Semantic Web Framework for Development of Very Large Ontologies

Sergey Yablonsky

Abstract—**This paper deals with the development of the Semantic Web framework for very large ontologies. The Semantic Web is often associated with specific XML-based standards for semantics, such as RDF and OWL. Application of lexical ontologies such as WordNet and others for different tasks on the Semantic Web requires their representation in RDF and/or OWL formats with possibility of the different ontology mappings, semantic workflows, services and other semantic technologies.**

Index Terms—**Semantic Web, OWL, RDF, Resource Description Framework.**

I. Introduction

THE Semantic Web, a Web with the meaning, is often associated with specific XML-based standards for semantics, such as RDF[1] and OWL. If HTML and the Web made all the online documents look like one huge book, RDF, schema, and inference languages will make all the data in the world look like one huge database [1]. The Semantic Web Layer Cake (Fig.1) shows that there are different layers in the Semantic Web and that they do different things. Some of the layers can take different forms. Each of the layers is less general than the layers below.

RDF (Resource Description Framework) is a markup language for describing information and resources on the web. RDF represents data as a set of statements consisting of a 'subject', a 'predicate', and an 'object'. Each statement is also known as a 'triple' or a 'relationship'. The Subject and the Predicate are named resources. A resource is represented by a URI. The Object can be a literal or another resource, see Table I.

TABLE I
Example of RDF Data

(Subject)	(Predicate)	(Object)
<SergeyYablonsky>	<name>	"Serge Yablonsky".
<SergeyYablonsky>	<email>	"serge_yablonsky@hotmail.com".
<SergeyYablonsky>	<PhDAdviser>	<AndreySukhonogov>.
<AndreySukhonogov>	<email>	<ASukhonogov@rambler.ru>.

Putting information into RDF files, makes it possible for computer programs ("web spiders") to search, discover, pick up, collect, analyze and process information from the web. The Semantic Web uses RDF to describe web resources.

Nowadays there exists a linked set of different Semantic Web resources as it is shown in Fig.2. In Fig.3 the Linking Open Data (LOD) Constellation is shown.

The objective of the Linking Open Data (LOD) community is to extend the Web with data commons by publishing various open datasets as RDF on the Web and by setting RDF links between data items from different data sources. All of the sources on these LOD diagrams are open data.

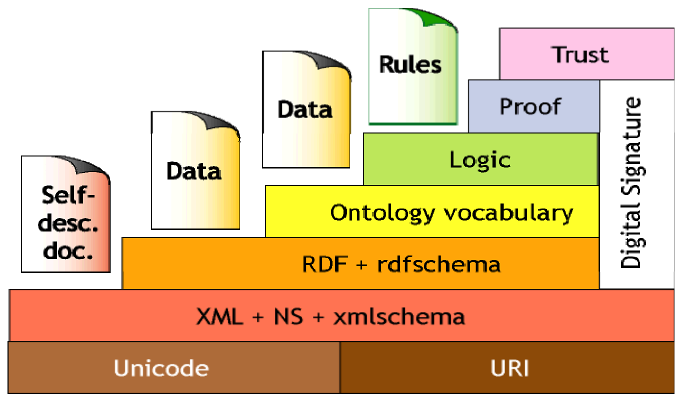

Fig. 1. The Semantic Web Layer Cake (http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html).

The Linking Open Data project is a community-led effort to create openly accessible, and interlinked, RDF Data on the Web. The data in question takes the form of RDF Data Sets drawn from a broad collection of data sources. There is a focus on the Linked Data style of publishing RDF on the Web. The project is one of several sponsored by the W3C's Semantic Web Education & Outreach Interest Group (SWEO).

OWL stands for Web Ontology Language. Web Ontology Language is designed to be used by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than that supported by XML and RDF by providing additional ology vocabulary along with a formal semantics. OWL is built on top of RDF. OWL has three increasingly-expressive sublanguages: OWL Lite (hierarchy with simple constraints), OWL DL (maximum expressiveness, computationally complete, compatible with Description Logics), and OWL Full (very expressive, no computation guarantees, RDF).

Among the most important Web resources are those that provide services. By "service" we mean Web sites that do not merely provide static information but allow one to effect some action or change in the world, such as the sale of a product or

Manuscript received November 15, 2008. Manuscript accepted for publication February 21, 2009.

Sergey Yablonsky is with Graduate School of Management, St. Petersburg State University, Information Technologies in Management Dpt., Volkhovsky Per. 3, St.-Petersburg, 199004, Russia, (e-mail: serge_yablonsky@hotmail.com).

[1] http://www.w3.org/RDF and http://www.w3.org/TR/owl-features

the control of a physical device. One of the key promises of the Semantic Web is that it will provide the necessary infrastructure for enabling services and applications on the Web to automatically aggregate and integrate information into a sum which is greater than the individual parts. So the Semantic Web should enable users to locate, select, employ, compose, and monitor Web-based services automatically.

To make use of a Web service a software agent needs a computer-interpretable description of the service, and the means by which it is accessed. An important goal for Semantic Web markup languages is to establish a framework within which these descriptions are made and shared. Web sites should be able to employ a standard ontology, consisting of a set of basic classes and properties, for declaring and describing services, while the ontology structuring mechanisms of OWL provide an appropriate, Web-compatible representation language framework within which to do this.

The Semantic Web services initiative has developed OWL-S (http://www.w3.org/Submission/OWL-S/) Semantic Markup for Web Services, which enables Web services to be described semantically and their descriptions to be processed and understood by software agents [2].

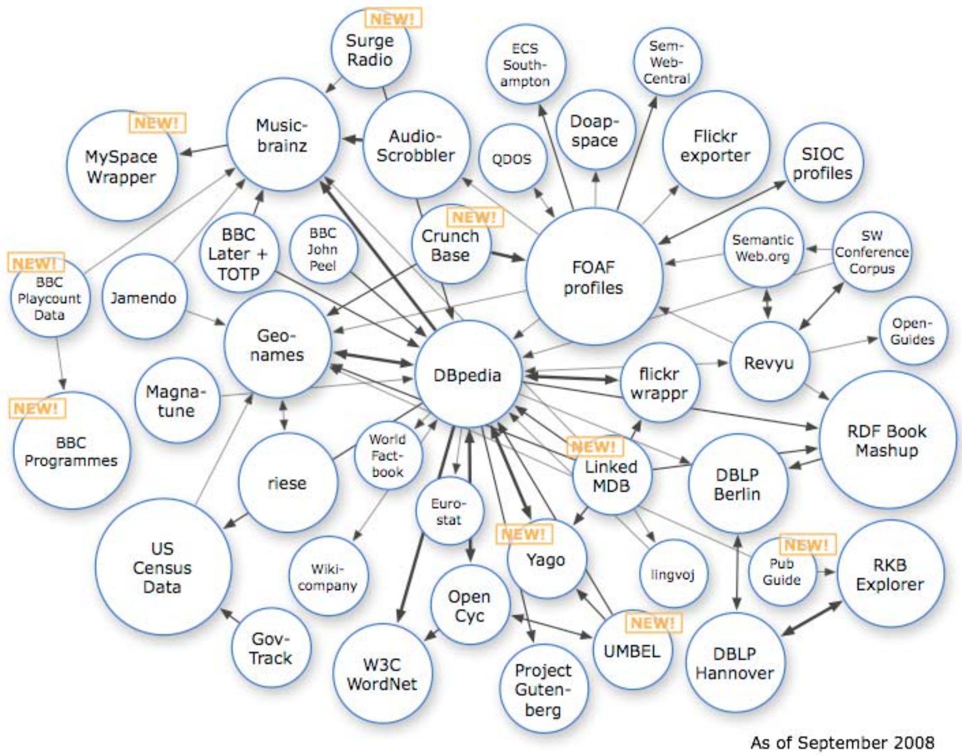

Fig. 2. Semantic Web Layer Cake (http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html).

The Semantic Web should enable greater access not only to content but also to services on the Web. Users and software agents should be able to discover, invoke, compose, and monitor Web resources offering particular services and having particular properties, and should be able to do so with a high degree of automation if desired. Powerful tools should be enabled by service descriptions, across the Web service lifecycle.

Ontologies provide the common vocabulary for the integration of the hundreds of different knowledge bases, meta-data formats and database schemas that are used in the different domains. An ontological framework enables researchers to access a knowledge base, appraise its content, determine if resources are relevant, and to integrate and aggregate the data with in-house resources and data. By linking external ontologies to such conceptual structure, the domain of the linked classes is exploded by leveraging conceptual structure [3].

For example, a new vocabulary for the Semantic Web UMBEL (Upper-level Mapping and Binding Exchange Layer) serves as a coherent reference structure of subject concept classes (http://www.umbel.org). UMBEL subject concepts are conceptually related together using the SKOS/OWL-Full ontologies. UMBEL defines "subject concepts" as a distinct subset of the more broadly understood concept such as used in the SKOS/OWL-Full controlled vocabulary, conceptual graphs, formal concept analysis or the very general concepts common to many upper ontologies. The subject concepts as a special kind of concepts: namely, those that are concrete, subject-related and non-abstract. The UMBEL subject concept structure is, in essence, a content graph of subject nodes related to one another via *skos:broaderTransitive* and *skos:narrowerTransitive* relations.

Computational lexicons (CL) provide machine understandable word knowledge. That is important for turning the WWW into a machine understandable knowledge base ─ Semantic Web. CL supply explicit representation of word meaning with word content accessible to computational agents. Word meaning in CL is linked to word syntax and morphology and has multilingual lexical links.

Computational lexicons are key components of HLT and usually have such typology:

- monolingual vs. multilingual;
- general purpose vs. domain (application) specific;
- content type (morpho-syntactic, semantic, mixed, terminological).

Today such types of CL are designed:

- network based (hierarchy/taxonomy ─ WordNet, heterarchy ─ EuroWordNet);
- frame based (Mikrokosmos, FrameNet);
- hybrid (SIMPLE).

Wordnets are databases of lexical data, including information on hypernyms, synonyms, polysemous terms, relations between terms, and sometimes multilingual equivalents. Wordnets are valuable resources as sources of ontological distinctions. The three core concepts in WordNet are the synset, the word sense and the word. Words are the basic lexical units, while a sense is a specific sense in which a specific word is used. Synsets group word senses with a synonymous meaning, such as {*car, auto, automobile, machine, motorcar*} or {*car, railcar, railway car, railroad car*}. There are four disjoint types of synset, containing exclusively nouns, verbs, adjectives or adverbs. There is one specific type of adjective, namely an adjective satellite.

Furthermore, WordNet defines seventeen relations, of which

- ten between synsets (hyponymy, entailment, similarity, member meronymy, substance meronymy, part meronymy, classification, cause, verb grouping, attribute);
- five between word senses (derivational relatedness, antonymy, see also, participle, pertains to);
- “gloss” (between a synset and a sentence);
- “frame” (between a synset and a verb construction pattern).

This paper additionally attempts to introduce results of an ongoing project of developing of the RDF versions of Russian WordNet and parallel English-Russian WordNet. The usage of the proposed Semantic Web framework is illustrated by developing a multilingual (monolingual Russian and bilingual English-Russian) RDF lexical database of mentioned above wordnets, which are structured along the same lines as the Princeton WordNet for English language.

II.-Framework Architecture

Proposed semantic Web framework is based on the following main parts (Fig. 4):

- RDF/OWL store;
- Tools for information extraction;
- Tools for Ontology Engineering Modeling Process;
- Knowledge mining, SPAROL/SQL search and analysis tools.

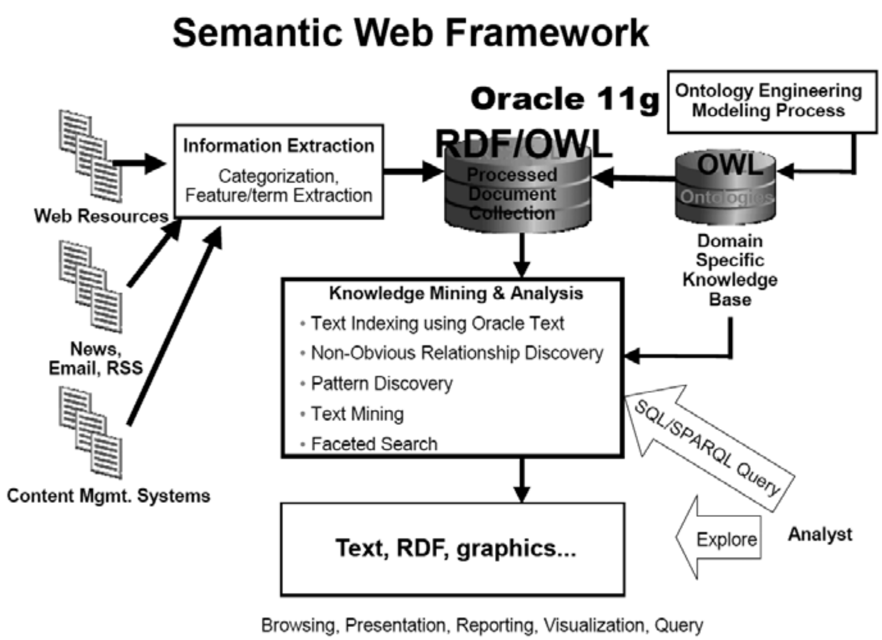

Fig. 4. Framework General Architecture using Oracle 11g.

A. *Oracle 11g RDF/OWL store*

Oracle 11g includes an open, scalable, secure and reliable RDF management platform. Based on a graph data model, RDF triples are persisted, indexed and queried, similar to other object-relational data types. The system also implements subsets of OWL Full.

different versions of WordNet in XML/RDF/OWL and how to define the relationship between them and how to integrate WordNet with sources in other languages. Main class/property and Data types of Russian WordNet OWL representation are shown in Table II. In Table III the correspondence between W3C WordNet and Russian WordNet RDF/OWL porting is listed.

TABLE II
RUSSIAN WORDNET OWL

N	Russian WordNet (OWL) Class/property	Data type
1.	**Synset**	owl:Class
2.	*owl:ObjectProperty***index**	#Synset/**&rdfs;Literal**
3.	*owl:ObjectProperty* **glossaryEntry**	#Synset/**&rdfs;Literal**
4.	*owl:ObjectProperty* **exampleSentences**	#Synset/**&rdfs;Literal**
5.	*owl:TransitiveProperty* **hyponymOf**	#Synset/#Synset
6.	*owl:TransitiveProperty* **hasHyponym**	#Synset/#Synset
7.	*owl:SymmetricProperty* **nearAntonym**	#Synset/#Synset
8.	*owl:SymmetricProperty* **seeAlso**	#WordSense/#WordSense
9.	*owl:ObjectProperty* **relatedForm**	#Synset/#Synset
10.	Noun	owl:Class
11.	Verb	owl:Class
12.	Adjective	owl:Class
13.	Adverb	owl:Class
14.	AdjectiveSatellite	owl:Class
15.	*owl:ObjectProperty* **meronymOf**	#Noun/#Noun
16.	*owl:ObjectProperty* **hasMeronym**	#Noun/#Noun
17.	*owl:ObjectProperty* **memberMeronymOf**	#Noun/#Noun
18.	*owl:ObjectProperty* **hasMemberMeronym**	#Noun/#Noun
19.	*owl:ObjectProperty* **substanceMeronymOf**	#Noun/#Noun
20.	*owl:ObjectProperty* **hasSubstanceMeronym**	#Noun/#Noun
21.	*owl:ObjectProperty* **partMeronymOf**	#Noun/#Noun
22.	*owl:ObjectProperty* **hasPartMeronym**	#Noun/#Noun
23.	*owl:ObjectProperty* **isCausedBy**	#Verb/#Verb
24.	*owl:ObjectProperty* **causes**	#Verb/#Verb
25.	*owl:SymmetricProperty* **sameGroupAs**	#Verb/#Verb
26.	*owl:ObjectProperty* **isDerivedFrom**	#WordSense/#WordSense
27.	*owl:ObjectProperty* **hasDerived**	#WordSense/#WordSense
28.	*owl:TransitiveProperty* **isSubeventOf**	#Verb/#Verb
29.	*owl:TransitiveProperty* **hasSubevent**	#Verb/#Verb
30.	*owl:SymmetricProperty* **similarTo**	#Adjective/#Adjective
31.	*owl:ObjectProperty* **attribute**	#Noun/#Adjective
32.	*owl:ObjectProperty* **valueOf**	#Adjective/#Noun
33.	*owl:ObjectProperty* **domainUsage**	#Synset/#Synset
34.	*owl:ObjectProperty* **domainUsageMember**	#Synset/#Synset
35.	*owl:ObjectProperty* **domainCategory**	#Synset/#Synset
36.	*owl:ObjectProperty* **domainCategoryMember**	#Synset/#Synset
37.	*owl:ObjectProperty* **domainRegion**	#Synset/#Synset
38.	*owl:ObjectProperty* **domainRegionMember**	#Synset/#Synset
39.	**WordSense**	owl:Class
40.	*owl:ObjectProperty* **inSynSet**	#WordSense/#Synset
41.	*owl:ObjectProperty* **containsWordSense**	#Synset/#WordSense
42.	**Word**	owl:Class
43.	*owl:ObjectProperty* **senseOf**	#WordSense/#Word
44.	*owl:ObjectProperty* **hasSense**	#Word/#WordSense
45.	*owl:ObjectProperty* **frequency**	#WordSense/**&xsd;double**
46.	*owl:ObjectProperty* **lemma**	#Word/ **&rdfs;Literal**
47.	*owl:ObjectProperty* **senseKey**	#WordSense/**&rdfs;Literal**
48.	*owl:ObjectProperty* **participleOf**	#WordSense/#WordSense
49.	*owl:ObjectProperty* **hasParticiple**	#WordSense/#WordSense
50.	*owl:SymmetricProperty* **antonym**	#WordSense/#WordSense
51.	**TopOntology**	owl:Class
52.	*owl:ObjectProperty* **hasItem**	#TopOntology/#Synset
53.	*owl:ObjectProperty* **index**	#TopOntology/&rdfs;Literal
54.	*owl:ObjectProperty* **name**	#TopOntology/&rdfs;Literal
55.	*owl:ObjectProperty* **broaderItem**	#TopOntology/#TopOntology
56.	*owl:ObjectProperty* **narrowerItem**	#TopOntology/#TopOntology

In Table III, the set of relations in different WordNet realization are summarized, where S – any synset, N – noun synset, V –verb synset, A – adjective synset, R - adverb synset, WS – any word sense, NS – noun sense, VS – verb sense, AS – adjective sense, RS – adverb sense

For managing WordNet Semantic Web models the Multilingual WordNet Editor [6] was used together with XMLSpy 2008 and Oracle 11g that provides important XML/RDF/OWL support for data modeling and editing of XML/RDF/OWL WordNet models.

IV. Experimental Results and Conclusion

As part of the general testing of the Framework General Architecture using Oracle 11g RDF store, we first re-ran the LUBM 8000 load test (1067 million triples). The result of the bulk–load:

- Time to load staging table: 3 to 12 hrs;
- Time using Bulk-load API: about 33 hrs;
- Storage: data 42 GB, indexes 95 GB, app table 23 GB.

Then we load RDF/OWL versions of WordNet and Russian WordNet. The Semantic Web Framework implementation:

- Stores RDF/OWL data and ontologies;
- Inferences new RDF/OWL triples via native inference;
- Provides Query RDF/OWL data and ontologies and Ontology-Assisted-Query of relational data;
- Conforms to W3C standards for storage, schema and rules.

There are many advantages to storing RDF data as an object type, rather than in flat relational tables. Benefits include making it easier to model and maintain RDF applications, simplifying the integration of RDF data with other enterprise data, reuse of RDF objects; moreover, no mapping is required between client RDF objects and database columns and tables that contain triples.

With the Oracle RDF Data Model triples are parsed and stored in the database as entries in the NDM nodes and links tables. Nodes in the RDF model are uniquely stored and reused when encountered in incoming triples. In user-defined application tables, only references are stored in the SDO_RDF_TRIPLE_S object to point to the triple stored in the central schema. The RDF Data Model also simplifies reification by utilizing an Oracle XML DB DBUri to directly reference the reified triple in the database, and thereby only requires one additional triple to be stored for each reification. Oracle provides an open, persistent, analytic semantic data management platform. Oracle Database Semantic Data Store is a feature of Oracle Spatial 11g Option for Oracle Database 11g Enterprise Edition.

The following Oracle Semantics Technology Benefits can be mentioned:

- Native Inference using W3C standards;
- Native Storage of RDF and OWL;
- Query of semantic data using SQL extensions and SPARQL;
- Innovative Ontology-Assisted Query of relational data;
- Embedded in database technology, stores up to 8 exabytes;
- Versioning and schema support;
- Programming language interfaces like PL/SQL and Java;
- Could use in-house expertise of DBAs and database developers;
- Scalability – Trillions of triples;
- Availability – tens of thousands of users;
- Security – protect sensitive business data;
- Performance – timely load, query & inference;
- Accessibility – to enterprise applications;
- Manageability – leverage IT resources.

References

[1] T. B. Lee, *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by its Inventor*, Harper One, 1st edition, 1999.

[2] S. McIlraith, T. C. Son, and H. Zeng, "Semantic Web Service," *IEEE Intelligent Systems*, 2001, 16(2):46-53.

[3] F. M. Suchanek, G. Kasneci, and G. Weikum, "YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia," in *Proceedings of the 16th international conference on World Wide Web*, 2007.

[4] C. Fellbaum, *WordNet: an Electronic Lexical Database*. MIT Press, Cambridge, MA, 1998.

[5] *WordNet OWL Ontology*. http://www2.unine.ch/imi/page11291_en.html

[6] V. Balkova, A. Suhonogov, and S. Yablonsky, "A. Russian WordNet. From UML-notation to Internet/Intranet Database Implementation," in *Proceedings of the Second International WordNet Conference. GWG-*2004, pp. 31–38, Brno, 2004.