Semantic Web Technologies II
SS 2009

20.05.2009

OWL - Semantik

Dr. Sudhir Agarwal
Dr. Stephan Grimm
Dr. Peter Haase
PD Dr. Pascal Hitzler
Denny Vrandecic

Übersicht

- OWL als Beschreibungslogik
- Modelltheoretische Semantik von DLs
- OWL Modellierungskonstrukte (semantisch)

Übersicht

- OWL als Beschreibungslogik
- Modelltheoretische Semantik von DLs
- OWL Modellierungskonstrukte (semantisch)

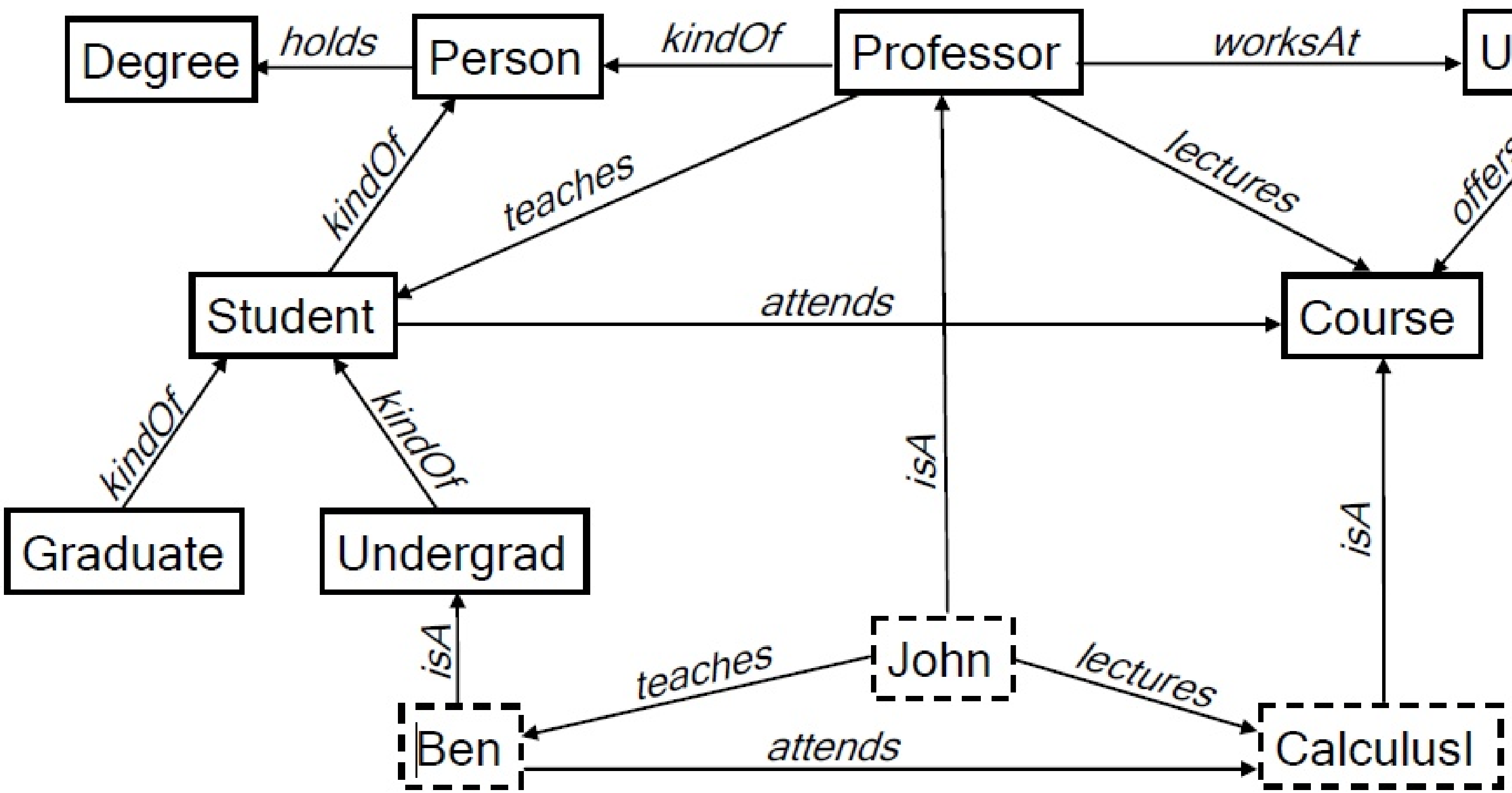
OWL als DL

- OWL-Varianten entsprechen semantisch DLs
  - OWL DL -> *SHOIN(D)*
  - OWL Lite -> *SHIF(D)*
  - OWL DL 2 -> *SROIQ(D)*

$\left.\begin{array}{l}\textit{SHOIN(D)}\\ \textit{SHIF(D)}\\ \textit{SROIQ(D)}\end{array}\right\} \supset \mathcal{ALC}$

- Beschreibungslogiken
  - engl. description logic (DL)
  - Fragmente der Prädikatenlogik 1. Stufe (FOL – first order logic)
  - Wissensrepresentationsformalismus für semantische Netzwerke

DLs als Formalisierung Semantischer Netzwerke



Formalisierung durch Prädikatenlogik

- Präzise Aussagen durch FOL-Formeln
  - Beispiel: Beziehungen zwischen Konzepten

$\boxed{Professor} \xrightarrow{lectures} \boxed{Course}$

$$\forall x, y : (lectures(x, y) \rightarrow Professor(x) \wedge Course(y))$$
$$\forall x : \exists y : (Professor(x) \rightarrow Course(y) \wedge lectures(x, y))$$

- Abbildung von Netzwerken auf FOL-Konstrukte
  - Klassen -> unäre Prädikate (Knoten im Graph)
  - Relationen -> binäre Prädikate (Kanten im Graph)
  - Individuen -> Konstanten (Knoten im Graph)

DLs als Fragment der Prädikatenlogik

- Ausgelegt auf Semantische Netzwerke
  - Klassen/Konzepte
    – DL: *Professor* , FOL: *Professor*(x)
  - Rollen/Relationen
    – DL: *lectures* , FOL: *lectures*(x,y)
  - Individuen
    – DL: *John* , FOL: *lectures*(*John*, *CalculusI*)
- Teile von Netzwerken als Axiome (variablen-frei)
  – DL: $\mathit{Professor} \sqsubseteq \mathit{Staff} \sqcap \exists \mathit{lectures}.\mathit{Course}$
  – FOL: $\forall x: \mathit{Professor}(x) \rightarrow \mathit{Staff}(x) \wedge \exists y: \mathit{lectures}(x,y)$

OWL Ontologie als DL Wissensbasis

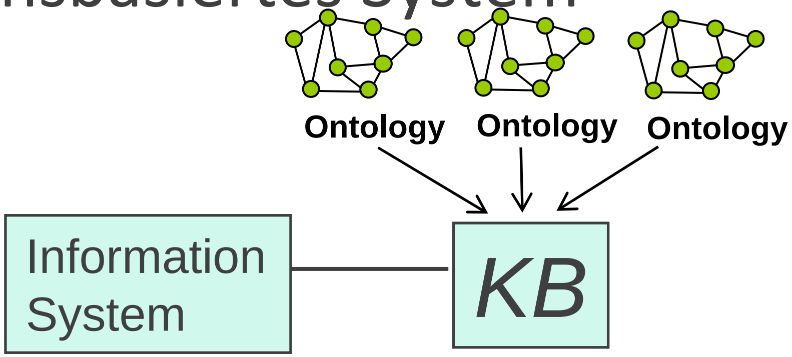
- Prädikatenlogik
  - Wissensbasis ist eine Menge von FOL-Formeln
- Beschreibungslogik
  - Gewisse FOL-Formeln sind DL *Axiome*
  - *Wissensbasis* KB ist Menge von Axiomen
  - *Signatur* von KB enthält Namen aller Klassen, Relationen, Individuen

Schlußfolgern (Reasoning)

- Unterscheidung von Wissen
  - Explizites (was wurde ausgesagt)
  - Implizites (was folgt)
- Schlußfolgern (Reasoning)
  - Implizites Wissen explizit machen
- Arten des Schlußfolgerns (mit DLs)
  - Deduktion
    – Ableiten impliziter Aussagen (logische Konsequenz)
  - Verifikation
    – Prüfung von Wissensbasen auf Konsistenz

Verwendung von Wissensbasen

- Wissensbasiertes System



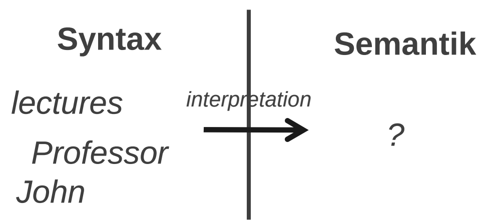
- Tell-Ask-Interface
  - Wissensbasis *KB* = { *Professor* $\sqsubseteq$ *Staff* }
  - `tell`-Operation fügt explizites Wissen hinzu
    – `tell`(*KB*, *Professor*(*John*))
  - `ask`-Operation fragt nach implizitem/explizitem Wissen
    – `ask`(*KB*, *Staff*(*John*)) -> true

Übersicht

- OWL als Beschreibungslogik
- Modelltheoretische Semantik von DLs
- OWL Modellierungskonstrukte (semantisch)
- Komplexere Modellierungsbeispiele in DL

Modelltheorie zu DLs

- Semantik
  - Assoziiert mit „Bedeutung“
- Formalisierung intuitiver Konzepte
  - Klassen, Individuen, Instanziierung, Subsumption, …
  - Interpretation der syntaktischen DL-Konstrukte durch Mengen und Relationen



- Zentrale Begriffe
  - Semantik: Interpretation, Modell
  - Inferenz: Erfüllbarkeit, logische Konsequenz

DL Syntax - Nomenklatur

name	*symbol*	*syntax*	*description*
top concept	$\mathcal{AL}$	$\top$	class of all objects
bottom concept		$\bot$	empty class
atomic concept negation		$\neg A$	complement of named class
concept conjunction		$C \sqcap D$	intersection of classes
universal restriction		$\forall r.C$	objects whose values for r all are in C
limited existential restr.		$\exists r$	objects with some value for r
concept disjunction	$\mathcal{U}$	$C \sqcup D$	union of classes
existential restriction	$\mathcal{E}$	$\exists r.C$	objects with some value for r in C
concept negation	$\mathcal{C}$	$\neg C$	complement of complex class
min cardinality restriction	$\mathcal{N}$	$\geq n\, r$	objects with at least n values for r
max cardinality restriction		$\leq n\, r$	objects with at most n values for r
qualified min card. restr.	$\mathcal{Q}$	$\geq n\, r.C$	objects w. at least n values for r in C
qualified max card. restr.		$\leq n\, r.C$	objects w. at most n values for r in C
nominal	$\mathcal{O}$	$\{o\}$	singleton class with individual o
inverse role	$\mathcal{I}$	r^-	relation with inverse direction
	$\mathcal{S}$		$\mathcal{ALC}$ with transitive roles
role inclusion	$\mathcal{H}$	$r \sqsubseteq s$	role hierarchies
concrete domains	$(\mathbf{D})$		datatype properties and predicates

Verschiedene DLs – Syntaktische Konstrukte

	Concepts	
$\mathcal{ALC}$	Atomic	A, B
	Not	¬C
	And	C ⊓ D
	Or	C ⊔ D
	Exists	∃R.C
	For all	∀R.C
$\mathcal{Q}$ ($\mathcal{N}$)	At least	≥n R.C (≥n R)
	At most	≤n R.C (≤n R)
$\mathcal{O}$	Nominal	$\{i_1,\ldots,i_n\}$

	Roles	
	Atomic	R
$\mathcal{I}$	Inverse	R^-

Ontology (=Knowledge Base)

	Concept Axioms (TBox)	
	Subclass	C ⊑ D
	Equivalent	C ≡ D

	Role Axioms (RBox)	
$\mathcal{H}$	Subrole	R ⊑ S
$\mathcal{S}$	Transitivity	Trans(S)

Assertional Axioms (ABox)	
Instance	C(a)
Role	R(a,b)
Same	a = b
Different	a ≠ b

ALC - Formale Syntax

- Wissensbasis
  - $KB =$ Tbox $\cup$ ABox
- Terminologisches Wissen
  - TBox = $\{ \ldots \underbrace{C \sqsubseteq D}_{\text{inclusion}}, \underbrace{C \equiv D}_{\text{equivalence}}, \ldots \}$ *C,D* – Klassen
- Faktenwissen
  - ABox = $\{ \ldots \underbrace{C(a)}_{\text{concept assertion}}, \underbrace{r(a,b)}_{\text{role assertion}}, \ldots \}$ *C* - Klasse, *r* - Rolle, *a*,*b* Individuen
- Komplexe Klassen

$$C, D \quad \longrightarrow \quad A \mid \top \mid \bot \mid C \sqcap D \mid C \sqcup D \mid \neg C \mid \forall\, r.C \mid \exists\, r.C$$

C,D – komplexe Klassen, *A* – atomare Klasse, *r* - Rolle

Interpretationen

- Ausgehend von Namen für
  - Individuen N_I, Klassen N_C, Rollen N_r
- Interpretation $I = (\Delta^I, \cdot^I)$ besteht aus
  - einer nicht-leeren Menge Δ^I, genannt Domäne und
  - einer Funktion $\cdot^I$ die abbildet von
    - Individuen auf Domänenelemente $a^I \in \Delta^I$
    - atomare Klassen auf Teilmengen der Domäne $A^I \subseteq \Delta^I$
    - atomare Rollen auf binäre Relationen über die Domäne $r^I \subseteq \Delta^I \times \Delta^I$
  - *I* entspricht konkreter Situation von Objekten in Δ^I und deren Zuordnung zu Klassen und Rollen

Interpretationen - Beispiel

- $I = (\Delta^I, \cdot^I)$

Individuen

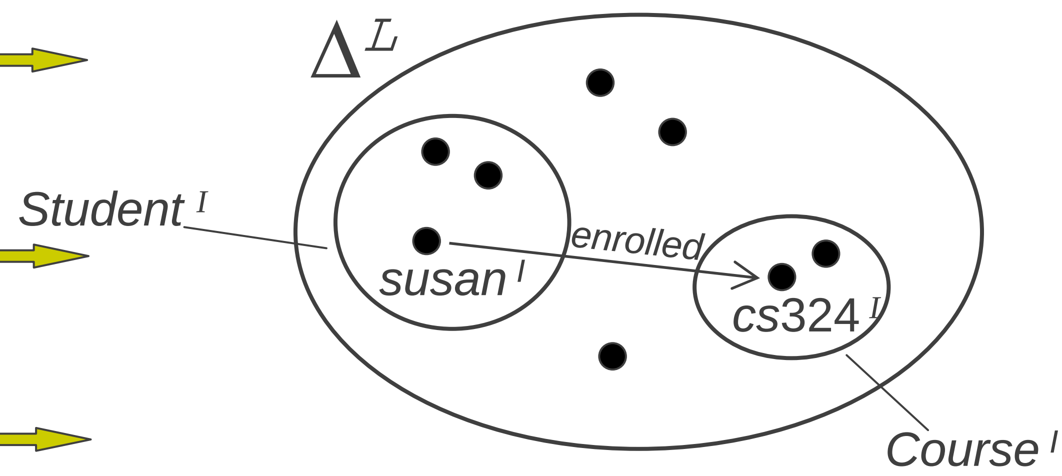
cs324 *susan*

Klassen

Student *Course*

Rolle

$cs324 \xrightarrow{enrolled} susan$



Interpretationen - Überlegungen

- Wie sehen Beispiele für eine Interpretation von folgenden Syntaxelementen aus?
  - Professor, Course $\in N_C$
  - John $\in N_I$
  - Lectures $\in N_r$
    -> konstruiere I

- Wieviele Interpretationen gibt es für gegebene Syntaxelemente?

Interpretation Komplexer Klassen ($\mathcal{ALC}$)

- Erweiterung der Interpretationsfunktion

$$\top^\mathcal{I} = \Delta^\mathcal{I} \qquad , \quad \bot^\mathcal{I} = \emptyset$$

$$\begin{aligned}
(C \sqcap D)^\mathcal{I} &= C^\mathcal{I} \cap D^\mathcal{I} \\
(C \sqcup D)^\mathcal{I} &= C^\mathcal{I} \cup D^\mathcal{I} \\
(\neg C)^\mathcal{I} &= \Delta^\mathcal{I} \setminus C^\mathcal{I} \\
(\forall\, r.C)^\mathcal{I} &= \{a \in \Delta^\mathcal{I} \mid \forall b.(a,b) \in r^\mathcal{I} \to b \in C^\mathcal{I}\} \\
(\exists\, r.C)^\mathcal{I} &= \{a \in \Delta^\mathcal{I} \mid \exists b.(a,b) \in r^\mathcal{I} \wedge b \in C^\mathcal{I}\}
\end{aligned}$$

Modelle

- Ausgehend von einer Wissensbasis *KB*
- Erfüllung von Axiomen durch Interpretationen
  - ABox-Axiome
    - $\mathcal{I}$ erfüllt $C(a)$ wenn $a^{\mathcal{I}} \in \mathrm{C}^{\mathcal{I}}$
    - $\mathcal{I}$ erfüllt $r(a,b)$ wenn $(a,b)^{\mathcal{I}} \in \mathrm{r}^{\mathcal{I}}$
  - TBox-Axiome
    - $\mathcal{I}$ erfüllt $C \sqsubseteq D$ wenn $C^{\mathcal{I}} \subseteq \mathrm{D}^{\mathcal{I}}$
    - $\mathcal{I}$ erfüllt $C \equiv D$ wenn $C^{\mathcal{I}} = \mathrm{D}^{\mathcal{I}}$
- Modellbegriff
  - $\mathcal{I}$ ist Modell von *KB* wenn $\mathcal{I}$ alle Axiome in *KB* erfüllt

Modelle - Beispiel

- $I = (\Delta^I, \cdot^I)$

Individuen

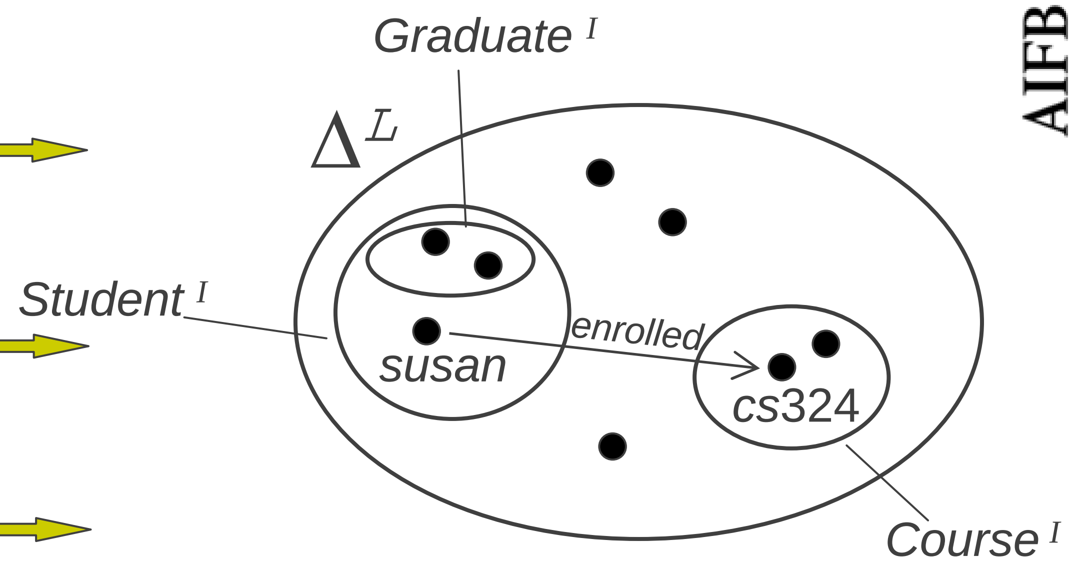
cs324 *susan*

Klassen

Student *Course* *Graduate*

Rolle

$cs324 \xrightarrow{enrolled} susan$



- I ist ein Modell von *KB* wenn es die Axiome in *KB* erfüllt

$$KB = \{ Graduate \sqsubseteq Student,\ Student(susan),\ enrolled(cs324, susan)\}$$

Überlegungen zu Modellen

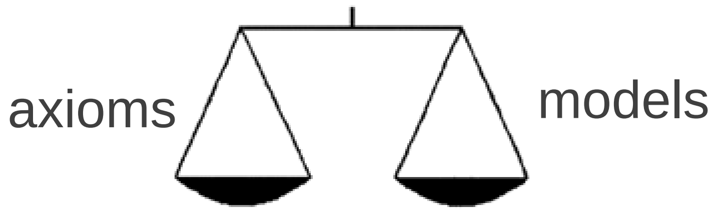
- Wie sehen Beispiele für Modelle der folgenden Wissensbasis aus?
  - KB = { $\mathit{Professor}(\mathit{John})$, $\mathit{Student}(\mathit{Ben})$, $\mathit{teaches}(\mathit{John},\mathit{Ben})$, $\forall \mathit{teaches}.\mathit{Graduate}(\mathit{John})$, $\mathit{Graduate} \sqsubseteq \mathit{Student}$, $\mathit{Undergrad} \sqsubseteq \mathit{Student}$, $\mathit{Graduate} \sqsubseteq \neg\mathit{Undergrad}$ }

- Wieviele Modelle gibt es für eine Wissensbasis?

- Gibt es mehr Modelle oder mehr Interpretationen?

Axiome vs. Modelle

- Axiome einer Wissensbasis schränken die möglichen Modelle ein (ausfiltern von Interpretationen)
- Je mehr Axiome desto weniger Modelle



- Was sind die Extreme?
  - *KB* = { … } ?

DL Inferenzprobleme – Übersicht

- Erfüllbarkeit der Wissensbasis
  - Ist KB erfüllbar? (Ist *KB* widerspruchsfrei?)
- Erfüllbarkeit von Klassen
  - Ist *C* erfüllbar bezüglich *KB*? (Darf *C* Individuen enthalten?)
- Logische Konsequenz
  - Folgt ein Axiom α aus einer Wissensbasis, $KB \models \alpha$?
    (Ist α eine logische Konsequenz des Wissens in *KB*?)
  - α kann sein
    - $C \sqsubseteq D$ Subsumption (bzw. $C \equiv D$ Equivalenz)
    - $C(a)$ Instanziierung
    - $r(a,b)$ Rollenzuweisung

Erfüllbarkeit

- Erfüllbarkeit der Wissensbasis
  - Definition der Erfüllbarkeit über Modellexistenz
    „Eine Wissensbasis *KB* ist genau dann erfüllbar wenn sie ein Modell hat."
  - Logischer Widerspruch entspricht dem Fehlen eines Modells
- Erfüllbarkeit einer Klasse
  - Definition der Erfüllbarkeit über Existenz von Instanzen
    „Eine Klasse C ist genau dann erfüllbar in Bezug auf eine Wissensbasis KB wenn KB ein Modell $\mathcal{I}$ hat mit $C^{\mathcal{I}} \neq \emptyset$"
  - Instanziierung unerfüllbarer Klassen führt zu logischem Widerspruch

Beispiel: Erfüllbarkeit

- Gegeben sei folgende Wissensbasis
  - $KB = \{\ \mathit{Professor} \sqsubseteq \exists \mathit{lectures}.\mathrm{Course},$
    $\mathit{LazyProfessor} \equiv \mathit{Professor} \sqcap \forall \mathit{lectures}.\bot\ \}$

- Ist KB erfüllbar?

- Ist die Klasse *LazyProfessor* erfüllbar?

- Was passiert nach Hinzufügen des Axioms *LazyProfessor*(*John*) ?

Logische Konsequenz

- Engl. *Entailment*, geschrieben $\models$
- Definition über Modellbegriff

  „Ein Axiom α ist eine logische Konsequenz aus einer Wissensbasis *KB*, geschr. $KB \models \alpha$, genau dann wenn alle Modelle von *KB* auch α erfüllen"

- Subsumption
  - $KB \models C \sqsubseteq D$ wenn $C^I \subseteq D^I$ für alle Modelle I von KB
- Instanziierung
  - $KB \models C(a)$ wenn $a^I \in C^I$ für alle Modelle I von KB

Beispiel: Logische Konsequenz

- Gegeben sei folgende Wissensbasis
  - $KB = \{ \mathit{Professor} \sqsubseteq \mathit{Lecturer}, \mathit{Lecturer} \sqsubseteq \mathit{Staff}$
    $\neg\mathit{Staff}(\mathit{Simon}), (\text{Professor} \sqcup \text{Student})(\mathit{Ben})\}$

- Gilt $KB \models \mathit{Professor} \sqsubseteq \mathit{Staff}$?

- Gilt $KB \models \neg\mathit{Professor}(\mathit{Simon})$?

- Gilt $KB \models \mathit{Staff}(\mathit{Ben})$?
  Gilt $KB \models \neg\mathit{Staff}(\mathit{Ben})$?

Rückführung von Inferenzproblemen

- Wichtige FOL-Equivalenz
  - $KB \models \alpha \quad \Leftrightarrow \quad KB \cup \{\neg\alpha\}$ ist unerfüllbar
  - Ermöglicht Zurückführen aller DL-Inferenzprobleme auf (Un-)Erfüllbarkeit der Wissensbasis
- Klassenerfüllbarkeit
  - C ist erfüllbar bezüglich $KB \Leftrightarrow KB \cup \{C(x)\}$ ist erfüllbar (x ist neues Individuum)
- Instanziierung
  - $KB \models C(a) \Leftrightarrow KB \cup \{\neg C(a)\}$ ist unerfüllbar
- Subsumption
  - $KB \models C \sqsubseteq D \Leftrightarrow KB \cup \{C \sqcap \neg D\}$ ist unerfüllbar

Praktische Inferenzprobleme

- Konsistenz einer Ontologie
  - Über Erfüllbarkeit der entspr. Wissensbasis
- Klassenkonsistenz
  - Über Erfüllbarkeit der entspr. Klasse
- Klassifikation
  - Subsumptionsbeziehungen zwischen allen atomaren Klassen
- Klassenzugehörigkeit
  - Über Instanziierung $KB \models C(a)$?
- Instanzgenerierung (Retrieval)
  - Instanziierung für alle Individuen $\{x : KB \models C(x)\}$

SHOIN-Syntax - Klassenkonstruktoren

Constructor	DL Syntax	Example	FOL Syntax
intersectionOf	$C_1 \sqcap \ldots \sqcap C_n$	Human $\sqcap$ Male	$C_1(x) \wedge \ldots \wedge C_n(x)$
unionOf	$C_1 \sqcup \ldots \sqcup C_n$	Doctor $\sqcup$ Lawyer	$C_1(x) \vee \ldots \vee C_n(x)$
complementOf	$\neg C$	$\neg$Male	$\neg C(x)$
oneOf	$\{x_1\} \sqcup \ldots \sqcup \{x_n\}$	{john} $\sqcup$ {mary}	$x = x_1 \vee \ldots \vee x = x_n$
allValuesFrom	$\forall P.C$	$\forall$hasChild.Doctor	$\forall y.P(x,y) \rightarrow C(y)$
someValuesFrom	$\exists P.C$	$\exists$hasChild.Lawyer	$\exists y.P(x,y) \wedge C(y)$
maxCardinality	$\leqslant nP$	$\leqslant$1hasChild	$\exists^{\leqslant n} y.P(x,y)$
minCardinality	$\geqslant nP$	$\geqslant$2hasChild	$\exists^{\geqslant n} y.P(x,y)$

Beliebig komplexes Schachteln von Konstruktoren erlaubt:

Person $\sqcap$ $\forall$hasChild.(Doctor $\sqcup$ $\exists$hasChild.Doctor)

SHOIN-Syntax - Axiome

Axiom	DL Syntax	Example
subClassOf	$C_1 \sqsubseteq C_2$	Human $\sqsubseteq$ Animal $\sqcap$ Biped
equivalentClass	$C_1 \equiv C_2$	Man $\equiv$ Human $\sqcap$ Male
disjointWith	$C_1 \sqsubseteq \neg C_2$	Male $\sqsubseteq$ $\neg$Female
sameAs	$\{x_1\} \equiv \{x_2\}$	{President_Bush} $\equiv$ {G_W_Bush}
differentFrom	$\{x_1\} \sqsubseteq \neg\{x_2\}$	{john} $\sqsubseteq$ $\neg${peter}
subPropertyOf	$P_1 \sqsubseteq P_2$	hasDaughter $\sqsubseteq$ hasChild
equivalentProperty	$P_1 \equiv P_2$	cost $\equiv$ price
inverseOf	$P_1 \equiv P_2^-$	hasChild $\equiv$ hasParent$^-$
transitiveProperty	$P^+ \sqsubseteq P$	ancestor$^+$ $\sqsubseteq$ ancestor
functionalProperty	$\top \sqsubseteq \leqslant 1 P$	$\top \sqsubseteq \leqslant$1hasMother
inverseFunctionalProperty	$\top \sqsubseteq \leqslant 1 P^-$	$\top \sqsubseteq \leqslant$1hasSSN$^-$

SHOIN – Semantik Komplexer Ausdrücke

$$
\begin{array}{rcl}
C, C_i & \longrightarrow & A \mid \bot \mid \top \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \exists r.C \mid \forall r.C \\
 & & \mid \geq n\, r \mid \leq n\, r \mid \{a_1, \ldots, a_n\} \\
r & \longrightarrow & s \mid s^-
\end{array}
$$

$$
\begin{array}{c}
\top^\mathcal{I} = \Delta^\mathcal{I} \quad , \quad \bot^\mathcal{I} = \emptyset \\
A^\mathcal{I} \subseteq \Delta^\mathcal{I} \quad , \quad r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}
\end{array}
$$

$$
\begin{array}{rcl}
(C_1 \sqcap C_2)^\mathcal{I} & = & C_1^\mathcal{I} \cap C_2^\mathcal{I} \\
(C_1 \sqcup C_2)^\mathcal{I} & = & C_1^\mathcal{I} \cup C_2^\mathcal{I} \\
(\neg C)^\mathcal{I} & = & \Delta^\mathcal{I} \setminus C^\mathcal{I} \\
(\forall r\,.C)^\mathcal{I} & = & \{x \in \Delta^\mathcal{I} \mid \forall y.(x,y) \in r^\mathcal{I} \to y \in C^\mathcal{I}\} \\
(\exists r\,.C)^\mathcal{I} & = & \{x \in \Delta^\mathcal{I} \mid \exists y.(x,y) \in r^\mathcal{I} \wedge y \in C^\mathcal{I}\} \\
(\geq n\, r)^\mathcal{I} & = & \{x \in \Delta^\mathcal{I} \mid \#\{y \in \Delta^\mathcal{I} \mid (x,y) \in r^\mathcal{I}\} \geq n\} \\
(\leq n\, r)^\mathcal{I} & = & \{x \in \Delta^\mathcal{I} \mid \#\{y \in \Delta^\mathcal{I} \mid (x,y) \in r^\mathcal{I}\} \leq n\} \\
(\{a_1, \ldots, a_n\})^\mathcal{I} & = & \{a_1^\mathcal{I}, \ldots, a_n^\mathcal{I}\} \\
(s^-)^\mathcal{I} & = & \{(y,x) \mid (x,y) \in s^\mathcal{I}\}
\end{array}
$$

Übersicht

- OWL als Beschreibungslogik
- Modelltheoretische Semantik von DLs
- OWL Modellierungskonstrukte (semantisch)
- Komplexere Modellierungsbeispiele in DL

OWL - Konjunktion

- OWL Sprachkonstrukt
  - `owl:intersection`
  - Bildet Schnitt von Klassen (sprachlich: und)
- DL Beispiel
  - Syntax
    - $\mathit{Professor} \sqsubseteq \mathit{Staff} \sqcap \mathit{Lecturer}$
  - *Semantik*
    - $\mathit{Professor}^I \subseteq \mathit{Staff}^I \cap \mathit{Lecturer}^I$
  - Intuition
    - "Professoren sind gleichzeitig Angestellte und Dozenten"

OWL - Disjunktion

- OWL Sprachkonstrukt
  - `owl:unionOf`
  - Bildet Vereinigung von Klassen (sprachlich: oder)
- DL Beispiel
  - Syntax
    - $\mathit{Lecturer} \sqsubseteq \mathit{Staff} \sqcup \mathit{External}$
  - *Semantik*
    - $\mathit{Lecturer}^I \subseteq \mathit{Staff}^I \cup \mathit{External}^I$
  - Intuition
    - "Dozenten sind entweder Angestellte oder Externe"

OWL – Komplement (Negation)

- OWL Sprachkonstrukt
  - `owl:complementOf`
  - Bildet Komplement von Klassen (sprachlich: nicht)
- DL Beispiel
  - Syntax
    - $Graduate \sqsubseteq \neg Undergrad$
  - *Semantik*
    - $Graduate^I \subseteq \Delta^I \setminus Undergrad^I$
  - Intuition
    - "Graduates sind keine Undergraduates"

OWL - Quantoren

- OWL Sprachkonstrukte
  - `owl:someValuesFrom, owl:allValuesFrom`
  - Beschränkt eine Rolle für eine Klasse nach Quantität
- DL Beispiel
  - Syntax
    - – $\mathit{Professor} \sqsubseteq \exists \mathit{lectures.Course}$
  - *Semantik*
    - – $\mathit{Professor}^{\mathcal{I}} \subseteq \{x \in \Delta^{\mathcal{I}} \mid \exists y: (x,y) \in r^{\mathcal{I}} \wedge y \in \mathit{Course}^{\mathcal{I}}\}$
  - Intuition
    - – “Jeder Professor liest einen Kurs”

OWL - Nominals

- OWL Sprachkonstrukt
  - `owl:oneOf`
  - Definiert Klassen durch Aufzählung ihrer Instanzen
- DL Beispiel
  - Syntax
    - $\forall lectures.\{cs1,cs2,cs3\}$ (John)
  - *Semantik*
    - $John^I \in \{x \in \Delta^I \mid \forall y\!: (x,y) \in r^I \rightarrow y \in \{cs1,cs2,cs3\}\}$
  - Intuition
    - "John liest nichts anderes als cs1, cs2 und cs2"

OWL - Kardinalitäten

- OWL Sprachkonstrukt
  - `owl:maxCardinality, owl:minCardinality`
  - Beschränkt Kardinalität einer Rolle für Klassen
- DL Beispiel
  - Syntax
    – $\mathit{Professor} \sqsubseteq\ \geq 2\ \mathit{lectures}$
  - *Semantik*
    – $\mathit{Professor}^{\mathcal{I}} \subseteq \{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x,y) \in \mathit{lectures}^{\mathcal{I}}\} \geq 2\}$
  - Intuition
    – “Professoren halten mindestens zwei Kurse”

OWL – Individuen(un)gleichheit

- OWL Sprachkonstrukt
  - `owl:sameAs, owl:differentFrom`
  - standardmäßig keine „unique name assumption“ in OWL
- DL Beispiel
  - Syntax
    – *Dean ≠ John, CommitteeMember(Dean)*
  - *Semantik*
    – $\mathrm{Dean}^{I} \in \mathit{CommitteeMember}^{I}$, $\mathrm{John}^{I} \notin \mathit{CommitteeMember}^{I}$
  - Intuition
    – “Dean und John sind verschiedene Individuen”

OWL – Rollenaxiome

- Rollenhierarchien
  - `rdfs:subPropertyOf`
  - $teaches \sqsubseteq knows$, $(x,y) \in \text{teaches}^{\mathcal{I}} \rightarrow (x,y) \in \text{knows}^{\mathcal{I}}$
- *Inverse Rollen*
  - `owl:inverseOf`
  - $isTaughtBy \equiv teaches^{-}$, $(x,y) \in \text{isTaughtBy} \leftrightarrow (y,x) \in \text{teaches}^{\mathcal{I}}$
- *Transitive Rollen*
  - `owl:TransitiveProperty`
  - $Trans(isReqFor)$, $(x,y) \in isReqFor^{\mathcal{I}} \wedge (y,z) \in isReqFor^{\mathcal{I}} \rightarrow (x,z) \in \text{isReqFor}^{\mathcal{I}}$

OWL – Rollenaxiome

- *Symmetrische Rollen*
  - `owl:SymmetricProperty`
  - $knows \equiv knows^{-}$ (Rollenequivalenz mit Inversen)

- *Funktionale Rollen*
  - `owl:FunctionalProperty`
  - $\top \sqsubseteq \leq 1\, isEmployedAt$ (Klasseninklusion mit Kardinalität)

- *Invers-Funktionale Rollen*
  - `owl:InverseFunctionalProperty`
  - $\top \sqsubseteq \leq 1\, lectures^{-}$ (Klasseninklusion mit Kardinalität auf inverse Rolle)

Ausblick

- Übung
  - Modellieren von Ontologien in OWL
  - Verwendung von Protégé 4
  - Praktische Beispiele zum Verständnis von logischen Konsequenzen