



Data, Information and Process Integration
with Semantic Web Services

DIP

Data, Information and Process Integration with Semantic Web Services

FP6 - 507483

Deliverable

WP 7: Technology Watch and Standardisation

D7.9

RIF contribution

Jos de Bruijn

Axel Polleres

Darko Anicic

Holger Lausen

Michael Stollberg

Martin Hepp



EXECUTIVE SUMMARY

One of the major outcomes of the DIP project is the WSML language, which pioneered the combination of Description Logics and rules in a unifying language framework for the Web. The World Wide Web Consortium (W3C) has standardized a Description Logic language for the Web, in the form of OWL. The W3C has recently started another working group, called the Rule Interchange Format (RIF) Working Group, to standardize a rules language for the Web. In order to benefit from the experiences with the WSML language, and the use of the project results beyond the project, several of the project members are participating in the RIF working group. This document reports on the activities of the DERI Innsbruck in the W3C RIF working group.

Document Information

IST Project Number	FP6 – 507483		Acronym	DIP
Full title	Data, Information, and Process Integration with Semantic Web Services			
Project URL	http://dip.semanticweb.org			
Document URL				
EU Project officer	Werner Janusch			
Deliverable	Number	7.9	Title	RIF contribution
Work package	Number	7	Title	Technology Watch and Standardisation

Date of delivery	Contractual	M 36	Actual	M36
Status	version. 1.0		final	<input checked="" type="checkbox"/>
Nature	Prototype <input type="checkbox"/> Report <input checked="" type="checkbox"/> Dissemination <input type="checkbox"/> Ontology <input type="checkbox"/>			
Dissemination Level	Public <input checked="" type="checkbox"/> Consortium <input type="checkbox"/>			

Authors (Partner)	Jos de Bruijn (UIBK), Axel Polleres (UIBK), Darko Anicic Holger Lausen (UIBK), Michael Stollberg (UIBK), Martin Hepp (UIBK)			
Responsible Author	Jos de Bruijn		Email	jos.debruijn@deri.org
	Partner	UIBK	Phone	+435125076475

Abstract (for dissemination)	One of the major outcomes of the DIP project is the WSML language, which pioneered the combination of Description Logics and rules in a unifying language framework for the Web. In order to benefit from the experiences with the WSML language, and the use of the project results beyond the project, several of the project members are participating in the RIF working group. This document reports on the activities of the DERI Innsbruck in the W3C RIF working group.		
Keywords	RIF, semantic Web rules language		

Version Log					
Issue Date	Rev No.	Author	Change		
2006-11-02	0.1	Jos de Bruijn	Initial version		
2006-12-29	1.0	Jos de Bruijn	Implementation of the reviewer comments		
Reviewer					
		Elmar Dörner	Email	elmar.dorner@sap.com	
		Partner	SAP	Phone	+49 721 6902 31
		Christian de Sainte Marie	Email	csma@ilog.fr	
		Partner	ILOG	Phone	+33 1 49082981

Project Consortium Information

Partner	Acronym	Contact
National University of Ireland Galway	NUIG  National University of Ireland, Galway <i>Ollscoil na hÉireann, Gaillimh</i>	Dr. Sigurd Harand Digital Enterprise Research Institute (DERI) National University of Ireland, Galway Galway Ireland Email: sigurd.harand@deri.org Tel: +353 91 495112
Fundacion De La Innovacion.Bankinter	Bankinter  .com	Monica Martinez Montes Fundacion de la Innovation. BankInter Paseo Castellana, 29 28046 Madrid, Spain Email: mmtnez@bankinter.es Tel: 916234238
British Telecommunications Plc.	BT 	Dr John Davies BT Exact (Orion Floor 5 pp12) Aadastral Park Martlesham Ipswich IP5 3RE, United Kingdom Email: john.nj.davies@bt.com Tel: +44 1473 609583
Swiss Federal Institute of Technology, Lausanne	EPFL 	Prof. Karl Aberer Distributed Information Systems Laboratory École Polytechnique Fédérale de Lausanne Bât. PSE-A 1015 Lausanne, Switzerland Email : Karl.Aberer@epfl.ch Tel: +41 21 693 4679
Essex County Council	Essex  Essex County Council	Mary Rowllatt, Essex County Council PO Box 11, County Hall, Duke Street Chelmsford, Essex, CM1 1LX United Kingdom. Email: maryr@essexcc.gov.uk Tel: +44 (0)1245 436524
Forschungszentrum Informatik	FZI  FZI	Andreas Abecker Forschungszentrum Informatik Haid-und-Neu Strasse 10-14 76131 Karlsruhe Germany Email: abecker@fzi.de Tel: +49 721 9654 0
Institut für Informatik, Leopold-Franzens Universität Innsbruck	UIBK  universität innsbruck	Prof. Dieter Fensel Institute of computer science University of Innsbruck Technikerstr. 25 A-6020 Innsbruck, Austria Email: dieter.fensel@deri.org Tel: +43 512 5076485

Partner	Acronym	Contact
ILOG SA	<p>ILOG</p>  <p>Changing the rules of business</p>	<p>Christian de Sainte Marie 9 Rue de Verdun, 94253 Gentilly, France Email: cma@ilog.fr Tel: +33 1 49082981</p>
inubit AG	<p>Inubit</p> 	<p>Torsten Schmale inubit AG Lützowstraße 105-106 D-10785 Berlin Germany Email: ts@inubit.com Tel: +49 30726112 0</p>
Intelligent Software Components, S.A.	<p>iSOCO</p> 	<p>Dr. V. Richard Benjamins, Director R&D Intelligent Software Components, S.A. Pedro de Valdivia 10 28006 Madrid, Spain Email: rbenjamins@isoco.com Tel. +34 913 349 797</p>
MDR Partners	<p>MDR</p> 	<p>Rob Davies MDR Partners 8 St. Andrew Street Hertford, Herts. United Kingdom, SG14 1JA, Email: rob.davies@mdrpartners.com +44 (0)208 8763121</p>
Hanival Internet Services GmbH	<p>HANIVAL</p> 	<p>Alexander Wahler Hanival Internet Services GmbH Kirchengasse 13/1a A-1070 Wien Email: wahler@niwa.at Tel:+43(0)1 3195843-11 </p>
The Open University	<p>OU</p> 	<p>Dr. John Domingue Knowledge Media Institute The Open University, Walton Hall Milton Keynes, MK7 6AA United Kingdom Email: j.b.domingue@open.ac.uk Tel.: +44 1908 655014</p>
SAP AG	<p>SAP</p> 	<p>Dr. Elmar Dörner SAP Research, CEC Karlsruhe SAP AG Vincenz-Priessnitz-Str. 1 76131 Karlsruhe, Germany Email: elmar.dorner@sap.com Tel: +49 721 6902 31</p>

Partner	Acronym	Contact
Sirma AI Ltd.	<p>Sirma</p> 	<p>Atanas Kiryakov, Ontotext Lab, - Sirma AI EAD Office Express IT Centre, 3rd Floor 135 Tzarigradsko Chausse Sofia 1784, Bulgaria Email: atanas.kiryakov@sirma.bg Tel.: +359 2 9768 303</p>
Vrije Universiteit Brussel	<p>VUB</p>  <p>Vrije Universiteit Brussel</p>	<p>Pieter De Leenheer Starlab- VUB Vrije Universiteit Brussel Pleinlaan 2, G-10 1050 Brussel ,Belgium Email: Pieter.De.Leenheer@vub.ac.be Tel.: +32 (0) 2 629 3749</p>

TABLE OF CONTENTS

EXECUTIVE SUMMARY	I
TABLE OF CONTENTS	VI
LIST OF ABBREVIATIONS.....	VIII
1 INTRODUCTION.....	1
2 WORKING GROUP PARTICIPATION	3
3 CREATION OF THE WORKING GROUP	3
3.1 W3C Workshop on Rule Languages for Interoperability.....	3
3.2 The Web Rule Language WRL	5
4 WORKING GROUP MEETINGS.....	6
4.1 Face-To-Face Meetings	6
4.2 Telephone Conferences	7
5 PROGRESS TO DATE	8
6 OUTLOOK	9
7 APPENDIX A. THE WSML RULE LANGUAGES FOR THE SEMANTIC WEB.....	10
7.1 Abstract.....	10
7.2 Introduction	10
7.3 WSML	11
7.3.1 Examples	12
7.3.2 Syntaxes for WSML.....	12
7.4 Relation with other efforts.....	13
7.4.1 SWSL	13
7.4.2 RuleML.....	13
7.4.3 SWRL.....	14
7.5 Summary of Position	14
7.6 References	15
7.7 Footnotes	16
8 APPENDIX B. WSML - A LANGUAGE FRAMEWORK FOR SEMANTIC WEB SERVICES	17
8.1 Abstract.....	17
8.2 Introduction	17
8.3 Web Service Modeling Language	18
8.3.1 Illustrating Example	18
8.3.2 WSML Layering.....	19

8.3.3 WSML and the Web	20
8.4 Summary of Position	21
8.5 References	22

LIST OF ABBREVIATIONS

OWL - Web Ontology Language (W3C standard)

RIF - Rule Interchange Formats (W3C Working Group)

W3C - World Wide Web Consortium

WRL - Web Rule Language (W3C submission)

WSML - Web Service Modeling Language

1 INTRODUCTION

With the standardization of the Web Ontology Language OWL, the ontology layer in the Semantic Web layer cake has been filled in. Formally, OWL is based on an expressive Description Logic language, namely SHOIN. This language is an expressive, though decidable, fragment of classical First-Order Logic (FOL). The language is based on the two variable fragment of FOL, which essentially means that only two variables may be used under the scope of any quantifier. This results in certain expressive limitations, because only formulas with the so-called tree shape are allowed in this language. On the other hand, there are many features in the language which make it very expressive, such as unlimited use of disjunction, classical negation and existential quantification, and the use of equality.

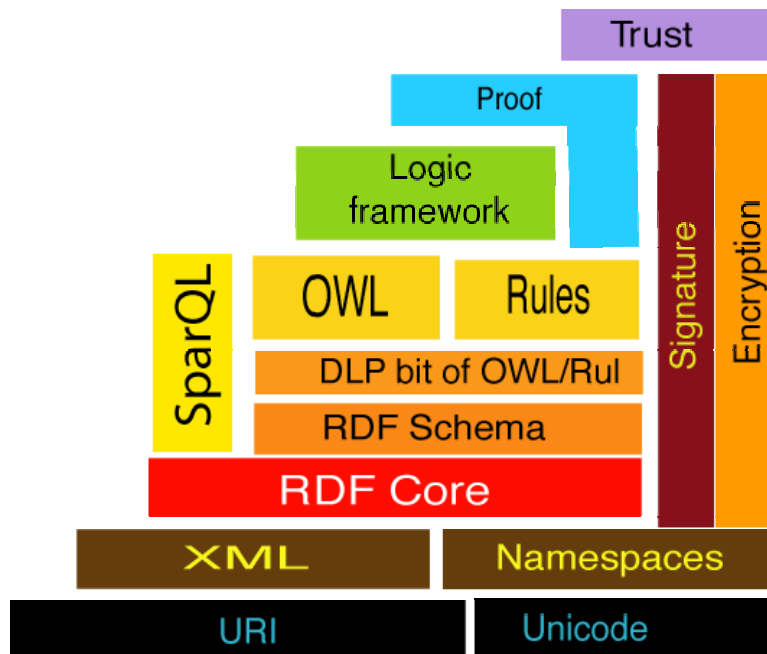
Description Logics have been used for some time in the knowledge representation community. Another language paradigm which has been used extensively in knowledge representation, even before Description Logics were around, is that of rules. Generally speaking, the expressiveness of rules is complementary to that of Description Logics. There is no limitation in the number of variables being used, and rules are not required to have a tree shape, but rather an "if...then" construction. Features which are generally not present in rules languages (with some exceptions) are disjunction and existential quantification. Also, the use of equality is generally not allowed, because this would influence reasoning in a negative way. Generally speaking, reasoning with rules is more efficient than reasoning with expressive Description Logics, except when expressive extensions are considered, such as the use of function symbols. Another, an extension for rules languages is default (nonmonotonic) negation, which is beyond classical logics.

There has been some investigation in the extension of Description Logics with rules. However, the straightforward extension of even a simple description logic (a subset of OWL) with a very simple rules language leads to an undecidable formalism. Combinations of Description Logics with more expressive rules languages which include negation are a topic of ongoing research, but such combinations are generally not straightforward and very hard to reason with. The reason for the difficulty in combining the formalisms, even in the simple case, is the earlier mentioned complementarity of the expressive power of both formalisms.

It has been recognized, from the early beginning, that rules play an important role on the Semantic Web. It is even the case that the earliest Semantic Web prototypes, Ontobroker and SHOE, used rules as their representation formalism. Still, stand-alone and embedded rules engines (e.g. Ontobroker, XSB, Jena, Sesame, TRIPLE, FLORA-2) are at the most used reasoners on the semantic Web and many applications use rules (e.g. F-Logic rules, RDF rules as in Jena2) for knowledge representation on the semantic Web. There is, however, no standard language for writing rules on the Semantic Web.

The WSML language, developed within the DIP project, has pioneered the use of rules and Description Logics within a common language framework, allowing interaction between the two formalisms through a common subset, thereby allowing to benefit from reasoning algorithms and tools created for both paradigms. This pioneering work has been picked up by the W3C at the W3C Workshop on Rule Languages for Interoperability, held in April 2005, and has led to the following revised layer cake,

which recognizes the existence of a rules layer besides the description logic layer, with a common core which enables interoperability:



As mentioned earlier, there is, at the time of writing, no standard language for writing rules on the semantic Web. Thus, the "Rules" box in the above figure is still empty. The recently formed Rule Interchange Format (RIF) working group at W3C aims to fill in this box. It is important for the DIP project, and the semantic Web community as a whole, that the ideas pioneered within DIP find their way into this working group.

In order to achieve this, a number of members of the DIP consortium have submitted a proposal for a Web Rule Language (WRL)¹, based on the rules component of WSMML. This proposal is considered as an input to the RIF working group, which started its work in December 2005. However, this is not enough to ensure that the ideas developed within DIP are taken up in this working group. An active participation by DIP consortium members has been, and is still, required.

This document reports the activities of DERI Innsbruck in the RIF working group, from the creation of the working group in 2005, up to the end of 2006. The two papers presented at the W3C rules workshop, in April 2005, are attached to this deliverable as an appendix.

¹ <http://www.w3.org/Submission/WRL/>

2 WORKING GROUP PARTICIPATION

Representatives of DERI Innsbruck are:

- Darko Anicic
- Jos de Bruijn (**principal**)
- Hong-Gee Kim
- Holger Lausen
- Axel Polleres

Additionally, a co-chair of the WSML working group, Michael Kifer, participates in the RIF working group as external expert.

The principal participant is responsible to represent the organization in case a vote is necessary. Each organization has exactly 1 vote.

Voting takes place in case a decision needs to be taken in the working group, and there is no consensus among the participants. Additionally, important decisions, such as the publication of a public working draft, are also voted upon.

3 CREATION OF THE WORKING GROUP

The WSML working group was very active in the process leading up to the creation of the RIF working group.

A W3C Workshop on Rule Languages for Interoperability (<http://www.w3.org/2004/12/rules-ws/>) was held in Washington DC, on April 27-28, 2005. The goal of this workshop was to find out whether there was enough interest in academia, as well as among rule vendors, to start a W3C working group on rules.

3.1 W3C Workshop on Rule Languages for Interoperability

DERI Innsbruck was represented by Dieter Fensel, Jos de Bruijn, and Holger Lausen. Two position papers were published in the proceedings of the workshop (<http://www.w3.org/2004/12/rules-ws/accepted>) and presented at the workshop. The two papers are attached to this deliverable in the form of an appendix. Abstracts of the papers are listed below:

H. Lausen, J. de Bruijn, A. Polleres, and D. Fensel. WSML - a Language Framework for Semantic Web Services

Abstract:

The Web Service Modeling Language (WSML) provides a framework of different language variants to describe semantic Web services. This paper presents the design

rationale and relation with existing language recommendations. WSML is a frame based language with an intuitive human readable syntax and XML and RDF exchange syntaxes, as well as a mapping to OWL. It provides different variants, allowing for open and closed world modeling; it is a fully-fledged ontology and rule language with defined variants grounded in well known formalisms, namely Datalog, Description Logic and Frame Logic. Taking the key aspects of WSML as a starting point, we rationalize the design decisions which we consider relevant in designing a proper layering of ontology and rule languages for the Semantic Web and semantic Web services.

URI of the full paper: <http://www.w3.org/2004/12/rules-ws/paper/44/>

J. de Bruijn, H. Lausen, A. Polleres, and D. Fensel. The WSML rule languages for the Semantic Web

Abstract:

The Web Service Modeling Language WSML provides a framework for the modeling of ontologies and semantic Web services based on the conceptual model of the Web Service Modeling Ontology. In this paper we describe the two rule-based WSML-variants and outline our position with respect to a rule language for the Semantic Web. The first rule-based WSML variant, WSML-Flight, semantically corresponds to the Datalog fragment of F-Logic, extended with inequality in the body and locally stratified negation under the Perfect model semantics. The second, WSML-Rule, is an extension of WSML-Flight to the logic programming subset of F-Logic which allows the use of function symbols and unsafe rules (i.e., there may be variables in rule heads which do not occur in the body).

URI of the full paper: <http://www.w3.org/2004/12/rules-ws/paper/128/>

The conclusion² of the workshop was that there was definitely interest in a W3C working group around rules. The mailing list was set up³ to further discuss the setup of the working group. DERI Innsbruck was very active in this discussion and helped writing a proposed charter porter working group, of which large parts are reflected in the final charter of the working group.

² Workshop report: <http://www.w3.org/2004/12/rules-ws/report/>

³ Public archive: <http://lists.w3.org/Archives/Public/public-rule-workshop-discuss/>

The charter⁴ was finalized in the fall of 2005 and the Rule Interchange Format (RIF) working group was established in December 2005, with the first face-to-face meeting in Burlingame, California, December 8-9, 2005.

3.2 The Web Rule Language WRL

In order to ensure that the rules working group would go in the right direction, a W3C member submission was created by the WSML working group and other interesting parties, and submitted to W3C. WRL was submitted to W3C by the following member organizations: DERI Innsbruck at the Leopold-Franzens-Universität Innsbruck, Austria, DERI Galway at the National University of Ireland, Galway, Ireland, The Open University, Software AG, Forschungszentrum Informatik (FZI), BT, and National Research Council Canada. Most of these are members of the DIP consortium.

The member submission was published online by W3C on September 9, 2005. This submission is considered by W3C as one of the inputs for to RIF working group.

The WRL submission⁵ consists of the following documents:

Jürgen Angele, Harold Boley, Jos de Bruijn, Dieter Fensel, Pascal Hitzler, Michael Kifer, Reto Krummenacher, Holger Lausen, Axel Polleres, and Rudi Studer. Web Rule Language (WRL)

Abstract:

The Web Rule Language WRL is a rule-based ontology language for the Semantic Web. The language is located in the Semantic Web stack next to the Description Logic based Ontology language OWL. WRL defines three variants, namely Core, Flight and Full. The Core variant marks the common fragment between WRL and OWL. WRL-Flight is a Datalog-based rule language. WRL-Full is a full-fledged rule language with function symbols and negation under the Well-Founded Semantics.

URI: <http://www.w3.org/Submission/WRL/>

Harold Boley, Jos de Bruijn, and Reto Krummenacher. WRL XML Schemas

Abstract:

This document contains a brief description of and references to the XML Schemas for the WRL XML syntax, as well as the relevant RuleML schemas for the XML syntax of the rules.

URI: <http://www.w3.org/Submission/WRL-xmlschemas/>

⁴ <http://www.w3.org/2005/rules/wg/charter>

⁵ <http://www.w3.org/Submission/2005/08/>

*Jos de Bruijn, Dieter Fensel, Pascal Hitzler, Michael Kifer, and Axel Polleres.
Relationship of WRL to relevant other technologies*

Abstract:

This document contains brief descriptions of the relationship between WRL and other selected relevant technologies.

URI: <http://www.w3.org/Submission/WRL-related/>

4 WORKING GROUP MEETINGS

The meetings of the RIF working group take place in two forms: there are weekly phone conferences and quarterly face-to-face meetings.

4.1 Face-To-Face Meetings

The working group has had four face-to-face meetings to date. Each of these meetings was attended by at least two participants from DERI Innsbruck.

Each meeting lasted two full days, which were filled with discussions about documents being produced by the working group, and technical discussions. A record of all face-to-face meetings can be found at: http://www.w3.org/2005/rules/wg/wiki/Face-to-Face_Meetings

At the time of writing, the following face-to-face meetings have taken place:

Burlingame, California, USA, December 8-9, 2005

Co-located with the OMG plenary

27-28 February 2006, Cannes-Mandelieu, France

At the W3C technical plenary

8-9 June 2006, Budva, Montenegro

Co-located with the European Semantic Web Conference (ESWC2006)

4-5 November 2006, Athens, Georgia, USA

Co-located with the International Semantic Web Conference (ISWC 2006)

The upcoming face-to-face meetings are planned for February and June 2007. The face-to-face meeting in June will most likely be in Innsbruck, co-located with the ESWC 2007.

4.2 Telephone Conferences

The RIF working group has weekly telephone conferences, which last one hour and a half. These telephone conferences are typically attended by several participants from DERI Innsbruck. The records of all telephone conferences can be found at: <http://www.w3.org/2005/rules/wg/wiki/Telecons>. At the time of writing, telephone conferences have taken place on the following dates:

Date	URI
2006-12-19	http://www.w3.org/2005/rules/wg/wiki/2006-12-19_Meeting
2006-12-12	http://www.w3.org/2005/rules/wg/wiki/2006-12-12_Meeting
2006-11-21	http://www.w3.org/2005/rules/wg/wiki/2006-11-21_Meeting
2006-11-14	http://www.w3.org/2005/rules/wg/wiki/2006-11-14_Meeting
2006-10-31	http://www.w3.org/2005/rules/wg/wiki/2006-10-31_Meeting
2006-10-24	http://www.w3.org/2005/rules/wg/wiki/2006-10-24_Meeting
2006-10-17	http://www.w3.org/2005/rules/wg/wiki/2006-10-17_Meeting
2006-10-10	http://www.w3.org/2005/rules/wg/wiki/2006-10-10_Meeting
2006-10-03	http://www.w3.org/2005/rules/wg/wiki/2006-10-03_Meeting
2006-09-26	http://www.w3.org/2005/rules/wg/wiki/2006-09-26_Meeting
2006-09-19	http://www.w3.org/2005/rules/wg/wiki/2006-09-19_Meeting
2006-09-12	http://www.w3.org/2005/rules/wg/wiki/2006-09-12_Meeting
2006-09-05	http://www.w3.org/2005/rules/wg/wiki/2006-09-05_Meeting
2006-08-29	http://www.w3.org/2005/rules/wg/wiki/2006-08-29_Meeting
2006-08-22	http://www.w3.org/2005/rules/wg/wiki/2006-08-22_Meeting
2006-08-15	http://www.w3.org/2005/rules/wg/wiki/2006-08-15_Meeting
2006-08-08	http://www.w3.org/2005/rules/wg/wiki/2006-08-08_Meeting
2006-08-01	http://www.w3.org/2005/rules/wg/wiki/2006-08-01_Meeting
2006-07-25	http://www.w3.org/2005/rules/wg/wiki/2006-07-25_Meeting
2006-07-18	http://www.w3.org/2005/rules/wg/wiki/2006-07-18_Meeting
2006-07-11	http://www.w3.org/2005/rules/wg/wiki/2006-07-11_Meeting
2006-07-04	http://www.w3.org/2005/rules/wg/wiki/2006-07-04_Meeting
2006-06-27	http://www.w3.org/2005/rules/wg/wiki/2006-06-27_Meeting
2006-06-20	http://www.w3.org/2005/rules/wg/wiki/2006-06-20_Meeting
2006-05-30	http://www.w3.org/2005/rules/wg/wiki/2006-05-30_Meeting
2006-05-23	http://www.w3.org/2005/rules/wg/wiki/2006-05-23_Meeting
2006-05-16	http://www.w3.org/2005/rules/wg/wiki/2006-05-16_Meeting

2006-05-09	http://www.w3.org/2005/rules/wg/wiki/2006-05-09_Meeting
2006-05-02	http://www.w3.org/2005/rules/wg/wiki/2006-05-02_Meeting
2006-04-25	http://www.w3.org/2005/rules/wg/wiki/2006-04-25_Meeting
2006-04-18	http://www.w3.org/2005/rules/wg/wiki/2006-04-18_Meeting
2006-04-11	http://www.w3.org/2005/rules/wg/wiki/2006-04-11_Meeting
2006-04-04	http://www.w3.org/2005/rules/wg/wiki/2006-04-04_Meeting
2006-03-28	http://www.w3.org/2005/rules/wg/wiki/2006-03-28_Meeting
2006-03-21	http://www.w3.org/2005/rules/wg/wiki/2006-03-21_Meeting
2006-03-14	http://www.w3.org/2005/rules/wg/wiki/2006-03-14_Meeting
2006-03-07	http://www.w3.org/2005/rules/wg/wiki/2006-03-07_Meeting
2006-02-21	http://www.w3.org/2005/rules/wg/wiki/2006-02-21_Meeting
2006-02-14	http://www.w3.org/2005/rules/wg/wiki/2006-02-14_Meeting
2006-02-07	http://www.w3.org/2005/rules/wg/wiki/2006-02-07_Meeting
2006-01-31	http://www.w3.org/2005/rules/wg/wiki/2006-01-31_Meeting
2006-01-24	http://www.w3.org/2005/rules/wg/wiki/2006-01-24_Meeting
2006-01-17	http://www.w3.org/2005/rules/wg/wiki/2006-01-17_Meeting
2006-01-10	http://www.w3.org/2005/rules/wg/wiki/2006-01-10_Meeting
2006-01-03	http://www.w3.org/2005/rules/wg/wiki/2006-01-03_Meeting
2005-12-20	http://www.w3.org/2005/rules/wg/wiki/2005-12-20_Meeting

5 PROGRESS TO DATE

The working group has published two consecutive working drafts of the Use Cases and Requirements document (<http://www.w3.org/TR/rif-ucr/>). The latest version contains a list of requirements to be addressed in a technical specification, as well as the RIF Rule systems Arrangement Framework (RIFRAF), which provides a means for characterizing rule languages and rules systems.

The major document to be produced by the working group is to technical specification of the language. During the face-to-face meeting in November 2006, a consensus was reached within the working group about the overall language design. There will be a common core (<http://www.w3.org/2005/rules/wg/wiki/CORE>), which corresponds to positive Horn logic (essentially, WSML-Rule without negation). This common core can be extended with different features, to create different dialects (e.g. a production rules dialect and logic programming dialect). It is expected that a first version of the technical specification, covering the core, will be published soon; the original goal was to have a last call working draft in the spring. This initial technical specification will only cover phase 1 of the working group, which essentially corresponds to the development of RIF core. Then, in phase 2 of the working group, the different dialects will be developed.

6 OUTLOOK

The next steps for the working group are finalizing the first version of technical specification (RIF core), outlining the connection to RDF and OWL, and then defining extensions of the core, to create useful RIF dialects, especially for use on the Semantic Web. With respect to all these aspects, WSML is still far ahead of RIF. Therefore, it is essential that members of the DIP consortium continued their participation in the working group. We plan to continue our participation in the RIF working group in the context of the SUPER⁶ project.

⁶ <http://ip-super.org/>

7 APPENDIX A. THE WSML RULE LANGUAGES FOR THE SEMANTIC WEB

Position Paper for the W3C rules workshop, Washington DC, USA, April 2005

Authors:

Jos de Bruijn

Holger Lausen

Axel Polleres

Dieter Fensel

Digital Enterprise Research Institute (DERI)
National University of Ireland, Galway, Ireland
University of Innsbruck, Austria
Email: {jos.debruijn, holger.lausen, axel.polleres} 'at' deri.org

7.1 Abstract

The Web Service Modeling Language WSML provides a framework for the modeling of ontologies and semantic Web services based on the conceptual model of the Web Service Modeling Ontology. In this paper we describe the two rule-based WSML-variants and outline our position with respect to a rule language for the Semantic Web. The first rule-based WSML variant, WSML-Flight, semantically corresponds to the Datalog fragment of F-Logic, extended with inequality in the body and locally stratified negation under the Perfect model semantics. The second, WSML-Rule, is an extension of WSML-Flight to the logic programming subset of F-Logic which allows the use of function symbols and unsafe rules (i.e., there may be variables in rule heads which do not occur in the body).

7.2 Introduction

The Web Service Modeling Language (WSML) [\[de Bruijn et al., 2005\]](#) provides an overall framework for different logical languages. The main language paradigms supported in WSML are Description Logics and Logic Programming. The language WSML-Core marks the intersection of both paradigms, based on DLP [\[Grosz et al., 2003\]](#). WSML-DL marks the Description Logic extension; WSML-Flight and WSML-Rule mark the rule extensions in the spirit of Logic Programming. In this paper, we discuss WSML-Flight and WSML-Rule as rule languages for the Semantic Web.

WSML-Flight and WSML-Rule are rule languages based on the Datalog and Logic Programming fragments of F-Logic, respectively, extended with inequality and default negation in rule bodies under the Perfect Model semantics. [\[Przymusiński, 1989\]](#). There exist efficient implementations for these paradigms. This allows us to use existing tools and reasoners to reason with WSML-Flight and WSML-Rule.

In the remainder of this paper, we briefly introduce WSML-Flight and WSML-Rule. We describe the relation between WSML and other initiatives for rule languages for the

Web. Finally, we summarize our positions with respect to rule languages for the Semantic Web.

7.3 WSML

WSML distinguishes between a conceptual and a logical expression syntax. The conceptual syntax is used for the modeling of web services, goals, mediators and ontologies. The logical expression syntax is used for the specification of axioms and constraints in an ontology and inside the pre- and post-conditions of goals and web services.

WSML-Flight and WSML-Rule both provide a conceptual syntax, but we will restrict the discussion to the logical expression syntax in this paper, since rules are defined as logical expressions in WSML. Basically, the logical expression syntax has its foundations in F-Logic [Kifer et al., 1995], but uses slightly different language keywords.

Every value in a WSML document is either a data value of a built-in datatype or an IRI (Internationalized Resource Identifier) [Duerst & Suignard, 2005]. Datatype wrappers are used to structure data values in WSML. Datatype support in WSML is based on XML Schema, but we allow more powerful constructs to reflect the structure of data values. For example, the date "March 15th, 2005" is represented as:

`_date(2005, 3, 15)`. IRIs, being a special and often used datatype in WSML, use a shorter notation: IRIs are delimited with '_' and ''. IRIs can also be abbreviated to sQNames (i.e., serialized QNames) using namespace prefixes. A document may have a default namespace, in which case the prefix may be omitted. Assuming the prefix 'ex' stands for 'http://example.org/', the sQName `ex#person` is an abbreviation for `_http://example.org/person`. Variables may be any alphanumeric string preceded by a question mark, e.g., `?x`, `?y`, `?myVariable`.

A WSML rule has the form:

head :- body.

WSML allows database-style constraints of the form:

!- body.

An atomic formula in WSML may be one of the following (with A_1, \dots, A_n denote data values, variables or identifiers, or, in the case of WSML-Rule, constructed terms, and p an identifier): (1) $p(A_1, \dots, A_n)$; (2) $A_1[A_2 \text{ op } A_3]$ with $op \in \{ \text{hasValue, ofType, impliesType} \}$; (3) $A_1 \text{ isa } A_2$ with $isa \in \{ \text{subConceptOf, memberOf} \}$.

The body of the rule may be any atomic formula or a combination of atomic formulae using the connectives **and**, **or** and **naf**. The head of the rule may be any atomic formula or a combination of atomic formulae using the connectives **and**, **implies** and **impliedBy**. The keywords **and** and **or** stand for conjunction and disjunction, respectively; **implies** and **impliedBy** denote the two directions (right and left,

respectively) of classical implication; **naf** (negation-as-failure) stands for default negation.

By allowing classical implication in the head of rules, we enable layering WSML-Flight on top of WSML-Core. WSML-Core only has classical implication. Every WSML-Core logical expression corresponds to a WSML-Flight rule with an empty body.

7.3.1 Examples

A common example for rules is a rule which models that the brother of your parent is your uncle. The statements in brackets are attribute value specifications. $a[b \text{ hasValue } c]$ intuitively means that object a has an attribute b with value c .

Example 1

```
?x[uncle hasValue ?z] :- ?x[father hasValue ?y] and
    ?y[brother hasValue ?z].
```

The following rule illustrates the use of signature molecules in rule bodies. The symbol **naf** is used for default negation (negation-as-failure); a statement of the form $a \text{ subConceptOf } b$ means that concept a is a subconcept of concept b .

Example 2

```
?x[directSuperConcept hasValue ?y] :- ?x subConceptOf ?y and
    naf ?x[indirectSuperConcept hasValue ?y] and ?x != ?y.
?x[indirectSuperConcept hasValue ?z] :- ?x subConceptOf ?y and
    ?y subConceptOf ?z and ?x != ?y and ?y != ?z.
```

This rule enumerates all direct and all indirect superconcepts for each concept. Actually, the attribute 'indirectSuperConcept' is used here to enable the definition of direct superconcepts. The rule also shows an application of default negation which ensures that for every superconcept for which it is not known to be an indirect superconcept, is a direct superconcept. The second rule ensures that every superconcept which is an indirect superconcept is known to be an indirect superconcept. The example shows an application of meta-modeling where a concept has an attribute value.

7.3.2 Syntaxes for WSML

Besides the mentioned normative human-readable syntax, WSML has an XML syntax for exchange between machines and an RDF syntax for exchange over the Semantic Web.

7.3.2.1 XML

WSML has an XML syntax associated with it. The XML syntax resembles the human-readable syntax, both in keywords and in structure. The XML version of the Example 1, in abbreviated form, looks like:

```
<impliedByLP xmlns="http://www.wsmo.org/wsml/wsml-syntax#">
```

```
<molecule>
  ...
</molecule>
<and>
  ...
</and>
</impliedByLP>
```

7.3.2.2 RDF

Statements with a complex structure can not be easily captured within the graph data model of RDF, because RDF essentially only allows representation of binary relations. Rules are in general very complex structures and do not allow for straightforward encoding in RDF. Furthermore, problems might occur when formulas should be interpreted according to their context (e.g. web service preconditions; cf. [1]), because all triples in RDF are global and do not have a context. We have not directly defined an RDF syntax for the WSML logical expression syntax. Instead, when using the RDF encoding for WSML, we use literals of the type `rdf:XMLLiteral` for the encoding of rules:

```
"<impliedByLP>...</impliedByLP>"^^rdf:XMLLiteral
```

7.4 Relation with other efforts

7.4.1 SWSL

SWSL is another recent proposal for a language for describing semantic Web services. SWSL has two parts, a rules language and a process ontology. The SWSL-Rules sublanguage is closely related to WSML-Flight and WSML-Rule. Both languages are largely based on F-logic and they mostly share the logical expression syntax. However, the two groups have pursued complimentary goals. WSML is focused on the end user and developed a "conceptual syntax" for top-level descriptions of services, which we believe, might make the specifications easier to read. WSML also paid special attention to the issue of OWL compatibility. To this end, WSML defines WSML-Core as a subset of both Description Logics and Logic Programming, which serves as a common ground for ontology interoperability. In contrast, SWSL's focus was on extending the functionality of their rule-based language. In particular, SWSL-Rules supports meta-reasoning with its HiLog and reification extensions. It also supports prioritized defaults and classical negation by incorporating Courteous Logic Programming.

7.4.2 RuleML

RuleML is an effort to develop a rule language for the Web. The syntax for RuleML is based on XML, although there exists an RDF syntax. RuleML provides different modules with different features for the rule language. For example, there is the Datalog module and also the `naf` (negation-as-failure) extension module which enables incorporating negation-as-failure in the document. Recently, the FOL (First-Order Logic) version of RuleML has been released. This version is an alternate syntax for First-Order Logic, based on rules. The XML syntax for WSML is partly inspired by RuleML. We do not rule out further alignment between the RuleML and WSML initiatives in the future.

7.4.3 SWRL

SWRL [[Horrocks et al., 2004](#)] is a recent proposal for a rule language for the Semantic Web, layered on top of the DL species of OWL. While a subset of SWRL falls inside Logic Programming, a SWRL knowledge base easily goes beyond this fragment, because of the possibility to include arbitrary OWL DL, which includes disjunction in the head and existentially quantified variables. Thus, SWRL does not allow for efficient implementation in current rule engines. Another major difference between SWRL and WSML is that SWRL is based on First-Order logic and is thus monotonic. WSML-Rule allows for nonmonotonic negation. SWRL itself poses a number of syntactic restrictions on the language. It is not possible to use predicates with an arity higher than 2 and function symbols are disallowed. These restrictions were not motivated by computational properties of the language; the satisfiability problem in SWRL is undecidable.

7.5 Summary of Position

Here follows our actual position with respect to a rule language for the Semantic Web:

Allow exchange with existing rules implementations

The Logic Programming and Deductive Database communities have developed many tools which are currently in use. A rule language for the Semantic Web should allow reuse of existing systems and existing techniques which have been developed in this area. Many systems implement default negation and therefore, we believe it is an essential component of any rule language.

Relation with a Description Logic language can be done through a common subset

Instead of directly extending a Description Logic, as SWRL [[Horrocks et al., 2004](#)] does, or defining an interface between the two paradigms, as in [[Eiter et al., 2004](#)], we propose to use a common subset, depicted by DLP [[Grosz et al., 2003](#)]. In this way, existing tools can still be used, and no special-purpose tools need to be built to allow interaction between the DL ontology and the rule base.

Different layers of expressiveness

It is useful to distinguish different layers of expressiveness in the rule language to suit different application domains. Some domains might require decidable reasoning; this can be ensured through a restriction to Datalog. Other applications might require greater expressiveness, in which case full Logic Programming (i.e. with function symbols and unsafe rules) could be used.

Human-readable syntax for understandability

In order to improve understandability of the language and in order to improve uptake of the language, the normative syntax for the language should be understandable for humans. Alternative syntaxes, such as XML and RDF, could be based on the human-readable syntax, in the same way the OWL DL RDF syntax is based on the OWL DL abstract syntax [[Patel-Schneider et al., 2004](#)].

Frame-based syntax

Frame Logic [Kifer et al., 1995] allows the use of frames in logical statements. This allows the user to work directly on the level of concepts, attributes, instances and attribute values. Furthermore, variables are allowed in place of concept and attribute identifiers, which enables meta-modeling and reasoning over the signature.

7.6 References

- [de Bruijn et al., 2005] J. de Bruijn, H. Lausen, R. Krummenacher, A. Polleres, L. Predoiu, M. Kifer, D. Fensel. *The Web Service Modeling Language WSML*. WSML Deliverable D16.1v0.2, 2005. <http://www.wsmo.org/TR/d16/d16.1/v0.2/>
- [Duerst & Suignard, 2005] M. Duerst and M. Suignard. *Internationalized Resource Identifiers (IRIs)*. IETF RFC3987. <http://www.ietf.org/rfc/rfc3987.txt>
- [Eiter et al., 2004] T. Eiter, T. Lukasiewicz, R. Schindlauer, and H. Tompits. Combining answer set programming with description logics for the semantic web. In *Proc. of the International Conference of Knowledge Representation and Reasoning (KR04)*, 2004.
- [Franconi and Tessaris, 2004] E. Franconi and S. Tessaris. Rules and queries with ontologies: a unified logical framework. In *Workshop on Principles and Practice of Semantic Web Reasoning (PPSWR'04)*, St. Malo, France, 2004.
- [Grosf et al., 2003] B. N. Grosf, I. Horrocks, R. Volz, and S. Decker. Description logic programs: Combining logic programs with description logic. In *Proc. of the Twelfth International World Wide Web Conference (WWW 2003)*, pages 48-57. ACM, 2003.
- [Horrocks et al., 2004] I. Horrocks, P.F. Patel-Schneider, H. Boley, S. Tabet, B. Grosf and M. Dean: *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*. W3C Member Submission 21 May 2004. Available from <http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/>.
- [Kifer et al., 1995] M. Kifer, G. Lausen, and J. Wu: Logical foundations of object-oriented and frame-based languages. *Journal of the ACM*, 42:741-843, July 1995.
- [Patel-Schneider et al., 2004] P. F. Patel-Schneider, P. Hayes, and I. Horrocks: *OWL web ontology language semantics and abstract syntax*. Recommendation 10 February 2004, W3C, 2004. Available from <http://www.w3.org/TR/owl-semantics/>.
- [Przymusinski, 1989] T. C. Przymusinski. On the declarative and procedural semantics of logic programs. *Journal of Automated Reasoning*, 5(2):167-205, 1989.
- [Roman et al., 2004] D. Roman, H. Lausen, and U. Keller (eds.): *Web Service Modeling Ontology - Standard (WSMO - Standard)*, WSMO deliverable D2 version 1.1. available from <http://www.wsmo.org/TR/d2/v1.1/>.

7.7 Footnotes

[1] See also the thread "Representation of SWRL expressions in OWL-S" on the public-sws-ig mailing list: <http://lists.w3.org/Archives/Public/public-sws-ig/2005Feb/0011.html>

8 APPENDIX B. WSML - A LANGUAGE FRAMEWORK FOR SEMANTIC WEB SERVICES

Position Paper for the W3C rules workshop, Washington DC, USA, April 2005

Authors:

Holger Lausen

Jos de Bruijn

Axel Polleres

Dieter Fensel

Digital Enterprise Research Institute (DERI)
National University of Ireland, Galway, Ireland
University of Innsbruck, Austria
Email: {holger.lausen, jos.debruijn, axel.polleres, dieter.fensel} 'at' deri.org

8.1 Abstract

The Web Service Modeling Language (WSML) provides a framework of different language variants to describe semantic Web services. This paper presents the design rationale and relation with existing language recommendations. WSML is a frame based language with an intuitive human readable syntax and XML and RDF exchange syntaxes, as well as a mapping to OWL. It provides different variants, allowing for open and closed world modeling; it is a fully-fledged ontology and rule language with defined variants grounded in well known formalisms, namely Datalog, Description Logic and Frame Logic. Taking the key aspects of WSML as a starting point, we rationalize the design decisions which we consider relevant in designing a proper layering of ontology and rule languages for the Semantic Web and semantic Web services.

8.2 Introduction

With the Web Service Modeling Language WSML [[de Bruijn et al., 2005](#)] we provide a language framework for semantic Web services, based on the conceptual model of WSMO [[Roman et al., 2005](#)]. WSML provides means to describe semantic Web services [[Fensel & Bussler, 2002](#)] and its related aspects, i.e. ontologies, web Services, goals, and mediators. Those descriptions aim at automating Web service related tasks such as discovery, mediation and invocation. Reasoning based on formal descriptions can be used to achieve automation. In this paper we will mainly focus on the modelling of ontologies and logical expressions.

We believe that different application scenarios require different expressiveness of the underlying logical formalism. For example in the application area of Web service discovery [[Keller et al., 2004](#)] it has been shown that depending on the concrete scenario different levels of expressiveness are required. However interoperability and reuse of common domain ontologies is required across those different layers. WSML

provides such a framework including a proper layering that enables the use of shared terminologies.

We believe that current Semantic Web language recommendations are not sufficient for the domain of semantic Web service modeling. Approaches such as OWL-S take different language recommendations (OWL, SWRL) and combine them without proper conceptual layering and lack of formal semantics. We take a different approach: with WSML we construct an overall framework for the specification of ontologies, Web services, goals, and mediators, with strict layering and a mapping to existing standards.

8.3 Web Service Modeling Language

WSML makes a clear distinction between the modeling of the different conceptual elements (ontologies, Web services, goals, and mediators). Furthermore WSML separates between conceptual syntax and logical expression syntax, where the latter is used for describing additional constraints and axioms. The separation allows for an easy adoption by non-experts, since the conceptual syntax allows an intuitive understanding for many readers, whereas complex logical expressions require more familiarity and training with the language. In this context, we emphasize the normative "human readable" syntax, which allows better legibility. For the purpose of exchange on the Web XML and RDF serializations are specified.

8.3.1 Illustrating Example

The example in Listing 1 illustrates some of the design principles of WSML. Identifiers are IRIs (the successor standard of URIs), delimited using `_` and `"` in WSML. Namespaces are used to abbreviate IRIs. In WSML every non-prefixed identifier is preceded by the default namespace (in our example `http://www.example.org/Family#`) and prefixed identifier such as `dc#source` are expanded to full IRIs by replacing the prefix with the corresponding namespace.

The conceptual model allows for each element the specification of non-functional properties for additional related information. We recommend the use of the Dublin Core metadata set [Weibel et al., 1998]. In the example one concept and one axiom is declared, the concept describes several aspects of a human being (lines 7 through 10). Attributes can have features such as being the inverse of another attribute, but in contrast to OWL WSML defines such features local to the concept. In the example the age of a human has to be given as an integer value (`_integer` corresponds to the integer datatype of XML Schema). Attributes can have cardinality restrictions. For instance `hasAgeInYears` is specified as optional attribute with the maximum cardinality of one (i.e. it is functional). The example uses closed world constraints on the type of attributes (keyword `ofType`), such that every attribute value is required to be provably of the type specified. Note that this is different to OWL where the type of an attribute value is inferred based on the given range, which corresponds to the use of the `impliesType` keyword in WSML.

```

1 wsmlVariant _"http://www.wsmo.org/wsml/wsml-syntax/wsml-flight"
2
3 namespace {_"http://www.example.org/Family#",
```

```

4  dc _"http://purl.org/dc/elements/1.1#" }
5
6  ontology Family
7  concept Human
8    hasParent inverseOf(hasChild) ofType Human
9    hasChild ofType Human
10   hasAgeInYears ofType (0 1) _integer
11
12  axiom DefinitionTeenager
13    nonFunctionalProperties
14    dc#source hasValue
15    _"http://dictionary.reference.com/search?q=teenager"
16    endNonFunctionalProperties
17    definedBy
18    forall {?teen,?age} (
19      ?teen memberOf Teenager impliedBy
20      ?age [hasAgeInYears hasValue ?age] memberOf Human and
        ?age >= 13 and ?age =< 19).

```

Listing 1: Example for a Concept and Axiom Description in WSML

Axioms allow the refinement of an ontology with the use of logical expressions. The axiom in the example defines a Teenager as a human of an age between 13 and 19. The logical expression syntax is based on F-Logic [Kifer et al., 1995], but using different keywords, such as `memberOf` instead of `:` to express the concept-membership relation. Logical expressions are interweaved with the conceptual syntax using the `axiom` keyword and an optional identifier. In the example, we use additional non-functional properties to specify the source of the information contained in the axiom.

8.3.2 WSML Layering

The WSML framework allows a number of variants with different expressiveness. All Variants and their interrelation are depicted in Figure 1. Whereby extension means that every valid specification of an extended variant is also a valid specification of the new variant. Furthermore, all consequences inferred from a "lower" WSML variant are also valid consequences of the extended variant. The layering is defined by syntactical restrictions on the language.

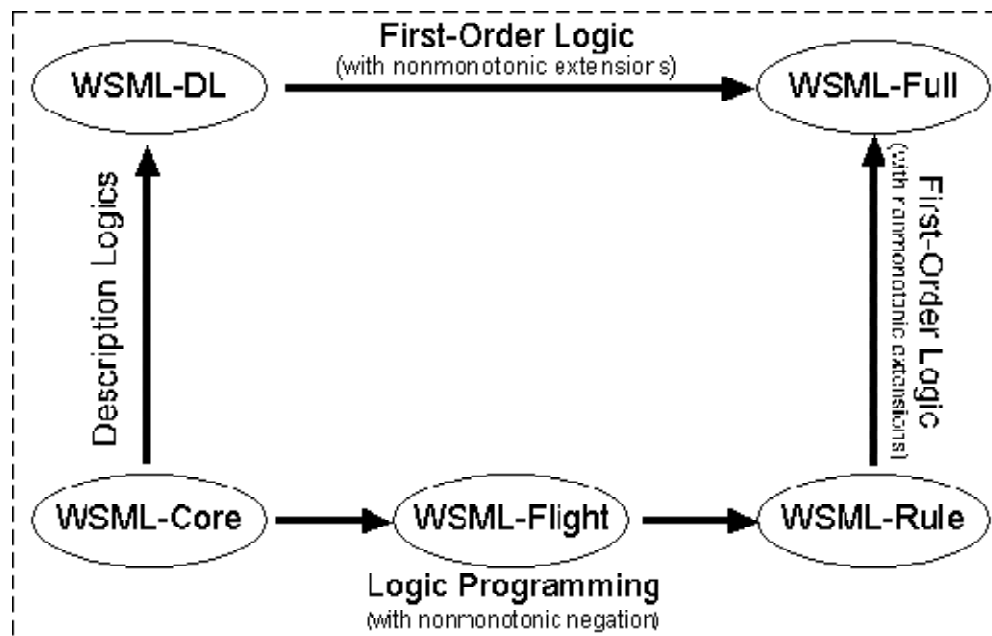


Figure 1: WSML Variants

WSML-Core corresponds with the intersection of Description Logic and Horn Logic [Grosz et al., 2003] (without function symbols and without equality), extended with datatype support in order to be useful in practical applications. WSML-Core is fully compliant with a subset of OWL.

WSML-DL extends WSML-Core to an expressive Description Logic, namely, *SHIQ*, thereby covering that part of OWL which is efficiently implementable.

WSML-Flight extends WSML-Core in the direction of Logic Programming. WSML-Flight has a rich set of modeling primitives for different aspects of attributes, such as value and integrity constraints. Furthermore, WSML-Flight incorporates a rule language, while still allowing efficient decidable reasoning. More precisely, WSML-Flight allows any Datalog rule, extended with inequality and (locally) stratified negation.

WSML-Rule extends WSML-Flight to a fully-fledged Logic Programming language, by allowing function symbols and unsafe rules.

WSML-Full unifies all WSML variants under a common First-Order umbrella with non-monotonic extensions which allow to capture non-monotonic negation of WSML-Rule.

8.3.3 WSML and the Web

WSML makes extensive use of existing Web standards. For example, it uses IRIs to identify entities, as illustrated in Listing 1.

The Extensible Markup Language (XML) is a markup language for interchange of structured data. WSML adopts the use of namespaces from XML and the datatypes from XML Schema. Furthermore, WSML specifies an XML exchange syntax for interoperability on the Web.

The Resource Description Framework (RDF) defines a triple data structure for meta data description on the Web and RDF Schema describes how to use RDF to describe vocabularies. WSML reuses and extends the RDF Schema vocabulary to allow existing RDF(S)-based tools to understand parts of a WSML specification. WSML puts viewer restrictions on the kind of allowed triples then OWL DL does. For logical expressions we reuse the WSML/XML serialization, since an encoding of complex constructs into triples results in many triples and complicated processing.

The Web Ontology Language (OWL) is not reused directly as basis for our language, given the goal of a strict semantic layering and computational beneficial properties. Instead, we provide mappings to those fragments of OWL that have desirable computational properties. Specifically, there exist bi-directional mappings for WSML-Core and WSML-DL to and from OWL.

8.4 Summary of Position

Given the design decisions taken and the relation to existing specifications, we summarize our rationale as follows:

One syntactic framework for a set of layered languages

We believe different semantic Web applications need languages of different expressiveness. Instead of retrofitting the layering to existing language proposals and recommendations we propose a coherent framework with strict layering and desirable computational properties. This is different than other approaches like SWRL, which layer on top of OWL DL. SWRL has syntactical restrictions such as allowing only binary relations, that are not motivated by computational properties.

Normative, human readable syntax

It has been argued that tools will hide language syntax from the user; however, as has been shown with SQL, an expressive but understandable syntax is one of the criteria for successful adoption of a language. Developers and users will have to deal with the concrete syntax and a human readable version (compared to XML or RDF) allows a faster uptake of the language.

Separation of conceptual model and logical expression

The conceptual syntax allows the modeling of ontologies on the level of concepts, relations and instances, while logical expression allow further refinements. This separation provides a clear way to include annotations and enhance the structure of a knowledge base, in the sense that the structure of an ontology can be easier recognized within the conceptual model then the encoding in logical expressions allows.

Semantics based on well known formalism

Our proposal unifies well known formalisms such as Datalog and Description Logics with a syntactical framework compliant with current Web standards, while maintaining the original well researched computational properties. Furthermore it allows the reuse of tools already developed for those formalisms.

Closed world and open world assumption

We believe it is beneficial to allow developers to choose between both assumptions for application within the semantic Web. Allowing the closed world assumption will allow users coming from the database area to adopt the technology more easily.

8.5 References

- [**de Bruijn et al., 2005**] J. de Bruijn, H. Lausen, R. Krummenacher, A. Polleres, L. Predoiu, M. Kifer, D. Fensel. *The Web Service Modeling Language WSMML*. WSMML Deliverable D16.1v0.2, 2005. Available from <http://www.wsmo.org/TR/d16/d16.1/v0.2/>
- [**Fensel & Bussler, 2002**] D. Fensel and C. Bussler: *The Web Service Modeling Framework WSMF*, Electronic Commerce Research and Applications, 1(2), 2002.
- [**Grosf et al., 2003**] B. N. Grosf, I. Horrocks, R. Volz, and S. Decker. Description logic programs: Combining logic programs with description logic. In *Proc. of the Twelfth International World Wide Web Conference (WWW 2003)*, pages 48-57. ACM, 2003.
- [**Keller et al., 2004**] U. Keller, R. Lara, A. Polleres (eds.): *WSMO Web Service Discovery*, WSMO deliverable D5.1 version 0.1. Available from <http://www.wsmo.org/2004/d5/d5.1/v0.1/>.
- [**Kifer et al., 1995**] M. Kifer, G. Lausen, and J. Wu: Logical foundations of object-oriented and frame-based languages. *Journal of the ACM*, 42:741-843, July 1995.
- [**Roman et al., 2004**] D. Roman, H. Lausen, and U. Keller (eds.): *Web Service Modeling Ontology - Standard (WSMO - Standard)*, WSMO deliverable D2 version 1.1. Available from <http://www.wsmo.org/TR/d2/v1.1/>.
- [**Weibel et al., 1998**] S. Weibel, J. Kunze, C. Lagoze, and M. Wolf: *RFC 2413 - Dublin Core Metadata for Resource Discovery*, September 1998.