Unit 3 — ASP for the Semantic Web

Giovambattista (aka GB) Ianni

Dipartimento di Matematica, Università della Calabria

Reasoning Web Summer School 2009

Unit Outline

1. DL-Programs
2. HEX-Programs
3. Other Linguistic Extension of ASP in the Direction of Semantic Web
4. Other Semantic Web Enabled Systems Based on ASP
5. Future Directions of ASP

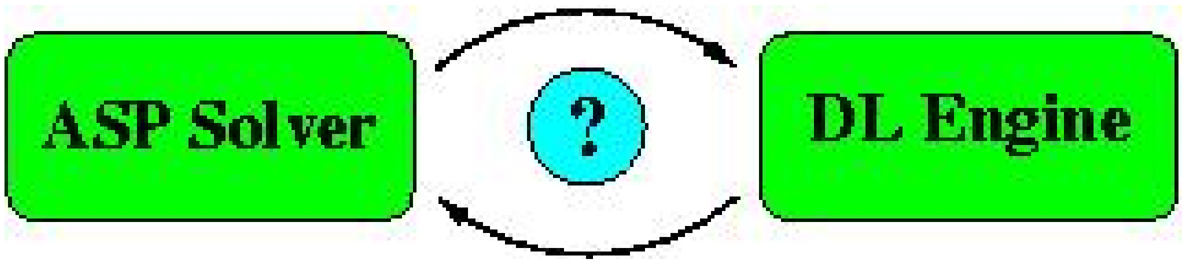
ASP vs DL: the realm of DL-Programs

Two main approaches

- Translate DLs to ASP programs
- Integrate ASP with DLs

A knowledge base KB formed by two sides:

- a logic program P
- a description logic knowledge base L.
- L is accessible from P by means of *dl-atoms*



DL-program example - I

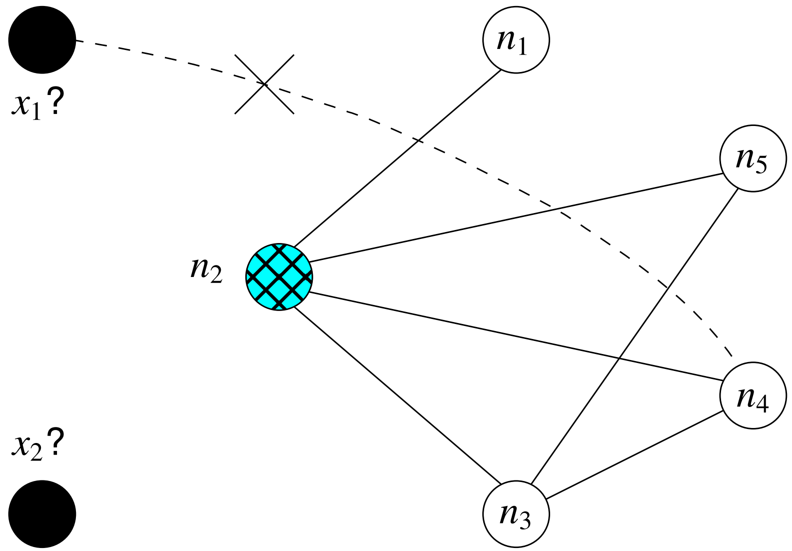


Figure: Hightraffic network

Example

Input: a DL ontology representing a network of nodes, possibly containing *high traffic nodes* (e.g. n_2), some new nodes to be added (e.g. x_1, x_2)

Output: A new connection for each of the new nodes, not causing new high traffic nodes.

DL-program example - II

L, describing the given network

$$
\begin{aligned}
&\geq 1\ wired \sqsubseteq Node; \quad \top \sqsubseteq \forall wired.Node; \quad wired = wired^{-};\\
&\geq 4\ wired \sqsubseteq HighTrafficNode; \quad n_1 \neq n_2 \neq n_3 \neq n_4 \neq n_5;\\
&Node(n_1); \quad Node(n_2); \quad Node(n_3); \quad Node(n_4); \quad Node(n_5);\\
&wired(n_1, n_2); \quad wired(n_2, n_3); \quad wired(n_2, n_4);\\
&wired(n_2, n_5); \quad wired(n_3, n_4); \quad wired(n_3, n_5).
\end{aligned}
$$

- Defines high traffic nodes
- Enforces unique name assumption on nodes

DL-program example - III

P, describing how to add nodes

$$
\begin{aligned}
newnode(x_1).& \\
newnode(x_2).excl(x_1, n_4).& \\
overloaded(X) &\leftarrow \mathrm{DL}[wired \uplus connect; HighTrafficNode](X). \\
connect(X,Y) &\leftarrow newnode(X), \mathrm{DL}[Node](Y), \\
&\quad\; not\ overloaded(Y), not\ excl(X,Y). \\
excl(X,Y) &\leftarrow connect(X,Z), \mathrm{DL}[Node](Y), Y \neq Z. \\
excl(X,Y) &\leftarrow connect(Z,Y), newnode(Z), newnode(X), Z \neq X.
\end{aligned}
$$

- $newnode(x) \Rightarrow x$ should be added to the network
- $overload(x) \Rightarrow x$ turns out to be overloaded in the current interpretation
- $connect(x,y) \Rightarrow x$ should be connected to y
- $excl(x,y) \Rightarrow x$ should *not* be connected to y

dl-atoms

The dl-atom device

- Can specify a query to L:

$$\mathrm{DL}[\mathit{Node}](Y)$$

$\mathrm{DL}[\mathit{Node}](y)$ true for y s.t. $L \models \mathit{Node}(y)$.

- Can push knowledge to L before querying:

$$\mathrm{DL}[\mathit{wired} \uplus \mathit{connect}; \mathit{HighTrafficNode}](X).$$

$\mathrm{DL}[\mathit{wired} \uplus \mathit{connect}; \mathit{HighTrafficNode}](x)$ true in an interpretation I of P for x s.t. $L \cup C \models \mathit{HighTrafficNode}(x)$ for $C = \{\mathit{wired}(x,y) \mid \mathit{connect}(x,y) \in I\}$.

also $\cup\!\!\!\!-$, $\cap\!\!\!\!-$ available.

Answer Sets of our example

Encoding of P

$$
\begin{aligned}
&newnode(x_1).\\
newnode(x_2).excl(x_1, n_4).&\\
overloaded(X) &\leftarrow \mathrm{DL}[wired \uplus connect; HighTrafficNode](X).\\
connect(X, Y) &\leftarrow newnode(X), \mathrm{DL}[Node](Y),\\
&\quad\; not\ overloaded(Y), not\ excl(X, Y).\\
excl(X, Y) &\leftarrow connect(X, Z), \mathrm{DL}[Node](Y), Y \neq Z.\\
excl(X, Y) &\leftarrow connect(Z, Y), newnode(Z), newnode(X), Z \neq X.
\end{aligned}
$$

- We get $M_1 = \{connect(x_1, n_1), connect(x_2, n_3), \ldots\}, M_2 = \{\ldots$

DL-programs properties

- Bidirectional: can push knowledge from P to L and pull knowledge from L to P;
- Clear separation between L and P: allows devising an integrated system based on existing ASP and DL reasoners.
- Two semantics available: strong/weak answer set & well-founded semantics.
- Can introduce on top of L most nonmonotonic features (e.g. (E)CWA [Reiter, 1978], Default Reasoning [Reiter, 1980])

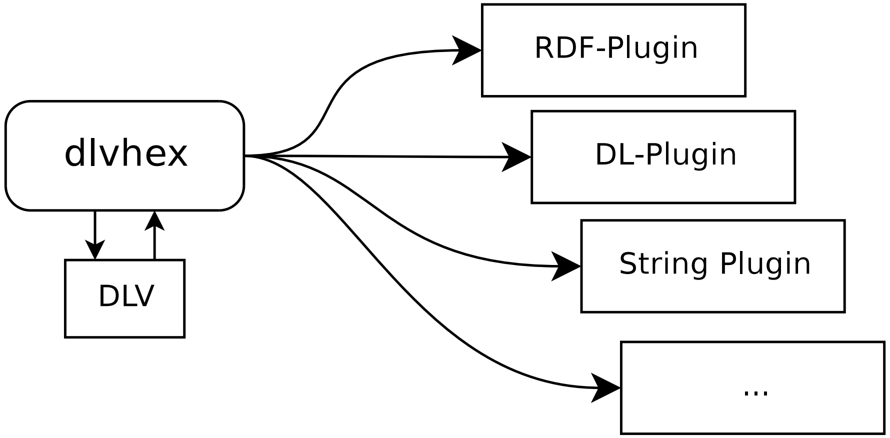
More information on [Eiter *et al.*, 2008b] and in previous Reasoning Web School lectures [Eiter *et al.*, 2006], [Eiter *et al.*, 2008a]

Semantics

New generalized definitions

- Herbrand base *not* defined in terms of constants explicitly mentioned in P
- For the answer set semantics case,
  - Least model of positive DL-program
  - Iterative least model of stratified DL-program
  - Answer sets of general DL-program
- For the well-founded semantics case, notion of unfoundedness lifted to dl-atoms.

The realm of HEX-programs



- Extends dl-programs from one-to-one coupling to many-one.
  - Outer Knowledge sources are not constrained to DL knowledge bases only.
- P can interface multiple external sources of knowledge of any sort via so called *external atoms*
- P has *higher order* atoms

An example

$$
\begin{aligned}
subRelation(brotherOf, relativeOf).\\
brotherOf(john, al).\\
relativeOf(john, joe).\\
brotherOf(al, mick).\\
invites(john, X) \vee skip(X) \leftarrow X \neq john, \&reach[relativeOf, john](X).\\
R(X, Y) \leftarrow subRelation(P, R), P(X, Y).\\
someInvited \leftarrow invites(john, X).\\
\leftarrow not\ someInvited.\\
\leftarrow \&degs[invites](Min, Max), Max > 2.
\end{aligned}
$$

Example

Input: Some data about *John*'s neighborhood

Output: Possible picks for persons John might want to invite, according to some constraints

Higher order atoms

$$\begin{aligned}
subRelation(brotherOf, relativeOf).\\
R(X,Y) \leftarrow subRelation(P,R), P(X,Y).\\
brotherOf(john, al).\\
relativeOf(john, joe).\\
brotherOf(al, mick).
\end{aligned}$$

The device of higher order atoms

- Predicate names can be variables
- Constants can appear both as terms values or as predicate values
- Allows (comfortable) meta-reasoning

$$\begin{aligned}
subRelation(brotherOf, relativeOf). \quad &\Rightarrow \quad relativeOf(X,Y) \leftarrow brotherOf(X,Y).\\
R(X,Y) \leftarrow subRelation(P,R), P(X,Y).
\end{aligned}$$

External atoms

$$\&reach[relativeOf, john](X) \tag{1}$$

$$\&degs[invites](Min, Max) \tag{2}$$

The device of external atoms

- Each external predicate is tied to a corresponding evaluation function
- E.g. $\&reach$ corresponds to $f_{\&reach}$.

For a given interpretation I, $I \models \&reach[relativeOf, john](x)$ if $f_{\&reach}(I, relativeOf, john) = 1$

Semantics

Higher order atoms

Generalized Herbrand base: an higher order atom $T_0(T_1, \ldots, T_n)$ can be grounded to $t_0(t_1, \ldots, t_n)$.

External atoms

Given a = $\&g[y_1, \ldots, y_n](x_1, \ldots, x_m)$, then $I \models a$, if and only if $f_{\&g}(I, y_1, \ldots, y_n, x_1, \ldots, x_m) = 1$.

Notion of answer sets

Similar to strong answer sets of dl-programs, but based on the FLP [Faber *et al.*, 2004] reduct.

- For interpretation I and program P, $fP^I = \{r \in P \mid$ the body of P is satisfied in $I\}$

Features and Applications of HEX-programs

- System available [1]
- Features a powerful SDK for developing own external atoms
- SW knowledge can be accessed via the $\&rdf$ atom. Access to reasoners via the $\&dlC$ and $\&dlR$ atom (similar to dl-atoms).
- Many other plugins available (string manipulation, aggregates etc.)

Applications

- ASP reasoning (possibly nonmonotonic) on ontologies [Hoehndorf *et al.*, 2007], [Bodenreider *et al.*, 2008]
- Planning [Nieuwenborgh *et al.*, 2007]

[1] `http://www.kr.tuwien.ac.at/research/systems/dlvhex/`

Loose vs Tight coupling

- HEX-programs take the loose coupling approach
  - L appears as a reasoning service for P.
- Knowledge bases of different semantics can be however integrated on different levels, e.g. on a *tight* coupling approach

Tight coupling

Tight coupling (e.g. $\mathcal{D}L+$log [Rosati, 2006]), features a single interpretative structure embracing both P and L.

Tight Coupling vs Loose Coupling - II

Example

$$p(X) \leftarrow \mathrm{DL}[Person](X)$$

Single answer set $M_1 = \{p(alice), p(bob), \ldots\}$, containing all the $p(x)$ s.t. $L \models Person(x)$

$$p(X) \leftarrow Person(X)$$

Has infinitely many stable models. A stable model M_I for each consistent interpretation I of L.

Other coupling methods exists. A thorough comparison can be found in [Eiter *et al.*, 2008a], [de Bruijn *et al.*, 2006].

Other SW Related ASP systems

OntoDLV

An extension of DLV with ontology modelling features living under native ASP semantics [Ricca *et al.*, 2008]

GiaBATA

A SPARQL enabled RDF store, based on dlvhex and DLV^{DB}
[_ et al. 2009a; _ et al. 2009b]

DLT

An ASP frontend allowing contexts, higher order, and frame syntax
[Calimeri and _ 2006; Alviano et al. 2008]

Current state-of-the-art

Semantics

- **Introduction of Function Symbols** [Syrjänen, 2001],[Bonatti, 2004],[Calimeri *et al.*, 2008],[Šimkus and Eiter, 2007],[Eiter and Šimkus, 2009]
- **Modularity** [Dao-Tran *et al.*, 2009],[Janhunen *et al.*, 2007],[Oikarinen and Janhunen, 2008],[Tari *et al.*, 2005],[Balduccini, 2007],[Baral *et al.*, 2006],[Calimeri and Ianni, 2006],[Polleres *et al.*, 2006],[Analyti *et al.*, 2008]
- **Study of equivalence** [Lifschitz *et al.*, 2001],[Gelfond, 2008],[Eiter *et al.*, 2007],[Eiter *et al.*, 2004],[Woltran, 2008]

Engineering

- **Debuggers** [El-Khatib *et al.*, 2005],[Brain and Vos, 2005] **and in-database evaluation** [Terracina *et al.*, 2008]
- **See the SEA workshop series** `http://sea07.cs.bath.ac.uk/`

Current state-of-the-art II

Scalability

- **Intelligent and lazy grounders** [Calimeri *et al.*, 2008],[Gebser *et al.*, 2008],[Palù *et al.*, 2008], [Lefévre and Nicolas, 2008]
- **See the Answer Set Programming competition**
  `http://www.cs.kuleuven.be/ dtai/events/ASP-competition/`

Session is over

Thanks!

You can play with ASP solver on this nice web interface
`http://asptut.gibbi.com`

References I

Mario Alviano, Giovambattista Ianni, Marco Marano, and Alessandra Martello.

Versatile semantic modeling of frame logic programs under answer set semantics.

In John Domingue and Chutiporn Anutariya, editors, *ASWC*, volume 5367 of *Lecture Notes in Computer Science*, pages 106–121. Springer, 2008.

Anastasia Analyti, Grigoris Antoniou, and Carlos Viegas Damásio.

A principled framework for modular web rule bases and its semantics.

In *Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning (KR2008)*. AAAI Press, September 2008.

References II

Marcello Balduccini.

Modules and Signature Declarations for A-Prolog: Progress Report.

In Marina de Vos and Torsten Schaub, editors, *Informal Proceedings of the 1st International Workshop on Software Engineering for Answer Set Programming, Tempe, AZ (USA), May 2007*, 2007.

Available at `http://sea07.cs.bath.ac.uk/downloads/sea07-proceedings.pdf`.

Chitta Baral, Juraj Dzifcak, and Hiro Takahashi.

Macros, Macro calls and Use of Ensembles in Modular Answer Set Programming.

In *Proceedings of the 22th International Conference on Logic Programming (ICLP 2006)*, number 4079 in LNCS, pages 376–390. Springer, 2006.

References III

- O. Bodenreider, Z. Coban, M. C. Doganay, E. Erdem, and H. Kosucu.
  A preliminary report on answering complex queries related to drug discovery using answer set programming.
  In *In Proc. of ALPSWS'08.*, 2008.
- Piero A. Bonatti.
  Reasoning with infinite stable models.
  Artificial Intelligence, 156(1):75–111, 2004.
- Martin Brain and Marina De Vos.
  Debugging logic programs under the answer set semantics.
  In *Answer Set Programming*, 2005.
- Francesco Calimeri and Giovambattista Ianni.
  Template programs for Disjunctive Logic Programming: An operational semantics.
  AI Communications, 19(3):193–206, 2006.

References IV

Francesco Calimeri, Susanna Cozza, Giovambattista Ianni, and Nicola Leone.

Computable Functions in ASP: Theory and Implementation.

In *Logic Programming, 24th International Conference, ICLP 2008, Udine, Italy, December 9-13 2008, Proceedings*, volume 5366 of *LNCS*, pages 407–424. Springer, 2008.

Minh Dao-Tran, Thomas Eiter, Michael Fink, and Thomas Krennwallner.

Modular nonmonotonic logic programming revisited.

In P. Hill and D.S. Warren, editors, *Proceedings 25th International Conference on Logic Programming (ICLP 2009)*, volume 5649 of *LNCS*, pages 145–159. Springer, July 2009.

References V

Jos de Bruijn, Thomas Eiter, Axel Polleres, and Hans Tompits.
On representational issues about combinations of classical theories with nonmonotonic rules.
In Jérôme Lang, Fangzhen Lin, and Ju Wang, editors, *KSEM*, volume 4092 of *Lecture Notes in Computer Science*, pages 1–22. Springer, 2006.

Thomas Eiter and Mantas Šimkus.
Bidirectional answer set programs with function symbols.
In C. Boutilier, editor, *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI-09)*. AAAI Press/IJCAI, 2009.

T. Eiter, M. Fink, H. Tompits, and S. Woltran.
Simplifying logic programs under uniform and strong equivalence.
In Ilkka Niemelä and Vladimir Lifschitz, editors, *Proceedings of the 7th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR 2004)*, number 2923 in LNCS, pages 87–99. Springer, 2004.

References VI

Thomas Eiter, Giovambattista Ianni, Axel Polleres, Roman Schindlauer, and Hans Tompits.

Reasoning with rules and ontologies.

In Pedro Barahona, François Bry, Enrico Franconi, Ulrike Sattler, and Nicola Henze, editors, *Reasoning Web 2006*, volume 4126 of *LNCS*, pages 93–127. Springer, September 2006.

Thomas Eiter, Michael Fink, and Stefan Woltran.

Semantical Characterizations and Complexity of Equivalences in Answer Set Programming.

ACM Trans. Comput. Log., 8(3), 2007.

Article 17 (53 + 11 pages).

References VII

Thomas Eiter, Giovambattista Ianni, Thomas Krennwallner, and Axel Polleres.

Rules and Ontologies for the Semantic Web.

In Cristina Baroglio, Piero A. Bonatti, Jan Maluszynski, Massimo Marchiori, Axel Polleres, and Sebastian Schaffert, editors, *Reasoning Web: 4th International Summer School 2008, Venice Italy, September 7-11, 2008, Tutorial Lectures*, volume 5224 of *LNCS*, pages 1–53. Springer, September 2008.

Slides available at `http://rease.semanticweb.org/`.

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, 2008.

References VIII

Omar El-Khatib, Enrico Pontelli, and Tran Cao Son.

Justification and debugging of answer set programs in asp.

In *AADEBUG*, pages 49–58, 2005.

W. Faber, N. Leone, and G. Pfeifer.

Recursive Aggregates in Disjunctive Logic Programs: Semantics and Complexity.

In *Proceedings JELIA-04*, pages 200–212, 2004.

Martin Gebser, Roland Kaminski, Benjamin Kaufmann, Max Ostrowski, Torsten Schaub, and Sven Thiele.

Engineering an Incremental ASP Solver.

In M.G. de La Banda and E. Pontelli, editors, *Proceedings 24th International Conference on Logic Programming (ICLP 2008)*, number 5366 in LNCS, pages 190–205. Springer, 2008.

References IX

M. Gelfond.

Answer sets.

In B. Porter F. van Harmelen, V. Lifschitz, editor, *Handbook of Knowledge Representation*, chapter 7, pages 285–316. Elsevier, 2008.

Robert Hoehndorf, Frank Loebe, Janet Kelso, and Heinrich Herre.

Representing default knowledge in biomedical ontologies: Application to the integration of anatomy and phenotype ontologies.

BMC Bioinformatics, 8(1):377, 2007.

Giovambattista Ianni, Thomas Krennwallner, Alessandra Martello, and Axel Polleres.

A Rule System for Querying Persistent RDFS Data.

In Lora Arroyo and Paolo Traverso, editors, *The Semantic Web: Research and Applications, 6th European Semantic Web Conference, ESWC 2009, Heraklion, Greece, May 31–June 4, 2009, Proceedings*, volume 5554 of *LNCS*, pages 857–862. Springer, 2009.

References X

- Giovambattista Ianni, Thomas Krennwallner, Alessandra Martello, and Axel Polleres.

  Dynamic Querying of Mass-Storage RDF Data with Rule-Based Entailment Regime.

  In Abraham Bernstein and David Karger, editors, *8th International Semantic Web Conference (ISWC 2009), Washington D.C., USA, 25–29 October, 2009*, LNCS. Springer, October 2009.

  To appear.

- Tomi Janhunen, Emilia Oikarinen, Hans Tompits, and Stefan Woltran.

  Modularity Aspects of Disjunctive Stable Models.

  In *Proceedings of the 9th International Conference on Logic Programming and Nonmonotonic Reasoning*, volume 4483 of *LNCS*, pages 175–187. Springer, May 2007.

References XI

Claire Lefévre and Pascal Nicolas.

Integrating grounding in the search process for answer set computing.

In *ASPOCP: Answer Set Programming and Other Constraint Paradigms*, pages 89–103, 2008.

Vladimir Lifschitz, David Pearce, and Agustín Valverde.

Strongly equivalent logic programs.

ACM Trans. Comput. Log., 2(4):526–541, 2001.

Davy Van Nieuwenborgh, Thomas Eiter, and Dirk Vermeir.

Conditional Planning with External Functions.

In Chitta Baral, Gerhard Brewka, and John S. Schlipf, editors, *Proceedings of the 9th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR 2007)*, volume 4483 of *LNCS*, pages 214–227. Springer, 2007.

References XII

Emilia Oikarinen and Tomi Janhunen.
Achieving compositionality of the stable model semantics for Smodels programs.
Theory and Practice of Logic Programming, 8(5–6):717–761, November 2008.

A. Dal Palù, A. Dovier, E. Pontelli, and G. Rossi.
Gasp: Answer set programming with lazy grounding.
In *LaSh 2008: LOGIC AND SEARCH - Computation of structures from declarative descriptions*, 2008.

Axel Polleres, Cristina Feier, and Andreas Harth.
Rules with Contextually Scoped Negation.
In *Proceedings of the 3rd European Conference on Semantic Web (ESWC 2006)*, volume 4011 of *LNCS*, pages 332–347. Springer, 2006.

References XIII

R. Reiter.

On Closed-World Databases.

In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 55–76. Plenum Press, New York, 1978.

Raymond Reiter.

A Logic for Default Reasoning.

Artificial Intelligence, 13(1–2):81–132, 1980.

Francesco Ricca, Lorenzo Gallucci, Roman Schindlauer, Tina Dell'armi, Giovanni Grasso, and Nicola Leone.

OntoDLV: An ASP-based System for Enterprise Ontologies.

Journal of Logic and Computation, 2008.

doi:10.1093/logcom/exn042.

References XIV

Riccardo Rosati.

$\mathcal{DL}+log$: Tight Integration of Description Logics and Disjunctive Datalog.

In *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning* (*KR 2006*), pages 68–78. AAAI Press, 2006.

Mantas Šimkus and Thomas Eiter.

FDNC: Decidable non-monotonic disjunctive logic programs with function symbols.

In N. Dershowitz and A. Voronkov, editors, *Proceedings 14th International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR 2007)*, number 4790 in LNCS, pages 514–530. Springer, 2007.

Extended Paper to appear in *ACM Trans. Computational Logic*.

References XV

Tommi Syrjänen.

Omega-restricted logic programs.

In *Proceedings of the 6th International Conference on Logic Programming and Nonmonotonic Reasoning*, Vienna, Austria, September 2001. Springer-Verlag.

Luis Tari, Chitta Baral, and Saadat Anwar.

A Language for Modular Answer Set Programming: Application to ACC Tournament Scheduling.

In *Proceedings of the 3Proceedings of the 3rd International ASP'05 Workshop, Bath, UK, 27th–29th July 2005*, volume 142 of *CEUR Workshop Proceedings*, pages 277–293. CEUR WS, July 2005.

Giorgio Terracina, Nicola Leone, Vincenzino Lio, and Claudio Panetta.

Experimenting with recursive queries in database and logic programming systems.

TPLP, 8(2):129–165, 2008.

References XVI

Stefan Woltran.

A common view on strong, uniform, and other notions of equivalence in answer-set programming.

Theory and Practice of Logic Programming, 8(2):217–234, 2008.