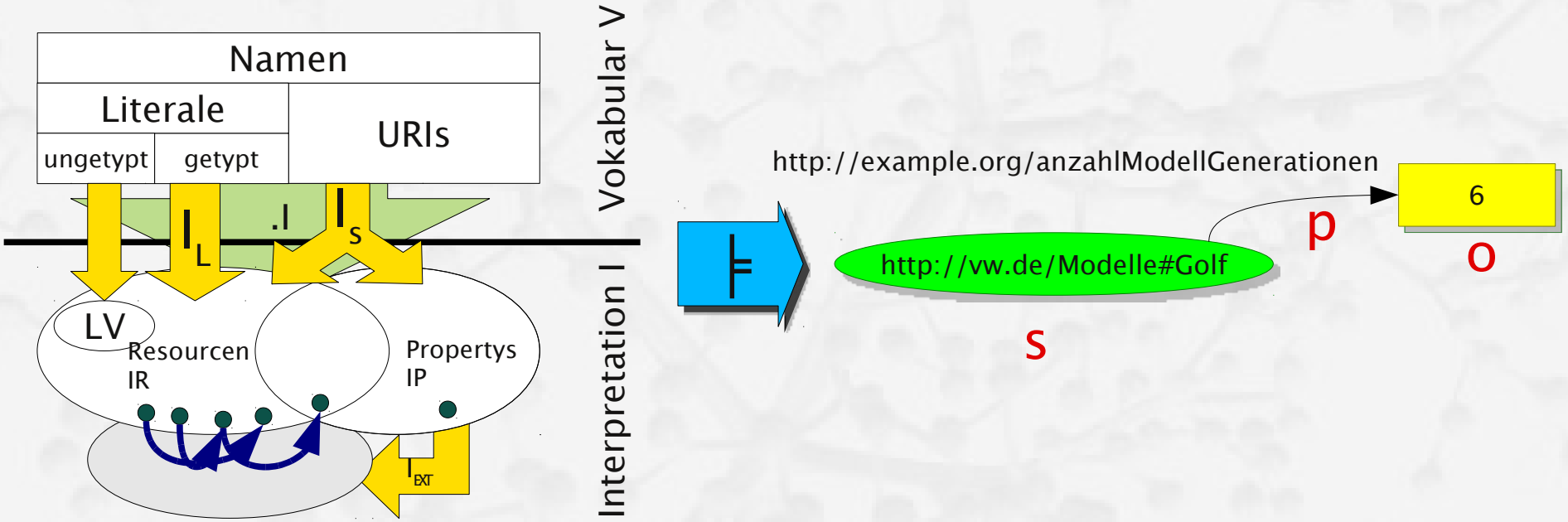
Wann ist ein Modell Interpretation eines Graphen?



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

Formale Semantik in RDF(S)

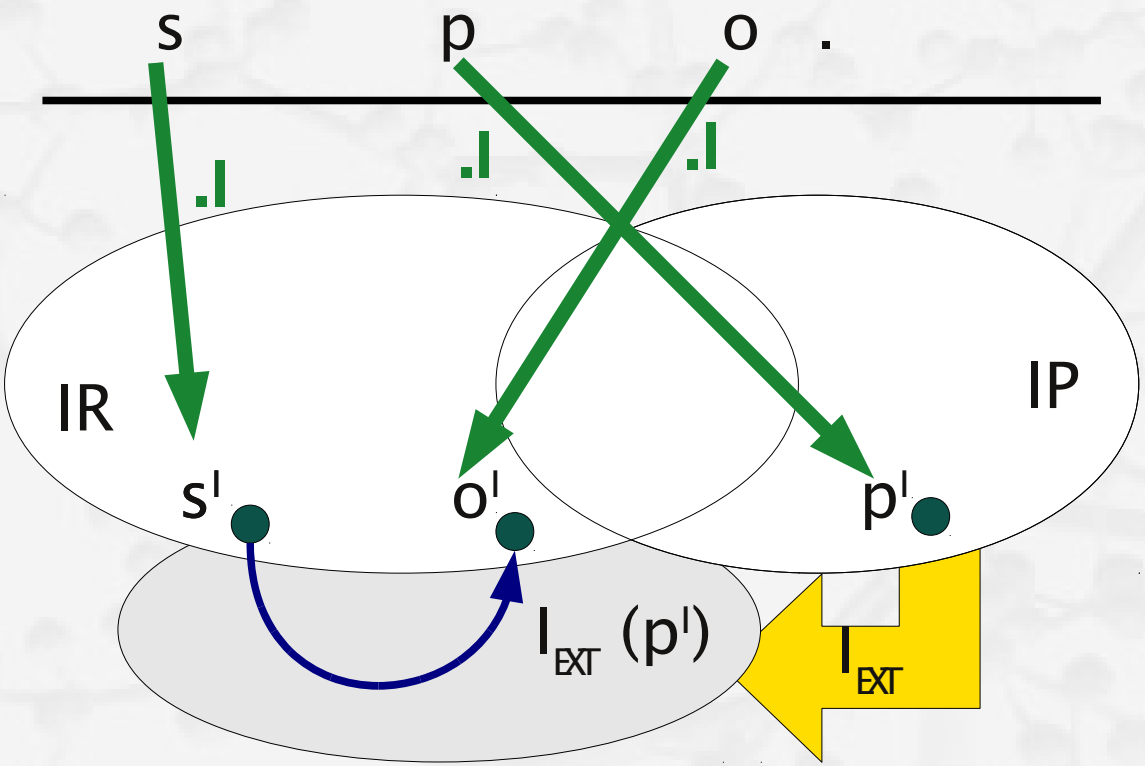
Wann ist eine Interpretation Modell eines Tripels?



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

Formale Semantik in RDF(S)

Einfache Interpretation eines Triples (Schematisch)



Die Interpretation .I ordnet dem Graph G einen Wahrheitswert zu.

G^I ist genau dann wahr, wenn für alle $T \in G$ T^I wahr ist

Formale Semantik in RDF(S)

Berücksichtigung von Blank Nodes

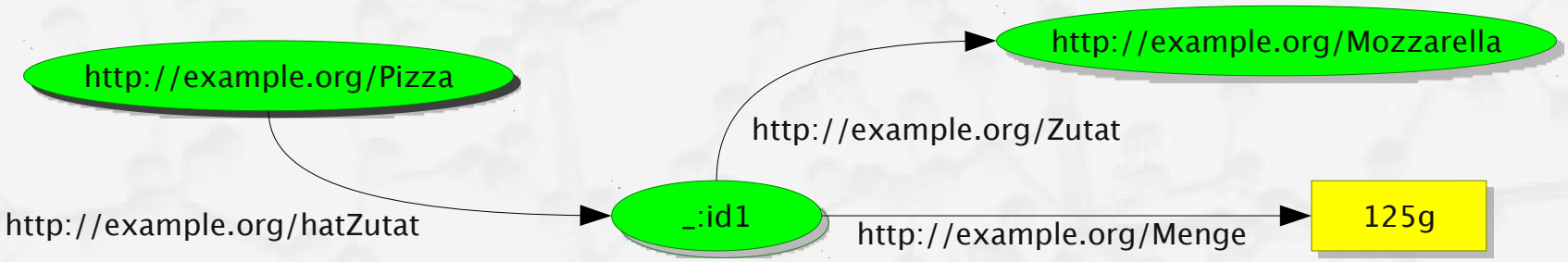
- sei A Funktion, die alle B-Nodes auf Elemente von IR abbildet
- für eine Interpretation *I*, sei *I+A* wie *I*, wobei zusätzlich für jeden B-Node b gilt:
  - $b^{I+A} = A(b)$
- eine Interpretation *I* ist jetzt Modell eines RDF-Graphen G, wenn es ein A gibt, so dass alle Tripel bezüglich *I+A* wahr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

Fazit:

Ein Graph G2 folgt einfach aus einem Graph G1, wenn jede einfache Interpretation, die Modell von G1 ist auch Modell von G2 ist.

Formale Semantik in RDF(S)

Beispiel



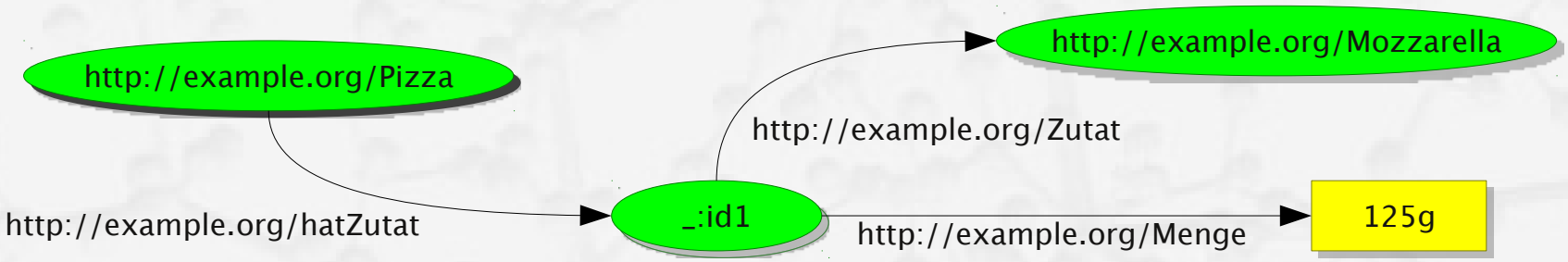
$IR = \{\chi, \upsilon, \tau, \nu, \epsilon, \iota, 125g\}$
$IP = \{\tau, \nu, \iota\}$
$LV = \{125g\}$
$I_{EXT} = \tau \rightarrow \{<\chi, \epsilon>\}$
$\nu \rightarrow \{<\epsilon, \upsilon>\}$
$\iota \rightarrow \{<\epsilon, 125g>\}$

$I_S =$ ex:Pizza $\rightarrow \chi$
ex:Mozzarella $\rightarrow \upsilon$
ex:hatZutat $\rightarrow \tau$
ex:Zutat $\rightarrow \nu$
ex:Menge $\rightarrow \iota$

I_L leer, da keine getypten Literale vorhanden

Formale Semantik in RDF(S)

Beispiel



Wählt man A: _:id1 → ϵ dann ergibt sich

$<ex:Pizza^{I+A}, \_:id1^{I+A}> = <\chi,\epsilon> \in I_{EXT}(\tau) = I_{EXT}(ex:hatZutat^{I+A})$

$<\_:id1^{I+A}, ex:Mozzarella> = <\epsilon,\upsilon> \in I_{EXT}(\nu) = I_{EXT}(ex:Zutat^{I+A})$

$<\_:id1^{I+A}, "125g"> = <\epsilon, 125g> \in I_{EXT}(\iota) = I_{EXT}(ex:Menge^{I+A})$

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

Formale Semantik in RDF(S)

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 VRDF:

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

Formale Semantik in RDF(S)

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)
  - unterscheide wohlgeformte / nicht-wohlgeformte Literale

Formale Semantik in RDF(S)

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

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

- 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).

(2) wenn "s"^^rdf:XMLLiteral in V enthalten und s ein wohlgeformtes XML-Literal ist, dann

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

(3) wenn "s"^^rdf:XMLLiteral in V enthalten und s kein wohlgeformtes XML-Literal ist, dann

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

Formale Semantik in RDF(S)

Zusätzlich gelten folgende “axiomatische” Tripel als 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:last	rdf:type	rdf:Property .
rdf:value	rdf:type	rdf:Property .
rdf:_1	rdf:type	rdf:Property .
rdf:_2	rdf:type	rdf:Property .
…		
rdf:Seq	rdf:type	rdf:List
rdf:nil	rdf:type	rdf:List .

- ein Graph G2 **RDF-folgt** aus einem Graph G1, wenn jede RDF-Interpretation, die Modell von G1 ist, auch Modell von G2 is

Formale Semantik in RDF(S)

RDFS-Interpretationen

- Zur korrekten Behandlung des RDFS-Vokabulars müssen zusätzliche Anforderungen an die Menge der zulässigen RDF-Interpretationen gestellt werden

- RDF Vokabular VRDFS:

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	

Formale Semantik in RDF(S)

Zur einfacheren Darstellung:

- Klassenextensionsfuktion I_{CEXT} : IR → 2^{IR}
- I_{CEXT} (y) enthalte genau diejenigen Elemente x, für die $\langle x,y \rangle \in I_{EXT}$ (rdf:type^I)
- IC = I_{CEXT} (rdfs:Class^I)
  - IC ist Extension der speziellen URI rdfs:Class

Formale Semantik in RDF(S)

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

(1) $IR = I_{CEXT}$ (rdfs:Resource^I)

- jede Ressource ist vom Typ rdfs:Resource

(2) $LV = I_{CEXT}$ (rdfs:Literal^I)

- jedes ungetypte und jedes wohlgeformte getypte Literal ist vom typ rdfs:Literal

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

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

- Ist x und y durch Property rdfs:domain/rdfs:range verbunden und verbindet das Property x die Ressourcen u und v, dann ist u/v vom Typ y

Formale Semantik in RDF(S)

(5) I_{EXT} (rdfs:subPropertyOf^I) ist reflexiv und transitiv auf IP

(6) Wenn $\langle x,y \rangle \in I_{EXT}$ (rdfs:subPropertyOf^I),

dann $x,y \in IP$ und $I_{EXT}(x) \subseteq I_{EXT}(y)$

(7) Wenn $x \in IC$, (Wenn x eine Klasse ist...)

dann $\langle x,$rdfs:Ressource$^I \rangle \in I_{EXT}$ (rdfs:subClassOf^I)

(...dann muss es eine Unterklasse der Klasse aller Ressourcen sein)

(8) Wenn $\langle x,y \rangle \in I_{EXT}$ (rdfs:subClassOf^I),

dann $x,y \in IC$ und $I_{CEXT}(x) \subseteq I_{CEXT}(y)$

(9) I_{EXT} (rdfs:subClassOf^I) ist reflexiv und transitiv auf IC

(10) Wenn $x \in I_{CEXT}$ (rdfs:ContainerMembershipProperty^I),

dann $\langle x,$rdfs:member$^I \rangle \in I_{EXT}$ (rdfs:subPropertyOf^I)

(11) Wenn $x \in I_{CEXT}$ (rdfs:Datatype^I),

dann $\langle x,$rdfs:Literal$^I \rangle \in I_{EXT}$ (rdfs:subClassOf^I)

Formale Semantik in RDF(S)

Zusätzlich noch zahlreiche axiomatische Tripel

- ```
  rdf:type            rdfs:domain rdf: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 .

  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:Liteal .
  rdf:value           rdfs:range  rdfs:Resource .
  ```

  ```
  rdfs:ContainerMembershipProperty
                      rdfs:subClassOf         rdf:Property .
  rdf:Alt             rdfs:subClassOf         rfs:Container .
  rdf:Bag             rdfs:subClassOf         rdfs:Container .
  rdf:Seq             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:typ
                            rdfs:ContainerMembershipProperty .
  Rdf:_1              rdfs:domain             rdfs:Resource .
  Rdf:_1              rdfs:range              rdfs:Resource .
  Rdf:_2              rdf:type
                            rdfs:ContainerMembershipProperty .
  Rdf:_2              rdfs:range              rdfs:Resource
  ...
  ```
  ```

Formale Semantik in RDF(S)

ein Graph G2 **RDFS-folgt** aus einem Graph G1, wenn jede RDFS-Interpretation, die Modell von G1 ist, auch Modell von G2 ist

Formale Semantik in RDF(S)

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 G1 $\models$ G2, 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

Formale Semantik in RDF(S)

Syntaktisches Schlussfolgern mit Ableitungsregeln

- Allgemeine Form von Ableitungsregeln (Deduktionsregeln):

$$\frac{s_1 \ldots s_n}{s}$$

- Sind Sätze s1,...,sn 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**

Formale Semantik in RDF(S)

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

Formale Semantik in RDF(S)

Ableitungsregeln für einfache Folgerung

- alle URIs werden gleich behandelt

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

$$\frac{u\ a\ x\ .}{\_:\mathrm{n}\ a\ x.}\ se\square$$

- Satz:

  Ein Graph G2 folgt einfach aus einem Graph G1, wenn G1 mit Hilfe der Regeln se1 und se2 zu einem Graphen G1‘ ergänzt werden kann, so dass G2 in G1‘ enthalten ist.

Formale Semantik in RDF(S)

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 B-Nodes 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 wohlgeforment XML-Literal l durch lg zugewiesen wurden

Formale Semantik in RDF(S)

Satz:

- Ein Graph G2 **RDF-folgt** aus einem Graph G1 genau dann, wenn es einen Graphen G1‘ gibt, der aus G1 mit Hilfe der Regeln rdfax, lg, rdf1 und rdf2 hergeleitet werden kann, aus dem G2 einfach folgt.

Formale Semantik in RDF(S)

Ableitungsregeln für RDFS-Folgerung

$$\frac{}{u\ a\ x.}$$ rdfsax

Jedes axiomatische Tripel “u a x .” kann immer abgeleitet werden

$$\frac{u\ a\ \_:n.}{u\ a\ l.}$$ gl

wobei _:n ein Blank Node ist, der durch vormalige Anwendung von lg enstanden ist

$$\frac{u\ a\ l.}{\_:n\ \text{rdf:type}\ \text{rdfs:Literal}}$$ rdfs1

wobei l ein ungetyptes Literal darstellt und _:n einen durch Anwendung von lg dem Literal l zugewiesenen Blank Node

Formale Semantik in RDF(S)

Ableitungsregeln für RDF-Folgerung

- Property Einschränkungen

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

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

- Alles ist eine Ressource

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

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

Formale Semantik in RDF(S)

Ableitungsregeln für RDF-Folgerung

- Sub-Properties

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

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

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

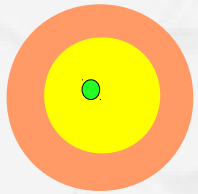
Formale Semantik in RDF(S)

Ableitungsregeln für RDF-Folgerung

- Sub-Klassen

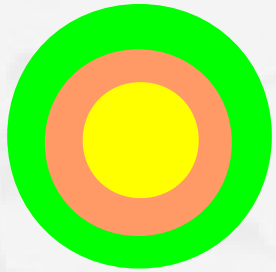
$$\frac{u \text{ rdf:type rdfs:Class .}}{u \text{ rdfs:subClassOf rdfs:Resource .}}$$ rdfs8

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



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

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



Formale Semantik in RDF(S)

Ableitungsregeln für RDF-Folgerung

- Container

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

- Literale

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

Formale Semantik in RDF(S)

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=wahr
- Allerdings gibt es in RDFS nur eingeschränkte Möglichkeiten zur Erzeugung von Inkonsistenzen

- Bsp. „XML-Clash“:

ex:hatSmiley rdfs:range rdf:Literal .
ex:böseBemerkung ex:hatSmiley „>:->“^^XMLLiteral .

Formale Semantik in RDF(S)

Satz:

Ein Graph G2 **RDFS-folgt** aus einem Graph G1 genau dann, wenn es einen Graphen G1‘ gibt, der aus G1 mit Hilfe der Regeln rdfax, lg, rdf1, rdf2, rdfs1 – rdfs13 und rdfsax hergeleitet werden kann, so dass

(1) G2 aus G1‘ einfach folgt, oder

(2) G1‘ einen XML-Clash enthält (inkonsistent ist)

Formale Semantik in RDF(S)

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

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

(mit Regel lg)

Formale Semantik in RDF(S)

Zusätzliche Regeln für externe Datentypen

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

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

rdfD2

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

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

rdfDAx

Formale Semantik in RDF(S)

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

```
ex:harald rdf:type ex:Nichtraucher .
ex:harald rdf:type ex:Raucher .
```

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

Sprachen des Semantic Web – RDF

Noch Fragen ?

Sprachen des Semantic Web – XML

- Literatur:
  - Buch “Semantic Web Grundlagen”, Springer Verlag 2008
    Pascal Hitzler, Markus Krötzsch, Sebastian Rudolph, York Sure
    ISBN: 978-3-540-33993-9
  - RDF Webseite
    http://www.w3.org/RDF/
  - Resource Description Framework Schema
    Dan Brickley, R.V. Guha, Brian McBride
    W3C Recommendation, 10th February 2004
    http://www.w3.org/TR/rdf-schema
  - RDF Planet – Blogs zum Thema RDF / Semantic Web
    http://www.planetrdf.com/