Evaluation of Ontology Creation Tools

Pallavi Grover, Sonal Chawla

***Abstract*—Representation of distributed information, with a well defined meaning understandable for different parties, is the major challenge of Semantic Web. Several solutions have been built up. Use of Ontologies is one of the solutions to challenges faced by semantic web. This paper highlights importance of ontologies. This paper has three fold objectives. Firstly the paper throws light on how a semantic web based tool helps producing information using ontologies. Secondly, paper highlights the importance of ontology. Lastly a comparison of various tools for ontology development has been presented on various parameters**

***Index Terms*—Ontology Ontology Tools, RDF, Semantic Web**

I. INTRODUCTION

Web holds s a massive amount of information which is accessed by different users every day using web browsers. A user inserts his query in search engine which produces multiple results which can be both relevant and irrelevant to the user. Although web can give user all the information that is needed but often produces irrelevant information. This is because of the voluminous amount of information present on the internet which requires human to understand and process it. This leads to increase in searching time and makes user responsible for organizing all the information manually. This becomes a huge obstacle for users who want to quickly find the desired information [1]

Semantic web comes as a solution to these problems as it provides information to user based on ontology. It converts the query information to machine process able form using RDF based libraries (ontologies) and produces refined results. This not only helps in reducing time and effort put in by the user but also allows precise in depth searching.

II. SEMANTIC WEB

Semantic web can be thought of as extension on World Wide Web that can be defined as a web of data that can be processed directly or indirectly by machines.[2]Information on the internet exists as distributed form and is diversified in nature i.e. No logical relation/link exists between two entities. So when a search engine processes a certain query it often gives irrelevant results as each word in query is treated as an individual component having no relation with another component which is part of the same query. Semantics web extends this diversified network of hyperlinked, human readable web pages by inserting machine readable information (or metadata) and their relationship. This, in turn, lets the automated agents to access the web more intelligently reducing user's task of organizing information manually.

Manuscript Received on May 2014.

Pallavi Grover, Department of Computer Science & Applications, Panjab University, Chandigarh, India.

Dr. Sonal Chawla, Department of Computer Science & Applications, Panjab University, Chandigarh, India.

The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries [2] Metadata of webpage is organized using RDF(Resource Description Framework).RDF allows us to frame relation among entities. These relations are then processed using different ontologies, which serve as knowledge base.

III. RESOURCE DESCRIPTION FRAMEWORK

Resource Description Framework (RDF) is a general method that decomposed knowledge into small pieces, using some rules about the semantics of those pieces. The idea is to eventually have a method so simple that it can express any fact in a structured manner that computer applications can do useful things with knowledge expressed in RDF.

Structure of RDF:

RDF information is presented in form of triplet i.e. it has three parts.

A subject: Start node of the edge.

A predicate: Describes a relation between the subject and object as a verb or type of edge or property.

An object: End node of the edge.

Triples are the core element of RDF. When RDF applications use RDF in any of the available formats, they see the triples. It can also be thought of as a labeled directed graph where each edge in the graph can be thought of as a fact or a relation between two things.

Everything at all mentioned in RDF means something. It may be a reference to something in the world, like a person or movie, or it may be an abstract concept, like the state of being friends with someone else. And by putting three such entities together, the RDF standard says how to arrive at a fact.

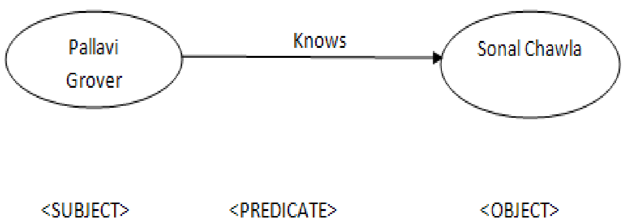

Fig.1 Representing structure of RDF

IV. ONTOLOGY

The term ontology has been widely used in recent years in the field of Artificial Intelligence, Computer and Information science especially in domains such as, cooperative information systems, intelligent information integration, information retrieval and extraction, knowledge representation, and database management systems.

Many different definitions of the term are proposed. One of the most widely quoted and well-known definition of ontology is Gruber's [8]: ontology is an explicit specification of a conceptualization [9].
Ontologies are the structural frameworks for organizing information .They find their application in various fields like Artificial Intelligence, Semantic Web, Biomedical Informatics, Library Science etc as a form of knowledge representation about the world or some part of it. To model the knowledge domain Ontology is designed for the system to incorporate the domain information in the form of instances and data type values, classes and object properties.
Ontologies are different from traditional keyword-based search engines. In that ontology is metadata, capable of providing the search engine with the functionality semantic matching. Ontologies are able to search more efficiently than traditional methods. Typically, ontology consists of hierarchical descriptions of important concepts in a domain and the descriptions of the properties of each concept.
Ontologies allow for machine-understandable semantics of data, and facilitate the search, exchange, and integration of knowledge. An ontology is always constructed with a specific purpose or certain task in mind because of which the content and structure of ontology is restricted. Several tools have been developed for implementing metadata of ontologies Protégé, SWOOP, OilEd, Apollo, RDFedt etc.

V. TOOLS FOR DEVELOPING ONTOLOGIES

Ontology development is by and large domain-oriented process. This complex process can be well benefited from tool support. An ontology can be used to support various types of knowledge management operations which includes its retrieval, storage, and sharing (Pundit & Bishr, 1999).XML is not suited to describe machine-understandable documents and interrelationships of resources in ontology (Gunther, 1998). Therefore, the W3C has recommended the use of RDF, RDFS, DAML+OIL and OWL. Since then many tools have been developed for implementing the metadata of ontologies by using these languages. Following are a few tools which have been used to develop ontologies

A. Apollo

Developed at the Knowledge Media Institute of Open University, United Kingdom [4].Apollo allows a user to model ontology with basic primitives such as classes, instances, functions, relations etc. The knowledge base of Apollo consists of hierarchically organized Ontologies. Ontologies can be inherited from other Ontologies and can be used as if they were their own Ontologies. Every ontology has a 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 does not support collaborative working but can be extended with plug-ins.

B. SWOOP

SWOOP was developed MND University of Maryland .It is a web based OWL ontology editor and browser. Since it follows OWL it contains OWL validations and allows several OWL presentation syntax views. It has reasoning support and provides Multiple Ontology environment i.e., it allows flawless ontology comparison, edit and merger of entities and relationships across various ontologies. Comparison is possible against their Description Logic-Based definitions, properties and instances related to them. Since the interface of SWOOP contains hyperlinked capabilities navigation is simple and easy navigation. SWOOP does not follow a methodology for ontology construction. Therefore it allows user to reuse external ontological data. This can be done either by linking to the external entity or by importing the entire external ontology. SWOOP does not allow partial imports of OWL but concept search across multiple ontologies is possible. Swoop uses ontology search algorithms that combine keywords with DLbased constructs to find related concepts in existing ontologies. This search is made along all the ontologies stored in the Swoop knowledge base [6].

C. Protege

Protégé (Noy, Sintek, Decker, Crubezy, Fergerson, &Musen, 2001) was developed by Stanford Medical Informatics [4] Protégé is based on Java and supports modeling ontologies via a web client or a desktop client. Protégé ontologies can be developed in a variety of formats including OWL, RDF(S), and XML Schema. It is a free, open source ontology editor and knowledge-base framework. Since it provides a plug n play environment it acts as flexible base for rapid prototyping and application development. Since knowledge-based system development is done by a team which includes both developers and domain experts who may be less familiar with computer software. Protégé works with an assumption that knowledge-based systems are usually very expensive to build and maintain. Protégé guides developers and domain experts through the process of system development.

D. RDFedt

RDFedt allows a user to build complex and structured RDF and RSS (RDF site summary) documents. RDFedt is developed by Jan Winkler of Germany [4]. RDFedt is a textual language editor. It is not a Java program, is not platform independent, and works only on Windows.
It provides an overview of complex data structures with element trees. Additionally, it allows a user to test data and to give comments and error messages with the help of some functions. RDFedt supports RDF, RDFS, and Dublin core elements.
Provision of modules like aggregation, notation, content, cut, organization, change of page, threading etc is given in RSS 1.0. RSS 0.91 supports declaration and levels of styles in XML, sets of imported elements, and the automatic generation of an RDF-based linked list from an HTML document.

E. OilEd

OilEd or OIL editor was primarily intended to demonstrate the use of DAML+OIL. It was developed by information Management Group of the Computer Science Department at the University of Manchester, United Kingdom[4].OilEd allows user to create and edit OIL ontologies. It doesn't support a full ontology-development environment. Also it doesn't support several activities like creation of large scale ontologies, versioning,

augmentation and the migration & integration of ontologies which are involved in ontology construction.
OilEd lacks extensibility provision but use of arbitrary class expressions, primitive and defined classes and concrete-type expressions is provided.

F. OntoLingua

OntoLingua is an ontology library and server that can be accessed using traditional Web Browser. It was developed by Knowledge Systems Lab in Stanford University in 1997[4].It also supports World Wide Web interface and translation into different formats. It provides a user-distributed collaborative environment, a suite of ontology-authoring tools, and a library of modular, reusable ontologies. Authorization can be provided by assembling and extending the ontologies obtained from the library and tools. Taxonomy reorganization and name conflict resolution in knowledge base is done using Chimaera. Write-only locking and user access-level assignment allows multiple users to use OntoLingua.

VI. COMPARATIVE ANALYSIS OF ONTOLOGY TOOLS

Various parameters have been listed, based on which a comparative study has been done amongst all six ontology tools that have been discussed before.
General description of the tools (Table I) includes information about developers and availability. We can see form (table 1) that: Apollo, Swoop, Protégé, RDFedt, OilEd and OntoLingua are open source or free available editors.

Table I

GENERAL DESCRIPTION OF THE TOOLS		
Tools	Developer	Availability
Apollo	KMI Open University	Open Source
Swoop Protégé	MND University of Maryland SMI Stanford University	Open Source Free/Open Source
RDFedt	Jan Winler	Free
OilEd	University of Manchester, United Kingdom	Free
OntoLingua	Knowledge Systems Lab, Stanford University	Free

Software architecture and tool evolution (Table II) details information about the necessary platforms required for tool use. The following information has been included: default architecture (standalone, client/server, n-tier application), extensibility, storage of the ontologies (databases, ASCII files, etc.) and extensibility. All of these tools are moving towards Java platforms and allow ontology storage as files except Swoop which is web based. Protégé and Swoop have client/server architecture where as Apollo supports standalone architecture.

Table II

SOFTWARE ARCHITECTURE AND TOOL EVOLUTION			
Tools	Semantic web Architecture	Extensibility	Ontology Storage
Apollo	Standalone	Plug-ins	Files
Swoop Protégé	Web based and Client server Standalone and client server	Yes via Plug-ins Plug-ins	As HTML Models Files and DBMS (JDBC)
RDFedt		Via Plug-ins	Files
OilEd		No	File
OntoLingua		Via Plug-ins	Files

Interoperability (Table III) gives information about the tools interoperability with other ontology development tools and languages, translations to and from some ontology languages. It is another important feature in the integration of ontologies in applications. Most of these tools support import and export to and from many languages in a variety of formats. Swoop supports RDF (S), OIL and DAML for import and OWL, XML, RDF and text formats for export. Apollo supports Apollo met language and OCML and CLOS for import and export respectively. Protégé supports the import of text files, database tables and RDF files and export of XML(S), RDF(S), OWL, Clips, SWRLIQ, Instance Selection, MetaAnalysis, OWLDoc, Queries and(RDF,UML,XML)backend. OntoLingua supports IDL, KIF for import and KIF, CLIPS, IDL, OKBC, Syntax, Prolog Syntax. Most of them support RDF(S) and XML(S). However, there is no comparative study on the quality of each of these translators. Moreover, there are no experimental results about the possibility of exchanging ontologies between different tools and knowledge on the loss in the translation processes.

Table III

INTEROPERABILITY			
Tools	With other Ontology Tools web Architecture	Import form Language	Exports to Languages Storage
Apollo Swoop	No No	Apollo Meta Language OWL,XML, RDF and text formats	OCML and CLOS RDF(S), OIL and DAML
Protégé	PROMPT, OKBC, JESS, FaCT and Jena	XML(S), RDF(S), OWL, (RDF,UML, XML)backend, Excel, BioPortal and DataMaster	XML(S), RDF(S), OWL, Clips, SWRL-IQ, Instance Selection, MetaAnalysis, OWLDoc, Queries and (RDF,UML, XML)backend
RDFedt	N/A	RDF(S),OIL, DAML, SHOE	RDF(S),OIL, DAML, SHOE
OilEd	No	RDF(S),OIL, DAML+OIL	RDF(S),OIL, DAML+OIL, SHIQ, dotty, HTML
OntoLingua	N/A	IDL,KIF	KIF,CLIPS,IDL,OKBC, Syntax, Prolog Syntax

Inference services are presented in (Table IV). This includes: built-in and other inference engines, constraint and consistency checking mechanisms, and exception handling. For built-in inference engine Protégé uses PAL and OilEd used FaCT. Protégé and Swoop have external attached inference engines. RDFedt uses exception handling.

Table IV

INFERENCE SERVICE			
Tools	Built-in inference engine	Constraint / Consistency Checking	Exception Handling
Apollo	No	Yes	No
Swoop	No	Only with reasoned plug-in	No
Protégé	Yes(PAL)	Yes via Plug-ins like PAL and FaCT	No
RDFedt	No	Only checks writing mistakes	Yes
OilEd	With FaCT	Via FaCT	No
OntoLingua	No	Via Chimaera	No

VII. CONCLUSION

A major challenge at present is the need for Ontology tools to support more expressive power and scalability with large knowledge base and reasoning in querying and matching.

In addition to this, they need to support the use of high-level language, modularity, visualization etc.

The is also some scope of research and applications about dynamic web pages consisting of database reports in e-commerce (Tarassenko & Bukharova, 2001).

Research on ontology-integration tasks in B2B ecommerce is also undergoing. The infrastructure of business documentation from the integration perspective and the identification of the integration subtasks were suggested (Monostori, Váncza, & Ali, 2003)

ACKNOWLEDGMENT

The authors are thankful to IJSCE Journal for the support to develop this document**.**

REFERENCES

1. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6245779
2. http://en.wikipedia.org/wiki/Semantic_Web
3. http://www.rdfabout.com/intro/#Introducing RDF
4. Ontology Development Tools for ontology-based knowledge management, S.Youn, D.McLeod, University of Southern California, 2006
5. http://www2012.wwwconference.org/proceedings/companion/p595.pdf
6. Kapoor, B., Sharma, S., "A Comparative Study Ontology Building Tools for Semantic Web Applications", International Journal 1, July (2010), 1-13.
7. Kalyanpur, A., Parsia, B., Sirin, E., Grau, B.C., Hendler, J., "Swoop: A „Web" Ontology Editing Browser", Mind, July 2005, 1-20.
8. Gruber, T. R. "A translation approach to portable ontology specification. Knowledge Acquisition" 5(2), 199-220, 1993.
9. Snae. C and Brueckner. M. Ontology-Driven E-Learning System Based on Roles and Activities for Thai Learning Environment. Interdisciplinary Journal of Knowledge