Knowledge Representation for the Semantic Web

Winter Quarter 2010

Slides 11 – 03/08/2010

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

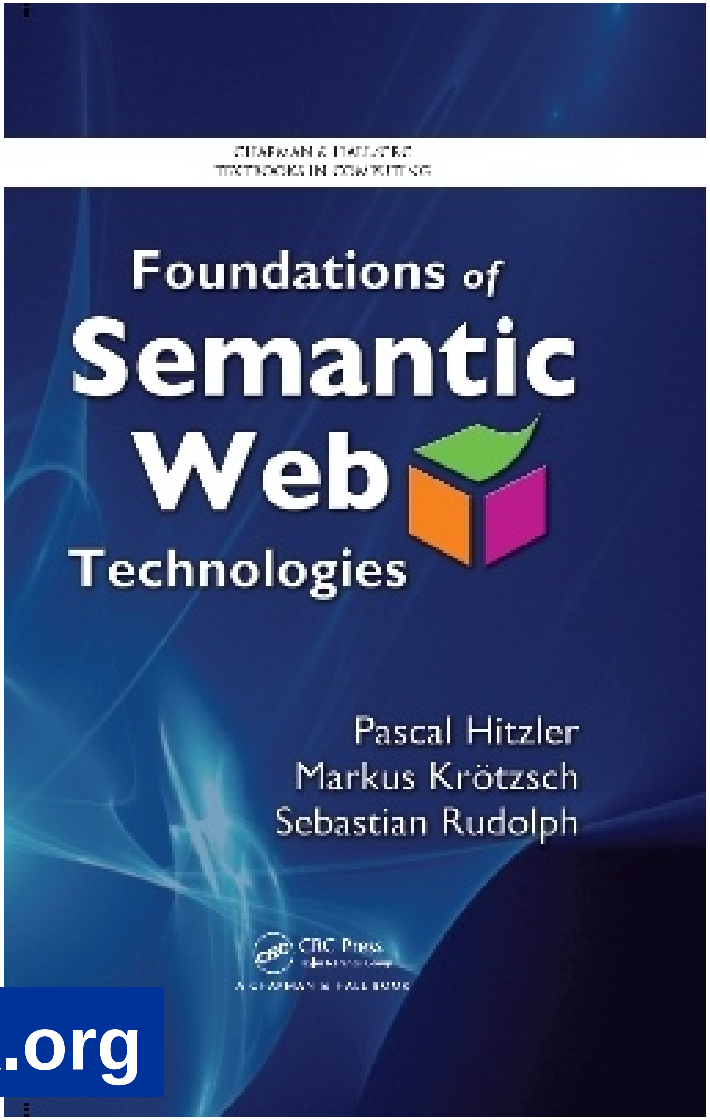
Slides are based on

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

Today: Wrap-Up

- Main Messages
- What Is Semantics – Revisited
- OWL At Its Expressive Limits
- A Very Personal Semantic Web History
- Making OWL Fit For Practice

Main Messages

- How to model in RDF and OWL
- What is model-theoretic semantics
- How to compute logical consequences in RDF and OWL

Today: Wrap-Up

- Main Messages
- What Is Semantics – Revisited
- OWL At Its Expressive Limits
- A Very Personal Semantic Web History
- Making OWL Fit For Practice

What Is Semantic Web Semantics?

- **Opinions Differ. Here's my take.**
- **Semantic Web requires a *computable* semantics.**
- **I.e., the semantics must be a formal entity which is clearly defined and automatically computable.**
- **Ontology languages provide this by means of their formal semantics.**
- **Semantic Web Semantics is given by a relation – the *logical consequence* relation.**
- **Note: This is considerably more than saying that the semantics of an ontology is the set of its logical consequences!**

Today: Wrap-Up

- Main Messages
- What Is Semantics – Revisited
- OWL At Its Expressive Limits
- A Very Personal Semantic Web History
- Making OWL Fit For Practice

OWL At Its Expressive Limits

- There are a lot of things that cannot be said in OWL.
- We will talk about a few such things and general ideas how to address them.

Simple Reasoning

covered by OWL

If I ask for soccer team members, I also want to get the goalkeepers listed ...

If I ask for cities, I also want all capitals listed ...

inheritance reasoning

Less Simple Reasoning

covered by OWL – given enough data

answering requires merging of knowledge from many websites and using background knowledge.

What was again the name of that russian researcher who worked on resolution-based calculi for EL?

Are lobsters spiders?

What is "Käuzchen" in english?

Sophisticated Application Needs

not covered by OWL

The conclusions from the sensor data are uncertain. How do I process that?

uncertainty reasoning

Merging different sources yields inconsistencies. How do we deal with that?

paraconsistent reasoning

Thinkpads run Windows, unless explicitly stated otherwise ...

default reasoning

Some things you can say in OWL

Rules **are often considered an intuitive form of knowledge representation**

- **Man(x) $\wedge$ hasBrother(x,y) $\wedge$ hasChild(y,z) $\rightarrow$ Uncle(x)**
  - **Man $\sqcap$ $\exists$hasBrother.$\exists$hasChild.$\top$ $\sqsubseteq$ Uncle**

- **ThaiCurry(x) $\rightarrow$ $\exists$contains.FishProduct(x)**
  - **ThaiCurry $\sqsubseteq$ $\exists$contains.FishProduct**

- **kills(x,x) $\rightarrow$ suicide(x)**
  - **$\exists$kills.Self $\sqsubseteq$ suicide**

suicide(x) $\rightarrow$ kills(x,x)

suicide $\sqsubseteq$ $\exists$kills.Self

Note: with these two axioms,

suicide **is basically the same as** ***kills***

Some things you can say in OWL

- NutAllergic(x) $\wedge$ NutProduct(y) $\rightarrow$ dislikes(x,y)
  - NutAllergic $\equiv$ $\exists$nutAllergic.Self
    NutProduct $\equiv$ $\exists$nutProduct.Self
    nutAllergic $\circ$ U $\circ$ nutProduct $\sqsubseteq$ dislikes

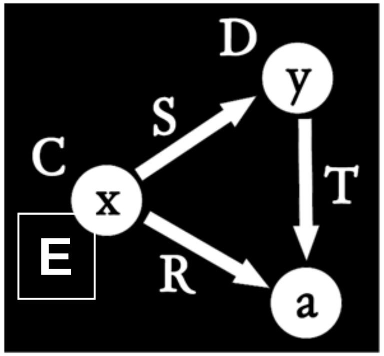
- dislikes(x,z) $\wedge$ Dish(y) $\wedge$ contains(y,z) $\rightarrow$ dislikes(x,y)
  - Dish $\equiv$ $\exists$dish.Self
    dislikes $\circ$ contains^{-} $\circ$ dish $\sqsubseteq$ dislikes

- worksAt(x,y) $\wedge$ University(y) $\wedge$ supervises(x,z) $\wedge$ PhDStudent(z)
  $\rightarrow$ professorOf(x,z)
  - $\exists$worksAt.University $\equiv$ $\exists$worksAtUniversity.Self
    PhDStudent $\equiv$ $\exists$phDStudent.Self
    worksAtUniversity $\circ$ supervises $\circ$ phDStudent $\sqsubseteq$ professorOf

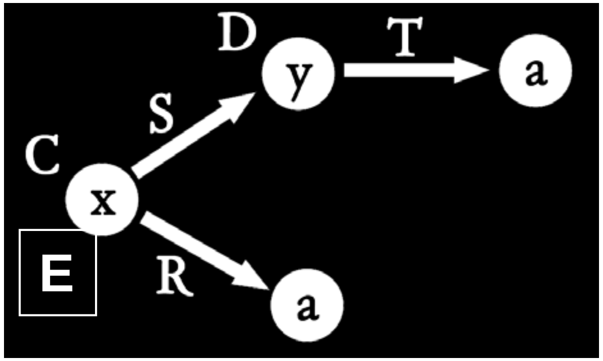
DL Rules: definition

- Tree-shaped bodies
- First argument of the conclusion is the root

- $C(x) \wedge R(x,a) \wedge S(x,y) \wedge D(y) \wedge T(y,a) \rightarrow E(x)$
  - $C \sqcap \exists R.\{a\} \sqcap \exists S.(D \sqcap \exists T.\{a\}) \sqsubseteq E$



duplicating nominals is ok

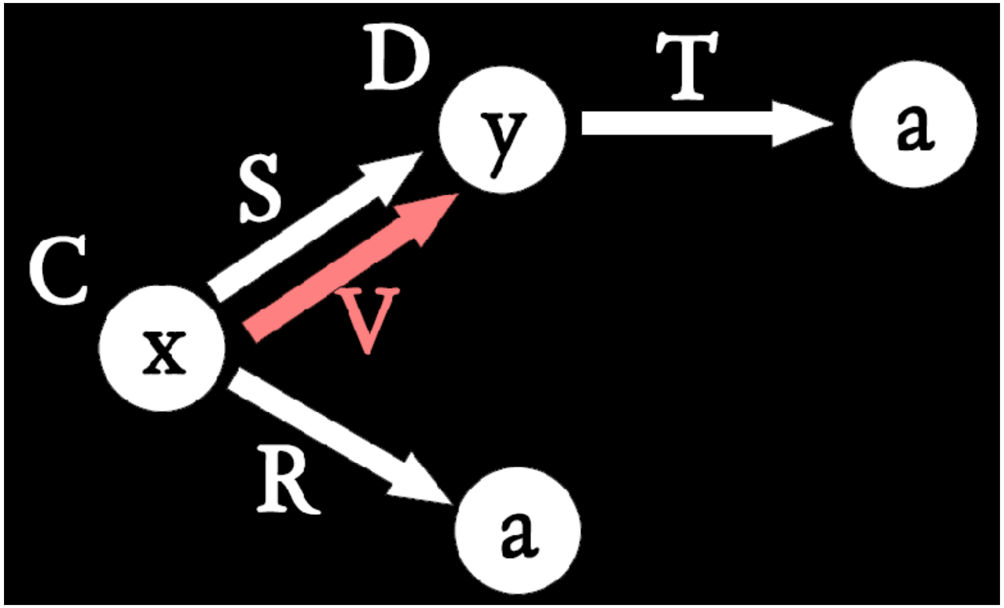


DL Rules: definition

- **Tree-shaped bodies**
- **First argument of the conclusion is the root**

- **$C(x) \wedge R(x,a) \wedge S(x,y) \wedge D(y) \wedge T(y,a) \rightarrow V(x,y)$**

$C \sqcap \exists R.\{a\} \sqsubseteq \exists R1.Self$
$D \sqcap \exists T.\{a\} \sqsubseteq \exists R2.Self$
$R1 \circ S \circ R2 \sqsubseteq V$



DL Rules: definition

- Tree-shaped bodies
- First argument of the conclusion is the root

- complex classes are allowed in the rules
  - Mouse(x) $\wedge$ $\exists$hasNose.TrunkLike(y) $\rightarrow$ smallerThan(x,y)
  - ThaiCurry(x) $\rightarrow$ $\exists$contains.FishProduct(x)

    Note: This allows to reason with unknowns (unlike rules)
  - allowed class constructors depend on the chosen underlying description logic!

SROIQ Rules can be transformed back into SROIQ!

Outside SROIQ Rules

$$\texttt{Pair}(p) \wedge \texttt{consistOf}(p, x) \wedge \texttt{consistOf}(p, y) \wedge \texttt{differentFrom}(x, y) \wedge \texttt{River}(r) \wedge \texttt{inBetween}(r, p) \wedge \texttt{rightBankOf}(x, r) \rightarrow \texttt{leftBankOf}(y, r).$$

- **Cannot be expressed in SROIQ (is not a SROIQ Rule).**
- **Extending OWL with such more general rules leads to undecidability.**

[Example due to Dong-Po Deng, presented at GeoS2009]

SWRL

$$\texttt{Pair}(p) \wedge \texttt{consistOf}(p, x) \wedge \texttt{consistOf}(p, y) \wedge \texttt{differentFrom}(x, y) \wedge \\ \texttt{River}(r) \wedge \texttt{inBetween}(r, p) \wedge \texttt{rightBankOf}(x, r) \rightarrow \texttt{leftBankOf}(y, r).$$

- **Read rule as a first-order predicate logic formula.**

  Semantically okay, but leads to undedicability in combination with OWL.

DL-safe SWRL

$$\mathtt{Pair}(p) \wedge \mathtt{consistOf}(p,x) \wedge \mathtt{consistOf}(p,y) \wedge \mathtt{differentFrom}(x,y) \wedge \mathtt{River}(r) \wedge \mathtt{inBetween}(r,p) \wedge \mathtt{rightBankOf}(x,r) \rightarrow \mathtt{leftBankOf}(y,r).$$

- **Semantically restrict rule, such that it applies only to individuals which are explicitly contained in the knowledge base.**
  I.e., those with known URIs.

- **DL-safe SWRL combined with OWL is decidable.**

- **Formalism supported, e.g., by Pellet.**

SWRL example

$NutAllergic(sebastian)$

$NutProduct(peanutOil)$

$\exists orderedDish.ThaiCurry(sebastian)$

$ThaiCurry \sqsubseteq \exists contains.\{peanutOil\}$

$\top \sqsubseteq \forall orderedDish.Dish$

$NutAllergic(x) \land NutProduct(y) \rightarrow dislikes(x,y)$

$dislikes(x,z) \land Dish(y) \land contains(y,z) \rightarrow dislikes(x,y)$

$orderedDish(x,y) \land dislikes(x,y) \rightarrow Unhappy(x)$

SWRL example

NutAllergic(sebastian)

NutProduct(peanutOil)

$\exists$orderedDish.ThaiCurry(sebastian)

ThaiCurry $\sqsubseteq$ $\exists$contains.{peanutOil}

$\top \sqsubseteq \forall$orderedDish.Dish

NutAllergic(x) $\wedge$ NutProduct(y) $\rightarrow$ dislikes(x,y)

dislikes(x,z) $\wedge$ Dish(y) $\wedge$ contains(y,z) $\rightarrow$ dislikes(x,y)

orderedDish(x,y) $\wedge$ dislikes(x,y) $\rightarrow$ Unhappy(x)

Conclusions:

dislikes(sebastian,peanutOil)

SWRL example

NutAllergic(sebastian)

NutProduct(peanutOil)

$\exists$orderedDish.ThaiCurry(sebastian)

ThaiCurry $\sqsubseteq$ $\exists$contains.{peanutOil}

$\top$ $\sqsubseteq$ $\forall$orderedDish.Dish

orderedDish rdfs:range Dish.

NutAllergic(x) $\wedge$ NutProduct(y) $\rightarrow$ dislikes(x,y)

dislikes(x,z) $\wedge$ Dish(y) $\wedge$ contains(y,z) $\rightarrow$ dislikes(x,y)

orderedDish(x,y) $\wedge$ dislikes(x,y) $\rightarrow$ Unhappy(x)

Conclusions:

dislikes(sebastian,peanutOil)

orderedDish(sebastian,y_s)

ThaiCurry(y_s)

Dish(y_s)

SWRL example

NutAllergic(sebastian)

NutProduct(peanutOil)

$\exists$orderedDish.ThaiCurry(sebastian)

ThaiCurry $\sqsubseteq$ $\exists$contains.{peanutOil}

$\top \sqsubseteq \forall$orderedDish.Dish

NutAllergic(x) $\wedge$ NutProduct(y) $\rightarrow$ dislikes(x,y)

dislikes(x,z) $\wedge$ Dish(y) $\wedge$ contains(y,z) $\rightarrow$ dislikes(x,y)

orderedDish(x,y) $\wedge$ dislikes(x,y) $\rightarrow$ Unhappy(x)

Conclusions:

dislikes(sebastian,peanutOil)

orderedDish(sebastian,y_s)

ThaiCurry(y_s)

Dish(y_s)

contains(y_s,peanutOil)

SWRL example

Does not work under DL-safety!

NutAllergic(sebastian)

NutProduct(peanutOil)

$\exists$orderedDish.ThaiCurry(sebastian)

ThaiCurry $\sqsubseteq$ $\exists$contains.{peanutOil}

$\top \sqsubseteq \forall$orderedDish.Dish

NutAllergic(x) $\wedge$ NutProduct(y) $\rightarrow$ dislikes(x,y)

dislikes(x,z) $\wedge$ Dish(y) $\wedge$ contains(y,z) $\rightarrow$ dislikes(x,y)

orderedDish(x,y) $\wedge$ dislikes(x,y) $\rightarrow$ Unhappy(x)

Conclusions:

dislikes(sebastian,peanutOil)
orderedDish(sebastian,y_s)
ThaiCurry(y_s)
Dish(y_s)

contains(y_s,peanutOil)
dislikes(sebastian,y_s)

SWRL example

$NutAllergic(sebastian)$

$NutProduct(peanutOil)$

$\exists orderedDish.ThaiCurry(sebastian)$

$ThaiCurry \sqsubseteq \exists contains.\{peanutOil\}$

$\top \sqsubseteq \forall orderedDish.Dish$

$NutAllergic(x) \land NutProduct(y) \rightarrow dislikes(x,y)$

$dislikes(x,z) \land Dish(y) \land contains(y,z) \rightarrow dislikes(x,y)$

$orderedDish(x,y) \land dislikes(x,y) \rightarrow Unhappy(x)$

Conclusions:

dislikes(sebastian,peanutOil)

orderedDish(sebastian,y_s)

ThaiCurry(y_s)

Dish(y_s)

contains(y_s,peanutOil)

dislikes(sebastian,y_s)

Unhappy(sebastian)

SWRL example

NutAllergic(sebastian)
NutProduct(peanutOil)
$\exists$orderedDish.ThaiCurry(sebastian)

ThaiCurry $\sqsubseteq$ $\exists$contains.{peanutOil}
$\top$ $\sqsubseteq$ $\forall$orderedDish.Dish

NutAllergic(x) $\wedge$ NutProduct(y) $\rightarrow$ dislikes(x,y)
dislikes(x,z) $\wedge$ Dish(y) $\wedge$ contains(y,z) $\rightarrow$ dislikes(x,y)
orderedDish(x,y) $\wedge$ dislikes(x,y) $\rightarrow$ Unhappy(x)

Conclusion: Unhappy(sebastian)

Alternative Semantics

- **SWRL and DL-safe SWRL are essentially based on the same style of model-theoretic semantics.**

- **If we want to deal with inconsistencies, uncertainty, or default reasoning, we have to modify the semantic approach.**

- **How to modify a semantics?**
  - **Redefine the notion of *model*!**

Sophisticated Application Needs

The conclusions from the sensor data are uncertain. How do I process that?

uncertainty reasoning

Merging different sources yields inconsistencies. How do we deal with that?

paraconsistent reasoning

Thinkpads run Windows, unless explicitly stated otherwise ...

default reasoning

Paraconsistent Reasoning

- Modification: Use four truth values instead of two.

  {true, false} $\rightarrow$ {true, false, none, both}

- Idea: “both” captures inconsistency.

- Unicorn(beauty)
  Unicorn $\sqsubseteq$ Fictitious
  Unicorn $\sqsubseteq$ Animal
  Animal $\sqsubseteq$ $\neg$Fictitious

  would, e.g., result in the truth value “both” for Fictitious(beauty).

- Problems: Paraconsistency or bugfixing? Which of various related approaches to take? How well does it work in practice?

Uncertainty Reasoning

- **Modification: Use the real unit interval as set of truth values.**

- **0 is interpreted as "false"**
- **1 is interpreted as "true"**

- **Define how to combine them. E.g.,**

  HighQuality(a) is 0.7 true
  Expensive(a) is 0.8 true
  HighQuality $\sqcap$ Expensive $\sqsubseteq$ Buyable

  how "much" true is Buyable(a)?

- **Problems: Different choices for combination. Does it match the intuition? Is this probabilistic or fuzzy? How reliable are the values? And it's computationally (much) more expensive.**

Default Reasoning

- Thinkpads "normally" run Windows.
  I.e., this is the default assumption (to be assumed unless there is evidence to the contrary).

- Thinkpad $\sqsubseteq$ $\forall$runsOS.WindowsOS
  Thinkpad(myThinkpad)
  Thinkpad(yourThinkpad)
  runsOS(yourThinkpad,linux)
  $\neg$WindowsOS(linux)

  is contradictory. How do we capture the default?

Default Reasoning

- **Thinkpad $\sqsubseteq$ ∀runsOS.WindowsOS $\sqcup$ ExceptionThing**
  Thinkpad(myThinkpad)
  Thinkpad(yourThinkpad)
  runsOS(yourThinkpad,linux)
  ¬WindowsOS(linux)

  + a semantics which "minimizes" ExceptionThing.
  IAW, something is only in ExceptionThing if it is necessarily contained in it (e.g., to avoid a contradiction).

- **This idea is called *circumscription* and is due to John McCarthy [1980] (not for DLs, obviously).**
  There exist other approaches which accomplish the same thing in other ways.

- **Problem: Computationally very expensive.**

Default Reasoning

- **Thinkpad $\sqsubseteq$ $\forall$runsOS.WindowsOS $\sqcup$ ExceptionThing**
  Thinkpad(myThinkpad)
  Thinkpad(yourThinkpad)
  runsOS(yourThinkpad,linux)
  $\neg$WindowsOS(linux)

  + a semantics which "minimizes" ExceptionThing.
  IAW, something is only in ExceptionThing if it is necessarily contained in it (e.g., to avoid a contradiction).

- **From all models I of the KB, select those models, for which ExceptionThing^I is minimal.**
  Take these as the *circumscribed* models.
  Define logical consequence as usual.

Today: Wrap-Up

- Main Messages
- What Is Semantics – Revisited
- OWL At Its Expressive Limits
- A Very Personal Semantic Web History
- Making OWL Fit For Practice

A Very Personal Semantic Web History

- **2002: Growing Semantic Web Hype (I wasn't there)**
- **2004: Will it every work?**
- **2006: It's probably not going to work.**
- **2008: Industry is catching on and RDF will work. But OWL won't.**
- **2010: Many major IT companies' R&D departments investigate OWL or even have their own OWL reasoner.**

Today: Wrap-Up

- Main Messages
- What Is Semantics – Revisited
- OWL At Its Expressive Limits
- A Very Personal Semantic Web History
- Making OWL Fit For Practice

Making OWL Fit For Practice

- **The use of formal semantics for RDF and OWL still hasn't produced prominent applications with clear-cut added value compared with other methods/technologies.**
  [More precisely: Such a thing hasn't been made public.]

- **Hindrances:**
  - **Scalability of reasoning isn't very good (yet).**
  - **Few people can really model well in OWL.**
  - **High-quality ontologies are expensive to produce.**
  - **Real-life data often isn't clean enough for reasoning.**

Making OWL Fit For Practice

- **Researchers have to work on:**
  - **Scalability, including alternative reasoning methods. Don't get fixed on soundness/completeness/decidability.**
  - **Dissemination and education.**
  - **Methods for making real-life data fit for formal semantics.**
  - **Developing clear-cut use cases for formal semantics.**

It is essential, to leave the ivory tower!

Some References

- Pascal Hitzler, Towards Reasoning Pragmatics. In: Krzysztof Janowicz, Martin Raubal, Sergei Levashkin (Eds.), GeoSpatial Semantics, Third International Conference, GeoS 2009, Mexico City, Mexico, December 3-4, 2009. Proceedings. Lecture Notes in Computer Science Vol. 5892 Springer 2009, pp. 9-25.
- Prateek Jain, Pascal Hitzler, Peter Z. Yeh, Kunal Verma, Amit P. Sheth, Linked Data is Merely More Data. Proceedings of Linked AI: AAAI Spring Symposium "Linked Data Meets Artificial Intelligence", March 22-24 2010. To appear.

- Markus Krötzsch, Sebastian Rudolph, Pascal Hitzler, ELP: Tractable Rules for OWL 2. In: Amit Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy Finin, Krishnaprasad Thirunarayan (eds.), The Semantic Web - ISWC 2008, 7th International Semantic Web Conference. Springer Lecture Notes in Computer Science Vol. 5318, 2008, pp. 649-664.
- Markus Krötzsch, Sebastian Rudolph, Pascal Hitzler, Description Logic Rules. In: Malik Ghallab, Constantine D. Spyropoulos, Nikos Fakotakis, Nikos Avouris (eds.), Proceedings of the 18th European Conference on Artificial Intelligence, ECAI2008, Patras, Greece, July 2008. IOS Press, 2008, pp. 80-84

Some References

- Stephan Grimm, Pascal Hitzler, A Preferential Tableaux Calculus for Circumscriptive ALCO. In: Axel Polleres, Terrance Swift (Eds.), Web Reasoning and Rule Systems, Third International Conference, RR 2009, Chantilly, VA, USA, October 20009, Proceedings. Lecture Notes in Computer Science Vol. 5837, Springer, pp. 40-54.
- Yue Ma, Pascal Hitzler, Paraconsistent reasoning for OWL 2. In: Axel Polleres, Terrance Swift (Eds.), Web Reasoning and Rule Systems, Third International Conference, RR 2009, Chantilly, VA, USA, October 20009, Proceedings. Lecture Notes in Computer Science Vol. 5837, Springer, pp. 197-211.
- Sebastian Rudolph, Tuvshintur Tserendorj, Pascal Hitzler , What Is Approximate Reasoning? In: D. Calvanese, G. Lausen (eds.), Web Reasoning and Rule Systems, Second International Conference, RR 2008, Karlsruhe, Germany, October/November 2008. Springer Lecture Notes in Computer Science Vol. 5341, 2008, pp. 150-164.