Standardization of Ontologies

Kore Nordmann <kore@php.net>

May 13, 2009

1 Introduction

Ontologies are a theoretical concept in Computer Science to formally represent knowledge in a way software can process the knowledge and reason about about it. Often also human readability is a central aspect when defining ontology definition languages so that definitions are easy to craft and maintain.

The last years showed different approaches of representing ontologies and integrating them with existing data. The process of integrating the different approaches and tools around ontologies can be viewed by different main characteristics. This paper will take a look at the historic evolution of ontologies and related technologies, the influence and development of related tools and the theoretical background of ontology definition languages together with the resulting inference complexity.

2 Historic evolution

The first key characteristic of the standardization of ontologies is the development of the ontology markup formats and associated standards over the time, which also show the evolving demands for semantic markup. The largest distributed pile of data currently is the internet, processed by a lot of different involved applications. This involves static data, as well as webservices interacting with each other and different data sources to build new services.

The development of the key technologies, which today build the internet, is shown in figure 1.

2.1 Beginning of markup

The development of what is today known as the "internet" started in 1989 with the development of HTML [BL89]. HTML has been developed together with the first version of HTTP by Tim Berners-Lee as a markup format for scientific documents, that should be shared using a common language and access protocol. HTML offered a domain specific semantic markup for documents - and can even today be used as such.

In 1994 one of the first ontologies in computer science has been developed, UNTANGLE, which tried to define semantics for a specific domain. The domain are catalogue card systems, for which a simple taxonomy has been implemented. [WJ00]. UNTANGLE was not intended as a generic language to define custom

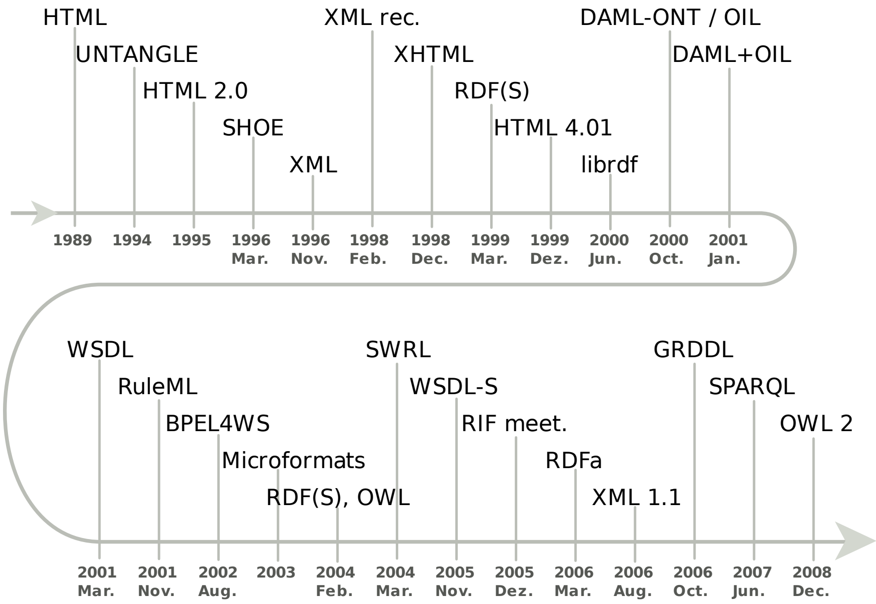

Figure 1: History of ontology related technologies

ontologies for your data, like the following ontology definition languages are, and has not been integrated with HTML or similar markup standards.

Since HTML proved to be a good idea it has been standardized in 1995 [BL95] after the initial working draft from 1989. In the meantime more and more applications evolved on the internet, which used HTML not only as a document markup language, but to present different kinds of data to users of web applications. This starts with forums, which at least contain "documents" as their posts and does not stop before online shops, which were not able to reuse the markup in the originally intended semantics. This resulted in the need for custom semantics modeling the application domain.

2.2 Defining first ontologies

To be able to annotate semantics of different applications, and embed those semantics inside of HTML the development of SHOE started in 1996 [BL96]. SHOE intended to be "an extension to HTML which provides a way to semantically describe important information about HTML" [BL96]. It offered a simple syntax, similar to DTD, to specify ontologies and relate them to existing documents. Its features will be described in further detail during the detailed discussion of ontologies in the third chapter of this paper.

HTML used a simplified version of Standard Simplified Markup Language (SGML), which proved to be a generic, easily readable syntax - for both, humans and software. To make this syntax usable for more applications then the plain document centric HTML markup the development of XML started in 1996. [TB96] XML was already intended as the foundation for future versions of HTML and especially focussed on extensibility. This led to the possibilities of embedding other markup languages inside existing documents, without violating the markup standard of the original document.

XML already reached its W3C recommendation state two years later, in February 1998. [TB96] With this base the development for a new variant of HTML started in 1998 - XHTML. [DRW98] XHTML was intended to offer different modules, which define subsets and supersets of the used HTML at this point in time. HTML itself has continuously has been extended by the developers of HTML clients, apart from the actual standards. XHTML was meant to lift this to a common base again - which never really happened.

2.3 Standardization

With XML and its extensible base, prepared to be used on websites with XHTML the development of RDF (Resource Description Framework) and RDF-Schema could start in 1999. [BG99] RDF-Schema can be considered a successor of SHOE, using the same theoretical base, which will be described later in more detail. RDF on the other is used to associate any kind of XML documents with semantic information, specified using RDF-Schema. These first specification drafts should still take five more years, until they reach recommendation state, but already were used as the base for the most following ontology definition languages and tools to handle semantic markup.

The currently last version of HTML, 4.0.1, followed in September 1999. [DRJ99] Its potential predecessors XHTML 2 and (X)HTML 5 are currently still under discussion, and it is still not clear which kind of semantic markup features they will allow to embed. Since HTML 4 is still not compatible with XML, it does not allow to embed RDF or any other semantic markup. Approaches like Microformats, which will be discussed later, are still the only possibility to embed semantics inside HTML without violating the standard.

Only a few months after the first working draft for RDF and RDF-Schema the first commits to librdf, also called Redland RDF library, occurred. [Bec00] The Redland library today is still one of the more popular RDF libraries, implementing RDF triple extraction, a RDF triple store and RDF triple store querying. Nowadays it offers bindings for several languages and application development platforms and thus facilitates the development of applications involving representations of semantic data.

During late 2000 two initiatives started to extend ontology definition languages, because RDF-Schema was not expressive enough for their purposes. [HPsH03] The DAML (Darpa Agent Markup Language) program therefore started to develop DAML-ONT with a first specification in October 2000. [BLvHH$^+$00] OIL, the Ontology Inference Layer, has been developed in parallel by a group of European researchers. [HPsH03] It intended to combine well known element from description logics, frame-logics and the already mentioned web standards, like XML and RDF. It was the first ontology definition language, which was explicitly designed to match a description logic style.

The efforts of both working groups were combined to DAML+OIL in early 2001. [HvHBL$^+$00] DAML+OIL mostly maintained the formal semantics of description logics from OIL. Influenced by DAML-ONT it was also more tightly integrated with RDF. [HPsH03] The usage of RDF as the syntax for inference rules, did not follow the explicit RDF Semantics defined three years later in

January 2003. [Hay03] Therefore a new language for ontology definitions had to be defined, which better integrates with existing standards.

OWL (Web Ontology Language), RDF and RDF-Schema had not been formally specified until February 2004. [BM04, BG04, MvH04] The main addition to RDF since the first working draft was the specification of the RDF Semantics in 2003. [Hay03] RDF-Schema has mostly left untouched, but extended by OWL, which also supersedes DAML-ONT and DAML+OIL by following the defined RDF semantics.

Since 2008 OWL 2 is under development, [BM08] which intends to define some more OWL variants with different complexity levels, like explained later in this paper.

2.4 Evolving tools

Since the definition of first standards for communication on the internet and the already mentioned development of ontology languages various tools evolved around those.

HTML has been used for web sites and web applications, but the more generic approach of XML made it also useful for other applications. WSDL, the Web Services Description Language, which has been approved 2001 by the W3C [ECW01] for example allowed the specification of web service interface using XML. Such web services can also be enriched with semantics using ontologies, like proposed in the Web Service Semantics (WSDL-S) membership submission in November 2005. BPEL, the Business Process Execution Language, first defined by OASIS in August 2002 [Oas02] specifies the interactions with web services, instead of only the API, like WSDL does. With semantic annotations like with WSDL-S such BPEL specifications can be automatically generated from the web service specifications. [RKM06]

In 2001 the development of RuleML started [Bol08] at the MIT, a generic language for defining inference rules from XML documents. It has been followed up by SWRL in May, 2004 [HPSB$^+$04], which focussed on integrating RuleML with OWL. In December 2005 the W3C started a new working group, to develop the Rule Interchange Format (RIF) [Haw05], which currently is still under discussion, but focusses from the very beginning on inferencing rules for ontologies, and especially OWL.

With Microformats, developed since 2003 [Tan06], and RDFa there are two more ways to annotate semantics in existing documents. Both are developed especially for annotating semantics in HTML. RDFa also still is under development and the first working draft has been published in March 2006. [AB06] To merge those different annotations of semantics in XML or HTML documents the W3C started to develop a mechanism for "Gleaning Resource Descriptions from Dialects of Languages" (GRDDL) in September 2006. [Dav06] GRDDL focusses to extract the semantics annotated using Microformats or RDF and offer RDF triples from that.

With RDF triple stores, like the earlier mentioned Redland RDF library, the semantic annotations like Microformats, RDF and RDFa languages for querying such data are required. RDQL (RDF Query Language), as first proposed to the W3C Query Languages Workshop in December 1998, [MS98] was one of the first approaches to develop a declarative language for querying RDF stores.

SPARQL, the "SPARQL Protocol and RDF Query Language", is its successor and a W3C recommendation since January 2008. [PS08]

3 Related technologies

As shown by the short historical overview there are several technologies directly associated with the ontologies, which influenced the standardization process itself.

3.1 Microformats

Microformats are an approach to embed more concrete semantics directly in HTML documents without breaking the existing markup or standard. Since HTML does not support name spaces existing elements or attributes had to be reused to accomplish this. Microformats do not aim at the specification of ontologies, but implement a set of standards, which each implement a concrete semantic.

One example is the hCalendar markup for events, for which no markup exists in HTML itself.

```
<div class="vevent">
    <span class="summary">Semantische Services</span>:
    <abbr class="dtstart" title="2009-02-09">February 9th</
        abbr>-
    <abbr class="dtend" title="2009-04-18">April 18th</abbr>,
     at <span class="location">TU Dortmund</span>
</div>
```

Like the example shows the HTML `class` attribute is used for most of the semantic annotations. It is available to most HTML elements, so that the markup can be embedded nearly anywhere. The used HTML elements itself are irrelevant to the Microformat readers, only the `class` attribute is taken care of. Since HTML allows to embed multiple values inside `class` attributes, separated by spaces existing values can be maintained and only extended by the semantic attributes. This is especially necessary, because the class attributes are often already used to associate layout constraints to HTML elements using CSS.

3.1.1 Defined Standards

The list of applications for Microformat based semantic markup is rather short at the current state. There is no dedicated organization, like the W3C, behind the standardization initiative.

hCalendar Microformat markup used to embed the iCalendar[1] standard inside XML and XHTML based languages. The iCalendar format is used as a markup language for distributing calendar events, like known from CalDAV servers.

hCard Microformat based markup to embed vCard[2] contact information inside XHTML.

[1] http://www.ietf.org/rfc/rfc2445.txt
[2] http://microformats.org/wiki/rfc-2426

rel Microformat standard using the rel attribute of links in XHTML to define the type of relations between documents. The relations yet specified are “nofollow”, “license” and “tag”.

- **nofollow** Actually not really semantic markup, but still included in the specification. This processing instruction specifies how search robots should weight the given link in their pagerank analysis.
- **license** A reference to the license of the current document.
- **tag** Defines the linked entity as a tag with a link to more resources using the same tag.

VoteLinks Markup for votings, specifying the type of voting (pro, contra) by the user and the resource the user votes on.

XFN Markup for relations between persons represented by documents in the Internet. Contains various tag like descriptors for the exact persons relations.

XMDP Document metadata descriptions, like already part of XHTML headers.

3.1.2 Usage

The main problem behind Microformats is the default of any formal specification of the semantics, or a direct association with existing ontology definition languages. Also there is no way to specify the structure of Microformat markup.

Even the used Microformats must be declared in the HTML header the names of the properties itself are not name spaced. This way it is not possible to develop custom Microformats in a forward compatible way, because at any point another Microformat may be specified which uses the same names as your custom definition, or an existing Microformat is extended to use them.

3.2 RDF

RDF, the Resource Description Framework, is a generic approach to reference semantics in documents specified in external ontologies. RDF always consists of `<Subject, Predicate, Object>` triples, where the subject is the described resource, the predicate defines the type of the description, and the object the value of the description. The object itself can again be described by by further RDF triple, which is called reification.

There exist different syntaxes for RDF, for example N3 (notation 3), defined by Tim Berners-Lee, which is easily readable for humans. The more popular syntax for RDF triples is its XML syntax, which allows RDF to be embedded in all documents, which also use XML as a syntax, which for example includes XHTML.

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
    <rdf:Description rdf:about="/blog/
        the_long_way_to_semantic_web.html">
```

```
        <dc:creator>Kore Nordmann</dc:creator>
        <dc:title>The long way to a semantic web</dc:title>
        <dc:date>Thu, 25 Oct 2007 11:20:13 +0200</dc:date>
        <dc:rights>CC by-sa</dc:rights>
        <dc:language>en</dc:language>
    </rdf:Description>
</rdf:RDF>
```

This simple example only contains one RDF specification with multiple RDF triples. The described subject is given as the `about` attribute of the `description` element. The `description` element again contains elements for the predicates, which again contain the describing object as its values.

This generic approach of triples fit other applications quite well, so that it also used as a “syntax” to specify DAML-ONT, RDF-Schema or OWL. Since 2003 an explicit semantic for RDF triples has been defined, which makes RDF unusable for some of those applications, which will be covered in more depth later.

3.3 RDFa

RDFa, which is still under development as a standard, tries to make it easier to embed RDF triples inside of XHTML. Embedding RDF directly in XHTML may lead to problems with current browsers, because CSS 2 can not define namespace specific style directives, so that it might occur that all elements from namespaces other then the XHTML namespace are displayed as plain text inside websites by current web clients.

For this RDFa specifies some new elements, which are not defined in the HTML or XHTML standards and defines an additional DTD, which adds those elements to the original XHTML DTDs. This way the documents embedding RDFa markup are not valid HTML or XHTML markup anymore, but the specification [AB06] states:

> The authors know of no deployed Web browser that will fail to present an HTML document as intended after adding RDFa markup to the document. However, publishers should be aware that RDFa will not validate in HTML4 at this time.

Ignoring this violation of existing standards RDFa offers the simplicity of Microformats when it comes to the integration with existing markup, like the following example shows.

```
<div xmlns:dc="http://purl.org/dc/elements/1.1/">
    <h2 property="dc:title">Das semantische Web</h2>
    <h3 property="dc:creator">Kore Nordmann</h3>
</div>
```

The `property` attribute in this example is one of the attributes and elements defined by the XHTML+RDFa-DTD. The two given RDF triples define information about the document they are embedded in. For the used ontologies any name space can be used, which enables the use of arbitrary ontologies, like the DublinCore ontology in this example. This way RDFa combines two advantages from RDF and Microformats, with the annoyance of invalidating existing documents.

3.4 GRDDL

The different methods to associate semantics with documents in the web, like Microformats, RDF and RDFa required a technology to extract such data in a unified way and offer a unique way to access it.

GRDDL (Gleaning Resource Descriptions from Dialects of Languages) aims to provide a mechanism for this, currently focussing on embedded RDF and Microformats. Using this Microformats are mapped to ontologies representing the respective Microformats standard.

Even GRDDL is still under development it will enable developers to use Microformats and still benefit from ontologies, their concepts and inference rules. The integration of the different annotation methods make it possible to focus on the actual ontologies and transparently switch between the annotation method.

3.5 SPARQL

SPARQL and its predecessor RDQL define query languages for RDF triples. A triple store can aggregate sets of RDF triples, and a SQARQL query can the be used to formulate constraints under which a set of those triples should be fetched from the database. The mentioned Redland RDF library for example can already evaluate SPARQL queries, and so do other libraries. A SPARQL query looks quite similar to SQL queries for relational databases.

```
PREFIX dc: <http://purl.org/dc/elements/1.1/#>
SELECT ?title
WHERE {
    ?doc dc.title ?title.
    ?doc dc.creator "Kore Nordmann".
}
```

The `PREFIX` defines the ontologies used in the query, similar to XML name spaces with their reference to the used XML schema. The variables after `SELECT`, which can either start with `?` or `$`, define the values, which should be returned from the query. The conditions in the `WHERE` section have to match the subject (resource), predicate, object structure of RDF and limit the subset of returned values. All conditions of the `WHERE` group must match, while more complex query structures using sub conditions, filters and optional matches can be defined.

Such query languages enable structured querying of data available by RDF triples. This data can originate from different ontologies, while either the subject or object obviously need to match to merge the data in one query. Together with GRDDL this will work for semantic information annotated in various formats. Some of the applications and libraries already implementing SPARQL querying also already provide different facilities for data extraction themselves.

3.6 RuleML

RuleML has been developed to perform inferencing on arbitrary XML documents, not related to semantics or ontology definition languages. It implements support for different styles of inference rules, like:

- reaction rules
- transformation rules

- derivation rules
- facts
- queries
- integrity constraints

Inference rules are specified explicitly for XML instances and generally do not relate to ontologies. An integration with DAML-ONT was planned, but never realized. An integration with OWL has been realized under the name SWRL, but the W3C currently works on RIF, which will most probably supersede RuleLM and SWRL.

3.7 RIF

The Rule Interchange Format (RIF) is in an early and very active planning state at the W3C and is intended to define an uniform interchange format for inference rules. The inference rules itself will most probably be based on Horn logic rules. [Haw05] The recommendation state is already scheduled for June 2009.

3.8 Summary

The Semantic Web Activity "layercake" diagram [Her09] in figure 2 shows how the W3C sees the relations between the mentioned technologies.

The very basic technologies like URIs and XML were not mentioned in this chapter. The next layer is the annotation of data semantics, where this layer cake diagram only shows RDF, while with GRDDL, RDFa and Microformats might be used too, in the future. To stay on the safe side and not rely on parsing invalid markup or ambiguous specifications RDF still should be the best way to go.

There are various means to interact with the annotated data. On the one hand there are already usable query languages like SPARQL. On the other hand there will be generic interchangeable inference rules for the data, to be specified using RIF. The ontologies definition languages like OWL and RDF-Schema, which can be used to define structure and axioms for the annotated data will be topic of the next chapter.

Applications may combine the given technologies and data from various sources to provide unified view on the data. Especially with inference rules, either defined in RIF or the ontology specification language itself and axioms it is possible to also proof on top of the existing data.

4 Ontology development

Even the development of the associated tools already shows the way ontology development takes there are some key characteristics, which especially involve the reasoning complexity, which are not covered yet. The logic background of ontologies, evolving during the development of new ontology definition languages, is important to understand the development and some characteristics of ontologies.

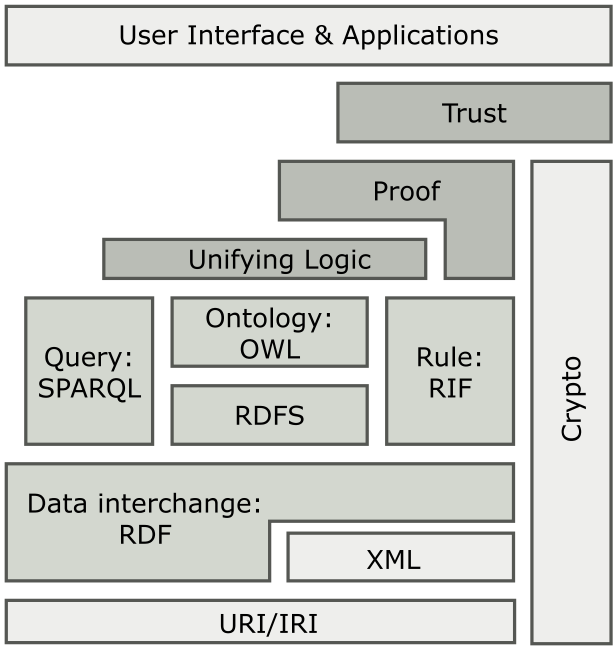

Figure 2: W3C Semantic Web Activity ”layercake” diagram [Her09]

4.1 SHOE

SHOE was one of the first attempts to create an ontology definition language to embed semantics inside HTML. [HPsH03] The general approach of SHOE is based on frame languages (F-logics). Therefore the ontology definitions mostly consists of:

- Class name
- More general class(es) (inheritance)
- List of ”slots”, property-value pairs or value constraints

This approach is very similar to Object Oriented Design. A simplified version of the “Personal Ontology”, an ontology for relations between persons and their personal data shows the principle of such an ontology definition: [Hef00]

```
[gen.base.SHOEEntity]
[...]
Address
Person
    Employee
```

This is a simplified syntax for the shortened class inheritance diagram. It shows two classes, where the class “Employee” inherits from the class “Person”, and an additional “Address” class. Between those classes relationships can be defined, as well as literal properties for classes.

```
addressCity(Address, .STRING)
[..]
homeAddress(Person, Address)
[..]
father(Person:"child", Person:"father")
friend(Person, Person)
[..]
```

This example first shows a simple property of the `STRING` literal type, which defines the city property of an “Address” class. Classes can be related with an aggregation relationship like the defined `homeAddress` for “Persons”. Relations between objects of the same class like the `father` relation in this example, are also possible.

Beside those it is possible to define simple inference rules in SHOE.

```
Child and parent are inverse relations.
    If child(x,y) then parent(y,x). If parent(x,y) then child(y,x).
[..]
```

SHOE already supported importing of other ontologies, and generally influenced the following development of ontologies.

4.2 RDF-Schema

RDF-Schema has first been proposed together with RDF. It follows the same frame-based paradigms as SHOE, but reuses RDF itself as a syntax for the ontology specification.

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">

    <rdf:Description ID="Person">
        <rdf:type resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
    </rdf:Description>

    <rdf:Description ID="Employee">
        <rdf:type resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
        <rdfs:subClassOf rdf:resource="#Person"/>
    </rdf:Description>
</rdf:RDF>
```

The example shows the same subclass relation, as defined before in SHOE. The features are mostly equivalent to the features in SHOE. The switch to a RDF like syntax is obvious from the example.

4.3 DAML-ONT

DAML-ONT has been developed because the available inferences in RDF-Schema weren’t sufficient for some applications. Even RDF-Schema already entered the standardization process at the W3C, DAML-ONT has been developed to extend

"RDF with language constructors from object-oriented and frame-based knowledge representation languages." [HPsH03] But it got merged rather quickly together with OIL into DAML+OIL.

4.4 OIL

In parallel to DAML-ONT OIL has been developed, the first time with a stronger focus on Description Logics (DL), which should be combined with the existing technologies like RDF and XML. It has been designed to explicitly match the semantics of the $\mathcal{SHIQ}$ Description Logics style. [HPsH03] Further information about description logics can be found in "The Description Logic Handbook: Theory, Implementation and Applications" by Franz Baader, et al. [BCM+03] The general structure of the definitions still stayed frame-based. If offered different syntaxes in XML and RDF, while the second one stayed closer to already known languages like RDF-Schema.

Because of the focus on Description Logics semantics it is the first ontology definition language which is not undecidable. The general complexity of the $\mathcal{SHIQ}$ Description Logics style is NExpTime. [wTAiST01]

4.5 DAML+OIL

The development of DAML-ONT and OIL has been merged into DAML+OIL very short after the initial development of both languages. The merged language did not entirely follow the DL style used by OIL, but used its own DL style. The frame structure of both languages has been discarded in favor of more DL style axioms, which better matched its RDF syntax. [HPsH03]

4.6 RDF Semantics

The meaning of the RDF syntax has first been specified 3 years after the development of DAML+OIL. [Hay03] Since then RDF triples were defined as monotone propositions like in model theory. Since DAML+OIL used RDF as a syntax for its inference rules, it had to obey these rules, which wasn't true for all cases.

For example DAML+OIL properties like `oil:hasSlotConstraint` did not follow these semantics [HPsH03], which required a new language for ontology definitions to be specified. This led to the development of a new ontology definition language at the W3C, OWL.

4.7 OWL

OWL, the Web Ontology Language, has been specified and standardized by the W3C one year later and entirely follows the RDF semantics. It also fully covers RDF-Schema, even though RDF-Schema contains several problematic constraints, like global cardinality constraints. [HPsH03] It supersedes DAML+OIL by also implementing a rich set of inferences.

With the rich set of inferences and the problematic constraints in RDF-Schema, which made RDF-Schema non-decidable, OWL itself is also non-decidable. To offer sensible ontology specification languages decidability is a key factor, so that subsets of OWL Full were defined, which retained decidability. The restricted OWL dialects OWL Lite and OWL DL again match known Description Logic styles.

- OWL Lite, $\mathcal{SHIF}(\mathcal{D})$ [HPsH03], ExpTime [wTAiST01]
- OWL DL, $\mathcal{SHOIN}(\mathcal{D})$ [HPsH03], NExpTime [wTAiST01]
- OWL Full, no DL equivalence, not decidable, covers RDF-Schema [HPsH03]

Currently a successor for OWL is in the work, OWL 2, and has been published as a first working draft. [BM08] OWL 2 even more trades expression power for effective reasoning and defines further subsets of OWL Full to be usable with common big datasets of semantically annotated data. On the other hand OWL 2 also specifies some small syntactic features, for which effective reasoning algorithms already exist. The currently developed OWL 2 dialects, with their complexity:

- OWL 2 EL, PSpace [BM08, BBL05]
- OWL 2 QL, LogSpace [BM08]
- OWL 2 RL, PSpace [BM08]

The reasoning still requires exponential time, but the amount of required memory could be significantly reduced.

4.8 Complexity comparison

Figure 3 shows the development of ontologies over time and the complexity of the respective reasoning algorithms.

The figure makes the attempt to reduce reasoning complexity obvious. In the same time span the semantics of RDF and its usage has been unified in the ontology definition languages. OWL and OWL 2 today offer evolved sets of features, while they still maintain the full expressiveness of RDF-Schema, even this has to be traded for more complexity in reasoning.

5 Conclusion

Not only the development of ontologies amount to OWL, but also the tools and specified languages focus more and more on RDF and OWL. The different languages like SPARQL and RIF are designed to meet their requirements. Frameworks like GRDDL make it possible to integrate technologies developed elsewhere, like Microformats.

Until now nobody knows if future web standards like HTML 5 will incorporate semantic markup like RDFa, but with CSS 3 and its capabilities to assign layout to elements based on their name space, it will be possible to use RDF directly in XHTML. XML based languages like WSDL or BPEL already offer full support for RDF annotations, without client problems.

From the current development of related standards it looks like RDF and OWL will be the technologies for annotating semantics on data of the near

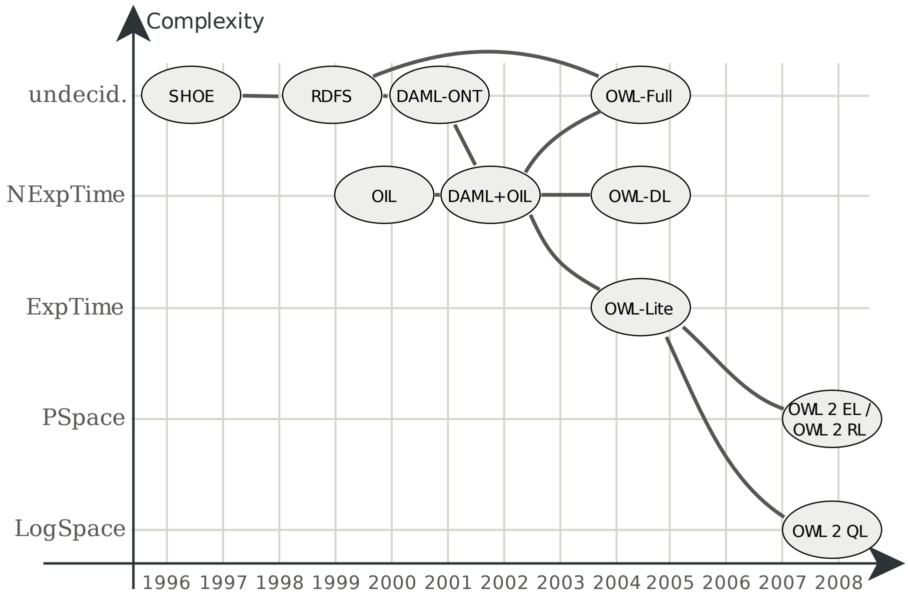

Figure 3: History and complexity of ontologies

future. Externally developed technologies like the Microformats seem to be a good starting point, and can still be integrated later. The ontology definition languages itself all seem to be superseded by OWL, because of their respective problems.

References

[AB06] Ben Adida and Mark Birbeck, *Rdf/a primer 1.0 - embedding rdf in xhtml*, `http://www.w3.org/TR/2006/WD-xhtml-rdfa-primer-20060310/`, March 2006.

[BBL05] Franz Baader, Sebastian Brandt, and Carsten Lutz, *Pushing the EL envelope*, IJCAI (Leslie Pack Kaelbling and Alessandro Saffiotti, eds.), Professional Book Center, 2005, pp. 364–369.

[BCM+03] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider (eds.), *The description logic handbook: Theory, implementation and applications*, Cambridge University Press, 2003.

[Bec00] Dave Beckett, *Redland rdf library - news*, `http://librdf.org/NEWS.html#D2000-06-21-V0.0.0`, June 2000.

[BG99] Dan Brickley and R.V. Guha, *Resource description framework rdf) schema specification*, `http://www.w3.org/TR/1999/PR-rdf-schema-19990303/`, March 1999.

[BG04] ———, *Rdf vocabulary description language 1.0: Rdf schema*, http://www.w3.org/TR/rdf-schema/, February 2004.

[BL89] Tim Berners-Lee, *Information management: A proposal*, http://www.w3.org/History/1989/proposal.html, March 1989.

[BL95] ———, *Rfc 1866 - hypertext markup language - 2.0*, http://www.ietf.org/rfc/rfc1866.txt, November 1995.

[BL96] ———, *Shoe 0.9 - proposed specification*, http://www.cs.umd.edu/projects/plus/SHOE/spec0.9.html, March 1996.

[BLvHH+00] Tim Berners-Lee, Frank van Harmelen, Ian Horrocks, Dan Brickley, Mike Dean, Stefan Decker, Richard Fikes, Pat Hayes, Jeff Heflin, Drew McDermott, and Ralph R. Swick, *Daml-ont initial release*, http://www.daml.org/2000/10/daml-ont.html, October 2000.

[BM04] Dave Beckett and Brian McBride, *Rdf/xml syntax specification (revised)*, http://www.w3.org/TR/rdf-syntax-grammar/, February 2004.

[BM08] et al. Boris Motik, *Owl 2 web ontology language - profiles*, http://www.w3.org/TR/2008/WD-owl2-profiles-20081202/#Introduction, December 2008.

[Bol08] Harold Boley, *The rule markup initiative*, http://www.ruleml.org/#Steps, September 2008.

[Dav06] Ian Davis, *Grddl primer*, http://www.w3.org/TR/2006/WD-grddl-primer-20061002/, September 2006.

[DRJ99] Arnaud Le Hors Dave Raggett and Ian Jacobs, *Html 4.01 specification*, http://www.w3.org/TR/html401/, December 1999.

[DRW98] Murray Altheim Dave Raggett, Frank Boumphrey and Ted Wugofski, *Reformulating html in xml*, http://www.w3.org/TR/1998/WD-html-in-xml-19981205/, December 1998.

[ECW01] Greg Meredith Erik Christensen, Francisco Curbera and Sanjiva Weerawarana, *Web services description language (wsdl) 1.1*, http://www.w3.org/TR/wsdl, March 2001.

[Haw05] Sandro Hawke, *Rule interchange format - working group charter*, http://www.w3.org/2005/rules/wg/charter.html, November 2005.

[Hay03] Patrick Hayes, *Rdf semantics*, http://www.w3.org/TR/2003/WD-rdf-mt-20030123/, January 2003.

[Hef00] Jeff Heflin, *Personal ontology (draft)*, http://www.cs.umd.edu/projects/plus/SHOE/onts/personal1.0.html, October 2000.

[Her09] Ivan Herman, *W3c semantic web activity*, http://www.w3.org/2001/sw/, Februrary 2009.

[HPSB+04] Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, and Mike Dean, *Swrl: A semantic web rule language - combining owl and ruleml*, `http://www.w3.org/Submission/SWRL/`, May 2004.

[HPsH03] Ian Horrocks, Peter F. Patel-schneider, and Frank Van Harmelen, *From shiq and rdf to owl: The making of a web ontology language*, Journal of Web Semantics **1** (2003), 2003.

[HvHBL+00] Ian Horrocks, Frank van Harmelen, Tim Berners-Lee, Dan Brickley, Dan Connolly, Mike Dean, Stefan Decker, Dieter Fensel, Richard Fikes, Pat Hayes, Jeff Heflin, Jim Hendler, Ora Lassila, Deb McGuinness, Peter Patel-Schneider, and Lynn Andrea Stein, *Daml+oil (december 2000)*, `http://www.daml.org/2000/12/daml+oil-index`, December 2000.

[MS98] Ashok Malhotra and Neel Sundaresan, *Rdf query specification*, `http://www.w3.org/TandS/QL/QL98/pp/rdfquery.html`, December 1998.

[MvH04] Deborah L. McGuinness and Frank van Harmelen, *Owl web ontology language - overview*, `http://www.w3.org/TR/owl-features/`, February 2004.

[Oas02] Oasis, *Oasis web services business process execution language tc*, `http://www.oasis-open.org/committees/wsbpel/charter.php`, August 2002.

[PS08] Eric Prud'hommeaux and Andy Seaborne, *Sparql query language for rdf*, `http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/`, January 2008.

[RKM06] Jinghai Rao, Peep Küngas, and Mihhail Matskin, *Composition of semantic web services using linear logic theorem proving*, Inf. Syst **31** (2006), no. 4-5, 340–360.

[Tan06] Tantek, *microformats history*, `http://microformats.org/wiki/index.php?title=history-of-microformats&oldid=6415`, May 2006.

[TB96] C. M. Sperberg-McQueen Tim Bray, *Extensible markup language (xml)*, `http://www.w3.org/TR/WD-xml-961114.html`, November 1996.

[WJ00] Christopher A. Welty and Jessica Jenkins, *Untangle: A new ontology for card catalog systems*, Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, AAAI Press / The MIT Press, 2000, pp. 1137–1138.

[wTAiST01] Rheinisch westfalischen Technischen, Hochschule Aachen, and Diplom informatiker Stephan Tobies, *Complexity results and practical algorithms for logics in knowledge representation*, June 15 2001.