Knowledge Representation for the Semantic Web

Winter Quarter 2010

Slides 7 – 02/11/2010

Pascal Hitzler
Kno.e.sis Center
Wright State University, Dayton, OH
http://www.knoesis.org/pascal/
Slides are based on

Pascal Hitzler, Markus Krötzsch, Sebastian Rudolph

Foundations of Semantic Web Technologies

Chapman & Hall/CRC, 2010

Flyer with special offer is available.

http://www.semantic-web-book.org
A Semantic Puzzle

From Horridge, Parsia, Sattler, From Justifications to Proofs for Entailments in OWL. In: Proceedings OWLED2009.
http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-529/

\[
\begin{align*}
\text{Person} & \sqsubseteq \neg \text{Movie} \\
\text{RRated} & \sqsubseteq \text{CatMovie} \\
\text{CatMovie} & \sqsubseteq \text{Movie} \\
\text{RRated} & \equiv (\exists \text{hasScript}. \text{ThrillerScript}) \sqcup (\forall \text{hasViolenceLevel}. \text{High}) \\
\text{Domain}(\text{hasViolenceLevel}, \text{Movie})
\end{align*}
\]

Fig. 1. A justification for Person \( \sqsubseteq \bot \)
Today: Model-theoretic Semantics
Today’s Session: DL Semantics

1. Model-theoretic Semantics of SROIQ(D)
2. The Description Logic EL++
3. Class Project
4. Class Presentations
Model-theoretic Semantics

• Recall:

  How does one make a model-theoretic semantics?

  What – which mathematical entity – actually captures the “meaning”?

  How would we get at this here?
Model-theoretic Semantics

- There are two semantics for OWL.

1. Description Logic Semantics
   also: Direct Semantics; FOL Semantics
   Can be obtained by translation to FOL.
   Some global restrictions apply! (see next slide)

2. RDF-based Semantics (requires RDF/XML syntax: done later)
   No syntax restrictions apply.
   Extends the direct semantics with RDFS-reasoning features.

In the following, we will deal with the direct semantics only.
Direct Semantics

To obtain decidability, syntactic restrictions apply.

• Type separation / punning

• No cycles in property chains.  
  (See global restrictions mentioned earlier.)

• No transitive properties in cardinality restrictions.  
  (See global restrictions mentioned earlier.)
Decidability

- A problem is \textit{decidable} if there exists an always terminating algorithm which determines, whether or not a solution exists.

- A problem is \textit{semi-decidable} if there exists an algorithm which, in case a solution exists, finds this out in finite time.

- A problem is \textit{undecidable} if it is not decidable.

- Note there exist problems which are semi-decidable and undecidable.
Decidability of DLs

- A description logic is decidable if “entailment of axioms” is decidable.

- Most description logics are decidable. Decidability is one of the design criteria for “good” description logics.
Direct Semantics

- model-theoretic semantics
- starts with interpretations
- an interpretation $\mathcal{I}$ maps
  individual names, class names and property names...

...into a domain
Interpretation Example

If we consider, for example, the knowledge base consisting of the axioms

\[
\text{Professor} \sqsubseteq \text{FacultyMember} \\
\text{Professor}(\text{rudiStuder}) \\
\text{hasAffiliation}(\text{rudiStuder}, \text{aifb})
\]

then we could set

\[
\Delta = \{a, b, \text{Ian}\} \\
I_I(\text{rudiStuder}) = \text{Ian} \\
I_I(\text{aifb}) = b \\
I_C(\text{Professor}) = \{a\} \\
I_C(\text{FacultyMember}) = \{a, b\} \\
I_R(\text{hasAffiliation}) = \{(a, b), (b, \text{Ian})\}
\]

Intuitively, these settings are nonsense, but they nevertheless determine a valid interpretation.
OWL Direct Semantics

- mapping is extended to complex class expressions:
  - \( T^I = \Delta^I \)
  - \( \bot^I = \emptyset \)
  - \( (C \cap D)^I = C^I \cap D^I \)
  - \( (C \cup D)^I = C^I \cup D^I \)
  - \( (\neg C)^I = \Delta^I \setminus C^I \)
  - \( (\forall R.C)^I = \{ x \mid \text{for all } (x,y) \in R^I \text{ we have } y \in C^I \} \)
  - \( (\exists R.C)^I = \{ x \mid \text{there is } (x,y) \in R^I \text{ with } y \in C^I \} \)
  - \( (\geq n R.C)^I = \{ x \mid \#\{ y \mid (x,y) \in R^I \text{ and } y \in C^I \} \geq n \} \)
  - \( (\leq n R.C)^I = \{ x \mid \#\{ y \mid (x,y) \in R^I \text{ and } y \in C^I \} \leq n \} \)
- ...and to role expressions:
  - \( U^I = \Delta^I \times \Delta^I \)
  - \( (R^-)^I = \{ (y,x) \mid (x,y) \in R^I \} \)
- ...and to axioms:
  - \( C(a) \) holds, if \( a^I \in C^I \)
  - \( R(a,b) \) holds, if \( (a^I,b^I) \in R^I \)
  - \( C \sqsubseteq D \) holds, if \( C^I \subseteq D^I \)
  - \( R \sqsubseteq S \) holds, if \( R^I \subseteq S^I \)
  - \( \text{Disjoint}(R,S) \) holds if \( R^I \cap S^I = \emptyset \)
  - \( S_1 \circ S_2 \circ \ldots \circ S_n \sqsubseteq R \) holds if \( S_1^I \circ S_2^I \circ \ldots \circ S_n^I \subseteq R^I \)
OWL Direct Semantics

• what’s below gives us a notion of model:

An interpretation is a model of a set of axioms if all the axioms hold (are evaluated to true) in the interpretation.

• Notion of logical consequence obtained as usual.

• ...and to axioms:
  – C(a) holds, if a \in C
  – C \subseteq D holds, if C \subseteq D
  – Disjoint(R,S) holds if R \cap S = \emptyset
  – S_1 \circ S_2 \circ \ldots \circ S_n \subseteq R holds if S_1 \circ S_2 \circ \ldots \circ S_n \subseteq R
Not a model!

If we consider, for example, the knowledge base consisting of the axioms

\[
\text{Professor} \sqsubseteq \text{FacultyMember} \\
\text{Professor}(\text{rudiStuder}) \\
\text{hasAffiliation}(\text{rudiStuder}, \text{aifb})
\]

then we could set

\[
\Delta = \{a, b, \text{Ian}\} \\
I_I(\text{rudiStuder}) = \text{Ian} \\
I_I(\text{aifb}) = b \\
I_C(\text{Professor}) = \{a\} \\
I_C(\text{FacultyMember}) = \{a, b\} \\
I_R(\text{hasAffiliation}) = \{(a, b), (b, \text{Ian})\}
\]

Intuitively, these settings are nonsense, but they nevertheless determine a valid interpretation.
A model

\[
\begin{align*}
\text{Professor} & \sqsubseteq \text{FacultyMember} \\
\text{Professor}(\text{rudiStuder}) & \\
\text{hasAffiliation}(\text{rudiStuder}, \text{aifb}) &
\end{align*}
\]

\[
\Delta = \{a, r, s\}
\]

\[
\begin{align*}
\mathcal{I}_I(\text{rudiStuder}) & = r \\
\mathcal{I}_I(\text{aifb}) & = a \\
\mathcal{I}_C(\text{Professor}) & = \{r\} \\
\mathcal{I}_C(\text{FacultyMember}) & = \{r, s\} \\
\mathcal{I}_R(\text{hasAffiliation}) & = \{(r, a)\}
\end{align*}
\]
Professor ⊆ FacultyMember
Professor(rudiStuder)
hasAffiliation(rudiStuder, aifb)

Is FacultyMember(aifb) a logical consequence?
Returning to our running example knowledge base, let us show formally that $\text{FacultyMember}(\text{aifb})$ is not a logical consequence. This can be done by giving a model $M$ of the knowledge base where $\text{aifb}^M \not\subseteq \text{FacultyMember}^M$. The following determines such a model.

$$\Delta = \{a, r\}$$

$$I_I(\text{rudiStuder}) = r$$

$$I_I(\text{aifb}) = a$$

$$I_C(\text{Professor}) = \{r\}$$

$$I_C(\text{FacultyMember}) = \{r\}$$

$$I_R(\text{hasAffiliation}) = \{(r, a)\}$$
• but often OWL 2 DL is said to be a fragment of first-order predicate logic (FOL) [with equality]...

• yes, there is a translation of OWL 2 DL into FOL

\[
\begin{align*}
\pi(C \subseteq D) &= (\forall x)(\pi_x(C) \rightarrow \pi_x(D)) \\
\pi_x(A) &= A(x) \\
\pi_x(\neg C) &= \neg \pi_x(C) \\
\pi_x(C \cap D) &= \pi_x(C) \land \pi_x(D) \\
\pi_x(C \cup D) &= \pi_x(C) \lor \pi_x(D) \\
\pi_x(\forall R.C) &= (\forall x_1)(R(x, x_1) \rightarrow \pi_{x_1}(C)) \\
\pi_x(\exists R.C) &= (\exists x_1)(R(x, x_1) \land \pi_{x_1}(C)) \\
\pi_x(\geq n S.C) &= (\exists x_1)\ldots(\exists x_n)\left( \bigwedge_{i \neq j} (x_i \neq x_j) \land \bigwedge_i (S(x, x_i) \land \pi_{x_i}(C)) \right) \\
\pi_x(\leq n S.C) &= \neg(\exists x_1)\ldots(\exists x_{n+1})\left( \bigwedge_{i \neq j} (x_i \neq x_j) \land \bigwedge_i (S(x, x_i) \land \pi_{x_i}(C)) \right) \\
\pi_x(\{a\}) &= (x = a) \\
\pi_x(\exists SSELF) &= S(x, x)
\end{align*}
\]

\[
\begin{align*}
\pi(R_1 \subseteq R_2) &= (\forall x)(\forall y)(\pi_{x,y}(R_1) \rightarrow \pi_{x,y}(R_2)) \\
\pi_{x,y}(S) &= S(x, y) \\
\pi_{x,y}(R^\text{\neg}) &= \pi_{y,x}(R) \\
\pi_{x,y}(R_1 \circ \ldots \circ R_n) &= (\exists x_1)\ldots(\exists x_{n-1}) \\
&\quad \left( \pi_{x_1}(R_1) \land \bigwedge_{i=1}^{n-2} \pi_{x_i,x_{i+1}}(R_{i+1}) \land \pi_{x_{n-1},y}(R_n) \right) \\
\pi(\text{Ref}(R)) &= (\forall x)(\pi_{x,x}(R)) \\
\pi(\text{Asy}(R)) &= (\forall x)(\forall y)(\pi_{x,y}(R) \rightarrow \neg \pi_{y,x}(R)) \\
\pi(\text{Dis}(R_1, R_2)) &= \neg(\exists x)(\exists y)(\pi_{x,y}(R_1) \land \pi_{x,y}(R_2))
\end{align*}
\]

• ...which (interpreted under FOL semantics) coincides with the definition just given.
Inconsistency and Satisfiability

- A set of axioms (knowledge base) is satisfiable (or consistent) if it has a model.
- It is unsatisfiable (inconsistent) if it does not have a model.
- Inconsistency is often caused by modelling errors.

```
Unicorn(beautyTheUnicorn)
Unicorn ⊑ Fictitious
Unicorn ⊑ Animal
Fictitious △ Animal ⊑ ⊥
```
Inconsistency and Satisfiability

• A knowledge base is incoherent if a named class is equivalent to ⊥.

• It usually also points to a modeling error.

\[
\begin{align*}
\text{Unicorn} & \sqsubseteq \text{Fictitious} \\
\text{Unicorn} & \sqsubseteq \text{Animal} \\
\text{Fictitious} & \sqcap \text{Animal} \sqsubseteq \bot
\end{align*}
\]
Rationale behind OWL

- Open World Assumption
- Favourable trade-off between expressivity and scalability
- Integrates with RDFS
- Purely declarative semantics

Features:
- Fragment of first-order predicate logic (FOL)
- Decidable
- Known complexity classes (N2ExpTime for OWL 2 DL)
- Reasonably efficient for real KBs
A Semantic Puzzle


```
Person ⊑ ¬Movie
RRated ⊑ CatMovie
CatMovie ⊑ Movie
RRated ≡ (∃hasScript.ThrillerScript) ∪ (∀hasViolenceLevel.High)
Domain(hasViolenceLevel, Movie)
```

**Fig. 1.** A justification for Person ⊑ ⊥
Today’s Session: DL Semantics

1. Model-theoretic Semantics of SROIQ(D)
2. The Description Logic EL++
3. Class Project
4. Class Presentations
• The OWL 2 spec describes three profiles (fragments, sublanguages) which have polynomial complexity.
  – OWL EL (the description logic EL++)
    we will talk about this next
  – OWL QL (the description logic DL Lite\textsubscript{R})
    forthcoming class presentation
  – OWL RL (the description logic DLP)
    skipped
    • inspired by intersecting OWL with Datalog
    • implemented e.g. in Oracle 11g
• *Pushing the EL Envelope.* Franz Baader, Sebastian Brandt, and Carsten Lutz. In Proc. of the 19th Joint Int. Conf. on Artificial Intelligence (IJCAI 2005), 2005
  – this introduces EL++

• *Pushing the EL Envelope Further.* Franz Baader, Sebastian Brandt, and Carsten Lutz. In Proc. of the Washington DC workshop on OWL: Experiences and Directions (OWLED08DC), 2008
  – this extends EL++. If people talk about EL++ better check if the extended version is meant.
• **EL**
  - existential quantification $\exists$
  - conjunction $\sqcap$
  - top concept $\top$
  - i.e. it’s a fragment/sublanguage of ALC

• **EL+**
  - bottom concept $\bot$ (this allows e.g. disjoint classes)
  - role chains $R \circ S \sqsubseteq T$
  - datatypes

• **EL++**
  - nominals with one individual \{o\}

• **EL++ extended**
  - reflexive roles
  - range of roles (domain is already in EL)

**note:** a global syntactic restriction applies to guarantee polynomiality
EL++ (ext.) global restriction

• IF
  – $R_1 \circ \ldots \circ R_n \subseteq S_1$
  – $S_1 \subseteq \ldots \subseteq S_n$
  – $\text{range}(S_n) \subseteq C$

• THEN
  – there are $R_{n+1}, \ldots, R_m$ with
  – $R_n \subseteq R_{n+1} \subseteq \ldots \subseteq R_m$ and
  – $\text{range}(R_m) \subseteq C$
• Work on EL++ initiated a research branch into polynomial description logics.
• Breakthrough was the classification of the SNOMED commercial ontology.
  – http://www.ihtsdo.org/snomed-ct/
• Most well-known reasoner: CEL
  http://lat.inf.tu-dresden.de/systems/cel/
  – performs classification only:
    computation of the class hierarchy of all named classes

• Pellet also has a specialized algorithm implemented

• It’s currently still unclear how to reason efficiently with nominals (and thus with ABoxes).
Other polynomial OWL profiles

• See
  http://www.w3.org/TR/owl2-profiles/
Today’s Session: DL Semantics

1. Model-theoretic Semantics of SROIQ(D)
2. The Description Logic EL++
3. Class Project
4. Class Presentations
Class Project

- Use the classes and properties from your ontology (if necessary, add some new ones).
- Use them as class names and role names, and write down (in DL notation) a number of SROIQ axioms which make sense in the context of your project ontology.
- Make sure you use each of the following constructs at least once:
  - $\cap$, $\cup$, $\neg$, $\exists$, $\forall$
  - a nominal
  - an inverse property
  - a qualified cardinality constraint
  - three of the property characteristics
Class Project

• Send me by Sunday 21\textsuperscript{st} of February:
  – Current version of your ontology in Turtle syntax.
  – The DL axioms.
    • Either on paper, handwritten (e.g. via Tonya Davis for me)
    • Or as a pdf (e.g. generated from LaTex).
    • Or via Protege (in one of the OWL 2 serializations).
      (We haven’t talked about OWL 2 syntax yet, so this is really optional.)
  – Each DL axioms accompanied with a natural language sentence which captures its meaning.
Today’s Session: DL Semantics

1. Model-theoretic Semantics of SROIQ(D)
2. The Description Logic EL++
3. Class Project
4. Class Presentations
Class presentations – scheduled

- RDFa – embedding RDF in HTML (W3C standard)
  Pavan, Thursday 28th of January

- Scalable Distributed Reasoning using MapReduce (Urbani, Kotoulas, Oren, van Harmelen, ISWC2009)
  Wenbo, Thursday 28th of January

All remaining presentations will be in the last week
- Semantic MediaWiki, Vinh, to be scheduled
- Linked Open Data, Ashutosh, to be scheduled
- FOAF, Hemant, to be scheduled
- Virtuoso, Pramod, to be scheduled
- Prateek, Conjunctive Queries for OWL
- Raghava, DL-Lite
Thursday 4th of February: OWL Part 1
Tuesday 9th of February: Campus Closed
Thursday 11th of February: OWL Part 2
Tuesday 23rd of February: Exercise Session
Thursday 25th of February: OWL Part 3
Week from March 8th: Class Presentations
Friday March 12th: most exams

Estimated breakdown of sessions:
Intro + XML: 2
OWL: 4
Class Project Session: 1
Exercise sessions: 2.7
RDF: 3.3
SPARQL: 1
Class Presentations: 3