Unit 3 – Languages and Systems

Thomas Krennwallner

KBS Group, Institute of Information Systems, TU Vienna

Reasoning Web Summer School 2008

Unit Outline

1. Introduction
2. RDF Stores with Rules Support
3. SWRL – OWL Reasoners with Rules Support
4. LP Engines with RDF Support
5. Systems for Hybrid Combinations
6. Overview
7. Conclusion

Introduction

- Current state of systems with ontology and rules support
- Presentation will do some kind of assessment, and will build the bridge to the theoretical part
- aim at implemented systems with rule support (only a small selection)
- how we distinguish the systems, classification (RDF/OWL)
  - RDF Frameworks + Rules
  - DL Reasoners + Rules
  - Prolog(-like) Systems + RDF
  - Hybrid Combinations
- systems and languages show case

RDF Stores – what are those?

- sophisticated frameworks for accessing RDF data
- database backends
- query interface, data management interface
- rules support adds additional query expressivity

Jena Rules

Conflicting reviewer in RIF:

```
Forall ?P ?A ?P1 ?N
 ( ?P#ex:ConflictingReviewer :-  And(
      <http://dblp.l3s.de/d2r/page/publications/conf/rweb/EiterIKP08>
         [dc:creator -> ?A]
      ?A[foaf:knows -> ?P1]
      ?P1[foaf:name -> ?N]
      ?P[foaf:name -> ?N]
      ?P#foaf:Person
     )
 )
```

and in Jena rules:

```
[ conflict1:
  (http://dblp.l3s.de/d2r/page/publications/conf/rweb/EiterIKP08 dc:creator ?A),
  (?A foaf:knows ?P1), (?P1 foaf:name ?N), (?P foaf:name ?N),
  (?P rdf:type foaf:Person)
  ->
  (?P rdf:type ex:ConflictingReviewer) ]
```

http://jena.sourceforge.net/inference/

Sesame/OWLIM

Conflicting Reviewers in RIF:

```
Forall ?P ?A ?Pub ?P1 ?N
 ( ?P#ex:ConflictingReviewer :- And(
     <http://dblp.l3s.de/d2r/page/publications/conf/rweb/EiterIKP08>
         [dc:creator -> ?A]
     ?Pub[dc:creator -> ?A]  ?Pub[dc:creator -> ?P1]   ?P1[ foaf:name -> ?N]
     ?P[ foaf:name -> ?N]    ?P#foaf:Person
    )
 )
```

and in OWLIM:

```
Id: conflict2
<http://dblp.l3s.de/d2r/page/publications/conf/rweb/EiterIKP08> <dc:creator> A
Pub <dc:creator> A
Pub <dc:creator> P1
P1 <foaf:name> N
P <foaf:name> N
P <rdf:type> <foaf:Person>
--------------------------
?P#ex:ConflictingReviewer
```

[Kiryakov *et al.*, 2005] `http://www.ontotext.com/owlim/`

Oracle $11g$

- extension of a Oracle database with a rule engine
- extension to SQL for accessing RDF data (SPARQL-like graph queries, SPARQL support planned)
- Native inferencing in the database for
  - RDF, RDFS, OWL-Prime (OWL subset)
  - User-defined rules
- New relationships/triples are inferred and stored ahead of query time
  - Minimizes on-the-fly computation and results in fast query times

[Wu *et al.*, 2008]

`http://www.oracle.com/technology/tech/semantic_technologies/`

Oracle $11g$ (ctd.)

Editing in RIF:

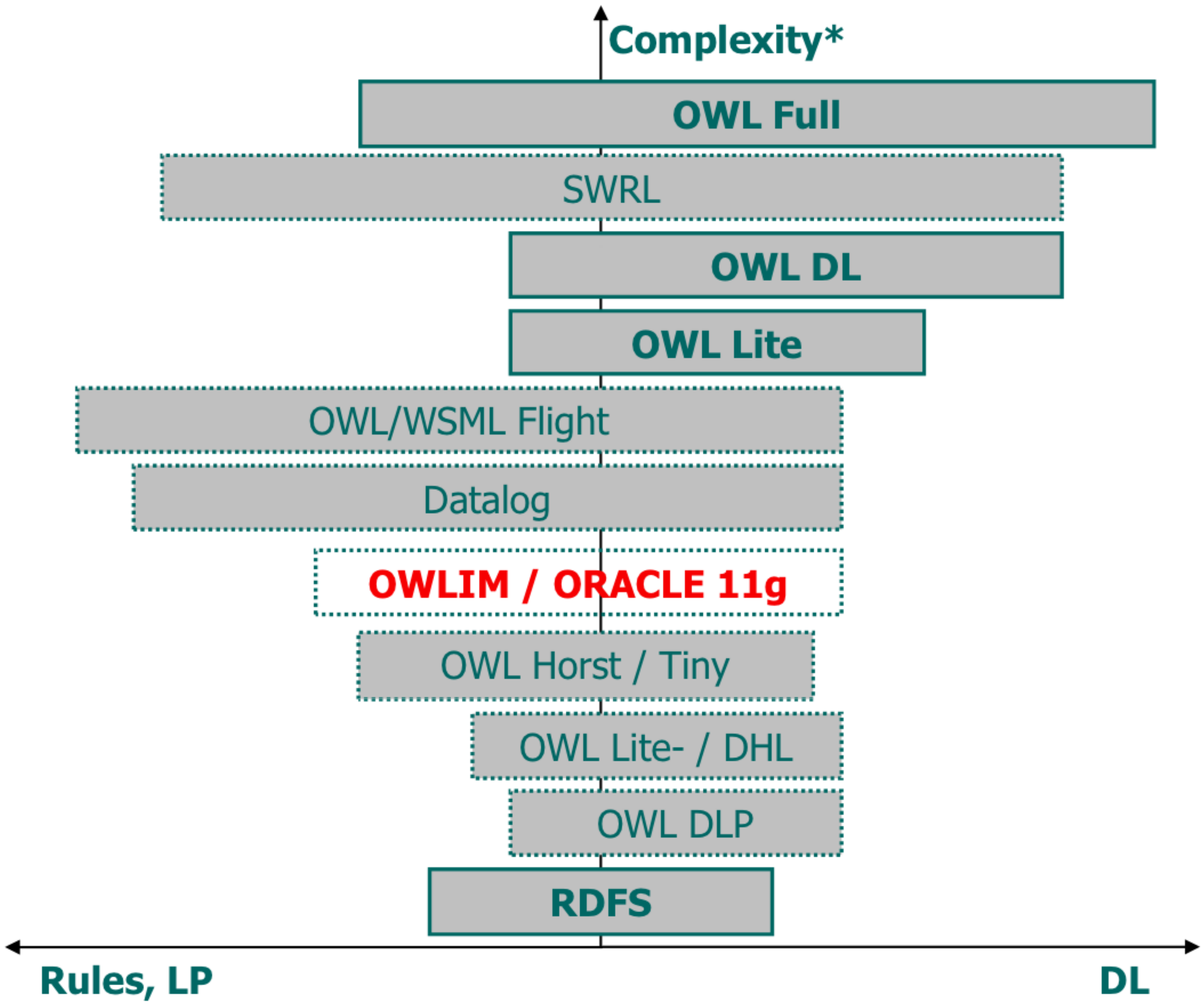
```
Forall ?A ?E ( ?A[ex:editedBy -> ?E] :- Exists ?C ( And (
                     ?A[dc:partOf -> ?C]
                     ?C[swrc:editor -> ?E] )
                 )
              )
```

and in Oracle:

```
INSERT INTO mdsys.semr_user_rulebase VALUES ('editedby_rule',
'(?a <http://purl.org/dc/elements/1.1/partOf> ?c)
 (?c <http://swrc.ontoware.org/ontology#editor> ?e)',
NULL, '(?a <http://www.example.org/editedBy> ?e)', null);

SELECT s,o FROM table(SEM_MATCH('(?s <http://www.example.org/editedBy> ?o)',
           SEM_MODELS('OWLTST'),
           SEM_RULEBASES('OWLPRIME','USER_RULEBASE'),  null, null ));
```

Comparison



(C) Ontotext Lab

SWRL – Semantic Web Rules Language

Observation: OWL DL has limited expressiveness

Example: Expressing editors?

Given the DL roles $partOf$, $editor$, and $editedBy$, one cannot express that

"an article, which is part of a collection with an editor, is edited by the editor of the collection"

by OWL DL *alone*.

but rules can easily express this "triangle:"

$$editedBy(A, E) \leftarrow partOf(A, C) \wedge editor(C, E)$$

Add rules to ontologies: Semantic Web Rules Language (SWRL)
W3C member submission [Horrocks *et al.*, 2004;
Horrocks *et al.*, 2005]

DL-safe Rules

- Well-known: reasoning in a formalism with unrestricted rules over a description logic is *undecidable* [Levy and Rousset, 1998]
- Adding a safety condition to the rules preserves decidability [Motik *et al.*, 2005]:

DL-safe SWRL

A rule r is called *DL-safe* if each variable in r occurs in a non-DL-atom in the rule body.

Example

The rule

$$editedBy(A, E) \leftarrow partOf(A, C) \wedge editor(C, E) \wedge o(A) \wedge o(C) \wedge o(E)$$

is DL-safe, if $o(\cdot)$ is a non-DL-atom (o does not appear in the DL-KB).

DL-safe Rules (ctd.)

DL-safe SWRL ontology

Ontology $\mathcal{O}$ describing $editedBy$, $partOf$, $editor$ plus the DL-safe rule

$$editedBy(A, E) \leftarrow partOf(A, C) \wedge editor(C, E) \wedge o(A) \wedge o(C) \wedge o(E)$$

and eventually adding facts $o(a)$ for every known individual a from $\mathcal{O}$.

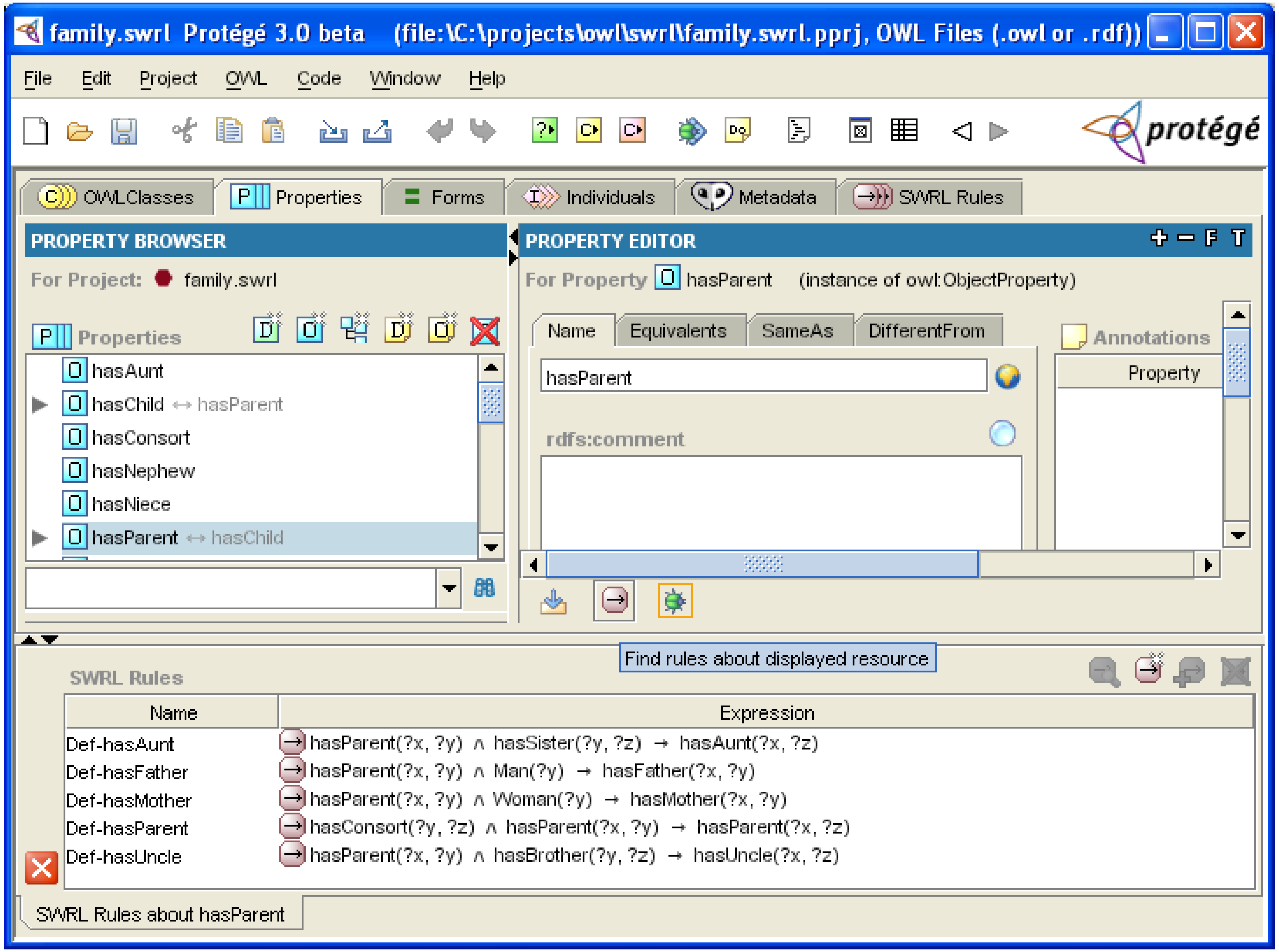
- restricts reasoning with DL-safe rules only to *known* individuals
- more details and other approaches to DL-safety in the next unit

KAON2, Pellet, RacerPro, Protégé

Important: standard DL reasoners support DL-safe SWRL

- KAON2 [Motik, 2006] `http://kaon2.semanticweb.org/`
  supports also F-Logic
- Pellet [Sirin *et al.*, 2007] `http://pellet.owldl.com/`
- RacerPro [Rac, 2007] `http://www.racer-systems.com/`
  also allows some forms of nonmonotonic reasoning
- Protégé support [O'Connor *et al.*, 2005]

SWRL in the Protégé Ontology Editor



(C) `http://protege.cim3.net/`

Logic Programming systems with RDF Support

- Logic Programming systems with additional RDF components
- realized by
  - creating a specialized RDF library to a Prolog engine,
  - compiling to Prolog, or
  - rule engine implemented from scratch with built-in RDF support

SWI-Prolog and $\mathcal{F}$LORA-2

- SWI-Prolog with Semantic Web library [Wielemaker *et al.*, 2008]
  `http://www.swi-prolog.org/`

Implementing RDFS semantics in SWI-Prolog

```
triple(O, rdf:type, C) :- rdf(S,P,O), rdf(P,rdfs:range,C).
```

- $\mathcal{F}$LORA-2 (F-Logic) [Kifer, 2005]
  `http://flora.sourceforge.net/`
  Frame Logic, based on XSB Prolog

Implementing RDFS semantics in $\mathcal{F}$LORA-2

```
?O[rdf:type -> ?C] :- ?S[?P -> ?O], ?P[rdfs:range -> ?C].
```

Combining Rules and Ontologies for the Semantic Web

- systems supporting complex integration of ontologies and rules
- anticipation of next unit
- show a sample combination approach: HEX-programs

HD-Rules and NLP-DL

- Hybrid programs [Drabent *et al.*, 2007]
  `http://www.ida.liu.se/hswrl/`
  HD-Rules: compiling Hybrid rules to prolog program

- dl-programs [Eiter *et al.*, 2008]
  `http://www.kr.tuwien.ac.at/research/systems/semweblp/`
  NLP-DL: implementation for dl-programs
  examples and theoretical details in the next unit

Motivation

- dl-programs: interfacing external source of (DL) knowledge
- Limited flexibility:
  - only one external KB possible
  - only one formalism allowed for KB (OWL)
- Spinning this idea further:
  - Access arbitrary external sources (solvers, services, different knowledge bases, etc.)
  - Standardized interface
  - Entire program: still ASP semantics

$\Rightarrow$ more general formalism needed!

Motivation (ctd.)

Other desirable features for Rules in the Semantic Web:

- Software Interoperability
  - Importing external knowledge
  - Easily extendable reasoning framework
- Higher-Order Reasoning: rules that talk about predicates
  - Stating generic rules
  - Defining ontology semantics in a program

Our extension: *__**H**__igher-order logic programs with __**EX**__ternal atoms* (HEX-programs) [Eiter *et al.*, 2005]

External Atoms

External Atom

$$\&rdf[\text{“http://www.ex.org/kb.rdf”}](S, P, O)$$

consists of

- an identifier: $\&rdf$
- an input list: “http://www.ex.org/kb.rdf”
- an output list: S, P, O

Not only constants, but also predicate extensions can be input:

$\&sum[p](X) \quad \Rightarrow \quad I : \{p(2), p(3), q(4)\} \quad \Rightarrow$ output list: 5

input list: p

$$f_{\&sum}(I, p, n) = 1 \text{ iff } \sum_{p(k) \in I} k = n$$

Higher-Order Atoms

Higher-Order Atoms are expressions of the form

$$(t_0, t_1, \ldots, t_n) \;\; \text{resp.} \;\; t_0(t_1, \ldots, t_n) \;\; ,$$

where $t_0, \ldots, t_n$ are (function-free) terms.

Variable predicate symbols, meta-reasoning:

$$C(X) \leftarrow subClassOf(D, C), D(X) \;\; .$$

Further features, inherited from DLV:

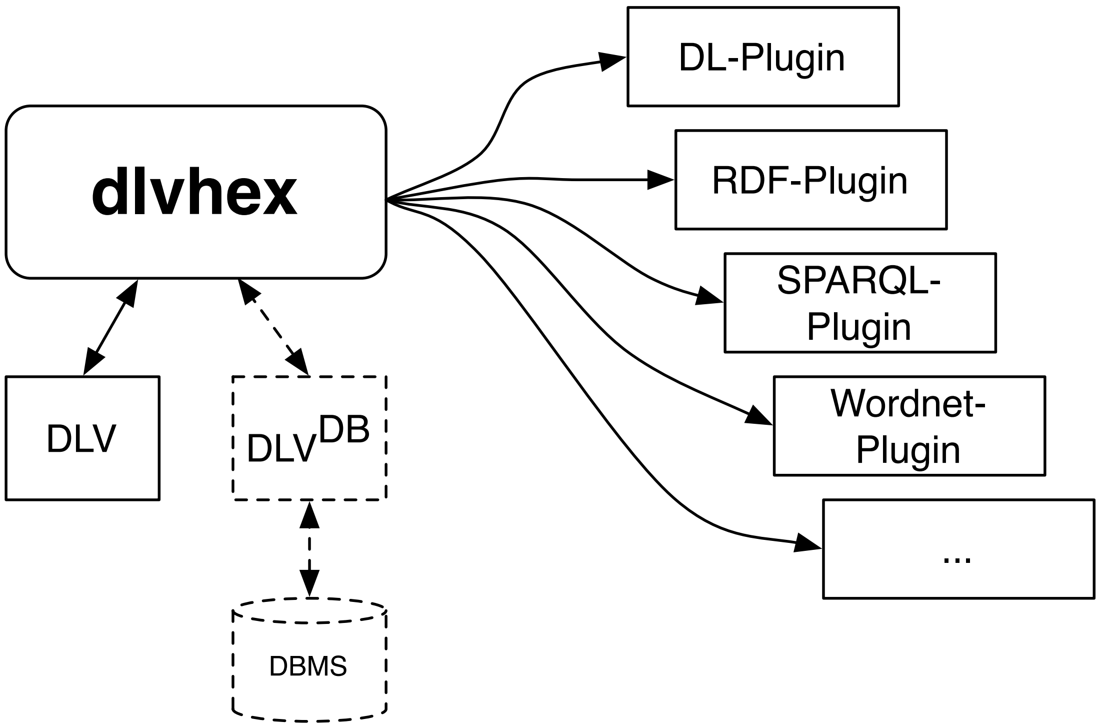
- Constraints, weak constraints
- Aggregates

Architecture

dlvhex [Eiter *et al.*, 2006]: a reasoner for HEX-programs

- Command line application
- interfaces DLV and plugins for external atoms used in a program
- alternatively, it can use DLV^{DB} to interface a database

Design principle:



Demo

see also `http://asptut.gibbi.com/` for a tutorial and demos on

- Answer Set Programming with dlv
- ASP with templates (dlt)
- dl-programs (NLP-DL)
- HEX-programs (dlvhex)

Overview of rule systems features

Features as shown in the previous unit

- Modules
- Function symbols
- Built-Ins
- Higher-order predicates
- Constraints
- Negation As Failure
- Disjunctive Rules

Next slide: current state of the systems

Overview of rule systems features

System (Language)	RDF(S)	OWL	M	F	BI	HO	C	NAF	∨
cwm (N3)	+	−	−	−	+	−	−	+	−
dlvhex (HEX)	+	+	−	−	+	+	+	+	+
$\mathcal{F}$LORA-2 (F-Logic)	+	−	+	+	+	+	−	+	−
HD-rules (Hybrid rules)	−	+	−	+	−	−	−	+	−
Jena (Jena Rules)	+	+	−	−	+	−	−	−	−
KAON2 (SWRL)	+	+	−	−	+	−	−	−	+
NLP-DL (dl-programs)	−	+	−	−	−	−	+	+	−
Oracle $11g$ (OWLPrime)	+	+∼	−	−	+	−	−	−	−
OWLIM (OWL Horst)	+	+∼	−	−	−	−	−	−	−
Pellet (SWRL)	+	+	−	−	−	−	−	−	−
RacerPro (SWRL)	+	+	−	−	+	−	−	−	−
SWI-Prolog (RDF(S))	+	−	+	+	+	−	−	+	−

Legenda: Modules, Functions, Built-Ins, Constraints, Higher Order, Negation As Failure
+ = yes, − = no, +∼ = yes, with some proviso

Conclusion

- current state of systems:
  - diverse syntax and
  - not so formal semantics
- demand for a standardized rule language like RIF (previous unit)
- support for RDF(S), OWL, and Built-ins is standard
- support for “LP-features” is not standard
- different approaches for combining ontologies with rules (next unit)

Tim Berners-Lee, Dan Connolly, Lalana Kagal, Yosi Scharf, and Jim Hendler.
N3Logic: A logical framework for the World Wide Web.
Theory and Practice of Logic Programming, 8:249–269, 2008.

Wlodek Drabent, Jakob Henriksson, and Jan Maluszynski.
HD-Rules: a hybrid system interfacing prolog with dl-reasoners.
In *ALPSWS2007*, 2007.

Thomas Eiter, Giovambattista Ianni, Roman Schindlauer, and Hans Tompits.
A Uniform Integration of Higher-Order Reasoning and External Evaluations in Answer Set Programming.
In L. P. Kaelbling and A. Saffiotti, editors, *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI-05)*, pages 90–96, Denver, USA, 2005. Professional Book Center.

Thomas Eiter, Giovambattista Ianni, Roman Schindlauer, and Hans Tompits.
Towards efficient evaluation of HEX programs.
In Jürgen Dix and Anthony Hunter, editors, *Proceedings of the 11th International Workshop on Nonmonotonic Reasoning (NMR-2006)*, pages 40–46. TR IfI-06-04, Institut für Informatik, TU Clausthal, Germany, May 2006.

Thomas Eiter, Giovambattista Ianni, Thomas Lukasiewicz, Roman Schindlauer, and Hans Tompits.
Combining answer set programming with description logics for the semantic web.
Artificial Intelligence, 172(12-13):1495–1539, August 2008.

Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, and Mike Dean.

SWRL: A Semantic Web Rule Language Combining OWL and RuleML.
W3c member submission, W3C, May 2004.

Ian Horrocks, Peter F. Patel-Schneider, Sean Bechhofer, and Dmitry Tsarkov.
Owl rules: A proposal and prototype implementation.
Journal of Web Semantics, 3(1):23–40, July 2005.

Michael Kifer.
Reasoning Web, chapter Rules and Ontologies in F-Logic, pages 22–34.
Springer, 2005.

Atanas Kiryakov, Damyan Ognyanov, and Dimitar Manov.
Owlim – a pragmatic semantic repository for owl.
In *Proceedings of the International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005), WISE 2005*, volume 3807 of *LNCS*, pages 182–192.
Springer, 2005.

Alon Y. Levy and Marie-Christine Rousset.
Combining horn rules and description logics in carin.
Artificial Intelligence, 104(1—2):165—209, September 1998.

Boris Motik, Ulrike Sattler, and Rudi Studer.
Query Answering for OWL-DL with Rules.
Journal of Web Semantics: Science, Services and Agents on the World Wide Web, 3(1):41–60, 2005.

Boris Motik.
Reasoning in Description Logics using Resolution and Deductive Databases.

PhD thesis, Univesität Karlsruhe (TH), Karlsruhe, Germany, January 2006.

Martin O'Connor, Holger Knublauch, Samson Tu, Benjamin Grosof, Mike Dean, William Grosso, and Mark Musen.
Supporting Rule System Interoperability on the Semantic Web with SWRL.
In *Proceedings of the 4th International Semantic Web Conference (ISWC 2005)*, volume 3729 of *LNCS*, pages 974–986. Springer, 2005.

Racer Systems GmbH & Co. KG.
RacerPro User's Guide Version 1.9.2, October 2007.

Evren Sirin, Bijan Parsia, Bernardo Cuenca-Grau, Aditya Kalyanpur, and Yarden Katz.
Pellet: A practical OWL-DL reasoner.
Journal of Web Semantics, 5(2):51–53, 2007.

Jan Wielemaker, Zhisheng Huang, and Lourens van der Meij.
SWI-Prolog and the Web.
Theory and Practice of Logic Programming, 8:363–392, 2008.

Zhe Wu, George Eadon, Souripriya Das, Eugene Inseok Chong, Vladimir Kolovski, Melliyal Annamalai, and Jagannathan Srinivasan.
Implementing and Inference Engine for RDFS/OWL Constructs and User-Defined Rules in Oracle.
In *Proccedings of the 24th International Conference on Data Engineering*, pages 1239–1248. IEEE, 2008.