Lightweight Reasoning and the Web of Data for Web Science

Edward Thomas
University of Aberdeen
Aberdeen, UK

Jeff Z. Pan
University of Aberdeen
Aberdeen, UK

Stuart Taylor
University of Aberdeen
Aberdeen, UK

Yuan Ren
University of Aberdeen
Aberdeen, UK

ABSTRACT

In the last five years, more than thirteen billion facts have been posted in public, open, semantically rich data sets on the World Wide Web. These data sets and the links between data sets contain an enormous amount of information which is of interest of scientists from all disciplines, but the sheer size of them, combined with the complexity of the underlying languages, makes these data sets unwieldy when tackled with traditional knowledge management tools. In this paper, we look at some new techniques which are available to deal with these problems, and see how and when they should be applied.

Keywords

Semantic Web, Reasoner, Ontology, RDF, Linked Data

1. INTRODUCTION

The Web of Data offers scientists unprecedented access to a huge corpus of data on countless different subjects. Much of this has been annotated with RDF(S) and OWL, which requires additional reasoning in a rule engine or a DL reasoner before complete results can be obtained, requiring heavyweight computation, particularly when the dataset numbers millions or billions of instances. Huge datasets covering many different subjects are being released into the public domain in machine readable RDF from governments (http://data.gov and the UK equivalent, http://data.gov.uk being the most visible), by corporations as diverse as Tesco[1] and Yahoo [2], by research and academic institutions, and by community projects like Wikipedia (via DBPedia [4]). This data covers a huge range of topics from census data [3], music[4], ICT, genetics[5], academic publications[6], and television programming[7]: a smorgasbord for scientists of every shade. We will outline here the various lightweight reasoning strategies which are available to reason with these data sets, and discuss how these can be used to enrich the knowledge on the Web of Data, without the computational overhead required of traditional reasoning mechanisms.

The most obvious area where lightweight reasoning is gaining attention is in the latest W3C recommendation for the second version of Web Ontology Language, i.e., OWL 2, which contains several tractable language profiles, including OWL2-EL, OWL2-QL ad OWL2-RL. Several reasoners have been developed to support these profiles, but currently there is little support for these profiles in ontologies and ontology development tools. We will see how various approximation techniques can be used to perform tractable reasoning on more expressive ontologies, preserving soundness and giving predictable and deterministic levels of completeness. Another area where lightweight reasoning is being performed is in the field of RDF(S), where a DL compatible subset of the language (RDF-DL) has been identified. Many existing RDF(S) schemas are compatible with RDF-DL, as the restrictions it places on RDFS are not onerous [10] and are in keeping with best practice [2]. We will look at how lightweight OWL2-QL reasoners can be adapted to become extremely high performance RDF-DL reasoners by sidestepping the consistency checks and much of the normalisation required in the OWL2-QL loading process [25].

Finally, we will use some common Linked Data ontologies to benchmark several of the most common reasoners, both lightweight and heavyweight. This will give an overview of the full landscape of web reasoning technologies. The paper will conclude with a look to future technologies, and how developments such as uncertainty reasoning (fuzzy and probabilistic logics), and spacio-temporal reasoning will affect this landscape.

2. BACKGROUND

Here we will give a brief overview of the technologies and projects covered by the paper.

[1]http://rdfa.info/2010/01/20/uk-retail-chain-tesco-adopts-rdfa/
[2]http://techcrunch.com/2008/03/13/yahoo-embraces-the-semantic-web-expect-the-web-to-organize-itself-in-a-hurry/
[3]http://www.rdfabout.com/demo/census/
[4]http://dbtune.org/musicbrainz/
[5]http://bio2rdf.wiki.sourceforge.net/
[6]http://dblp.l3s.de/d2r/
[7]http://bbc-hackday.dyndns.org:2825/

Web Science Conf. 2010, April 26-27, 2010, Raleigh, NC, USA.
.

2.1 Semantic Web

The Semantic Web [7] is an evolution of the World Wide Web where the meaning of information is formally defined. This makes it possible for computers to automatically process this data, and for them to understand the content and the context in which it is presented. The World Wide Web Consortium has defined several knowledge representation languages, and has specified formal semantics for these languages, this makes it possible for reasoners to infer additional knowledge from the ground facts presented. Resources on the Semantic Web are represented by uniform resource indicators (URIs), similar to documents on the World Wide Wed, but these may also represent things in the real world (cf. the "Web of Things"), abstract concepts such as a disease or an academic field, or relationships between resources.

The core language for the Semantic Web is the Resource Description Framework (RDF). RDF is a directed, labelled graph language, where resources are related to each other using named properties.

2.2 Linked Data

Linked data is a movement that encourages organisations (corporate, academic, and governmental) to release data in an open format on the Semantic Web. These datasets are linked so that artifacts described in one data set can be related to other artifacts in other sets. An overview of the state of the Linked Open Data Cloud, including the relative size of each dataset, and the links between the datasets, can be seen in figure 1.

Linked data is highly heterogeneous, different datasets use different schemes to describe their data, and the linking is often incomplete or inadequately described [13], however, the quantity and diversity of the data available, as well as the ease by which this data can be accessed, is unprecedented and, if properly managed, could revolutionise how science is performed [6].

2.3 Schema Languages

As we have seen, the main data format used on the Semantic Web is RDF. RDF is an unstructured language, which allows any combination of URIs in its graphs. In order to start to reason with the data available in RDF, it is helpful to impose some sort of structure or schema over it. The most common languages for this are RDF Schema (RDFS) and the Web Ontology Language (OWL). RDFS and OWL share some common elements, they define hierarchies of classes (unary predicates that group individuals with common attributes) and properties (binary predicates which link individuals with other individuals, or with literal data values), however, the underlying philosophies of the two languages differ greatly.

RDFS [9] defines a set of forward chaining rules which can be executed over an RDF graph which will assert new facts based on those already known. In the simplest case, if we define a class "Cat" as a subclass of "Animal", and assert that "Felix" is an instance of the class Cat, the RDFS rules will also assert that Felix is also a member of the class Animal. The full set of rules is part of the published specification of RDFS [9], and there are several implementations of these rules for various knowledge bases and rule engines.

OWL and OWL2 are based on the $\mathcal{SHOIN}(\mathcal{D})$ and $\mathcal{SROIQ}$ description logics [22, 17] respectively. Description logics (DLs) are a family of knowledge representation languages which provide a formal logic and semantics to ontologies on the Semantic Web.

OWL 2 defines a set of tractable fragments of the language which are aimed at mitigation the complexity of the full language by allowing knowledge engineers to use a subset of the language which has known, tractable, computational properties [16]. These are OWL2-EL, based on the EL++ [5] DL, OWL2-QL, based on the DL-Lite [3] DL, and OWL2-RL which is based on DLP [11] and pD* [24]. Each of these fragments is targeted at a particular niche.

EL supports expressive class descriptions and is therefore very useful for describing taxonomic data. It is used in the SNOMED ontology[8], which is a very large medical ontology aimed at providing a common vocabulary for doctors working in differing fields and geographic locations. If this ontology had used the full expressivity of OWL or OWL2, then it would require impractical computational resources to reason over it in a sound manner, since it uses the EL fragment of OWL, modern EL++ reasoners can classify the ontology on a normal desktop computer [20] (also see [15] for a similar result reasoning over SNOMED with the CB Horn-$SHIQ$ reasoner).

QL is a profile of OWL2 for ontologies with very large amounts of instance data, it supports database-style queries over the knowledge base. These queries (as well as the underlying ontology) are rewritten into a set of database queries, meaning that query performance on a QL ontology is comparable to database query performance. Using synthetic benchmarks, QL reasoners have shown to be able to reason over (and query) ontologies containing millions or billions of individuals [25].

RL is the fragment of OWL2 which is conducive to reasoning using forward chaining rules. Because of this, it is possible to execute additional arbitrary rules over the knowledge base. The most expressive decidable OWL 2 language is OWL2-DL, which has direct semantics. The different profiles of OWL and their computational properties can be seen in figure 2.

3. MOTIVATION

The motivation for this work is twofold: firstly, the vast amount of data which is being published in easily accessible and usable formats through such projects as Linked Open Data; secondly, where this data uses a schema or ontology (which is the case for almost all LOD datasets) reasoning or inference must be performed in order to get a valid result for any query. The size of many of these datasets (the LOD cloud contains many datasets which exceed one billion triples[9]) means that reasoning using traditional heavyweight reasoners or in memory rule entailment engines is impractical for all but the most powerful supercomputers. This lead us to evaluate several techniques that can facilitate this reasoning with commodity computing hardware.

Looking more deeply into the problem, we will examine a hypothetical situation involving DBPedia[10]. A (somewhat simplistic) description of DBPedia is that it consists of a set of RDF file generated automatically from the infoboxes on Wikipedia. These infoboxes contain facts and statistics

[8] http://www.snomed.org

[9] http://esw.w3.org/DataSetRDFDumps

[10] http://dbpedia.org

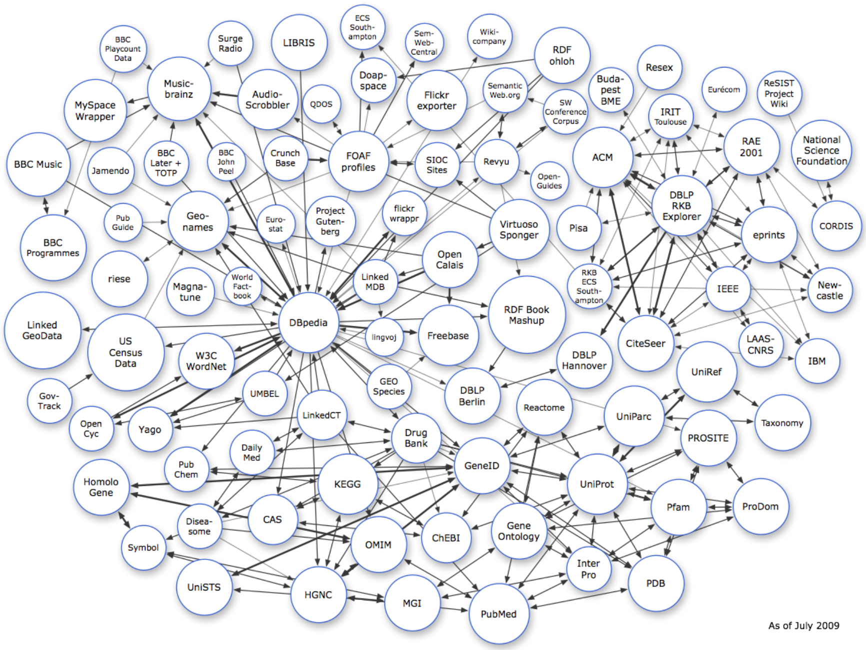

Figure 1: The linked open data cloud as of June 2009 (image ©Richard Cyganiak).

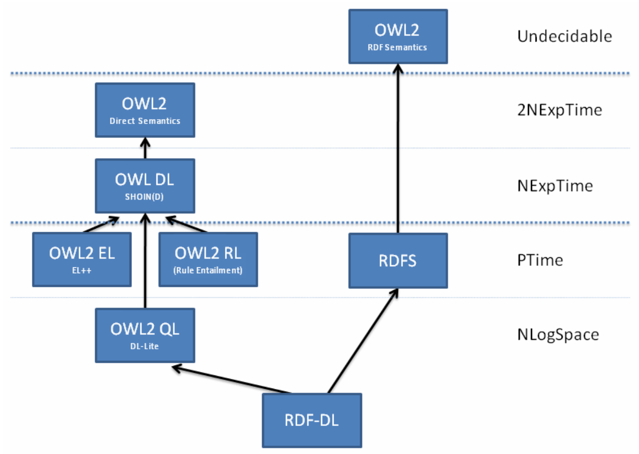

Figure 2: An overview of the Semantic Web language family tree. Note that RDFS becomes undecidable when blank nodes (b-nodes) are permitted. OWL 2 with direct semantics is also called OWL2-DL.

which relate to the subject of each article. The content of the infoboxes is converted into RDF properties which link the URI representing the subject of the article to data constants (in the case of numerical or textual links) or to other URIs representing related subjects. DBPedia has been identified as an important component of the LOD cloud [4]. It is highly heterogeneous and covers a vast number of topics, it benefits from the large public involvement in creating, updating, and correcting articles on Wikipedia, and the generation process is relatively cheap in terms of manpower and computing resources compared to manually curated datasets of a similar size. The current size of DBPedia is 247 million triples, covering content in fourteen languages.

By loading the data from DBPedia into an RDF query engine, we can very easily see the limitations of RDF semantics for the DBPedia data, when it is combined with one of the ontologies which are part of the DBPedia project. One such ontology is YAGO [23] which unifies the categories in Wikipedia and Wordnet to produce a taxonomy of concepts which can be used to categorise the instances in DPBedia. A simple query of this ontology can be made in SPARQL to find all instances of books known by Wikipedia:

```
PREFIX yago:  <http://dbpedia.org/class/yago/>
SELECT *
WHERE { ?book a yago:Book106410904 .  }
```

Executing this query will return a total of 4,481 results (using DBPedia release 3.2); this is only a fraction of the total number of books that are in the DBPedia dataset. The reason for this discrepancy is that most books in Wikipedia are classified in some sub category of book, for example "Books about Baseball" which are defined as sub classes of Book by Yago. Unless each possible subclass (and there are a total of 237,159 sub-class axioms in the ontology) are specified in the query, an RDF only query engine will fail to return a complete result, and may in many cases return an empty result.

This is only one facet of Linked Open Data which requires reasoning. The core of the LOD philosophy is that instances in one dataset that describe the same thing as an instance in another dataset should have this explicitly specified by using the owl:sameAs property [8]. This is part of OWL DL, and as such, requires inference with a reasoner capable of supporting this construct. In the next section we will look at some techniques which can help to deal with the reasoning task.

4. DEALING WITH LARGE SEMANTIC DATASETS

Working with large data set which use the full expressivity of RDFS or OWL is problematic. Datasets which are on the order of dozens of gigabytes will not fit into the main memory of any but the most powerful computer systems. Performing rule entailment on millions of triples on disk is extremely slow. Work has been ongoing on rule entailment techniques which support RDFS and OWL2-RL reasoning using clusters of commodity computers [26, 18] but these still require large computer resources. Here we will look at some techniques which make it easier to perform useful and pragmatic reasoning with a more limited computing infrastructure.

4.1 Syntactic Approximation

Syntactic approximation is the process of converting an intractable and expressive language into a tractable, but less expressive language by examining and modifying the syntax of the language. In the simplest case, one can look at each axiom of the dataset, and discard those axioms which cannot be expressed in the target language. This naïve approach will result in a lot of information being lost, but every result that can be derived from the resulting knowledge base will be true with respect to the original data set, i.e. it will be sound but incomplete (in fact, an "approximation" which discards every axiom in the knowledge base is also sound and incomplete).

4.1.1 OWL2-DL to OWL2-EL

A more refined syntactic approximation has been developed for converting OWL2 ontologies into OWL2-EL ontologies [20]. It has been shown that for many benchmark ontologies, this approach is produces both sound and complete results (ibid), although at the time of writing, this cannot be guaranteed for any arbitrary ontology. The benefit of this approximation is that the resulting ontology has the tractable properties of OWL2-EL, and the approximation process takes linear time ($O(n)$), therefore the total time required to load, approximate, and reason over the ontology is far less than that of a traditional expressive reasoner. The two approaches can be used in parallel to provide anytime reasoning, where the initial results (quickly) returned from the EL reasoner is supplemented by the (slower) results from the complete reasoner as they become available.

4.1.2 OWL-DL to Rule Entailment

Several rule based reasoners allow reasononing to be performed on OWL ontologies by using a syntactic approximation to the some set of entailment rules. In general these are some fragment of Horn-$\mathcal{SHIQ}$ [14] such as DLP [11] or pD* [ibid]. For example Jena, OWL-IM, MARVIN, Virtuoso and many other reasoners use some variant of this technique. This modifies the semantics of OWL from a description logic to a set of forward chaining rules which can provide an incomplete, simulation of the inferences made under OWL semantics. There currently is no proof that this approach can be made to guarantee sound results for all ontologies, although it can be proven on a case by case basis by comparing the results of reasoning tasks with a sound and complete reasoner such as Fact++ or Hermit. The benefit of such an approximation is that the output of such inferences is a finite set of axioms which can be queried using a graph query language such as SPARQL.

4.1.3 RDFS to RDF-DL

As we saw earlier, the complexity of reasoning and querying RDFS is undecidable, furthermore, full rule entailment of RDFS requires a lot of time (particularly when the dataset is too big to fit in memory), and it results in a dataset which is polynomially large with respect to the input dataset ($O(n^k)$). One technique which can be used to mitigate this is to approximate the RDFS knowledge base into RDF-DL. RDF-DL is the intersection of the syntax of RDFS with the semantics of OWL2-QL. The benefits over full RDFS are that the dataset can be stored without storing redundant knowledge, and by using QL query rewriting, sound and complete results can be obtained at query time. Since RDFS does not contain negation, it is impossible to create an inconsistent knowledge base in RDFS, therefore the time consuming consistency checking required in QL can be avoided; this allows a reasoner to work with very large knowledge bases in a streaming manner, working with individual RDF triples, and saving memory. The only structures which need to be stored in memory as the knowledge base is loaded are class and property hierarchies which are typically much smaller than the complete knowledge base. This technique has been proven to be capable of working with billions of triples (several hundred gigabytes of RDF taken from the Semantic Web, along with various RDFS and OWL schemas) on commodity hardware [25]. The performance of the reasoner in this case allowed data to be loaded at >10,000 RDF triples per second; this is comparable to the performance of a plain RDF triple store loading the data on the same hardware (tested against Jena-TDB).

4.2 Semantic Approximation

Semantic approximation works at the semantic level of the OWL languages. It typically requires the use of a heavyweight (OWL2) reasoner to ensure that the approximation is sound, and as complete as possible with respect to the input ontology. The reasoner is used to derive every expressible axiom in the target language which is true with respect to the input language. Semantic Approximation is an offline process which can calculate a tractable representation of an input ontology, in order that online queries can be performed more efficiently.

4.2.1 OWL2-DL to OWL2-QL

Semantic Approximation from OWL2 to OWL2-QL calculates the OWL2-QL approximation of an expressive OWL2 ontology [19]. The resulting ontology can be stored in a relational database, and guarantees sound results for any conjunctive query, as well as sound and complete results for any query which is expressible in OWL2-QL. The approximation process has similar complexity to the underlying ontology, NExpTime in the case of OWL-DL, and 2NExpTime in the case of OWL2 with complex roles. The approximation process uses the heavyweight reasoner to calculate the least upper bound of OWL2-QL axioms which are valid with respect to the original ontology. This represents the minimal set of axioms which, under OWL2-QL inference, can express the maximum number of inferable axioms in the source ontology.

4.3 Summary

Source Language	Target Language	Method	Sound	Complete	Online/Offline	Main Application
RDFS	RDF-DL	Syntactic	Yes	No	Online	Conjunctive query answering
OWL2	OWL2-EL	Syntactic	Yes	Case Specific	Online	TBox reasoning
OWL2	OWL2-QL	Semantic	Yes	Yes[11]	Offline	Conjunctive query answering
OWL2	Rules	Syntactic	unknown[12]	No	Offline	Expressive query answering

Table 1: A summary of language transformations for Semantic Web languages

We have introduced four possible language transformations which make reasoning on the Semantic Web more tractable. The main properties of each transformation are summaries in table 1. Each of these transformations has particular strengths and weaknesses. For dealing with extremely large data sets, an approximation to RDF-DL allows the data to be loaded very quickly into a single machine, and balances this with good query performance for simple conjunctive queries. Syntactic approximation to OWL2-EL allows TBox reasoning to be performed on extremely large ontologies which could not otherwise be reasoned over within a reasonable amount of time (cf. the FMA ontology as shown in [20]).

5. NOTES ON PERFORMANCE

Here we will briefly review the published and theoretical performance of the methods summarised above. In each case we will be examining the performance figures from published papers, so in each case it may not be possible to directly compare different methods. The main comparison will be against other reasoning methods.

In summary, all the reasoning methods here have demonstrated significant performance advantages over the standard reasoning methods for each language, either in terms of time required, or computing resources consumed.

5.1 RDFS to RDF-DL

RDFS to RDF-DL approximation has two major benefits. Firstly, only TBox inference is performed at load time, so for standard synthetic benchmarks such as LUBM [12], where the TBox does not change for larger datasets, performance scales linearly with the size of the input. Secondly, minimal redundant data is stored, meaning that the stored representation of the ABox is smaller than for other methods.

5.2 OWL2-DL to OWL2-EL

The approximation to OWL2-EL reduces the complexity of OWL inference from a worst case of 2NExpTime to PTime. This can clearly be seen in benchmarks against the leading OWL2 reasoners in [21], where the approximation is able to outperform by one or two orders of magnitude, whilst consistently delivering recall (ie, completeness) of over 99%.

5.3 OWL2-DL to OWL2-QL

The sound and complete approximation of OWL2 to OWL2-QL allows for very fast conjunctive query answering. In benchmarks against contemporary reasoners, the approximation was shown to be far more scalable, and also capable of handling much more data than the memory based reasoners [19].

[11]Completeness guaranteed reasoning requires additional pre-processing

[12]As far as we know, there are no published soundness proofs for the mentioned approaches

5.4 OWL-DL to Rule Entailment

The approximation from OWL2 to some fragment of Horn-SHIQ rules results in a finite set of RDF triples which represent the entailed ontology. This means that a single RDF triple store can be used for storage and querying. Modern RDF stores can scale to tens of billions of triples [1] and perform expressive SPARQL queries which go beyond what is possible with conjunctive queries over OWL2-QL.

6. CONCLUSIONS AND FUTURE WORK

The availability of semantically rich data on the Web has the potential to open up new avenues for science. For the first time it is possible to tap into very large data sets without having to perform tedious manipulation of that data to align it to a particular idiom. Ontologies in particular simplify greatly the process of mapping between diverse data sources. The scale of the data available has brought with it new problems, the computational complexity of the languages used makes it impossible to reason efficiently with much of this data without applying some transformation to mitigate this complexity. In this paper we have presented a number of techniques which have been proven to work efficiently to solve these problems.

Research into knowledge representation, reasoning, and query answering for the Semantic Web is very active. Much of it is directed at dealing with dynamic data, which can constantly update the representation stored in a reasoner. Sources of such dynamic data on the Web are as diverse as a Facebook or Twitter feed to the output from a scientific instrumentation such as the information released to the Seti@Home project. With dynamic data, it is important that a knowledge base can maintain a consistent picture of the data at any one time, and in many areas, being able to query the state of the knowledge base at some point in the past is essential. Other research is being directed at fuzzy and probabilistic extensions of Semantic Web languages. This makes it easier to model data where there is no exact true or false value (in the case of fuzzy data), or where there is disagreement or uncertainty as to whether some axiom is true or not (probabilistic data). Integration of this research into the Semantic Web toolkit will make it easier to apply reasoning techniques to real world data. Finally, several extensions have been proposed for spacio-temporal extensions to Semantic Web languages. These add constructors to locate knowledge within a one, two, three, or four dimensional space; extending query languages and the knowledge itself with constructors for specifying volumes in which data lies.

The common factor to all these extensions is that they further increase the complexity of reasoning and querying knowledge. This makes it even more imperative that novel techniques are developed and employed to mitigate this complexity wherever possible.

7. REFERENCES

[1] Large triple stores. http://esw.w3.org/LargeTripleStores, 2010.

[2] D. Allemang and J. Hendler. *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. Morgan Kaufmann, May 2008.

[3] A. Artale, D. Calvanese, R. Kontchakov, and M. Zakharyaschev. The dl-lite family and relations. *J. Artif. Int. Res.*, 36(1):1–69, 2009.

[4] S. Auer1, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. DBpedia: A Nucleus for a Web of Open Data. In *Proceedings of the 6th International Semantic Web Conference*. Springer Berlin / Heidelberg, 2008.

[5] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence IJCAI-05*, Edinburgh, UK, 2005. Morgan-Kaufmann Publishers.

[6] T. Berners-Lee, W. Hall, J. Hendler, N. Shadbolt, and D. J. Weitzner. Computer science: Enhanced: Creating a science of the web. *Science*, 313(5788):769–771, August 2006.

[7] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001.

[8] C. Bizer, R. Cyganiak, and T. Heath. How to publish linked data on the web. http://www4.wiwiss.fu-berlin.de/bizer/pub/LinkedDataTutorial/, 2007.

[9] D. Brickley and R. Guha. Rdf vocabulary description language 1.0: Rdf schema. http://www.w3.org/TR/rdf-schema/, Feb 2004.

[10] B. C. Grau. A possible simplification of the semantic web architecture. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 704–713, New York, NY, USA, 2004. ACM.

[11] B. Grosof, I. Horrocks, R. Volz, and S. Decker. Description logic programs: Combining logic programs with description logics. In *Proceedings of the World Wide Web Conference (WWW2003)*, Budapest, Hungary, 05 2003.

[12] Y. Guo, Z. Pan, and J. Heflin. LUBM: A Benchmark for OWL Knowledge Base Systems. *Journal of Web Semantics*, 3(2):158–182, 2005.

[13] A. Hogan and R. Cyganiak. Frequently observed problems on the web of data. http://pedantic-web.org/fops.html, 2009.

[14] U. Hustadt, B. Motik, and U. Sattler. Data complexity of reasoning in very expressive description logics. In *IJCAI'05: Proceedings of the 19th international joint conference on Artificial intelligence*, pages 466–471, San Francisco, CA, USA, 2005. Morgan Kaufmann Publishers Inc.

[15] Y. Kazakov. Consequence-driven reasoning for horn shiq ontologies. In *IJCAI'09: Proceedings of the 21st international jont conference on Artifical intelligence*, pages 2040–2045, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[16] B. Motik, B. C. Grau, I. Horrocks, Z. Wu, A. Fokoue, and C. Lutz. Owl 2 web ontology language profiles. http://www.w3.org/TR/owl2-profiles/, Oct 2009.

[17] B. Motik, P. F. Patel-Schneider, and B. Parsia. Owl 2 web ontology language structural specification and functional-style syntax. http://www.w3.org/TR/owl2-syntax/, Oct 2009.

[18] E. Oren, S. Kotoulas, G. Anadiotis, R. Siebes, A. Ten Teije, and F. Van Harmelen. Marvin: distributed reasoning over large-scale semantic web data. *Journal of Web Semantics*, 2009.

[19] J. Z. Pan and E. Thomas. Approximating OWL-DL Ontologies. In *the Proc. of the 22nd National Conference on Artificial Intelligence (AAAI-07)*, pages 1434–1439, 2007.

[20] Y. Ren, J. Z. Pan, and Y. Zhao. Soundness preserving approximation for tbox reasoning in r. In B. C. Grau, I. Horrocks, B. Motik, and U. Sattler, editors, *Description Logics*, volume 477 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009.

[21] Y. Ren, J. Z. Pan, and Y. Zhao. Soundness preserving approximation for tbox reasoning. In *Proc. of the 25th Nat. Conf. on Artificial Intelligence (AAAI 2010)*, 2010.

[22] M. K. Smith, C. Welty, and D. L. McGuiness. http://www.w3.org/TR/owl-guide/. http://www.w3.org/TR/owl-guide/, Feb 2004.

[23] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A large ontology from wikipedia and wordnet. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(3):203 – 217, 2008. World Wide Web Conference 2007Semantic Web Track.

[24] H. J. ter Horst. Completeness, decidability and complexity of entailment for rdf schema and a semantic extension involving the owl vocabulary. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(2-3):79 – 115, 2005. Selcted Papers from the International Semantic Web Conference, 2004 - ISWC, 2004.

[25] E. Thomas and J. Z. Pan. R-quill: Reasoning with a billion triples. In *8th International Semantic Web Conference (ISWC2009)*, 2009.

[26] J. Urbani, S. Kotoulas, E. Oren, and F. van Harmelen. Scalable distributed reasoning using mapreduce. In *8th International Semantic Web Conference (ISWC2009)*, October 2009.