

Comparative Analysis of Open Source Ontology Editors: A Study

Kaushal Giri*, Pratibha Gokhale**

Abstract

Currently computers are changing from single isolated devices to entry points into a worldwide network of information exchange and business transactions. Therefore, support in the exchange of data, information, and knowledge is becoming the key issue in computer technology today. Ontologies provide a shared and common understanding of a domain that can be communicated between people and across application systems. Ontologies will play a major role in supporting information exchange processes in various areas. To build Ontologies, ontology editors play an important role. Ontology editors are software tools that allow the creation and maintenance of ontologies through a graphical user interface. The paper describes comparative analysis of open source ontology editors.

Keywords: Semantic Web; Ontologies; Ontology Editors.

Introduction

Ontologies were developed in Artificial Intelligence to facilitate knowledge sharing and reuse. Since the beginning of the nineties ontologies have become a popular research topic investigated by several Artificial Intelligence research communities, including Knowledge Engineering, natural language processing, Information retrieval and knowledge representation [1]. More recently, the notion of ontology is also becoming widespread in fields such as intelligent information integration, cooperative information systems, information retrieval, electronic commerce, and knowledge management. Currently, ontologies are applied to the World Wide Web creating what is called the semantic web. Originally, the web grew mainly around the language HTML, that provide a standard for structuring documents that was translated by browsers in a canonical way to render documents [2]. On the one hand, it was the

simplicity of HTML that enabled the fast growth of the WWW. On the other hand, its simplicity seriously hampered more advanced web application in many domains and for many tasks. Still, XML is basically a defined way to provide a serialized syntax for tree structures. Therefore, it is just an important first step in the direction of a semantic web, where application programs have direct access to the semantics of an important additional step has been taken by RDF which defines a syntactical convention and a simple data model for representing machine-processable semantics of data. Therefore Ontologies will play a major role in supporting information exchange processes in various areas.

Semantic Web

Semantic Web is the new-generation Web that tries to represent information such that it can be used by machines not just for display purposes, but for automation, integration, and reuse across applications [3]. The Semantic Web is a Web with a meaning. It describes things in a way that computers can understand. It is an extension to the normal Web and is not about links -relationships between things and its properties. It is the efficient way of representing data on the World Wide Web. Tim Berner Lee (Inventor of Web, HTTP, & HTML) says that

Author's Affiliation: *Manager, Export-Import Bank of India (EXIM Bank), Mumbai, Maharashtra, India. **Former Head & I/C University Librarian, Department of Library & Information Science, University of Mumbai, Mumbai, Maharashtra, India.

Reprint's Request: Kaushal Giri, Manager, Export-Import Bank of India (EXIM Bank), Mumbai, Maharashtra, India.
E-mail: kaushal.giri@eximbankindia.in

Received on 22.02.2017, Accepted on 04.03.2017

Semantic web will be the next generation of Current Web and the next IT revolution [4].

Ontology

The term ontology can be defined in many different ways. Ontology is an explicit specification of a set of objects, concepts, and other entities that are presumed to exist in some area of interest and the relationships that hold them. Ontology defines a common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them. It enables the Web for software components can be ideally supported through the use of Semantic Web technologies. This helps in understanding the concepts of the domain as well as helps the machine to interpret the definitions of concepts in the domains and also the relations between them. Ontologies can be broadly divided into two main types: *lightweight* and *heavyweight*. Lightweight Ontologies involve taxonomy (or class hierarchy) that contains classes, subclasses, attributes and values. Heavy weight Ontologies model domains in a deeper way and include axioms and constraints [5].

Why to Develop Ontology?

Some of the Reasons are

- To share common understanding of the structure of information among people or software agents.
- To enable reuse of domain knowledge.
- To make domain assumptions explicit.
- To separate domain knowledge from the operational knowledge.
- To analyze domain knowledge.

Ontology Editors

An ontology editor provides a graphical user interface (GUI) abstraction over various operations involved in ontology creation. A user can create a class and its sub-classes; add properties and relationships; create instances; and specify rules etc. The generated ontology can then be exported into different ontology languages (SHOE, RDF, DAML and OWL etc.) without letting the user to worry about the intricate details of ontology languages. Most of the research proposals adopt their own techniques and criteria for evaluation of their advocated editors. This paper qualitatively evaluated the ontology editors based on certain features. Many ontology editors

could be found on Internet. Some of them (like: Apollo, OntoStudio, Protégé, Swoop and Top Braid Composer Free Edition, Comparison could be done by using different criterions: generality, expressiveness, complexity, documentation, scalability etc. In this paper the main criterion is easiness to use and spreading of editors [6].

Ontology Editor's Development Tools

This paper gives a broad overview of some available editors and environments that can be used for the building of ontologies. Comparison could be done by considering different properties of editors. The following are the characteristics [7].

- *General description of the tools:* This includes information about developers and availability.
- *Software architecture and tool evolution:* This includes information about the tool architecture (standalone, client/server, n-tier application).
- *Interoperability with other ontology development tools and languages:* This includes information about the interoperability of the tool. Tool's interoperability with other ontology tools can be recognized by functionalities like (merging, annotation, storage, inference, etc.), in addition to translations to and from ontology languages.
- *Usability:* The existence of the graphical editors for the creation of concept taxonomies and relations.

The following editors have been selected for comparison:

- Apollo;
- OntoStudio;
- Protégé;
- Swoop and
- TopBraid Composer (Free Edition).

All these tools are widespread in the ontology design and development sector and are accepted by relatively large semantic web communities. These tools also provide the minimum necessary functionality supporting the ontology development process. The ontology editors are tools that allow users to visually manipulate, inspect, browse, code ontologies, support the ontology development and maintenance task.

Apollo

Apollo is a user friendly knowledge modeling application. Apollo allows a user to model ontology

with basic primitives, such as classes, instances, functions, relations and so on. The internal model is a frame system based on the OKBC protocol. The knowledge base of Apollo consists of a hierarchical organization of ontologies. Each ontology is the default ontology, which includes all primitive classes. Each class can create a number of instances, and an instance inherits all slots of the class. Each slot consists of a set of facets. (Apollo) Apollo does not support graph view, web, information extraction and multi-user capabilities or collaborative processing but it features strong (files only) and import/export format (I/O plug-in architecture-export plug-ins to CLOS and OCML). Apollo is implemented in Java and it is available for a download from <http://apollo.open.ac.uk/index.html>

Onto Studio

OntoStudio is the most widespread modeling environment for creating and maintaining ontologies. It stands out due to its comprehensive functions in intuitive ontology modeling. OntoStudio is also able to import many structures, schemas and models. OntoStudio is based on IBM Eclipse framework. It can be downloaded for one month free from <http://www.semafora-systems.com/en/products/ontostudio/>. It is an Ontology Engineering Environment supporting the development and maintenance of ontologies by using graphical means. It is based on client/server architecture, where ontologies are managed in a central server and various clients can access and modify these ontologies. It supports multilingual development, and the knowledge model is related to frame-based languages. OntoStudio is based on an open plug-in structure. The internal representation data model can be exported to DAML+OIL, F-Logic, RDF(S), and OXML. Additionally, ontologies can be exported to relational databases via JDBC.

Protégé

Protégé is a free, open-source platform that provides a growing user community with a suite of tools to construct domain models and knowledge-base applications with ontologies. It implements a rich set of knowledge-modeling structures and action that support the creation, visualization and manipulation of ontologies in various representation formats. It can be customized to provide domain-friendly support for creating knowledge models and entering data. Also, it can be extended by a plug-in architecture and Java based application programming interface (API) for building

knowledge-base tools and applications. The significant advantage of Protégé is its scalability and extensibility. Protégé (Escórcio & Cardoso, 2007) allows to build and to process large ontologies in an efficient manner. Protégé is freely available for download from <http://protege.stanford.edu/>

Swoop

Swoop is an open-source, Web based OWL ontology editor and browser. Swoop (Kapoor & Sharma, 2010) contains OWL validation and offers various OWL presentation syntax views. It has reasoning (RDFS-like and Pallet) support (OWL Inference Engine), and provides a Multiple Ontology environment, by which entities and relationships across various ontologies can be compared, edited and merged seamlessly. Navigation could be simple and easy due to the hyperlinked capabilities in the interface of Swoop. Swoop does not follow a methodology for ontology construction. The users can reuse external ontological data either by simply linking to the external entity, or by importing the entire external ontology. It is not possible to do partial imports of OWL, but it is possible to search concepts across multiple ontologies. Swoop uses ontology search algorithms that combine keywords with DL-based constructs to find related concepts in existing ontologies.

TopBraid Composer Free Edition

TopBraid Composer comes in three editions: Free Edition (FE) is an introductory version with only a core set of features. Standard Edition (SE) includes all features of FE plus graphical viewers, import facilities, advanced refactoring support and much more. Maestro Edition (ME) includes all features of SE plus support for TopBraid Live, EVN and Ensemble as well as SPARQL Motion and many other power user features.

Top Braid Composer (FE) a component of Top Braid Suite is a professional development tool for semantic models (ontologies). It is based on the Eclipse platform and the Jena API. It is a complete editor for RDF(S) and OWL models, as well as a platform for other RDF-based components and services. TopBraid Composer (FE) can (Knublauch, 2009) loads and save any OWL2 file in formats such as RDF/XML or Turtle. TopBraid Composer (FE) (W3C, 2001) supports various reasoning and consistency checking mechanisms. It also supports the SPARQL inference Notation (SPIN). TopBraid Composer (FE) can download and evaluate the full version for a 30 days evaluation period.

Comparison of Ontology Editors (tools)

The comments concerning this section are based on tools that have been described above. The tools are

specified in alphabetical order: Apollo, OntoStudio, Protégé, Swoop and TopBraid Composer (FE).

- *General Description of the Tools*

Table 1: It includes information about developers and availability.

Feature	Apollo	OntoEdit	Protégé	Swoop	TopBraid Compser
Developers	KMI (Open University)	Ontoprise	SMI (Stanford University)	MND (University of Maryland)	TopQuadrant
Availability	Open Source	Software license (One Month free Trail period)	Open Source	Open Source	Software licence (One Month free Trail period)

- *Software Architecture and Tool Evolution*

Table 2: It includes information about the necessary platforms to use the tool

Feature	Apollo	OntoEdit	Protégé	Swoop	TopBraid Composer
Semantic Web architecture	Standalone	Eclipse client / server	Standalone and client- server	Web- based and client server	Standalone Eclipse plug-in
Extensibility	Plug-ins	Plug-ins	Plug-ins	Plug-ins	Plug-ins
Backup management	No	No	No	No	Yes
Ontology Storage	Files	DBMS	Files and DBMD	As HTML models	DBMS

- *Interoperability*

Table 3: It includes information about the tools interoperability with other ontology development tools.

Feature	Apollo	OntoEdit	Protégé	Swoop	TopBraid Composer
With Other Ontology Tool	No	OntoAnnotate, OntoBroker, OntoMat	PROMPT, OKBC, Jess, FaCT, and Jena	No	Seasame, Jena and AllegroGraph
Imports from Languages	Apollo Meta language	XML, OWL, Excel, RDF (S), UML 3.0, Databases Schemas (Oracle, MS-SQL)	XML, OWL, HTML, RDF, UML, XML, backend text file, RDF file, Excel Bio Portal and Data master	OWL, XML, RDF and text formats	RDFa, WOL, XML, RDF, UML, RDB and D2Rq, SPIN, Oracle Database
Export from languages	OCML and CLOS	OWL, RDF, RIF, SPARQL, F- logic and excel	XML, OWL, HTML, RDF, UML, XML, Java, F- logic, SWRL	RDF, OIL and DAML	HTML, UML, XSD, Excel, RDB, RDF file.

- *Usability*

Table 4: It is related to Graphical editors, collaborative working and the provision of reusable ontology libraries.

Feature	Apollo	OntoEdit	Protégé	Swoop	TopBraid Composer
Graphical Taxonomy	No	Yes	Yes	Yes	Yes
Graphical Prunes (Views)	No	Yes	Yes	No	Yes
Collaborative working	No	Yes	Yes	Yes	Yes
Ontology libraries	Yes	Yes	Yes	No	Yes

To sum up, the Apollo, Protégé and Swoop tools are open source ontology editors and OntoStudio and TopBraid composer (FE) they are available for free trial version but software license is must. The tools: Protégé, TopBraid Composer (FE) and OntoStudio use databases for storing ontologies. The

same applies to backup management functionality, which is just provided by TopBraid Composer (FE). Protégé and OntoStudio are more graphical ontology tools. The Swoop is Web-based application. OntoStudio gives support to the Onto Knowledge methodology. TopBraid Composer (FE) uses the

Exception Handling. Some of the tools only support the joint edition of the functions of browsing. Protégé, TopBraid Composer (FE), Swoop and OntoStudio editors provide documentation ontology, ontology import / export to different formats, graphical view of ontologies, ontology libraries and attached inference engines and Apollo supports Apollo meta language. It is quite clear that Ontology development is mainly an ad-hoc approach. Among several viable alternatives, a user needs to find which one would work better for the projected task and which one easily and effectively can be maintained and expressed. The foundation of ontology is logic, but in same time it is a model of reality and the concepts in the ontology must reflect this reality. The above study has described a tool-assisted method for building the basis for ontologies adopted from domain analysis. After analysing the researcher found protégé as the best Ontology editor.

Conclusion

Ontology represents information in a manner so that this information can also be used by machines not only for displaying, but also for automating, integrating, and reusing the same information across various applications. There is lot of work going on to adapt the technology and build ontologies. In near future the above ontology editors will come out with new and existing features which will help the

developers/researchers to build ontologies in a better manner.

References

1. Fensel, D. *Ontologies: Silver bullet for knowledge management and electronic commerce*. Berlin, Germany: Springer-Verlag. 2001.
2. Berners-Lee T, Hendler J and Lassila O, *The semantic web*, *Scientific American*, 2002;24(3):35-43.
3. Grigoris Antoniou and Frank Von Hormelen, "A Semantic Web primer", The MIT Press Cambridge, Massachusetts London, England.
4. Christoph Bussler, Dieter Fensel, Alexander Maedche, "A Conceptual Architecture for Semantic Web Enabled Web Services", NSF-EU Workshop on Database and Information Systems Research for Semantic Web and Enterprises, April 3 - 5, 2002 Amicalola Falls and State Park, Georgia.
5. Sergey Sosnovsky, Darina Dicheva, "Ontological technologies for user modeling", *Int. J. Metadata, Semantics and Ontologies*, 2010; 5(1).
6. Karim, S., & Tjoa, A.M. *Towards the Use of Ontologies for Improving User Interaction for People with Special Needs*. *Computers Helping People with Special Needs*. 2006;40(61):77-84. Berlin: Springer-Heidelberg.
7. Kashyap, V., Bussler, C., & Matthew, M. *The Semantic Web: Semantics for Data and Services on the Web (Data-Centric Systems and Applications)*. Berlin: Springer-Verlag. 2008.