

DLP isn't so bad after all^{*}

Pascal Hitzler, Peter Haase, Markus Krötzsch, York Sure, and Rudi Studer

AIFB, Universität Karlsruhe, Germany

Abstract. We discuss some of the recent controversies concerning the DLP fragment of OWL. We argue that it *is* a meaningful fragment and *can* serve as a basic interoperability layer between OWL and logic programming-based ontology languages.

1 Introduction

DLP — Description Logic Programs — has originally been conceived in [1] as a fragment of OWL DL.¹ Since then it has been talked about a lot, but has also been a source of confusion, controversies, and heated discussions. This is only partly due to the fact that [3] leaves ambiguities as to what DLP actually is (see [4]²). Most of it is indeed caused by the sharp discussions on the relationship between the first-order predicate logic-based Web Ontology Language OWL and logic programming-based ontology languages such as the W3C member submission Web Rule Language WRL³, which is based on F-Logic [5]. Rather explicit manifestations of these controversies are e.g. [3] and [6], and accompanying presentations.

We think that DLP isn't so bad after all. It is limited, yes, but useful to understand the relationships between ontology languages such as OWL and F-Logic. Further it is a constructive approach to establishing interoperability among them on a well-defined level. Indeed, we claim that the semantic issues concerning logic programming, OWL, and their intersection DLP are not as fuzzy as they seem and are often displayed as. We believe that DLP *can* serve as a very basic interoperability layer between OWL and logic programming-based ontology languages, although in a way which yet remains to be worked out in detail, and which will most likely be of a very restricted nature.

^{*} The authors acknowledge support by the German Federal Ministry of Education and Research (BMBF) under the SmartWeb project, and by the European Commission under contract IST-2003-506826 SEKT and under the KnowledgeWeb Network of Excellence. The expressed content is the view of the authors but not necessarily the view of any of the projects as a whole.

¹ The dissertation [2] delivers a more comprehensive treatment.

² In [4] concrete proposals for definitions are given.

³ <http://www.w3.org/Submission/WRL>

2 DLP and Horn logic programming

An exhaustive and entirely satisfactory definition of DLP is not straightforward, as argued in [4]². Three things, however are entirely clear and unarguable, as this is the way DLP was originally conceived:

- DLP is syntactically a fragment of OWL in the sense that every DLP knowledge base is a syntactically valid OWL knowledge base.
- DLP carries the semantics it inherits from OWL.
- Each DLP knowledge base is semantically equivalent to a set of Horn clauses under first-order predicate logic semantics.

As such, any DLP ontology can be converted syntactically into a set of Horn clauses. A source of confusion comes from the fact that Horn clauses are also used for the syntactic representation of logic programs, more precisely, of definite logic programs (i.e. not containing any form of negation). This in turn motivates reasoning with DLP using logic programming systems. Whether this form of reasoning is reasonable, is the central point which is being discussed. It is being argued, for example, that logic programming semantics is based on the closed world assumption (CWA), while OWL (and therefore DLP) is based on the open world assumption (OWA) — which may lead one to the arguable consequence that the semantics are fundamentally different and thus incompatible.

The relation between Horn logic programming semantics and the first-order predicate logic semantics, however, is not as complicated as the *OWA vs. CWA* debate might suggest. If we go back to the roots of logic programming, then it indeed came into existence due to efforts of automatizing *resolution*, which is a sound and complete proof theory for first-order predicate logic. Proof procedures implemented in standard Prolog systems (like SLD- or SLG-resolution) build on this and trade completeness of reasoning for speed. The proof-theoretic semantics of such systems is thus an *approximation* of the first-order semantics in the sense that it is sound but not complete, and this is entirely reasonable as predicate logic is not decidable (but only semi-decidable) to start with. It can indeed be understood as approximation in a very formal (and logical!) sense by means of standard perspectives of denotational semantics i.e. domain theory (see e.g. [7]).⁴

An issue likely to be brought into the discussion is that of negation. Standard logic programming systems tend to answer *No* to queries if they (finitely) fail to prove them. Naively understood, this may prompt the user to think that the negation of the query be true, although under first-order predicate logic semantics the query simply fails to be a logical consequence of the knowledge base. Interpreting the *No* as truth of the negation of the query indeed is the step which leads us into CWA reasoning, and thus differs from OWA reasoning.

The problem, however, is not on the side of the semantics of Horn logic programming. It is rather in *our interpretation of procedural system behaviour*. A *No* answer to a query simply means that the query cannot be proven, and

⁴ The omission of occurs check and corresponding lazy unification causes systems to be unsound. But this is not a problem in practice and can be rectified easily.

that furthermore, it can be *shown* that the query cannot be proven. It is thus a system response which can also be interpreted in an OWA fashion, namely as *Query cannot be proven*, together with some information on the procedural aspects of the underlying reasoning. As such, Horn logic programming is fully compatible with OWA, and there is no fundamental difference to OWA reasoning as applied e.g. in OWL.

But what about negation as failure, or default negation, or other kinds of non-monotonic negation? They don't play a role in our context, as DLP — properly translated into Horn clause syntax — does not contain any such negation. If non-monotonic negation is used on top of DLP, then this should be perceived as a non-monotonic extension of DLP, in the sense in which non-monotonic extensions of OWL are being studied, e.g. in [8, 9]. In either case — non-monotonic extensions of DLP or OWL — the goal is to add closed world reasoning aspects on top of the underlying open world reasoning system, which results in an *extension* of expressibility, but not in an incompatible semantics!

3 Interoperability?

Some of the currently discussed proposals for a standardized rule language for the Semantic Web⁵ aim at establishing a logic programming-based rule ontology language as an alternative to the already defined standard OWL. This in part mirrors the actual situation within the Semantic Web community, where both paradigms are currently being used. An obvious question being asked in this case is to what extent interoperability across the paradigms would be possible in case such an alternative standard were to be defined. Some proposals, such as WRL, state that DLP would serve as interoperability layer between paradigms.

We have already noted that semantic compatibility between the two perspectives on DLP — Horn logic programming and first-order predicate logic — is basically given. However, it can still be argued whether DLP is a reasonable interoperability layer between OWL and F-Logic-based languages. Indeed, DLP appears to be a rather restricted fragment which lacks in expressivity.

There are two aspects to this. The first is that many available ontologies use only very few constructs outside the DLP fragment (see [2]). As such, DLP can be perceived as a transitional ontology language which can be used for developing and experimenting with technologies available at the current state-of-the-art. This perspective also implies that DLP will probably disappear naturally as technology advances and ontologies become more expressive.

The second argument is that DLP serves only as a common *fragment* of OWL and F-Logic, and rather *not* as something which makes interoperability between the frameworks possible in a reasonable sense. Agreed! — so far. But we believe that this is only a first step towards a larger interoperability which can be established by using DLP as a base, e.g. by using DL-safe rules [10] for

⁵ See e.g. the website of the W3C Workshop on Rule Languages for Interoperability at <http://www.w3.org/2004/12/rules-ws>.

encoding in F-Logic rules in the style of [5], or by using the KAON2⁶ algorithms [11, 12] for converting a large fragment of OWL DL to disjunctive datalog.

Indeed, as OWL can be perceived as an extension of DLP, F-Logic (or WRL) can be perceived as an extension of DLP in a different direction. Indeed, OWL semantics is that of first-order predicate logic, and F-Logic semantics is different, as it involves CWA and well-founded (non-monotonic) negation.⁷ But the latter has been developed on the basis of classical logic and relates to it in a formal way. Admittedly, the study of the exact relationships between non-monotonic semantics and first-order predicate logic semantics has been neglected in the past, and consequently the relations are not well understood. However, there do exist investigations, e.g. [13, 14], which can be expanded on in order to work out the exact relationships.⁸

The underlying vision is straightforward: Given ontologies from different sources, written in different languages (say, OWL and WRL), it is reasonable to expect that a merging of the knowledge is possible in some reasonable way, namely such that at least *some* meaningful consequences can be drawn from the combined knowledge, which cannot be derived from any of the ontologies looked at in isolation. Realizing interoperability in this sense has to be approached by studying the exact relationships between the semantics of the frameworks, and it is reasonable to expect that such efforts would produce meaningful – albeit restricted – outcomes.

4 Conclusions: Be reasonable

Summarizing, we conclude that OWL and F-Logic indeed are basically compatible if restricted to DLP. Furthermore, we see that interoperability between frameworks is possible and in our opinion needs to be researched and realized.

We would indeed be very happy to see more efforts in establishing constructive relationships between paradigms as the Semantic Web can and should be expected to be very heterogenous. We would also be happy to see many more *constructive* discussions as they may lead to improved systems and interoperability, and ultimately to improved perception of our community in other research communities and the industry.

References

1. Grosz, B., Horrocks, I., Volz, R., Decker, S.: Description logic programs: Combining logic programs with description logics. In: Proc. of WWW 2003, Budapest, Hungary, May 2003, ACM (2003) 48–57

⁶ <http://kaon2.semanticweb.org>

⁷ Note that the different semantics proposed for F-Logic coincide on ground consequences of the Horn fragment.

⁸ We remark here that there is a natural limit to possible interoperability due to the fact that the well-founded semantics is in general not semi-decidable [15].

2. Volz, R.: Web Ontology Reasoning with Logic Databases. PhD thesis, AIFB, University of Karlsruhe (2004)
3. de Bruijn, J., Polleres, A., Lara, R., Fensel, D.: OWL DL vs. OWL Flight: Conceptual modeling and reasoning for the Semantic Web. In: Proceedings of the 14th International World Wide Web Conference, WWW2005, Chiba, Japan. (2005)
4. Hitzler, P., Studer, R., Sure, Y.: Description logic programs: A practical choice for the modelling of ontologies. In: 1st Workshop on Formal Ontologies Meet Industry, FOMI'05, Verona, Italy, June 2005. (2005)
5. Kifer, M., Lausen, G., Wu, J.: Logical foundations of object-oriented and frame-based languages. *Journal of the ACM* **42** (1995) 741–843
6. Horrocks, I., Parsia, B., Patel-Schneider, P., Handler, J.: Semantic web architecture: Stack or two towers? In Fages, F., Soliman, S., eds.: Proceedings of Principles and Practice of Semantic Web Reasoning. Third International Workshop, PPSWR 2005, Dagstuhl Castle, Germany, September 2005. Volume 3703 of Lecture Notes in Computer Science., Springer, Berlin (2005) 37–41
7. Smyth, M.B.: Topology. In Abramsky, S., Gabbay, D.M., Maibaum, T.S., eds.: Handbook of Logic in Computer Science. Volume 1. Clarendon, Oxford (1994) 641–761
8. Donini, F.M., Nardi, D., Rosati, R.: Description logics of minimal knowledge and negation as failure. *ACM Trans. Comput. Logic* **3** (2002) 177–225
9. Eiter, T., Lukasiewicz, T., Schindlauer, R., Tompits, H.: Combining answer set programming with description logics for the Semantic Web. In: Proc. KR-2004. (2004) 141–151
10. Motik, B., Sattler, U., Studer, R.: Query answering for OWL-DL with rules. *Journal on Web Semantic* **3** (2005) 41–60
11. Hustadt, U., Motik, B., Sattler, U.: Reducing SHIQ-description logic to disjunctive datalog programs. In Dubois, D., Welty, C.A., Williams, M.A., eds.: Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR2004), Whistler, Canada, June 2-5, 2004, AAAI Press (2004) 152–162
12. Hustadt, U., Motik, B., Sattler, U.: Reasoning in description logics with a concrete domain in the framework of resolution. In de Mántaras, R.L., Saitta, L., eds.: Proceedings of the 16th European Conference on Artificial Intelligence, ECAI'2004, including Prestigious Applicants of Intelligent Systems, PAIS 2004, Valencia, Spain, August 22-27, 2004, IOS Press (2004) 353–357
13. Makinson, D.: Bridges between classical and nonmonotonic logic. *Logic Journal of the IGPL* **11** (2003) 69–96
14. Hitzler, P., Wendt, M.: A uniform approach to logic programming semantics. *Theory and Practice of Logic Programming* **5** (2005) 123–159
15. Schlipf, J.: The expressive power of logic programming semantics. *Computer Systems and Science* **51** (1995) 64–86