OWL 1.1 – Syntax und Semantik

Pascal Hitzler Markus Krötzsch Sebastian Rudolph

Institut AIFB · Universität Karlsruhe

Semantic Web Technologies 1 (WS07/08)

30. Januar 2008

http://semantic-web-grundlagen.de

AIFB

Outline

AIFB

Semantic Web Technologies 1

1. Einleitung und Ausblick
2. XML und URIs
3. Einführung in RDF
4. RDF Schema
5. Logik – Grundlagen
6. Semantik von RDF(S)
7. OWL – Syntax und Intuition
8. OWL – Semantik und Reasoning
9. SPARQL – Syntax und Intuition
10. Semantik von SPARQL
11. Konjunktive Anfragen und Regelsprachen
12. OWL 1.1 – Syntax und Semantik ($\rightarrow$ Webseite)
13. Bericht aus der Praxis
14. Semantic Web – Anwendungen

Literatur zu dieser Vorlesung online siehe $\rightarrow$ Vorlesungswebseite

AIFB

Rückblick: Erweiterungen von OWL

OWL für viele Aufgaben noch zu schwach (siehe $\rightarrow$ letzte Vorlesung).

- OWL als Anfragesprache ungenügend
  $\rightsquigarrow$ Konjunktive Anfragen, SPARQL für OWL
- OWL als Ontologiesprache ungenügend
  $\rightsquigarrow$ Prädikatenlogische Regelerweiterungen, SWRL
- OWL als Programmiersprache ungenügend
  $\rightsquigarrow$ Logikprogrammierung im Semantic Web
  ($\rightarrow$ Vorlesung SWebT2)

AIFB

Rückblick: Erweiterungen von OWL

OWL für viele Aufgaben noch zu schwach (siehe $\rightarrow$ letzte Vorlesung).

- OWL als Anfragesprache ungenügend
  $\rightsquigarrow$ Konjunktive Anfragen, SPARQL für OWL
- OWL als Ontologiesprache ungenügend
  $\rightsquigarrow$ Prädikatenlogische Regelerweiterungen, SWRL
- OWL als Programmiersprache ungenügend
  $\rightsquigarrow$ Logikprogrammierung im Semantic Web
  ($\rightarrow$ Vorlesung SWebT2)

Sollte auch der OWL-Standard selbst erweitert werden?

$\rightsquigarrow$ OWL 1.1

Entwicklung von OWL 1.1

OWL 1.1 als „nächste Version“ von OWL

Erweiterungen aufgrund von Praxiserfahrung mit OWL 1.0:

- zusätzliche Ausdrucksstärke durch neue ontologische Axiome
- nicht-logische Erweiterunen (Syntax, Metadaten, . . .)
- Überarbeitung der OWL-Varianten (Lite/DL/Full)

Zielstellungen:

- weitestgehende Kompatibilität zum existierenden OWL-Standard
- Erhaltung der Entscheidbarkeit von OWL DL
- Behebung von Problemen im OWL-1.0-Standard

AIFB

Outline

AIFB

Von $\mathcal{SHOIN}$ zu $\mathcal{SROIQ}$

OWL DL basiert auf Beschreibungslogik $\mathcal{SHOIN}(D)$:

- Axiome:
  - Tbox: Subklassenbeziehungen $C \sqsubseteq D$
  - Rbox: Subrollenbeziehungen $R \sqsubseteq S$ ($\mathcal{H}$), Inverse Rollen R^- ($\mathcal{I}$), Transitivität
  - Abox: Fakten zu Klassen $C(a)$, Rollen $R(a, b)$, und Gleichheit $a \approx b$ bzw. $a \not\approx b$
- Klassenkonstruktoren:
  - Konjunktion $C \sqcap D$, Disjunktion $C \sqcup D$, Negation $\neg C$ von Klassen
  - Rollenrestriktionen: universell $\forall R.C$ und existenziell $\exists R.C$
  - Zahlenrestriktionen ($\mathcal{N}$): $\leq n\, R$ und $\geq n\, R$ (n nicht-negative Zahl)
  - Nominale ($\mathcal{O}$): $\{a\}$
- Datentypen (D)

Erweiterung in OWL 1.1 zu $\mathcal{SROIQ}(D)$

AIFB

Abox

$\mathcal{SHOIN}$ unterstützt verschiedene Abox-Fakten:

- Klassenzugehörigkeit $C(a)$ (C komplexe Klasse),
- Sonderfall: negierte Klassenzugehörigkeit $\neg C(a)$ (C komplexe Klasse),
- Gleichheit $a \approx b$,
- Ungleichheit $a \not\approx b$
- Rollenbeziehungen $R(a, b)$

AIFB

Abox

$\mathcal{SHOIN}$ unterstützt verschiedene Abox-Fakten:

- Klassenzugehörigkeit $C(a)$ (C komplexe Klasse),
- Sonderfall: negierte Klassenzugehörigkeit $\neg C(a)$ (C komplexe Klasse),
- Gleichheit $a \approx b$,
- Ungleichheit $a \not\approx b$
- Rollenbeziehungen $R(a, b)$
- *negierte Rollenbeziehungen?*

AIFB

Abox

$\mathcal{SHOIN}$ unterstützt verschiedene Abox-Fakten:

- Klassenzugehörigkeit $C(a)$ (C komplexe Klasse),
- Sonderfall: negierte Klassenzugehörigkeit $\neg C(a)$ (C komplexe Klasse),
- Gleichheit $a \approx b$,
- Ungleichheit $a \not\approx b$
- Rollenbeziehungen $R(a, b)$
- *negierte Rollenbeziehungen?*

$\rightsquigarrow$ $\mathcal{SROIQ}$ erlaubt auch negierte Rollen in der Abox: $\neg R(a, b)$

Zahlenrestriktionen

$\mathcal{SHOIN}$ unterstützt nur einfache Zahlenrestriktionen ($\mathcal{N}$):

$$\text{Person} \sqcap \geq 3\, \text{hatKind}$$

„Klasse aller Personen mit 3 oder mehr Kindern.“

Zahlenrestriktionen

$\mathcal{SHOIN}$ unterstützt nur einfache Zahlenrestriktionen ($\mathcal{N}$):

$$\text{Person} \sqcap \geq 3\, \text{hatKind}$$

„Klasse aller Personen mit 3 oder mehr Kindern.“

$\rightsquigarrow \mathcal{SROIQ}$ erlaubt auch qualifizierte Zahlenrestriktionen ($\mathcal{Q}$):

$$\text{Person} \sqcap \geq 3\, \text{hatKind}.(\text{Frau} \sqcap \text{Professor})$$

„Klasse aller Personen mit 3 oder mehr Töchtern, die Professoren sind.“

Das Konzept Self

Modellierungsaufgabe: „Jeder Mensch kennt sich selbst.“

Das Konzept Self

Modellierungsaufgabe: „Jeder Mensch kennt sich selbst.“

- $\mathcal{SHOIN}$:

$$\text{kennt}(\text{tom}, \text{tom}) \quad \text{kennt}(\text{tina}, \text{tina}) \quad \text{kennt}(\text{udo}, \text{udo}) \quad \ldots$$

Das Konzept Self

Modellierungsaufgabe: „Jeder Mensch kennt sich selbst.“

- $\mathcal{SHOIN}$:

  $$\text{kennt}(\text{tom}, \text{tom}) \quad \text{kennt}(\text{tina}, \text{tina}) \quad \text{kennt}(\text{udo}, \text{udo}) \quad \ldots$$

  $\rightsquigarrow$ nicht allgemein anwendbar

- $\mathcal{SROIQ}$: spezieller Ausdruck Self

  $$\text{Mensch} \sqsubseteq \exists \text{kennt}.\text{Self}$$

Rollenaxiome und die universelle Rolle

$\mathcal{SROIQ}$ bietet zusätzliche Aussagen über Rollen:

- Tra(R): R ist transitiv (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Tra(liegtIn)

Rollenaxiome und die universelle Rolle

$\mathcal{SROIQ}$ bietet zusätzliche Aussagen über Rollen:

- Tra(R): R ist transitiv (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Tra(liegtIn)
- Sym(R): R ist symmetrisch (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Sym(verwandtMit)

Rollenaxiome und die universelle Rolle

$\mathcal{SROIQ}$ bietet zusätzliche Aussagen über Rollen:

- Tra(R): R ist transitiv (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Tra(liegtIn)
- Sym(R): R ist symmetrisch (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Sym(verwandtMit)
- Ref(R): R ist reflexiv, $(x, x) \in R^{\mathcal{I}}$ für alle Domänenindividuen x
  Beispiel: Ref(kennt)

AIFB

Rollenaxiome und die universelle Rolle

$\mathcal{SROIQ}$ bietet zusätzliche Aussagen über Rollen:

- Tra(R): R ist transitiv (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Tra(liegtIn)
- Sym(R): R ist symmetrisch (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Sym(verwandtMit)
- Ref(R): R ist reflexiv, $(x, x) \in R^{\mathcal{I}}$ für alle Domänenindividuen x
  Beispiel: Ref(kennt)
- Irr(R): R ist irreflexiv, $(x, x) \notin R^{\mathcal{I}}$ für alle Domänenindividuen x
  Beispiel: Irr(hatKind)

AIFB

Rollenaxiome und die universelle Rolle

$\mathcal{SROIQ}$ bietet zusätzliche Aussagen über Rollen:

- Tra(R): R ist transitiv (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Tra(liegtIn)
- Sym(R): R ist symmetrisch (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Sym(verwandtMit)
- Ref(R): R ist reflexiv, $(x, x) \in R^{\mathcal{I}}$ für alle Domänenindividuen x
  Beispiel: Ref(kennt)
- Irr(R): R ist irreflexiv, $(x, x) \notin R^{\mathcal{I}}$ für alle Domänenindividuen x
  Beispiel: Irr(hatKind)
- Dis(R, S): R und S sind disjunkt, $(x, y) \notin R^{\mathcal{I}} \cap S^{\mathcal{I}}$ für alle x, y
  Beispiel: Dis(hatVater, hatSohn)

AIFB

Rollenaxiome und die universelle Rolle

$\mathcal{SROIQ}$ bietet zusätzliche Aussagen über Rollen:

- Tra(R): R ist transitiv (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Tra(liegtIn)
- Sym(R): R ist symmetrisch (definiert wie in $\mathcal{SHOIN}$)
  Beispiel: Sym(verwandtMit)
- Ref(R): R ist reflexiv, $(x, x) \in R^{\mathcal{I}}$ für alle Domänenindividuen x
  Beispiel: Ref(kennt)
- Irr(R): R ist irreflexiv, $(x, x) \notin R^{\mathcal{I}}$ für alle Domänenindividuen x
  Beispiel: Irr(hatKind)
- Dis(R, S): R und S sind disjunkt, $(x, y) \notin R^{\mathcal{I}} \cap S^{\mathcal{I}}$ für alle x, y
  Beispiel: Dis(hatVater, hatSohn)
- Universelle Rolle U: $(x, y) \in U^{\mathcal{I}}$ für alle x, y
  Beispiel: $\top \sqsubseteq \leq 7000000000\, U$.Menschen (nicht empfohlen!)
  $\rightsquigarrow$ U ist vor allem als Gegenstück zu $\top$ sinnvoll, z.B. als Wurzel der Rollenhierarchie in grafischen Editoren

AIFB

Allgemeine Rolleninklusion

„Die Freunde meiner Freunde sind auch meine Freunde.“

$\rightsquigarrow$ Kann in $\mathcal{SHOIN}$ ausgedrückt werden: hatFreund ist transitiv.

„Die Feinde meiner Freunde sind auch meine Feinde.“

$\rightsquigarrow$ Kann nicht in $\mathcal{SHOIN}$ ausgedrückt werden!

AIFB

Allgemeine Rolleninklusion

„Die Freunde meiner Freunde sind auch meine Freunde.“

$\rightsquigarrow$ Kann in $\mathcal{SHOIN}$ ausgedrückt werden: hatFreund ist transitiv.

„Die Feinde meiner Freunde sind auch meine Feinde.“

$\rightsquigarrow$ Kann nicht in $\mathcal{SHOIN}$ ausgedrückt werden!

Rolleninklusion

- Rbox-Ausdrücke der Form $R_1 \circ R_2 \circ \ldots \circ R_n \sqsubseteq S$,
  Beispiel: hatFreund $\circ$ hatFeind $\sqsubseteq$ hatFeind
- Semantik: wenn $(x_0, x_1) \in R_1^{\mathcal{I}}$, $(x_1, x_2) \in R_2^{\mathcal{I}}$, $\ldots$, $(x_{n-1}, x_n) \in R_n^{\mathcal{I}}$,
  dann gilt auch $(x_0, x_n) \in S^{\mathcal{I}}$
  Beispiel: wenn $(x, y) \in \text{hatFreund}^{\mathcal{I}}$ und $(y, z) \in \text{hatFeind}^{\mathcal{I}}$,
  dann gilt auch $(x, z) \in \text{hatFeind}^{\mathcal{I}}$

Weitere Beispiele:

teilVon $\circ$ gehört $\sqsubseteq$ gehört hatBruder $\circ$ hatKind $\sqsubseteq$ istOnkelVon

AIFB

Ausdrucksstärke der Rolleninklusion

Wie kompliziert ist Rolleninklusion?

Mit Rboxen kann man formale Sprachen kodieren: (Skizze!)

Grammatik für Sprache der Wörter ab, aabb, aaabbb, ...:

$$\begin{aligned} \mathtt{L} &::= \mathtt{ab} \\ \mathtt{L} &::= \mathtt{aLb} \end{aligned} \quad \text{wird zu Rbox} \quad \begin{aligned} R_a \circ R_b &\sqsubseteq L \\ R_a \circ L \circ R_b &\sqsubseteq L \end{aligned}$$

$\rightsquigarrow \exists L.\top \neq \bot$ ("$\exists L.\top$ notwendig nicht-leer") bedeutet*:
„Es gibt eine Kette aus R_a und R_b, die zur Sprache gehört."

$\rightsquigarrow \exists L_1.\exists L_2^- \neq \bot$ für zwei kodierte Sprachen L_1 und L_2 bedeutet:
„Es gibt ein Wort, das zu L_1 und zu L_2 gehört."

*) bei entsprechender Tbox!

Ausdrucksstärke der Rolleninklusion

Wie kompliziert ist Rolleninklusion?

Mit Rboxen kann man formale Sprachen kodieren: (Skizze!)

Grammatik für Sprache der Wörter ab, aabb, aaabbb, ...:

$$\begin{array}{rcl} \mathtt{L} & ::= & \mathtt{ab} \\ \mathtt{L} & ::= & \mathtt{aLb} \end{array} \quad \text{wird zu Rbox} \quad \begin{array}{r} R_a \circ R_b \sqsubseteq L \\ R_a \circ L \circ R_b \sqsubseteq L \end{array}$$

$\rightsquigarrow \exists L.\top \neq \bot$ ("$\exists L.\top$ notwendig nicht-leer") bedeutet*:
„Es gibt eine Kette aus R_a und R_b, die zur Sprache gehört."

$\rightsquigarrow \exists L_1.\exists L_2^- \neq \bot$ für zwei kodierte Sprachen L_1 und L_2 bedeutet:
„Es gibt ein Wort, das zu L_1 und zu L_2 gehört."

*) bei entsprechender Tbox!

Leider gilt: Leerheit der Überschneidung kontextfreier Sprachen ist unentscheidbar.

$\rightsquigarrow$ OWL mit Rolleninklusionen ist unentscheidbar

Reguläre Rboxen

Kann man Rolleninklusion zwecks Entscheidbarkeit einschränken?

- Rboxen sind wie Grammatiken für kontextfreie formale Sprachen
- Überschneidungen von kontextfreien Sprachen problematisch

$\rightsquigarrow$ Einschränkung auf reguläre Sprachen!

AIFB

Reguläre Rboxen

Kann man Rolleninklusion zwecks Entscheidbarkeit einschränken?

- Rboxen sind wie Grammatiken für kontextfreie formale Sprachen
- Überschneidungen von kontextfreien Sprachen problematisch

$\rightsquigarrow$ Einschränkung auf reguläre Sprachen!

Reguläre Rboxen

Rollennamen werden mit $\prec$ geordnet (strenge totale Ordnung).
Jede Rbox-Inklusion muss eine der folgenden Formen haben:

- $R \circ R \sqsubseteq R$
- $R^- \sqsubseteq R$
- $S_1 \circ S_2 \circ \ldots \circ S_n \sqsubseteq R$
- $R \circ S_1 \circ S_2 \circ \ldots \circ S_n \sqsubseteq R$
- $S_1 \circ S_2 \circ \ldots \circ S_n \circ R \sqsubseteq R$

Dabei gilt: $S_i \prec R$ für alle $i = 1, 2, \ldots, n$.
Rbox ist regulär, wenn es so eine Ordnung $\prec$ gibt.

Reguläre Rboxen – Beispiel

Beispiel:

$R \circ S \sqsubseteq R \qquad S \circ S \sqsubseteq S \qquad R \circ S \circ R \sqsubseteq T$

Reguläre Rboxen – Beispiel

Beispiel:

$R \circ S \sqsubseteq R \qquad S \circ S \sqsubseteq S \qquad R \circ S \circ R \sqsubseteq T$

$\rightsquigarrow$ ist regulär mit Ordnung $S \prec R \prec T$

Beispiel:

$R \circ T \circ S \sqsubseteq T$

Reguläre Rboxen – Beispiel

Beispiel:

$R \circ S \sqsubseteq R \qquad S \circ S \sqsubseteq S \qquad R \circ S \circ R \sqsubseteq T$

$\rightsquigarrow$ ist regulär mit Ordnung $S \prec R \prec T$

Beispiel:

$R \circ T \circ S \sqsubseteq T$

$\rightsquigarrow$ ist nicht regulär (unzulässige Inklusions-Form)

Beispiel:

$R \circ S \sqsubseteq S \qquad S \circ R \sqsubseteq R$

AIFB

Reguläre Rboxen – Beispiel

Beispiel:

$R \circ S \sqsubseteq R \qquad S \circ S \sqsubseteq S \qquad R \circ S \circ R \sqsubseteq T$

$\rightsquigarrow$ ist regulär mit Ordnung $S \prec R \prec T$

Beispiel:

$R \circ T \circ S \sqsubseteq T$

$\rightsquigarrow$ ist nicht regulär (unzulässige Inklusions-Form)

Beispiel:

$R \circ S \sqsubseteq S \qquad S \circ R \sqsubseteq R$

$\rightsquigarrow$ ist nicht regulär (keine gültige Ordnung möglich)

AIFB

Beschränkung einfacher Rollen

- Einfache Rollen in $\mathcal{SHOIN}$ = Rollen ohne transitive Unterrollen
- In $\mathcal{SROIQ}$: Beachtung der Rolleninklusionen nötig!

Einfache Rollen sind alle Rollen . . .

- die nicht auf der rechten Seite einer Rolleninklusion vorkommen,
- die Inverse von anderen einfachen Rollen sind,
- die nur auf der rechten Seite von Rolleninklusionen vorkommen, bei denen links ausschließlich einfache Rollen stehen.

(Achtung: induktive Definition)

$\rightsquigarrow$ nicht-einfach sind Rollen, die direkt oder indirekt von sich selbst abhängen (und deren Überrollen)

Warum ist das wichtig?

AIFB

Beschränkung einfacher Rollen

- Einfache Rollen in $\mathcal{SHOIN}$ = Rollen ohne transitive Unterrollen
- In $\mathcal{SROIQ}$: Beachtung der Rolleninklusionen nötig!

Einfache Rollen sind alle Rollen . . .

- die nicht auf der rechten Seite einer Rolleninklusion vorkommen,
- die Inverse von anderen einfachen Rollen sind,
- die nur auf der rechten Seite von Rolleninklusionen vorkommen, bei denen links ausschließlich einfache Rollen stehen.

(Achtung: induktive Definition)

$\rightsquigarrow$ nicht-einfach sind Rollen, die direkt oder indirekt von sich selbst abhängen (und deren Überrollen)

Warum ist das wichtig?
Ausdrücke $\leq n\,R.C$, $\geq n\,R.C$, $\mathrm{Irr}(R)$, $\mathrm{Dis}(R,S)$, $\exists R.\mathrm{Self}$ $\neg R(a,b)$
nur für einfache Rollen R und S erlaubt!
(Grund: Sicherstellung von Entscheidbarkeit)

AIFB

Überblick $\mathcal{SROIQ}$

Klassenausdrücke

Klassennamen	A, B
Konjunktion	$C \sqcap D$
Disjunktion	$C \sqcup D$
Negation	$\neg C$
Exist. Rollenrestr.	$\exists R.C$
Univ. Rollenrestr.	$\forall R.C$
Self	$\exists S.\mathrm{Self}$
Größer-als	$\geq n\ S.C$
Kleiner-als	$\leq n\ S.C$
Nominale	$\{a\}$

Rollen

Rollennamen	R, S, T
einfache Rollen	S, T
Inverse Rollen	R^-
Universelle Rolle	U

Tbox (Klassenaxiome)

Inklusion	$C \sqsubseteq D$
Äquivalenz	$C \equiv D$

Rbox (Rollenaxiome)

Inklusion	$R_1 \sqsubseteq R_2$
Allgemeine Inkl.	$R_1^{(-)} \circ \ldots \circ R_n^{(-)} \sqsubseteq R$
Transitivität	$\mathrm{Tra}(R)$
Symmetrie	$\mathrm{Sym}(R)$
Reflexivität	$\mathrm{Ref}(R)$
Irreflexivität	$\mathrm{Irr}(S)$
Disjunktheit	$\mathrm{Dis}(S, T)$

Abox (Fakten)

Klassenzugehörigkeit	$C(a)$
Rollenbeziehung	$R(a, b)$
Neg. Rollenbeziehung	$\neg S(a, b)$
Gleichheit	$a \approx b$
Ungleichheit	$a \not\approx b$

AIFB

Outline

AIFB

Wie kompliziert ist $\mathcal{SROIQ}$?

Rückblick: $\mathcal{SHOIN}$ (OWL DL) ist sehr komplex (NExpTime)

Wie komplex ist $\mathcal{SROIQ}$?

Wie kompliziert ist $\mathcal{SROIQ}$?

Rückblick: $\mathcal{SHOIN}$ (OWL DL) ist sehr komplex (NExpTime)

Wie komplex ist $\mathcal{SROIQ}$?

Beobachtung: einige Ausdrucksmittel sind nicht wirklich nötig!

- Tra(R) durch $R \circ R \sqsubseteq R$ ausdrückbar
- Sym(R) durch $R^- \sqsubseteq R$ ausdrückbar
- Irr(S) durch $\top \sqsubseteq \neg\exists S.\text{Self}$ ausdrückbar
- Universelle Rolle durch transitive, reflexive Überrolle aller Rollen ersetzbar (hier nicht vertieft)
- Abox durch Nominale darstellbar, z.B. $R(a, b)$ durch $\{a\} \sqsubseteq \exists R.\{b\}$

Qualifizierte Zahlenrestriktionen kaum problematisch (bekannt und implementiert, siehe Vorlesung zu OWL)

$\rightsquigarrow$ Hauptproblem Rollenaxiome (Rbox)

AIFB

Rolleninklusion, Sprachen, Automaten

Wie geht man mit Rboxen um?

- Rbox-Regeln ähneln formalen Grammatiken
- jede Rolle R definiert eine reguläre Sprache:
  die Sprache der Rollen-Ketten, aus denen R folgt
- reguläre Sprachen $\equiv$ reguläre Ausdrücke $\equiv$ endliche Automaten

$\rightsquigarrow$ Ansatz: Tableauverfahren werden mit „Rbox-Automaten“ erweitert

Details siehe Literaturangaben zu SROIQ

Entscheibarkeit von $\mathcal{SROIQ}$

Tableauverfahren von $\mathcal{SROIQ}$ zeigt:

$\mathcal{SROIQ}$ ist entscheidbar.

- Algorithmus hat gute Anpassungseigenschaften: ungenutze Merkmale belasten die Abarbeitung kaum („pay as you go“)
- Tableau-Verfahren ungeeignet für enge Komplexitätsabschätzungen
  $\rightsquigarrow$ Komplexität von $\mathcal{SROIQ}$ bisher nicht bekannt (30.1.2008)

AIFB

Outline

OWL DL 1.1: Weitere Aspekte

$\mathcal{SROIQ}$ ist „nur“ logische Grundlage von OWL DL 1.1

Weitere nicht-logische Aspekte:

- Syntax (Erweiterung nötig)
- Datentypdeklaration und Datentypfunktionen, neue Datentypen?
- Metamodellierung: „Punning“
- Kommentarfunktionen und ontologische Metadaten
- Invers-funktionale konkrete Rollen (DatatypeProperties): Keys?
- Mechanismen zu Ontologieimport?
- . . .

$\rightsquigarrow$ viele laufende Änderungen
Hier: erste Übersicht für einige dieser Punkte

AIFB

Metamodellierung

Spezifikation ontologischen Wissens *über* einzelne Elemente der Ontologie (einschließlich Klassen, Rollen, Axiome).

Beispiele:

- „Die Klasse *Person* wurde am 30.1.2008 von *MarkusK* angelegt.“
- „Für die Klasse *Stadt* wird die Property *Einwohnerzahl* empfohlen.“
- „Die Aussage ‚Dresden wurde 1206 gegründet‘ wurde maschinell ermittelt mit einer Sicherheit von 85%.“

(Vergleich auch Reifikation in RDF Schema)

AIFB

Wortspiele in OWL: Punning

Metamodellierung in ausdrucksstarken Logiken ist gefährlich und teuer!

OWL 1.1. unterstützt zurzeit einfachste Form von Metamodellierung:

Punning

- Bezeichner für Klassen, Rollen, Individuen müssen nicht disjunkt sein
- keine *logische* Beziehung zwischen Klasse, Individuum und Rolle gleichen Namens
- Beziehung nur relevant für pragmatische Interpretation

Beispiel:

$$\text{Person}(\text{Sebastian}) \quad \text{klasseErstelltVon}(\text{Person}, \text{Markus})$$

AIFB

Kommentare und Metadaten

Punning unterstützt einfache Metadaten mit (schwacher) semantischer Bedeutung

Wie kann man rein syntaktische Kommentare zu einer Ontologie machen?

- Kommentare in XML-Dateien: `<!-- Kommentar -->`

Kommentare und Metadaten

Punning unterstützt einfache Metadaten mit (schwacher) semantischer Bedeutung

Wie kann man rein syntaktische Kommentare zu einer Ontologie machen?

- Kommentare in XML-Dateien: `<!-- Kommentar -->`
  $\rightsquigarrow$ kein Bezug auf OWL-Axiome dieser Datei
- nicht-logische Annotationen in OWL:
  `owl:AnnotationProperty`

AIFB

Kommentare und Metadaten

Punning unterstützt einfache Metadaten mit (schwacher) semantischer Bedeutung

Wie kann man rein syntaktische Kommentare zu einer Ontologie machen?

- Kommentare in XML-Dateien: `<!-- Kommentar -->`
  $\rightsquigarrow$ kein Bezug auf OWL-Axiome dieser Datei
- nicht-logische Annotationen in OWL:
  `owl:AnnotationProperty`
  $\rightsquigarrow$ fest verknüpft mit (semantischem) ontologischem Element, kein syntaktischer Bezug

OWL 1.1 soll „echte“ syntaktische Kommentare unterstützen

AIFB

Syntaktische Fragen

Syntax für OWL 1.1 zum Teil noch unklar

Wahrscheinlich zwei grundlegende Varianten:

- Funktionale Syntax: ersetzt „Abstrakte Syntax“ von OWL 1.0
- RDF-Syntax: Erweiterung von OWL/RDF 1.0

⇝ funktionale Syntax einfacher zu definieren (keine RDF-Beschränkungen), kompakter

⇝ RDF-Syntax für Abwärtskompatibilität wichtig

⇝ verschiedene Ausdrucksmittel bisher ohne RDF-Syntax

Outline

AIFB

Quo vadis, OWL Lite?

Quo vadis, OWL Lite?

OWL Lite als Fehlschlag:

- beinahe so komplex wie OWL DL
- komplizierte Syntax gibt keinen direkten Zugang zu wahrer Ausdrucksstärke
- Verwendung in Ontologien heute praktisch nur „zufällig“, nicht bewusst

Ursprüngliches Ziel:
einfach und effizient implementierbarer Teil von OWL

$\rightsquigarrow$ offene Diskussion für OWL 1.1

AIFB

$\mathcal{EL}^{++}$ – OWL polynomiell

Beschreibungslogik $\mathcal{EL}^{++}$

- Konzpete nur mit Konjunktion $C \sqcap D$, Existenz $\exists R.C$, $\top$ und $\bot$
- Nominale
- allgemeine Rolleninklusionen (Rbox), Transitivität

$\mathcal{EL}^{++}$ – OWL polynomiell

Beschreibungslogik $\mathcal{EL}^{++}$

- Konzpete nur mit Konjunktion $C \sqcap D$, Existenz $\exists R.C$, $\top$ und $\bot$
- Nominale
- allgemeine Rolleninklusionen (Rbox), Transitivität

Vorteile:

- polynomielle Komplexität
- schnelle Implementierungen verfügbar
- einfache Definition
- unterstützt praktisch relevante Ontologien (SNOMED)

AIFB

DL Lite – Die Datenbank-DL

Beschreibungslogik DL Lite$_R$

- Konzpete nur mit Konjunktion $C \sqcap D$, unqualifizierter Existenz $\exists R.\top$, und $\bot$
- Inverse Rollen, einfache Rollenhierarchien
- Abox wie in $\mathcal{SROIQ}$

DL Lite – Die Datenbank-DL

Beschreibungslogik DL Lite$_R$

- Konzpete nur mit Konjunktion $C \sqcap D$, unqualifizierter Existenz $\exists R.\top$, und $\bot$
- Inverse Rollen, einfache Rollenhierarchien
- Abox wie in $\mathcal{SROIQ}$

Vorteile:

- sub-polynomielle Komplexität (verwandt mit relationalen Datenbanken)
- schnelle Implementierungen verfügbar
- relativ einfache Definition
- unterstützt RDFS

AIFB

Outline

Was tun mit OWL Full?

Ziel von OWL 1.1 DL: viele OWL-Full-1.0-Ontologien als DL interpretierbar machen (siehe z.B. Punning)

Was soll aus OWL Full 1.1 werden?

- Erweiterung von OWL Full im Sinne von OWL 1.1 wird durch verschiedene Anwender unterstützt
- Definition zurzeit nicht ausgearbeitet $\rightsquigarrow$ Spezifikation noch in Arbeit
- Erweiterung von OWL Full Teil des Mandates der OWL-Arbeitsgruppe

AIFB

Offene Fragen zu OWL Full 1.1

Viele Fragen um OWL Full 1.1 zurzeit unklar

- Komplette Neuformulierung der Spezifikation? (Konsistenz der *Spezifikation* von OWL Full 1.0 konnte bis heute nicht nachgewiesen werden [30.1.2008])
- Probleme mit OWL-1.1-Erweiterungen (z.B. Punning)?
- Gewünschter semantischer Bezug zu OWL Full 1.0 und OWL DL 1.1?
- Sollte es „OWL Lite 1.1“ und „OWL Lite 1.1 Full“ geben? (Erweiterung syntaktischer Fragmente um OWL-Full-Semantik)
- . . .

Relativ sicher ist:

- Wenn technisch möglich, so wird die OWL-Arbeits OWL Full 1.1 definieren.
- Es wird (weiterhin) kleinere(?) inkompatible Unterschiede der Semantik von OWL Full und OWL DL geben.

AIFB

Outline

AIFB

Status von OWL 1.1

Aktueller Status:

- OWL-Arbeitsgruppe besteht seit September 2007 (erstes Treffen Anfang Dezember)
- erste Arbeitsentwürfe für Syntax, Semantik und RDF-Umsetzung am 8. Januar veröffentlicht
- Softwareunterstützung für Inferenz und Ontologieerstellung zum Teil bereits verfügbar

Offizieller Zeitplan:

- Dezember 2008: fertig ausgearbeitete Standardisierungsempfehlung (*Candidate Recommendation*)
- März 2009: Verabschiedung des neuen Standards

AIFB

Zusammenfassung

OWL 1.1 als erste Weiterentwicklung des OWL-Standards

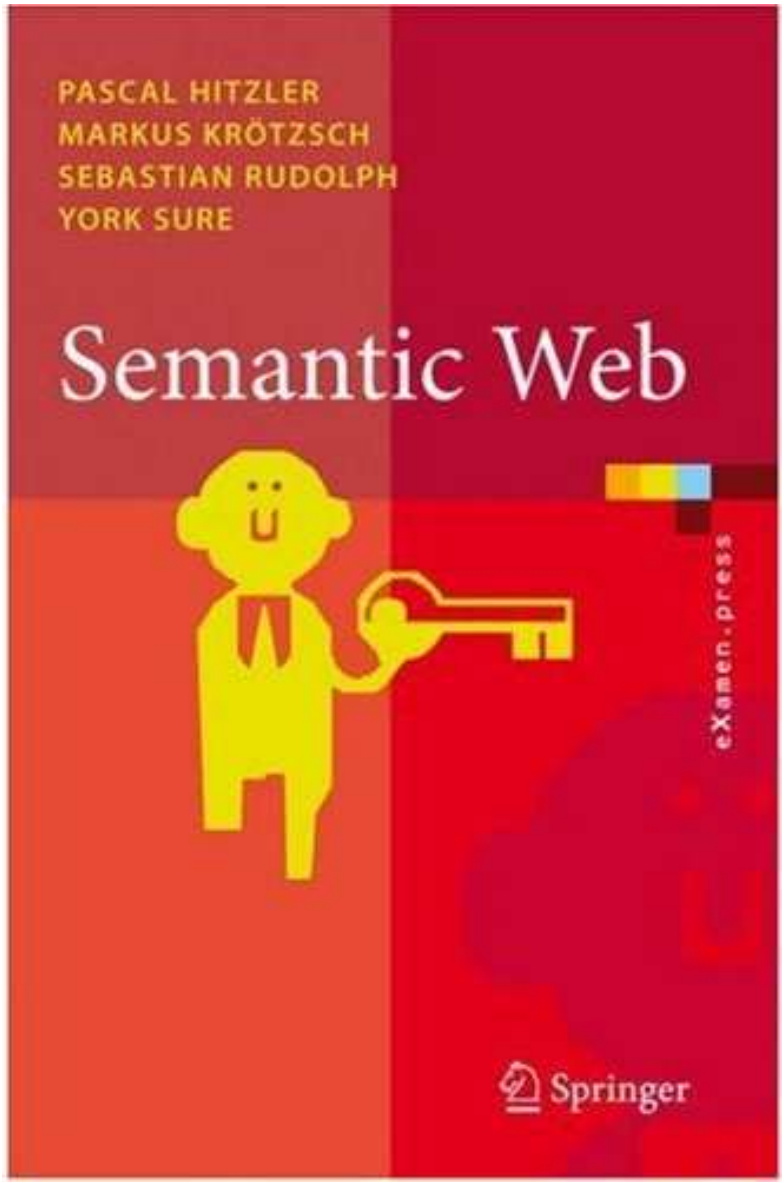
- logische Erweiterung: Beschreibungslogik $\mathcal{SROIQ}$ als Grundlage
- neue Ausdrucksmittel vor allem Rollenaxiome, qualifizierte Zahlenrestriktionen
- nicht-logische Erweiterungen: Punning, Kommentare, Datentypen, u.a.
- Entwicklung von OWL Lite und OWL Full zurzeit noch offen
- Verabschiedung bis März 2009 geplant

AIFB

Literatur

Pascal Hitzler
Markus Krötzsch
Sebastian Rudolph
York Sure

Semantic Web
Grundlagen



Springer 2008, 277 S., Softcover
ISBN: 978-3-540-33993-9
Aktuelle Literaturhinweise online

AIFB