

Semantic Web Technologies II

SS 2008

21.05.2008

Formale Semantik von OWL

Dr. Peter Haase
PD Dr. Pascal Hitzler
Dr. Steffen Lamparter
Denny Vrandečić



Content licensed under Creative Commons
<http://creativecommons.org/licenses/by/2.0/de/>

Einleitung

- Wozu dient Semantik?

Beispiel:

Schreibtisch \sqsubseteq Tisch

Möbelstück \sqsubseteq Gegenstand

Einleitung

- Wie definiert man eine Semantik?

Beispiel:

Schreibtisch \sqsubseteq Tisch

Möbelstück \sqsubseteq Gegenstand

Topics

- **Warum und was ist formale Semantik?**
- Beschreibungslogiken (und OWL)
- Semantik von OWL
- Aktuelle Forschungsthemen:
Erweiterung der Ausdrucksmächtigkeit von OWL

Syntax und Semantik

Syntax: Bedeutungsleere Zeichenfolgen

Semantik: Bedeutung der Syntax

IF cond(A,B)
THEN display(_354)

Syntax

Zeige Pixelmenge „_354“ auf
dem Bildschirm, wenn die
Blume „A“ rot ist.

Bedeutung
z.B. „die Welt“

Zuweisung von Bedeutung

Was ist Semantik? Beispiel Programmiersprache

Syntax

```
FUNCTION f (n:natural) :natural;  
BEGIN  
  IF n=0 THEN f:=1  
  ELSE f:=n*f (n-1);  
END;
```

Berechnung der Fakultät

Intendierte Semantik

$f : n \mapsto n!$

Formale Semantik

Verhalten des Programms
bei Ausführung

Prozedurale Semantik

Semantik von Logik/Wissensrepräsentationssprachen

Syntax

$$\forall x (p(x) \rightarrow q(x))$$

Alle Menschen
sind sterblich

Intendierte Semantik

Logische
Konsequenz

\models

Modelltheoretische Semantik

\vdash

Beweisbarkeit
in einem Kalkül

Beweistheoretische Semantik

Topics

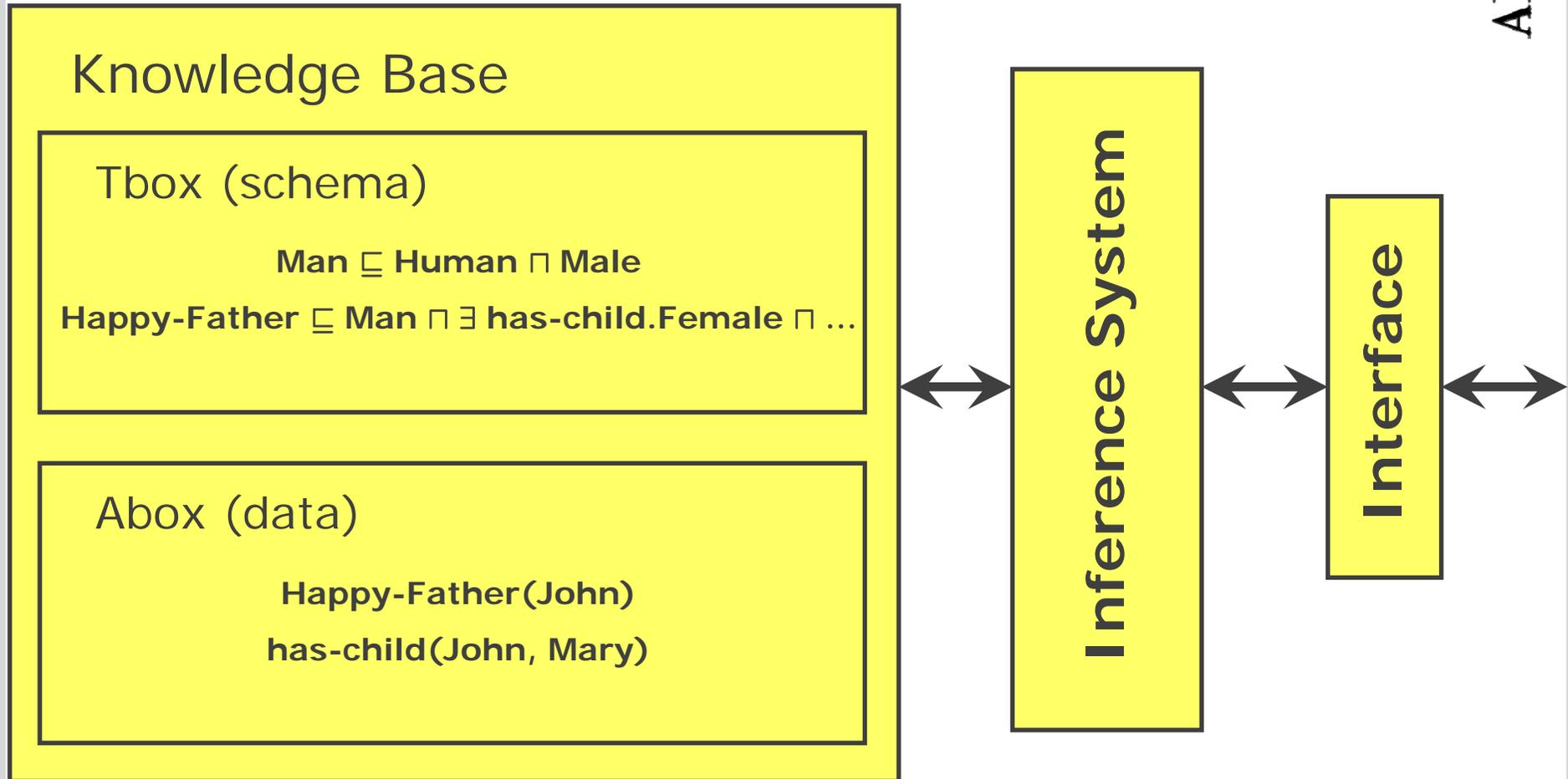
- Warum und was ist formale Semantik?
- **Beschreibungslogiken (und OWL)**
- Semantik von OWL
- Aktuelle Forschungsthemen:
Erweiterung der Ausdrucksmächtigkeit von OWL

DLs – allgemeiner Aufbau

- DLs sind eine **Familie** logikbasierter Formalismen zur Wissensrepräsentation
- Spezielle Sprachen v.a. charakterisiert durch:
 - Konstruktoren für komplexe Konzepte und Rollen aus einfacheren.
 - Menge von Axiomen um Fakten über Konzepte, Rollen und Individuen auszudrücken.
- ALC ist die kleinste DL, die aussagenlogisch abgeschlossen ist
 - Konjunktion, Disjunktion, Negation sind Konstruktoren, geschrieben \sqcap , \sqcup , \neg .
 - Quantoren schränken Rollenbereiche ein:

$\text{Man} \sqcap \exists \text{hasChild.Female} \sqcap \exists \text{hasChild.Male} \sqcap \forall \text{hasChild.}(\text{Rich} \sqcup \text{Happy})$

Allgemeine DL Architektur



ALC: Grundbausteine

- Grundbausteine:
 - Klassen
 - Rollen
 - Individuen
- Professor(RudiStuder)
 - Individuum RudiStuder ist in Klasse Professor
- Zugehoerigkeit(RudiStuder,AIFB)
 - RudiStuder ist dem AIFB zugehörig

ALC: Subklassenbeziehungen

- Professor \sqsubseteq Fakultaetsmitglied
 - entspricht $(\forall x)(\text{Professor}(x) \rightarrow \text{Fakultaetsmitglied}(x))$
 - entspricht owl:subClassOf
- Professor \equiv Fakultaetsmitglied
 - entspricht $(\forall x)(\text{Professor}(x) \leftrightarrow \text{Fakultaetsmitglied}(x))$
 - entspricht owl:equivalentClass

ALC: komplexe Klassenbeziehungen

- Konjunktion \sqcap entspricht owl:intersectionOf
- Disjunktion \sqcup entspricht owl:unionOf
- Negation \neg entspricht owl:complementOf

- Professor \sqsubseteq (Person \sqcap Unversitaetsangehoeriger)
 \sqcup (Person \sqcap \neg Doktorand)

$(\forall x)(\text{Professor}(x) \rightarrow$

$((\text{Person}(x) \wedge \text{Unversitaetsangehoeriger}(x))$

$\vee (\text{Person}(x) \wedge \neg \text{Doktorand}(x)))$

ALC: Quantoren

- Prüfung $\sqsubseteq \forall \text{hatPruefer. Professor}$
 $(\forall x)(\text{Pruefung}(x) \rightarrow (\forall y)(\text{hatPruefer}(x,y) \rightarrow \text{Professor}(y)))$
 - entspricht owl:allValuesFrom
- Prüfung $\sqsubseteq \exists \text{hatPruefer. Person}$
 $(\forall x)(\text{Pruefung}(x) \rightarrow (\exists y)(\text{hatPruefer}(x,y) \wedge \text{Person}(y)))$
 - entspricht owl:someValuesFrom

Modellierung in ALC

- owl:nothing:
- owl:thing:
- owl:disjointWith:

- rdfs:range:
- rdfs:domain:

$$\perp \equiv C \sqcap \neg C$$

$$\top \equiv C \sqcup \neg C$$

ALC: Syntax

- Folgende Syntaxregeln erzeugen Klassen in ALC. Dabei ist A eine atomare Klasse und R eine Rolle.
 $C, D \rightarrow A \mid \top \mid \perp \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \forall R.C \mid \exists R.C$
- Eine *ALC TBox* besteht aus Aussagen der Form $C \sqsubseteq D$ und $C \equiv D$, wobei C, D Klassen sind.
- Eine *ALC ABox* besteht aus Aussagen der Form $C(a)$ und $R(a,b)$, wobei C eine komplexe Klasse, R eine Rolle und a, b Individuen sind.
- Eine *ALC-Wissensbasis* besteht aus einer ABox und einer TBox.

DL Wissensbasen

- DL Wissensbasen bestehen aus 2 Teilen:
 - TBox: Axiome, die die Struktur der zu modellierenden Domäne beschreiben (konzeptionelles Schema):
 - $\text{HappyFather} \equiv \text{Man} \sqcap \exists \text{hasChild.Female} \sqcap \dots$
 - $\text{Elephant} \sqsubseteq \text{Animal} \sqcap \text{Large} \sqcap \text{Grey}$
 - $\text{transitive}(\text{hasAncestor})$
 - ABox: Axiome, die konkrete Situationen (Daten) beschreiben:
 - $\text{HappyFather}(\text{John})$
 - $\text{hasChild}(\text{John}, \text{Mary})$
- Unterscheidung TBox/ABox hat keine logische Bedeutung
... ist aber konzeptionell einfacher.

Einfaches Beispiel

Terminologisches Wissen (*TBox*):

Human $\sqsubseteq \exists \text{hasParent.Human}$

Orphan $\equiv \text{Human} \sqcap \neg \exists \text{hasParent.Alive}$

Wissen um Individuen (*ABox*):

Orphan(harrypotter)

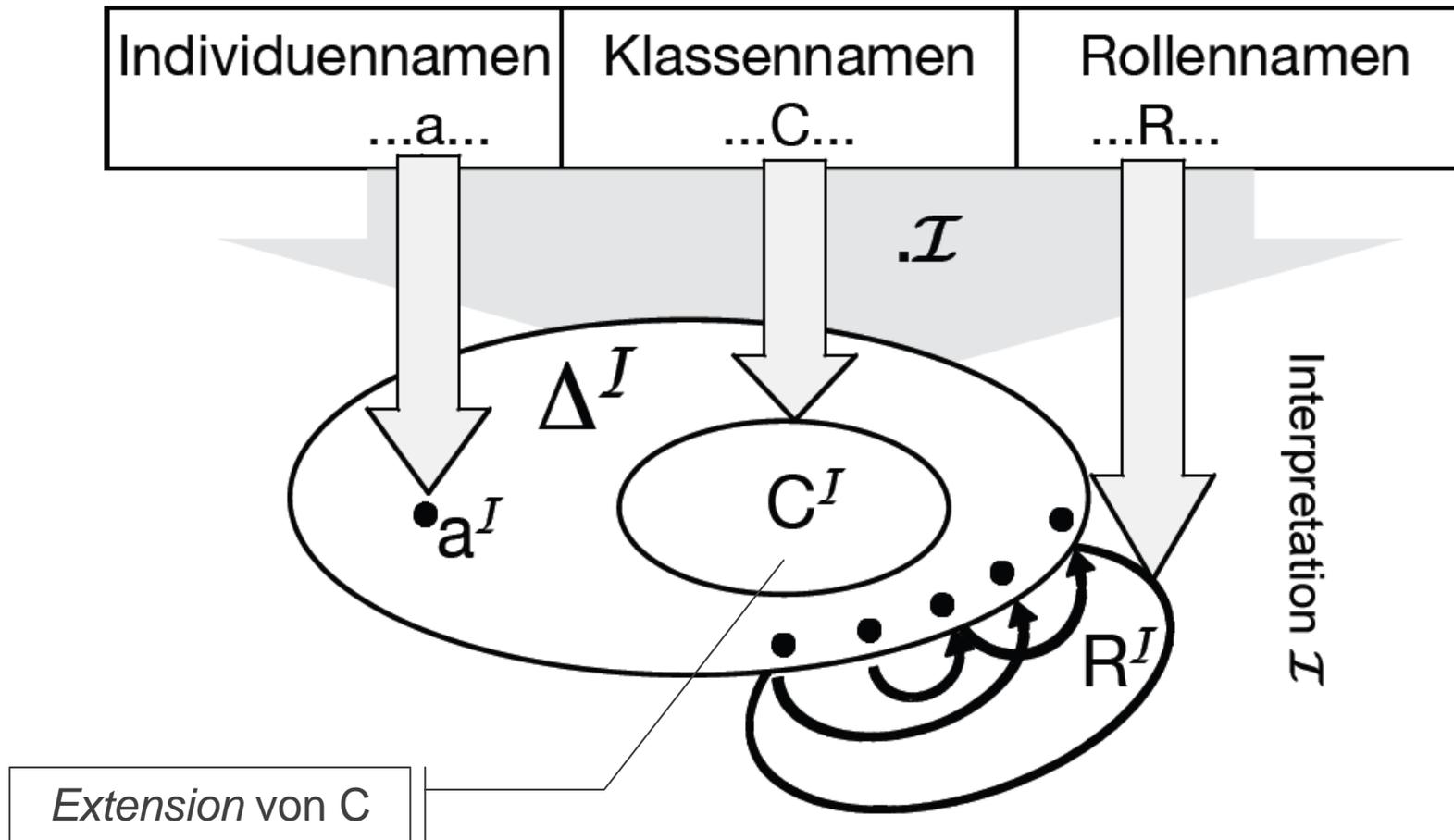
hasParent(harrypotter,jamespotter)

Semantik und logische Konsequenzen klar, da
übersetzbar nach FOL.

Topics

- Warum und was ist formale Semantik?
- Beschreibungslogiken (und OWL)
- **Semantik von OWL**
- Aktuelle Forschungsthemen:
Erweiterung der Ausdrucksmächtigkeit von OWL

Formale Semantik von ALC

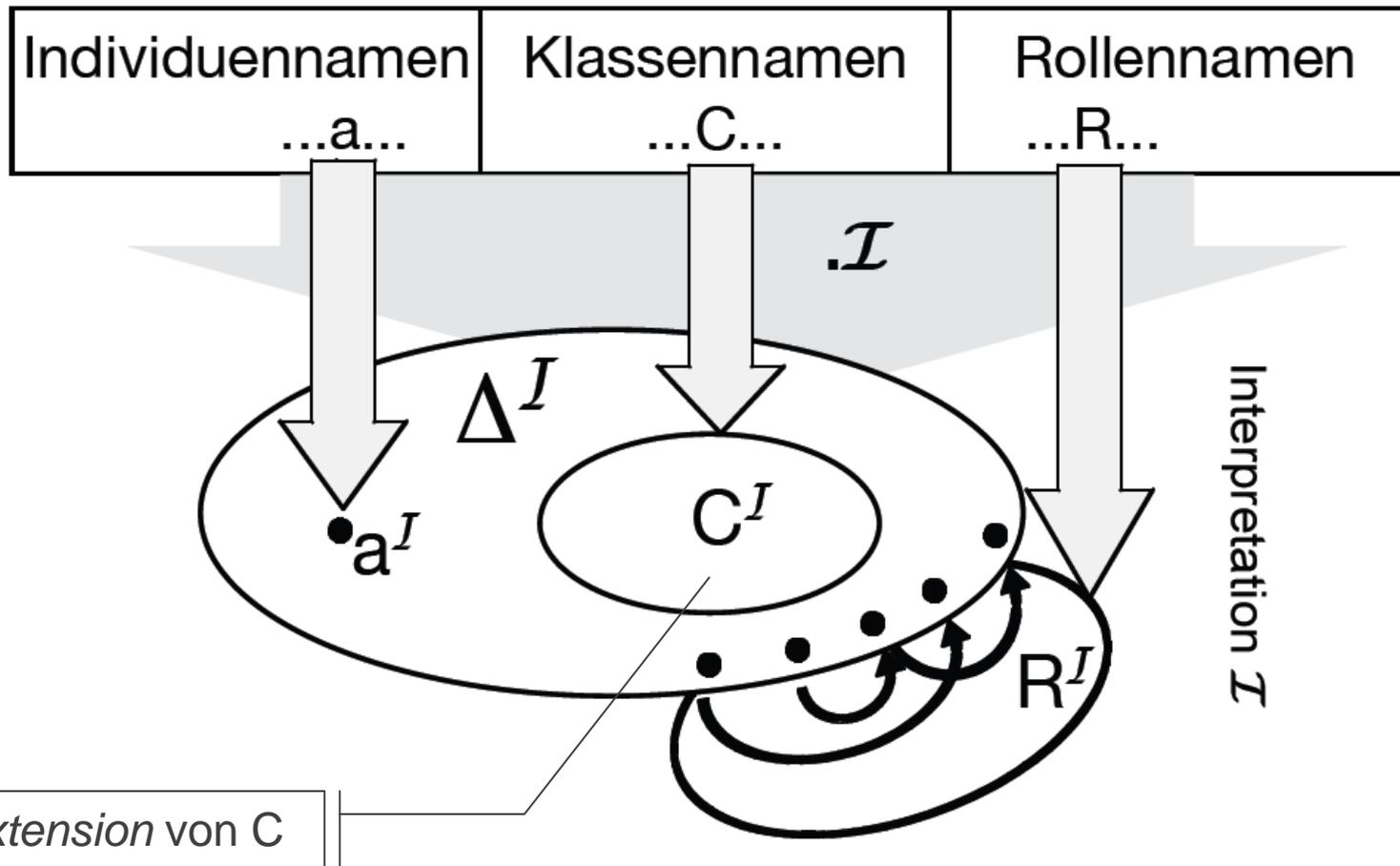


Formale Definition von "Interpretation"

Ein **Interpretation** $I = (D, \cdot^I)$ besteht aus

- einer Menge D , genannt Interpretationsbereich
- einer Funktion \cdot^I , die
 - Individuen a auf Elemente von D abbildet: $a^I \in D$
 - Klassennamen C auf Teilmengen von D abbildet: $C^I \subseteq D$
 - Rollennamen R auf Mengen von Paaren in D abbildet:
 $R^I \subseteq D \times D$

Formale Semantik von ALC



"Hochziehen" von I auf komplexe Klassen

- $\top^I =$
- $\perp^I =$
- $(C \sqcap D)^I =$
- $(C \sqcup D)^I =$
- $(\neg C)^I =$
- $(\forall R.C)^I =$
- $(\exists R.C)^I =$

"Hochziehen" von I auf komplexe Klassen

- $\top^I = D$
- $\perp^I = \emptyset$
- $(C \sqcap D)^I = C^I \cap D^I$
- $(C \sqcup D)^I = C^I \cup D^I$
- $(\neg C)^I = D \setminus C^I$
- $(\forall R.C)^I = \{x: \text{wenn } (x,y) \in R^I, \text{ dann } y \in C^I\}$
- $(\exists R.C)^I = \{x: \text{es existiert } (x,y) \in R^I \text{ mit } y \in C^I\}$

"Hochziehen" auf Axiome

- $C(a)$ gilt, wenn $a' \in C'$
- $R(a,b)$ gilt, wenn
- $C \sqsubseteq D$ gilt, wenn
- $C \equiv D$ gilt, wenn

"Hochziehen" auf Axiome

- $C(a)$ gilt, wenn $a^I \in C^I$
- $R(a,b)$ gilt, wenn $(a^I, b^I) \in R^I$
- $C \sqsubseteq D$ gilt, wenn $C^I \subseteq D^I$
- $C \equiv D$ gilt, wenn $C^I = D^I$

I ist ein **Modell** einer Wissensbasis (ABox+TBox) K ,
wenn jedes Axiom von K bezüglich I gilt.

Definition des Schlussfolgerns

Gegeben eine Wissensbasis (ABox+TBox) K .

Gegeben eine in ALC formulierte Aussage A .

A ist eine logische Konsequenz von K

geschrieben $K \models A$

genau dann, wenn

jedes Modell von K auch ein Modell für A ist.

OWL und ALC

Folgende OWL DL Sprachelemente sind in ALC repräsentierbar:

- Klassen, Rollen, Individuen
- Klassenzugehörigkeit, Rolleninstanzen
- owl:Thing und owl:Nothing
- Klasseninklusion, -äquivalenz, -disjunktheit
- owl:intersectionOf, owl:unionOf
- owl:complementOf
- Rollenrestriktionen
- rdfs:range und rdfs:domain

OWL als SHOIN(D): Individuen

- owl:sameAs
 - DL: $a=b$
 - FOL: Erweiterung durch Gleichheitsprädikat

- owl:differentFrom
 - DL: $a \neq b$
 - FOL: $\neg(a=b)$

OWL als SHOIN(D): abgeschlossene Klassen

Abgeschlossene Klassen

- owl:oneOf
 - DL: $C \equiv \{a,b,c\}$
 - FOL: $(\forall x) (C(x) \leftrightarrow (x=a \ \zeta \ x=b \ \zeta \ x=c))$
- owl:hasValue
 - Darstellbar mittels owl:someValuesFrom und owl:oneOf

OWL als SHOIN(D): Zahlenrestriktionen

Zahlenrestriktionen mittels Gleichheitsprädikat

```
<owl:Class rdf:about="#Pruefung">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hatPruefer"/>
      <owl:maxCardinality rdf:datatype="&xsd;nonNegativeInteger">2</owl:maxcardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Eine Prüfung kann *höchstens zwei* Prüfer haben.

- DL: $\text{Pruefung} \sqsubseteq \leq 2 \text{ hatPruefer}$

- In FOL: $(P \dots \text{Prüfung}, h \dots \text{hatPruefer})$

$$(\forall x)(P(x) \rightarrow \neg(\exists x_1)(\exists x_2)(\exists x_3)(x_1 \neq x_2 \wedge x_2 \neq x_3 \wedge x_1 \neq x_3 \\ \wedge h(x,x_1) \wedge h(x,x_2) \wedge h(x,x_3)))$$

Entsprechend für die anderen Zahlenrestriktionen

OWL als SHOIN(D): Rollenkonstruktoren

- Rdfs:subPropertyOf
 - DL: $R \sqsubseteq S$
 - FOL: $(\forall x)(\forall y)(R(x,y) \rightarrow S(x,y))$
- Entsprechend Rollenäquivalenz
- Inverse Rollen: $R \equiv S^{-}$
 - FOL: $(\forall x)(\forall y)(R(x,y) \leftrightarrow S(y,x))$
- Transitive Rollen: $\text{Trans}(R)$
 - FOL: $(\forall x)(\forall y)(\forall z)(R(x,y) \wedge R(y,z) \rightarrow R(x,z))$
- Symmetrie: $R \equiv R^{-}$
- Funktionalität: $\top \sqsubseteq \leq 1 R$
- Inverse Funktionalität: $\top \sqsubseteq \leq 1 R^{-}$

Datentypen

- Erlaubt ist die Verwendung von Datentypen im zweiten Argument konkreter Rollen in der ABox.
- Ausserdem kann eine Menge konkreter Daten eine abgeschlossene Klasse bilden
- Datentypen lassen sich nicht ohne Weiteres in FOL ausdrücken. Man kann die FOL Semantik aber entsprechend erweitern.

OWL DL als SHOIN(D): Überblick

Erlaubt sind:

- ALC
- Gleichheit und Ungleichheit zwischen Individuen
- Abgeschlossene Klassen
- Zahlenrestriktionen
- Subrollen und Rollenäquivalenz
- Inverse und transitive Rollen
- Datentypen

Bezeichner für Beschreibungslogiken

- ALC: Attribute Language with Complement
- S: ALC + Rollentransitivität
- H: Subrollenbeziehung
- O: abgeschlossene Klassen
- I: inverse Rollen
- N: Zahlenrestriktionen $\leq n$ R etc.
 - Q: Qualifizierende Zahlenrestriktionen $\leq n$ R.C etc.
- (D): Datentypen
- F: Funktionale Rollen
- R: Rollenkonkatenation hasParent \circ hasBrother

- OWL DL ist SHOIN(D)
- OWL 2 ist SROIQ(D)
- OWL Lite ist SHIF(D)

Komplexitäten (worst-case)

OWL Variante	Datenkomplexität	Kombinierte Komplexität
OWL Full	unentscheidbar	unentscheidbar
OWL DL	unbekannt	NExpTime
OWL 2	unbekannt	N ² ExpTime
OWL DL ohne Nominals	NP	ExpTime
OWL Lite	NP	ExpTime

Datenkomplexität: nur bezüglich ABox

Kombinierte Komplexität: bezüglich ABox und TBox

Beispiel: DL und RDF Syntax

Person $\sqcap \forall \text{hasChild} . (\text{Doctor} \sqcup \exists \text{hasChild} . \text{Doctor})$:

```
<owl:Class>
  <owl:intersectionOf rdf:parseType="collection">
    <owl:Class rdf:about="#Person"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasChild"/>
      <owl:allValuesFrom>
        <owl:unionOf rdf:parseType="collection">
          <owl:Class rdf:about="#Doctor"/>
          <owl:Restriction>
            <owl:onProperty rdf:resource="#hasChild"/>
            <owl:someValuesFrom rdf:resource="#Doctor"/>
          </owl:Restriction>
        </owl:unionOf>
      </owl:allValuesFrom>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

Topics

- Warum und was ist formale Semantik?
- Beschreibungslogiken (und OWL)
- Semantik von OWL
- **Aktuelle Forschungsthemen:
Erweiterung der Ausdrucksmächtigkeit von OWL**

Warum kann OWL so wenig?

Anforderungen aus der Praxis für an Ontologiesprachen

- Closed World vs. Open World
- Umgang mit Inkonsistenzen
- Umgang mit unsicherem Wissen
- Verwendung von Regeln

Closed World

z.B. Wissensbasis:

hasChild(john, peter)

hasChild(john, paul)

male(peter)

male(paul)

folgt daraus: $(\forall \text{hasChild.male})(\text{john})?$

Ist das sinnvoll?

Closed World in OWL – wie?

- viele Möglichkeiten (klassisches Forschungsgebiet in der KI: nichtmonotones Schließen, eng verknüpft mit Logikprogrammierung)
- ein Blickwinkel/Beispiel:
Man will aus einer Wissensbasis *mehr* Schlußfolgerungen ziehen als klassisch möglich wäre.
→ betrachten von *weniger* Modellen!

Betrachten von weniger Modellen

z.B. Wissensbasis:

hasChild(john, peter)

hasChild(john, paul)

male(peter)

male(paul)

folgt daraus: $(\forall \text{hasChild.male})(\text{john})?$

- betrachte nur Modelle, für die die Extension von hasChild *minimal* ist

→ *Circumscription*

Inkonsistenzen

- vogel \sqsubseteq fliegt
pinguin \sqsubseteq vogel
pinguin \sqcap fliegt $\sqsubseteq \perp$
pinguin(tweety)
- adler \sqsubseteq fliegt
(frischGeschlüpft \sqcap adler)(charlie)
frischGeschlüpft \sqcap fliegt $\sqsubseteq \perp$

Was tun mit Inkonsistenzen?

1. debugging
finde maximale konsistente Teilmengen
Problem: Wahlmöglichkeiten
 - adler \sqsubseteq fliegt
(frischGeschlüpft \sqcap adler)(charlie)
frischGeschlüpft \sqcap fliegt \sqsubseteq \perp

Was tun mit Inkonsistenzen?

2. Modifikation der Semantik
erlaube Wahrheitswert für "inkonsistent"
wie konstruiert man sowas?
 - benötige neuen Modellbegriff
 - Problem: Wahlmöglichkeiten bei der Konstruktion der Semantik

Unsicheres Wissen

- z.B. Informationsaustausch zwischen Fahrzeugen: Sensordaten sind unscharf
- z.B. Wissen, das durch automatische Textanalyse aus Webseiten gewonnen wird: Wissen ist unsicher
- Wie löst man das?

Lösung für unsicheres Wissen

- z.B. erlaube reelle Werte im Intervall $[0,1]$ als Wahrheitswerte (\rightarrow Fuzzy Logik)
- Benötige neuen Modellbegriff!
- Probleme:
 - Freiheitsgrade bei Definition der Semantik
 - Schlussfolgern wird sehr kompliziert

Regeln

- Viele Anwendungen regelbasierter Systeme
- Wie verhalten sich Regeln zu OWL?

→ Wichtiges Forschungsgebiet. Sehr viele verschiedene Ansätze. Situation noch recht unklar.

Verschiedene Ansätze für Regeln + OWL

- hybrid: loose integration
regelbasierte Wissensbasis + OWL Wissensbasis
hybride Semantik definiert, wie Information zwischen ihnen "fließt"
- hybrid: tight integration
vereinheitlichte Semantik für die hybride Wissensbasis
- Regelfragmente von OWL
Welche Teile von OWL lassen sich durch Regeln darstellen?
- OWL-artige Regelerweiterungen von OWL
z.B. durch prädikatenlogische Implikation (+ Semantik)

Regeln: Alternative zu OWL?

z.B. Ontologiesprache F-Logik

- rein regelbasiert
- closed-world Semantik aus der Logikprogrammierung
- verwendet in kommerziellen Systemen z.B. durch ontoprise GmbH

Rückblick auf die Stunde

- Beschreibungslogiken (in Beziehung zu OWL)
- Semantik von OWL
Wichtigkeit des Modellbegriffs
- Aktuelle Forschungsthemen um fortgeschrittene Modellierungsmöglichkeiten

Literatur

- Pascal Hitzler, Markus Krötzsch, Sebastian Rudolph, York Sure. **Semantic Web – Grundlagen.** Springer 2008. (ISBN 9783540339939)
- Steffen Staab, Rudi Studer (Editors). **Handbook on Ontologies.** Springer 2003 (ISBN 3540408347).
- Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, Peter Patel-Schneider (eds.), **The Description Logic Handbook.** Cambridge University Press, 2007. (ISBN 9780521781763)