

Ontology Development Tools for Ontology-Based Knowledge Management

Seongwook Youn

University of Southern California, Los Angeles, USA

Dennis McLeod

University of Southern California, Los Angeles, USA

INTRODUCTION

Ontologies play a key role in realizing the full power of e-technology. Ontologies allow for machine-understandable semantics of data, and facilitate the search, exchange, and integration of knowledge for B2B (business-to-business) and B2C (business-to-consumer) e-commerce. Ontology is defined as “the specification of shared knowledge” (Waterson & Preece, 1999). By using semantic data, the usability of e-technology can be facilitated. There are several languages like XML (extensible markup language), RDF (resource description framework), RDFS (RDF schema), DAML+OIL, and OWL. Many tools have been developed for implementing metadata of ontologies using these languages. However, current tools have problems with interoperability and collaboration. The primary goal of this survey is to introduce and understand several tools through their use. Therefore, we can develop a new generation of tools that not only support more capabilities, but also solve the problems of current tools.

BACKGROUND

Ontology tools can be applied to all stages of the ontology life cycle including the creation, population, implementation, and maintenance of ontologies (Polikoff, 2003). An ontology can be used to support various types of knowledge management including knowledge retrieval, storage, and sharing (Pundit & Bishr, 1999). In one of the most popular definitions, an ontology is “the specification of shared knowledge” (Waterson & Preece, 1999). For a knowledge-management system, an ontology can be regarded as the classification of knowledge. Ontologies are different from traditional keyword-based search engines in that they are metadata, able to provide the search engine with the functionality of semantic matching. Ontologies are able to search more efficiently than traditional methods. Typically, an ontology consists of hierarchical descriptions of important concepts in a domain and the descriptions of the properties of each concept.

Traditionally, ontologies are built by both highly trained knowledge engineers and domain specialists who may not be familiar with computer software. Ontology construction is a time-consuming and laborious task. Ontology tools also require users to be trained in knowledge representation and predicate logic.

XML is not suited to describe machine-understandable documents and interrelationships of resources in an 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.

MAIN THRUST OF THE ARTICLE

There are several ontology languages like XML, RDF(S), DAML+OIL, and OWL that are used to implement tools for implementing the metadata of ontologies.

Protégé 2000

Protégé (Noy, Sintek, Decker, Crubezy, Ferguson, & Musen, 2001) is developed by Stanford Medical Informatics. With Protégé, a user can construct domain ontologies, customize data entry forms, and enter data. Protégé has an extensible plug-in architecture, allowing users to add functionality by using plug-ins. Hence, Protégé can be easily extended to use knowledge-based embedded applications. Tables and diagrams are constructed using graphical widgets. However, new basic types cannot be added in Protégé.

Protégé 2000 assumes that knowledge-based systems are usually very expensive to build and maintain because knowledge-based system development is done by a team including both developers and domain experts who may be less familiar with computer software. Protégé 2000 guides developers and domain experts through the process of system development. Developers can reuse domain ontologies and problem-solving methods with

Protégé 2000, shortening the time for development and program maintenance. One domain ontology that solves different problems can be used in several applications, and different ontologies can use the same problem-solving methods.

OilEd

OilEd (Bechhofer, Horrocks, Goble, & Stevens, 2001) is developed by the Information Management Group of the CS Department at the University of Manchester, United Kingdom. OilEd is an OIL editor that allows a user to create and edit OIL ontologies. OilEd is primarily intended to demonstrate the use of DAML+OIL, but it does not support a full ontology-development environment. OilEd does not support many activities such as the creation of large-scale ontologies, versioning, augmentation, and the migration and integration of ontologies that are involved in ontology construction. OilEd has no extensibility, but arbitrary class expressions, primitive and defined classes, and concrete-type expressions can be used.

Apollo

Apollo (Koss, 2002) is developed by the Knowledge Media Institute of Open University, United Kingdom. Apollo allows a user to model an ontology with basic primitives such as classes, instances, functions, relations, and so forth. The internal model is a frame system based on the OKBC protocol. 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 can be extended with plug-ins, but it does not support collaborative working.

RDFedt

RDFedt is developed by Jan Winkler of Germany. RDFedt allows a user to build complex and structured RDF and RSS (RDF site summary) documents. It provides an overview of complex data structures with element trees. Also, it allows a user to test data and to give comments and error messages with the help of additional functions.

RDFedt supports RDF, RDFS, and Dublin core elements. RSS 1.0 provides modules like aggregation, notation, content, cut, organization, change of page, threading, and so forth. 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 (hypertext markup language) document.

RDFedt is a textual language editor. It is not a Java program, is not platform independent, and works only on Windows.

OntoLingua

OntoLingua (Fikes, Farquhar, & Rice, 1997) is developed by the Knowledge Systems Lab of Stanford University. It provides a user-distributed collaborative environment, a suite of ontology-authoring tools, and a library of modular, reusable ontologies. It also supports a World Wide Web (WWW) interface and translation into different formats. OntoLingua is an ontology library and server that can be accessed with a traditional Web browser. By assembling and extending the ontologies obtained from the library and tools in OntoLingua, authorization can be provided. Using Chimaera, the taxonomy is reorganized and name conflicts in the knowledge base are resolved. Multiple users can use OntoLingua via write-only locking and user access-level assignment.

OntoEdit

OntoEdit (Sure, Angele, & Staab, 2003) is developed by Ontoprise of Germany. There are freeware and professional versions. Our survey focused on freeware. OntoEdit offers export interfaces to all major ontology-representation languages and has a flexible plug-in framework. This feature allows a user to customize the tool in a user-friendly fashion. Several functions are modularized, so it can be easily extended.

An ontology requirements-specification document describing what an ontology should support is needed for ontology development. According to the ontology requirements-specification document, an ontology engineer determines relevant concepts and their hierarchical structure in the ontology. OntoEdit can be used in this phase using two plug-ins, OntoKick and Mind2Onto5, for metaontology description with the automatic calculation of statistic information.

WebODE

WebODE (Arpirez, Corcho, Fernandez-Lopez, & Gomez-Perez, 2001; Corcho, Fernandez-Lopez, Gomez-Perez, & Vicente, 2002) is developed by the Technical School of Computer Science in Madrid, Spain. It was made to use and test the methontology methodology. The motivation of WebODE is to support an integrated ontological engineering workbench, which has three activity groups: ontology-development, -management, and -population activities; ontology middleware services; and ontology-

based application development. WebODE supports varied ontology-related services and most of the activities involved in the ontology-development process. It is not an isolation tool for ontology development. The WebODE architecture consists of three tiers. The first tier provides the user interface, the second tier provides the business logic, and the third tier consists of the data.

KAON

KAON (Volz, Oberle, Staab, & Motik, 2003) is developed by the FZI Research Center and the AIFB Institute of the University of Karlsruhe, Germany. KAON is designed to manage business applications. KAON has two user-level applications: OiModeler and KAON PORTAL. All other modules except for these two applications are for software development. OiModeler is an ontology editor for ontology creation and maintenance. We can navigate and search ontologies with a Web browser using KAON PORTAL. KAON supports scalable and efficient reasoning with ontologies; RDFS extension with symmetric, transitive, and inverse relations; and metamodeling using axiom patterns.

ICOM

ICOM is developed by Free University of Bozen-Bolzano, Italy. The purpose of the development of ICOM is to provide a freeware conceptual modeling tool that can be used for knowledge representation in database and ontology design. It is useful in the conceptual modeling of databases and the design of various ontologies. It supports interontology mapping with the GUI (graphical user interface). ICOM is created with Java 1.2 and can be run on Linux and Windows machines. The CORBA protocol with the FaCT description logic server is used for ICOM communication.

DOE

DOE (Differential Ontology Editor; Isaac, Troncy, & Malais, 2003) is developed by the National Audiovisual Institute in France. It follows a methodology proposed by Bruno Bachimont. DOE is not developed for full ontology development, so many activities in traditional ontology construction are not supported. DOE allows a user to attach a lexical definition to concepts and relations and justify their hierarchy.

DOE has problems importing Dublin core metadata. In DOE, interoperability can be applied using both RDFS and OWL, and concepts cannot be defined intentionally with constraints. Individuals cannot be imported during the running of WebODE.

WebOnto

WebOnto (Domingue, Motta, & Corcho Garcia, 1999), developed by the Knowledge Media Institute of the Open University in England, supports the collaborative browsing, creation, and editing of ontologies, represented in the knowledge-modeling language OCML without suffering from the interface problems. WebOnto attempts to be easy to use and yet scalable to large ontologies. The main features of WebOnto are the management of ontologies using a graphical interface, the automatic generation of instance-editing forms from class definitions, the inspection of elements taking into account the inheritance of properties, and consistency checking and support for collaborative work using broadcasting and receiving and making annotations. WebOnto consists of a Java-based central server and clients.

Medius Visual Ontology Modeler

Medius Visual Ontology Modeler (VOM) is a UML-based ontology-modeling tool that enables ontology development and management for use in collaborative applications and interoperability, and is available as an add-on to the Rational Rose Enterprise Edition. It is developed by Sandpiper Software. It has a set of ontology-authoring wizards that create and maintain the required UML model elements for the user, substantially reducing construction errors and inconsistencies. That is to say, it supports limited consistency checking.

LinKFactory Workbench

LinKFactory Workbench (Deray & Verheyden, 2003) is designed for very large medical ontologies. The LinKFactory client-server structure is an information system constructed using a three-tier architecture. First, LinKFactory Workbench is a client application on the user side to manage the LinKBase ontology database. Second, LinKFactory is the server interface that receives and answers user requests, holds the business logic, and requests data using the application-server tier. Third, the data layer accesses the underlying database. This database contains all information (user information, ontology contents, and maintenance information).

K-Infinity

K-Infinity is a tool developed by Intelligent Views, a German company, for the creation, maintenance, and use of a knowledge network. It is a knowledge editor with broad support for object-oriented knowledge modeling.

Table 1. Comparison table

	Import format	Export format	GUI	Consistency check	Multi-user	Web support	Merging
Protégé 2000	XML, RDF(S), XML schema	XML, RDF(S), XML schema, FLogic, CLIPS, Java, HTML	Via plug-ins like GraphViz and Jambalaya	Via plug-ins like PAL and FaCT	Limited (multi-user capability added to it in 2.0 version)	Via Protégé-OWL plug-in	Via Anchor-PROMPT plug-in
OilEd	RDF(S), OIL, DAML+OIL	RDF(S), OIL, DAML+OIL, SHIQ, doty, HTML	No	Via FaCT	No	Very limited name spaces	No
Apollo	OCML, CLOS	OCML, CLOS	No	Yes	No	No	No
RDFedt	RDF(S), OIL, DAML, SHOE	RDF(S), OIL, DAML, SHOE	No	Only checks writing mistakes	No	Via RSS	N/A
OntoLingua	IDL, KIF	KIF, CLIPS, IDL, OKBC syntax, Prolog syntax	No	Via Chimaera	Via write-only locking, user access levels	Yes	N/A
OntoEdit (free version)	XML, RDF(S), Flogic, DAML+OIL	XML, RDF(S), Flogic, DAML+OIL	Yes	Yes	No	Yes	N/A
WebODE	RDF(S), UML, DAML+OIL, OWL	RDF(S), UML, DAML+OIL, OWL, Prolog, X-CARIN, Java/Jess	Form-based graphical user interface	Yes	By synchronization, authentication, and access restriction	Yes	Via ODEmerge
KAON	RDF(S)	RDF(S)	No	Yes	By concurrent access control	Via KAON PORTAL	No
ICOM	XML, UML	XML, UML	Yes	Via FaCT	No	No	With inter-ontology mapping
DOE	XSLT, RDF(S), OIL, DAML+OIL, OWL, CGXML	XSLT, RDF(S), OIL, DAML+OIL, OWL, CGXML	No	Via type inheritance and detection of cycles in hierarchies	No	Load ontology via URL	No
WebOnto	OCML	OCML, GXL, RDF(S), OIL	Yes	Yes	With global write-only locking	Web based	N/A
Medius VOM	XML schema, RDF, DAML+OIL	XML schema, RDF, DAML+OIL	UML diagrams via Rose	With a set of ontology-authoring wizards	Network based	Via read-only browser support from Rose	Limited (only native Rose model)
LinKFactory	XML, RDF(S), DAML+OIL, OWL	XML, RDF(S), DAML+OIL, OWL, HTML	No	Yes	Yes	Yes	Yes
K-Infinity	RDF	RDF	With graph editor	Yes	Network based	No	N/A

Knowledge Builder is K-Infinity's main component. Using Knowledge Builder, knowledge engineers and lexicographers can create, delete, rename, and edit both objects and relations, as well as relate objects to each other according to defined relations. This can be done in two different workspaces: (a) the graph editor that enables the user to see a graphical view of the network of objects and the relations of them, and (b) the concept editor that is a supplemental data viewer to the graph editor.

In addition to tools for editing the knowledge, K-Infinity provides the K-Organizer, which provides administration, navigation, search, and query formulation.

FUTURE TRENDS

There are still some research challenges. Ontology tools have to support more expressive power and scalability

Table 1. Comparison table (continued)

	Collaborative working	Ontology library	Inference engine	Exception handling	Ontology storage	Extensibility	Availability
Protégé 2000	No	Yes	With PAL	No	File & DBMS (JDBC)	Via plug-ins	Free
OilEd	No	Yes	With FaCT	No	File	No	Free
Apollo	No	Yes	No	No	Files	Via plug-ins	Free
RDFedt	No	No	No	Yes	Files	No	Free
OntoLingua	Yes	Yes	No	No	Files	No	Free
OntoEdit	No	No	No	No	File	Via plug-ins	Free
WebODE	Yes	No	Prolog	No	DBMS (JDBC)	Via plug-ins	Free
KAON	No	Yes	Yes	No	DBMS	No	Free
ICOM	No	No	Yes	No	DBMS	Yes	Free
DOE	No	No	Yes	No	File	No	Free
WebOnto	Yes	Yes	Yes	No	File	No	Free Web access
Medius VOM	Yes	Yes (IEEE SUO)	Yes	N/A	N/A	Yes	Commercial
LinKFactory	Yes	Yes	Yes	No	DBMS	Yes	Commercial
K-Infinity	Yes	Yes	Yes	N/A	DBMS	No	Commercial

with a large knowledge base, and reasoning in querying and matching. Also, they need to support the use of high-level language, modularity, visualization, and so forth.

There are also research and applications about dynamic Web pages consisting of database reports. Database reports are indispensable for every e-commerce transaction (Tarassenko & Bukharova, 2001).

Research on ontology-integration tasks in B2B e-commerce 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)

There is research on a generic e-business model ontology for the development of tools for e-business management and IS requirements engineering. Based on an extensive literature review, the e-business model Ontology describes the logic for a business system (Osterwalder, Parent, & Pigneur, 2004).

CONCLUSION

So far, we have described 14 ontology tools. All of these tools are ontology-development tools. To compare each tool's features, we have chosen the tools with similar purpose. Of course, there are many other tools that have different purposes. For example, Chimaera, FCA-merge, and PROMPT are ontology-merge and -integration tools. AeroDAML, COHSE, MnM, and OntoAnnotate are ontology-annotation tools. Sesame, Inkling, rdfDB, Redland, jena, and cerebra are ontology-storing and -querying tools.

Several important aspects when we analyze tools exist. Most of the tools are moving toward Java platforms and extensible architectures as well. Interoperability and storage in databases are still weak points of ontology tools.

Polikoff (2003) used Protégé 2000, OilEd, and OntoEdit to test the interoperability of ontology tools. For example, plain RDFS from OilEd could be used in OilEd and Protégé 2000, but it was not working well in OntoEdit. Plain RDFS from Protégé 2000 could be used in Protégé 2000 and OilEd version 19990303, but it could not be used in OntoEdit. Also, standard Oil RDFS from Protégé 2000 and OWL from OilEd could not be used in all tools (OilEd, Protégé 2000, and OntoEdit). DAML+OIL from OilEd could be used in OilEd and OntoEdit with minor problems, but it could not be used in Protégé 2000. Interoperability with other ontology-development tools, -merging tools, and databases, as well as translations to and from some ontology languages, is an important factor in order to integrate ontologies in applications. However, there are few comparative studies about the quality of all translators, the possibility of exchanging ontologies between different tools, and the loss of knowledge in the translation processes.

In case of storage, only a few of the surveyed tools use databases for storing ontologies: LinKFactory, OntoEdit Professional Version, Protégé 2000, and WebODE. Also, only a few have backup-management functionality.

Although a lot of similar ontology tools exist for ontology creation, they do not interoperate well and do not cover all the activities of the ontology life cycle. The lack of interoperability between all these tools creates

important problems, for example, when we integrate an ontology into the ontology library of a different tool, or when two ontologies built using different tools or languages are integrated using merging tools.

REFERENCES

- Arpirez, J., Corcho, O., Fernandez-Lopez, M., & Gomez-Perez, A. (2001). WebODE: A scalable workbench for ontological engineering. *First International Conference on Knowledge Capture (KCAP'01)* (pp. 6-13).
- Bechhofer, S., Goble, C., & Horrocks, I. (2001). DAML+OIL is not enough. *SWWS-1, Semantic Web Working Symposium*.
- Bechhofer, S., Horrocks, I., Goble, C., & Stevens, R. (2001). OilEd: A reasonable ontology editor for the semantic Web. In *Lecture notes on artificial intelligence: Vol. 2174. Proceedings of KI2001* (pp. 396-408). Springer-Verlag.
- Corcho, O., Fernandez-Lopez, M., Gomez-Perez, A., & Vicente, O. (2002). WebODE: An integrated workbench for ontology representation, reasoning and exchange. In *Lecture notes on artificial intelligence: Vol. 2473. 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW'02)* (pp. 138-153). Springer-Verlag.
- Das, A., Wu, W., & McGuinness, D. L. (2001). Industrial strength ontology management. *Semantic Web Working Symposium* (pp. 17-37).
- Deray, T., & Verheyden, P. (2003). Towards a semantic integration of medical relational databases by using ontologies: A case study. *OTM Workshops* (pp. 137-150).
- Domingue, J., Motta, E., & Corcho Garcia, O. (1999). *Knowledge modelling in WebOnto and OCML: A user guide*.
- Fikes, R., Farquhar, A., & Rice, J. (1997). Tools for assembling modular ontologies in Ontolingua. *AAAI/IAAI* (pp. 436-441).
- Gunther, O. (1998). *Environment information systems*. Springer.
- Isaac, A., Troncy, R., & Malais, V. (2003). Using XSLT for interoperability: DOE and the travelling domain experiment. *Proceedings of the Second International Workshop on Evaluation of Ontology-based Tools (EON 2003)*, Sanibel Island, FL.
- Koss, M. (2002). *Apollo: The user guide*.
- Macris, A. (2003). *CULTOS: Cultural units of learning tools and services*.
- Monostori, L., Váncza, J., & Ali, M. (2003). Ontology integration tasks in business-to-business e-commerce. *Proceedings of the 14th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE 2001)*, Budapest, Hungary.
- Noy, N., Sintek, M., Decker, S., Crubezy, M., Ferguson, R., & Musen, M. (2001). Creating semantic Web contents with Protege-2000. *IEEE Intelligent Systems*, 16(2), 60-71.
- Osterwalder, A., Parent, C., & Pigneur, Y. (2004). Setting up an ontology of business models. *CAiSE Workshops*, 3, 319-324.
- Polikoff, I. (2003). Ontology tool support. In *TopQuadrant Technology Briefing*.
- Pundt, H., & Bishr, Y. (1999). Domain ontologies for data sharing: An example from environmental monitoring using field GIS. *Computer & Geosciences*, 28, 98-102.
- Sure, Y., Angele, J., & Staab, S. (2003). OntoEdit: Multifaceted inferencing for ontology engineering. *Data Semantics*, 1, 128-152.
- Tarassenko, P. F., & Bukharova, M. S. (2001). System for database reports generating. *Science and Technology, KORUS'01: Proceedings of the Fifth Russian-Korean International Symposium (Vol. 1)*, pp. 84-88.
- Volz, R., Oberle, D., Staab, S., & Motik, B. (2003). KAON SERVER: A semantic Web management system. *Alternate Track Proceedings of the Twelfth International World Wide Web Conference, WWW2003* (pp. 20-24).
- Waterson, A., & Preece, A. (1999). Verifying ontological commitment in knowledge-based systems. *Knowledge-Based System*, 12, 45-54.
- Weibel, S., Kunze, J., Lagoze, C., & Wolf, M. (1998). *Dublin core metadata for resource discovery* (IETF No. 2413). The Internet Society.

KEY TERMS

B2B (Business-to-Business): Business that provides some kind of services or sells some product to other businesses.

E-Technology: Technology including hardware and software, and their development using Internet, multimedia, mobile, wireless, and security technologies, and so forth.

Ontology Development Tools for Ontology-Based Knowledge Management

Knowledge Management: Systematic process of finding, organizing, and presenting information according to a specific area of interest.

Metadata: Data about data, or machine-understandable information for the Web that describes other data.

Ontology: “[T]he specification of shared knowledge” (Waterson & Preece, 1999) and the relationships of the collected concepts.

Ontology-Management System: System for querying, storing, creating, modifying, loading, accessing, and manipulating ontologies.

Semantic Web: Extension of the current Web with well-defined meaning, facilitating machines and people to work in cooperation.

