A Comparison of Ontology Servers

M. Nazir Ahmad, Robert M. Colomb
School of Information Technology and Electrical Engineering
The University of Queensland
AUSTRALIA
{nazir, colomb}@itee.uq.edu.au

Mohd Taib Abd Wahid
Faculty of Computer Science and Information Systems,
The University of Technology
MALAYSIA
taib@fsksm.utm.my

Abstract

Recently, an ontology is increasingly becoming an essential tool for solving problems in many research areas. The ontology is a complex information object. It is used in practice and can contain millions of concepts in complex relationship. When we want to manage complex information objects, we generally turn to information systems technology. There are many tens of millions of information systems in use around the world doing many things ranging from helping a business manage its billing to keeping track of the design and parts inventory of large aircraft. An information system intended to manage ontology is called an ontology server. The ontology server technology is at the time of writing quite immature. Therefore, in this paper, we necessarily try to remain somewhat speculative in nature by reviewing several ontology servers in mainstream literatures. As a result, we point out some important opportunities that would trigger several research questions related to the server technology.

Keywords: Ontology, Ontology Server, Knowledge Sharing, Information Systems Interoperability

1. Introduction

An ontology is an explicit specification of a conceptualization [58]. It is a designed artifact that formally represents agreed semantics of a domain interest in computer resources [58][60]. This enables the sharing and reuse of information and allows for the interoperation of information systems [111]. Although not a new field, ontology research has recently received renewed interest and attracted many other fields such as the semantic web (e.g., [15] [64] [96]), databases (e.g., [132][130] [62]), electronic commerce (e.g., [10][29][63]), knowledge management (e.g., [4][70][113][32]), electronic learning (e.g., [116][3][124]), agent technology (e.g., [54][122][82]), information retrieval (e.g., [61][5] [6]), digital library (e.g., [21][129][115]), bioinformatics (e.g., [13][85][14]), geographical information systems (e.g., [53][112][52][23][123]), software engineering (e.g., [131][38]), intelligent systems (e.g., [11][8]) and natural language processing (e.g., [79][48]). Thus, we can classify the ontology applications as reported in Jasper et al. [76], Pisanelli et al. [110], Fensel [49], Mizoguchi [98] and the most comprehensive survey by Hart et al. [69].

2. Ontology Applications

Mizoguchi [98] defines five typical types of ontology application including: (a) ontology as a common vocabulary, (b) ontology as assisting of information access, (c) ontology as the medium for mutual understanding (d) ontology as specification and (e) ontology as foundation of knowledge systematization. However, we are interested in (c), which ontology plays an important role for mutual understanding. Mutual understanding is always necessary between (a) humans and humans (b) humans and software agents (c) software agents and software agents. In communication amongst humans, the ontology can be useful especially for knowledge-intensive engineering such as concurrent engineering, business process engineering, where interdisciplinary collaboration is required. Understanding between humans and software agents could be seen in the context of WWW information searching. The requirements specified by the users have to be properly understood by search engines through the shared ontology. The semantic web is the biggest application of this type. The same happens in the case of communication between software agents. In order to enable software agents to communicate with each other, they need a common protocol and vocabulary. An ontology plays a role in a common vocabulary in a more advanced way than that in (a) because ontology agents manage such vocabularies in a formal way to process queries about ontologies by other agents.

3. Ontologies and Semantic Web

Ontologies are the most important part of the Semantic Web. The semantic web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation [15]. In the semantic web, ontologies can be used to encode meaning into a web page, so that intelligent agents can understand what the web page is about, and therefore provide humans with more useful cooperative services [16]. However, most of the main problem with ontologies is how to manage them. Due to this problem, there are related research trends in the field of ontologies which include (a) creating large-scale ontologies (e.g., [91]), (b) defining expressive languages for representing ontological knowledge (e.g., [93][84] [50][17][86][22]) and (c) implementing systems that support ontology-based applications (e.g., [55][126][27] [33][92][39][105][20][114][71][125][78]). However, the authors give more emphasis to (c), which relates to the development of systems, specifically the ontology server that focuses on ontology supporting information systems interoperation. In addition, we should be aware of several ontology management issues such as ontology representation, reusability, usability, storage, versioning and security as discussed in (e.g., [43] [103] [78]) and take them challenging tasks are tailored to our research context (i.e. ontology usability). Therefore, in the scope of ontology server development and supporting runtime interoperation, we will concentrate on some aspects relating to how ontology should be represented in the server, how the server will manage those ontologies (i.e. server functions) and how we should design its repository (i.e. ontology repository). Hence, to address these aspects, we will investigate more on theoretical and practical issues in designing ontology server context.

4. Model Driven Architecture (MDA) to Ontologies

Ontologies and MDA are two technologies being developed in parallel, but by different communities [65]. They have common points and issues and can be brought closer together [28][65][40]. Therefore, to bring software engineering practitioners and ontologies closer, many researchers suggest the use of Unified Modeling Languages (UML) in ontology development (e.g., [31][18][87][51] [66][28][40][7]). The main question they want to answer is how to use UML as well-accepted modeling languages for developing and using ontologies in real world applications. Although the ontology concepts are coincidently similar to object-oriented paradigms, it has some limitations mainly regarding the concept of property (e.g., see [19]). Because of these discrepancies, initially, we could only use UML in the beginning of ontology development. However, there is a significant movement in this research to overcome these limitations using UML extensions (i.e. UML profiles) as implemented in [28][40]. As a result, the Object Management Group (OMG) has established Ontology Definition Metamodel (ODM) as a MDA standard metamodel for modeling ontology [28]. The ODM defines concrete abstract syntaxes (i.e. OWLDataTypeProperty, OWLClass) for modeling ontology that can be represented by using UML profiles [28]. The ODM is centrally based on UML and the W3C Web Ontology Language (OWL) recommendation [17]. In terms of ontology modeling, on one hand, the UML provides powerful graphical modeling capabilities and widely supported tools (i.e. Rational Rose, Poseidon, Magic Draw, ArgoUML, etc). In addition, since the UML and ODM are MOF-compliant languages, it is possible to store ontologies in MOF-based repositories, to store ontologies diagrams in a standard way (UML2 XMI), as well as to share and interchange ontologies using XMI [28][40]. However, on the other hand, we note that not all OWL features could be represented by UML. We will use ODM and UML profiles defined in [28] for representing ontologies and designing the server. In addition, UML is currently a de facto standard modeling language. There is a growing interest in its adoption as a language for conceptual modeling and ontological representation (e.g., [31][18][87][51][90][66] [28][83][40][7][67][136][68]). Also, the UML is widely accepted as a universal modeling language for modeling any kind of information systems [24] such as efforts by [97][26][88][59][72][34][37][75][117][101][9][95][94].

5. A Comparison of Ontology Servers

The ontology server is used at design, commit and runtime. At design time, the ontologist uses the server for developing ontologies. At commit-time, players (i.e. applications) will commit to part of the entire ontology for joining the exchanges of messages. At runtime, players exchange messages mediated by shared ontologies. Zachman [135] has proposed an information system architecture framework. This framework is well-accepted and has achieved international recognition (e.g., [46][73][81][45][47][107][133][80]) for describing an information system. There are three main concepts (knowledge, process and communication) and related enabling technologies described by Zachman [135]. Since the ontology server is a kind of information system, thus, it can be described in terms of this framework. We do not tend to discuss the framework (see details in [135]). With respect to Zachman's framework, we will concentrate on process and knowledge aspects of the ontology server. These are two main server components: (a) Server's functionality (b) Server's repository. Furthermore, we will look into the use of the server at commit-time. This work is centrally devoted to how the server can be used to assist players for joining the exchanges.

5.1 Ontology Server Functions

Unfortunately, much literature (e.g., [105][106][118] [125][114]) does not directly use term ontology server in their works though some of facilities discussed basically relate to ontology server. Most of literature discusses mainly the server's functions but they describe them in terms of a special purpose tool for managing ontologies (e.g., [119][55][102][106][118][89][44][100]). Some of those tools perform single or a few server functions as shown in [100] and [44] for storing and browsing ontology. Some others tools provide a sophisticated suite of application tools like editing, browsing, publishing, storing and reasoning ontologies such as Farquhar et al.

[55]. Besides developing a respective tool, the server's functions could be implemented by Application Programming Interface (API) such as Oberle et al. [108]. Furthermore, the server's functions can also be implemented by both developing a significant tool and API like Starlab [127]. Moreover, there are also several literatures that discuss ontology management frameworks but very few address the server's role in their framework (e.g., [121][74][33]). These literatures have only discussed the use of the server but have not explained in details how the server is developed to meet their goals. We assume that they are not really concerned about the server developments but emphasize the framework or architecture of the ontology management environment. In addition, these literatures relate to specific large research projects which eventually focus on building sophisticated tools for developing and maintaining ontologies. However, several literatures of these projects really address the issues of the ontology server development such as Ontolingua [55], KAON [108] and Starlab [127]. In term of server functions, to avoid confusion, we should be aware of many terms appearing in the literatures regarding server functions such as server functionality and functions [127], ontology service [57], server process [27], server services [55] server operation and ontology operation [39] and server facilities [109].

5.2 **Ontology Server Repository**

Besides the matter of server's functions, there are literatures discussing more on the server's repository aspects such as (e.g., [109][71]). For example, Harrison et al. [71] have proposed a generic representation of ontology repository. Similar to Pan et al. [109], they have introduced a so-called lightweight ontology repository for enabling shareable and maintainable ontologies. Repository is a prototype implementation of a design to allow ontology designers and agents to use open Web standards to publish and retrieve ontologies and metadata about them. Key features of the system include the use of the HTTP protocol following the REST (Representational State Transfer) architectural style, the representation of the recorded information about ontologies and the repository information schema using RDF and its schema language RDFS, and the use of URNs to identify ontologies. The use of Web standards for communication between agents and Web-based resources such as ontology repositories provides a more lightweight and open architecture for agent interaction with these resources. Harrison et al. [71] discuss two methods of storing the ontologies. The first method involves storing individual ontology in a separate flat file. The file provides a more straightforward view where its contents can be inspected. However, the main problem with this method is that a search engine would need to be developed to search the contents of the ontology files [71]. The second method uses a database to store the ontologies. Databases have indexing and securing capabilities that enable faster searching. Therefore, to ensure scability and maintainable large ontologies, the database method rather than a flat file we think would preferable for storing the ontologies. However, how the ontologies are stored depend on how ontology representation should appear in the server and what types of database are considered (i.e. relational, object-relational). For example, Harrison et al. [71], Mauger [100], Starlab [127] use relational database to store ontologies. Harrison et al [71] use MySQL [104] because it has (a) excellent performance and (b) it can accommodate object-oriented relational data.

5.3 **Ontology Server Implementation**

Section 5.1 outlines related works on ontology server functions in the literatures, while section 5.2 overviews examples of the server repository discussed in the literatures. However, we are interested in a server that used in the worldwide web, particularly supporting semantic web vision. We would say that the implementation of the server depends on the purpose of the server development. For example, several servers developed are for supporting B2B and electronic commerce (e.g., [27][33][108][39][127]), knowledge-based systems (e.g., [128][25]), enterprise systems (e.g., [137]), in open environment like world wide web and agent-based systems (e.g., [55][57][56][44][92][109]), semantic web (e.g., [33][56][108] [125][114][71][127]). However, in this paper, we will look into several ontology servers that (a) are mostly cited in literatures (b) they are explicitly called ontology server and address the development issues (c) we assume that several of them are still progressing and (d) we are interested in servers used in open environment like web and semantic web. Thus, those are Ontolingua Server [55], ACOS [92], Starlab [127], KAON [108], OntoRama [44], OWS [39], FIPA Ontology Server [57] and Adapted Ontology Server [27].

5.3.1 **Several Ontology Servers: Ontolingua, ACOS, Starlab, KAON**

The first is an Ontolingua server [55], a kind of a knowledge repository management system that has been running since 1995. It is a tool that supports distributed, collaborative editing, browsing and creation of ontolingua ontologies [1][27][49][33][92][103]. The ontolingua ontology uses the representation languages, both Ontolingua Frame Ontology and KIF, which are wide spectrum languages capable of representing fine features of concepts. The Ontolingua server supports ontology inclusion and circular dependencies [55]. Its consistency-check capability, however, is restricted to the functions similar to database schema checking [55][92]. For instance, "all slots, slot values, facets and facets values are checked to make sure that they conform to the constraints that they apply (i.e. domain, range, slot value type, and cardinality constraints)" [92]. Meaning, the consistency checking is done in domain experts' heads. It is a large project focusing on ontology development [55]. It has built sophisticated tools for developing and maintaining frame-based ontologies. It focuses on formal ontology specifications, and reuse and translation to different ontology implementation systems [55]. However, it does not address problems related to legacy systems and tools to merge ontologies [33]. The users interact with the single server through a web browser to create, edit and browse ontologies. Users have to tolerate the network delays and server response delays [134]. The ontolingua server uses ontolingua-based repository for storing ontologies and developing a set of tools for demonstrating server functions.

The second server is ACOS [92]. Li et al. [92] claim it to be a community-oriented ontology server. It provides the way for the so-called "community" to construct ontologies. The concept of "community" in ontology management enables everyone to have the opportunities of influencing ontology construction. This depends on the users' importance score, and such score is computed mechanically based on how active these users are in contributing to the knowledge base. They came up this kind of vision because in an open environment, ontology is the asset of all participants; every user can join, contribute and leave such community. The opposite of this approach is called "central". The central-controlled mechanism would not appear to be neutral for this situation [92]. However, most of the ontology server implementations in the literatures apply the "central" approach to ontology development (e.g., [55][27][56][44][127]). The "central" means a centralized ontology server, only group of fixed users have the rights to modify ontologies, which is similar to the situation in database management [92]. Similar to Ontolingua server, the ACOS [92] has also provided consistency checking but not limited to humans. It also can be shared by a wide range of software agent communities and it facilitates on-line ontology construction, consistency-check and use [92]. Li et al. [92] claim that ACOS is designed to be an online community in which a diverse group of software agent can contribute and use ontologies at runtime. A key enabler in this scenario is a high degree of "shareness" of the ontologies maintained by the server. This relates to designing an appropriate knowledge representation, which is the first step towards building an ontology server. In this context, Li et al. [92] believes that to achieve "shareness", it is required that local features of ontology are removed. Examples of constructors that can bring in local features include *property* in DAML-OIL [35] or *slot* in Ontolingua [55]. Other examples of constructors that bring in local effect are *part-of* in Framework [55] and *disjoint-with* in Descriptions Logic [12]. In other words, Li et al. [92] accept the shared ontologies should have minimal expressivities (a coarse ontology) which consist of a minimal set of axioms written in a language of minimal expressivities. Thus, to achieve a high degree of shareness, Li et al. [92] believe that constructors *Class, subClassOf, SameClassAs, SuperClassOf* and *InstanceOf* offer more stability in the process evolution and minimal expressivities. Compared to Ontolingua server, the ACOS use DAML-OIL as ontology representation languages [92]. The ACOS server uses file-based repository for storing DAML-OIL ontologies and developing a set of APIs for implementing server functions. To facilitate collaborative, online ontology development, this server implemented import mechanism that is similar to the *inclusion model* in Ontolingua server and *extension relationship* in FIPA ontology server [57]

Third is a Starlab's ontology server [127]. This server is also still under investigation and progressing several deliverables. This research initially comes from works done by Jarrar [78] in Dogma's framework [77][120][78][127]. At the time of writing, we would say that this is a new ontology server research project. The mission of its ontology server is to assist the gathering and incremental growth of ontologies [138]. In terms of ontology representation, this server is inline with the rationale of ACOS server previously discussed. The proposed ontology model consists of five basic elements: context, terms, concepts, roles and lexicon. Starlab [127] claims that constraints and derivation rules are intentionally left outside the ontology. At this moment, they are still experimenting with the ontology model and an early version of ontology server has been implemented. In the first prototype, consistency-check and user control are not included [127]. This server uses MSQL server to store the ontology and an ontology objects are expressed in XML. Its server functions are implemented based on developed tools and basic APIs (e.g., [127][92]). In addition, the server's ontology representation also benefits from graphical notation based on Object Role Modeling (ORM) languages. Fourth, is KAON server [108]. Similarly to Ontolingua server [55] and Starlab [127], the KAON server is also a result of sophisticated tools delivered by a large research project. This server provides its functions through a set of APIs and store RDF-based ontology using relational database [108][103]. In terms of ontology representation, it is similar to Ontolingua server [55] the complexity of ontology (i.e. property) is not left outside the ontology, which contrast with Starlab [127] and ACOS [92].

5.3.2 Several Ontology Servers: Others

Fifth is the OntoRama server [44]. The main functions of OntoRama include search, compare and modify WebKB ontologies. It does not support consistency-check or cross-ontology queries; as a result, its capability of supporting online collaborative ontology construction is restricted [92]. In our opinion, this server closely demonstrates functions of storing and browsing the RDF-based ontologies while supporting ontology development is not its main purpose. Sixth is the OWS server proposed by Dameron et al. [39]. This server does not generally deal with ontology development. It is used for all other tasks, when the ontologies already exist. The implementation of this server is bit tricky. In fact, the server described in [39] does not serve as ontology repository (i.e. the server itself does not store the ontologies). The server provides services, taking ontologies as inputs. Hence, they claim that ontology evolution is not a problem here, because only simply give the server a new ontology as inputs. In our opinion, we would say that this server supports runtime interoperation. Its server functions are implemented using web services and assume that ontologies naturally reside in any player's sites. In other words, the ontologies are stored based on file-based, which hold by different players' sites. However, we think the underlying assumption in the Dameron et al. [39] that ontologies can be pieced together dynamically is not workable. Fortunately, most of the content of the paper does not depend on dynamic piecing together ontologies. Seventieth is the FIPA ontology server. The FIPA (Foundation for Physical Agents) is a standard body, which has developed interaction standards for agents in open environment. In the FIPA's ontology server specification, its server's ontological representation is divided into a *fine-grained ontology* or called heavyweight ontology in [110] and *a coarse ontology* that consists of minimal set of axioms written in a language of minimal expressivity so called lightweight ontology in [110]. However, the FIPA's server supports these two kinds of ontology representation but in different scope of use and level of details [57].

Finally is the server proposed in [27]. This server is developed specifically for ontology developed for electronic commerce applications. In terms of ontology representation, Chung et al. [27] define three main criteria for ontology: (a) ontology can be translated and (b) ontology should be practical. Its server functions include gathering information from web, creating a relation, modifying and rebuilding the standard ontology and servicing the standard ontology. This server uses MySQL database to store ontologies and using Java APIs for implementing server functions. We assume that this server allow complex ontological representation. This because the standard ontology is built from the local terms used in sites, and then it provide editor tool for making relation between ontologies [27]. In addition, they claim that a standard ontology is necessary have the objective and concrete property [27]. In general, this server developed for supporting ontology development but limited to the electronic commerce and deal with the current web. In order to have a clearer understanding and better view of the ontology server implementation, we summarize several servers as discussed in section 5.3.1 and 5.3.2 in the Table 1 (see appendix).

6. Conclusions and Future Works

The servers related works have previously been developed for different purposes such as enterprise systems (e.g., [137]), knowledge-based systems (e.g., [128][25]), in open environment like world wide web and agent-based systems (e.g., [55][57][56][44][92][109]), semantic web (e.g., [33][56][108][125][114][71][127]) and electronic commerce and B2B (e.g., [27][33][108][39][127]). However, we are only interested in the use of server for supporting semantic web vision. In the semantic web movement, the ontology server is increasingly becoming important for supporting many purposes of semantic web applications. Thus, it is believed to be a key component of the semantic web [2]. The server is used at design, commit and runtime. However, we note that the server development is closely determined by the ontology requirements and purposes. For example, the ontologies supporting runtime interoperation is slightly different from ontology for supporting engineering applications [28][69]. Thus, the server to be developed should tailor to what ontology is made for. In [69] provides the most comprehensive survey of ontology usage and requirements for determining the scope of ontology server development. In conclusion, we highlight several important points of ontology server development:

(a) A standard methodological approach for designing ontology servers is still missing.
(b) Most of the ontology server usage is mainly focused on design time issues, or, most of the servers are used for supporting ontology development.
(c) Only few ontology servers benefited from the graphical ontological representations.
(d) Since the server mostly used to support ontology development, mainly, the reusability aspect considered important in many literatures.
(e) There is almost no explicit discussion of servers for supporting information systems interoperation in the mainstream literature.
(f) The server functions to support many aspects of Semantic web are still missing.
(g) Still the main issues in ontological representation of the server, which related to the ontological repository of the server.

Our future works is to define several strategies of useful methodological approach for designing ontology server in the context of runtime interoperation problems.

7. References

[1] Abugessaisa, I. E A and Sivertun, A. Ontological Approach to Modeling Information Systems, *Proceedings of the Fourth International Conference on Computer and Information Technology (CIT'04) IEEE,* 2004

[2] Agrawal, R. Making Semantic Web Real: Some Building Blocks. Invited Talk, *Proceedings of International Workshop on Semantic Web,* Hawaii, 2002.

[3] Angelova, G., Kalaydjiev, O., Strupchauska, A. Domain Ontology as a Resource Providing Adaptivity in eLearning. *Proceedings of on the Move to Meaningful Internet Systems 2004, OTM 2004, Workshops*, LNCS 3292, Cyprus, pp. 700-712, 2004.

[4] Abdul Majid H. Mohamed, Lee, S.P., Siti Salwah Salim. An Ontology-Based Knowledge Model For Software Experience Management, *Journal of Knowledge Management Practice*, Vol. 5, May 2004.

[5] Aitken, S., Reid, S. Evaluation of an Ontology-Based Information Retrieval Tool, *Proceedings of ECAI'00*, Berlin, Germany, 2000.

[6] Abdelali, A., Cowie, J., Farwell, D., Ogden, B., Helmreich, S. Cross-Language Information Retrieval Using Ontology, *Proceedings of TALN Batz-Sur-Mer*, France, 2003.

[7] Aßmann, U., Zschaler, S. and Wagner, G. *Ontologies, Meta-Models, and the Model-Driven Paradigm.* In Coral Calero, Francisco Ruiz, and Mario Piattini (eds.): Ontologies for Software Engineering and Technology. Springer, 2006.

[8] Akkermans, J., Baida, Z., Gordijn, J., Pe˜na, N., Altuna, A., Laresgoiti, I. Value Webs: Using ontologies to Bundle Real-World Services. *IEEE Intelligent Systems*, 19(4), pp. 57–66, 2004

[9] Apvrille, L., Courtiat, J-P., Lohr, C., Saqui-Sannes, P. TURTLE: A Real-Time UML Profile Supported by a Formal Validation Toolkit. *IEEE Trans. Software Eng.* 30(7): 473-487 (2004)

[10] Barley, M., Clark, P., Williamson, K., Woods, S. The Neutral Representation Project. *Proceeding of AAAI'97 Spring Symposium on Ontological Engineering*, AAAI Press, 1997.

[11] Borst, W., Akkermans, J., Top, J. Engineering Ontologies. *International Journal of Human-Computer Studies*, 46, pp. 365–406, 1997

[12] Borgida, A., Brachman, R.J., McGuiness, D.L. and Resnick, L.A. CLASSIC: A Structural Data Model for Objects. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 58-67, Portland, OR, 1989.

[13] Baker, G., Brass, A., Bechhofer, S., Goble, C., Paton, N., Stevens, R. TAMBIS: Transparent Access to Multiple Bioinformatics Information Sources. In: Glasgow, J., Littlejohn, T., Major, F., Lathrop, R., Saukoff, D., Sensen, S. (eds): *In the 6th International Conference on Intelligent Systems for Molecular Biology*, AAAI Press, Menlo Park, Montreal, Canada, pp. 25-34, 1998.

[14] Bada, M., Stevens, R., Goble, C., Gil, Y., Ashburner, M., Blake, J. A., Cherry, M., Harris, M., Lewis, S. A Short Study on the Success of Gene Ontology. *Web Semantics: Science, Services and Agents on the World Wide Web*, Vol. 1, pp. 235–240, 2004

[15] Berners-Lee, T. *Weaving the Web.* Orion Business Books, London, 1999.

[16] Berners-Lee, T., Hendler, J., and Lassila, O. The Semantic Web. *Scientific American*, 284(5):34-43, 2001

[17] Bechhofer, S., Van Harmelen, F., Hendler, J., Horrocks, I., McGuiness, D.L. Patel-Schneider, P.F. and Stein, L.A. OWL Web Ontology Language Reference. W3C Recommendation, http://www.w3.org/TR/2004/REC-owl-ref-20040210/, 2004

[18] Baclawski, K., Kokar, M., Kogut, P., Hart, L., Smith, J., Holmes, W., Letkowski, J., Aronson, M. Extending UML to Support Ontology Engineering for the Semantic Web. *UML 2001*, Toronto, CA, Oct 2001.

[19] Baclawski, K., Kokar, M., Kogut, P., Hart, L., Smith, J., Holmes, W., Letkowski, J., Aronson, M., Emery, P. Extending the UML for Ontology Development. *International Journal of Sofware and Systems Modeling* ,Vol 1. No. 2, pp. 142-156, 2002.

[20] Bhatt, M. and Taniar, D. A Distributed Approach to Sub-Ontology Extraction. *Proceedings of the 18th International Conference on Advanced Information Networking and Application (AINA'04)*, IEEE, 2004.

[21] Birmingham, W., Durfee, E., Mullen, T., Wellman, M. The Distributed Agent Architecture of the University of Michigan Digital Library (UMDL). *AAAI Spring Symposium Series on Software Agents*, 1995.

[22] Brickley, D., Guha, R.V. and McBride, B. RDFS Resource Description Framework Schema. W3C Recommendation, http://www.w3.org/TR/2004/REC-rdf-schema-20040210/, 2004

[23] Bishr, Y & Kuhn, W. Ontology-Based Modelling of Geospatial Information, *Proceeding of 3rd AGILE Conference on Geographic Information Science*, Helsinki, Finland, May 25-27, 2000

[24] Booch, G., Jacobson, I. And Rumbaugh, J. *the Unified Modeling Language, Notation Guide Version 1.1.* September 1997.

[25] Chang, C.K. (eds). *Handbook of Software Engineering and Knowledge Engineering.* World Scientific Publishing Co., 2002.

[26] Conallen, J. *Building Web Applications With UML.* Addison-Wesley, USA, 2000

[27] Chung, H., Choi, J., Yi, J., Han, J., Lee, E. A Construction of the Adapted Ontology Server in EC, *Lecture Notes in Computer Science*, Vol. 1983, pp. 355-360, 2000.

[28] Colomb, R.M., Raymond, K., Hart, L., Emery, P., Chang, D., Kendall, E. and Frankel, D. Draft Version 2.0: The Object Management Group Ontology Definition Metamodel, 2004

[29] Corcho, O., Gemez-Perez, A. Solving Integration Problems of e-Commerce Standards and Initiatives through Ontological Mappings. *Proceedings of IJCAI'01*, 2001.

[30] Cranfield, S., Haustein, S., Purvis, M. UML-Based Ontology Modeling for Software Agents, *Proceeding of the Workshop on Ontologies in Agents Systems, 5th International Conference on Autonomous Agents*, Montreal, pp. 21-28, 2001

[31] Cranfield, S. and Purvis, M. UML as an Ontology Modeling Language, *In Workshop on Intelligent Information Integration, 16th International Joint Conference on Artificial Intelligence, IJCAI-99,* 1999

[32] Chi, Y.L., Hsu, T.Y., Yang, W.P. Ontological Techniques for Reuse and Sharing Knowledge in Digital Museums. *Journal of Electronic Library,* Vol. 24, No. 20, pp. 147-159, 2006.

[33] Cui, Z. and O'Brien, P. Domain Ontology Management Environment, *Proceedings of the 33rd Hawaii International Conference on System Sciences*, IEEE, 2000

[34] Carlson, D. *Modeling XML Applications with UML: Practical e-Business Applications.* Addison-Wesley, USA, 2001.

[35] DAML+OIL Specification. www.daml.org/2001/03/daml+oil-index.html, 30 December 2001

[36] Ding, Y., Fensel, D. Ontology Library Systems: the Key to Successful Ontology Reuse. *1st Semantic Web Symposium (SWWS1),* Stanford, USA, July 29-August 1 2001.

[37] Douglass, B.P. *Real-Time UML: Developing Efficient Objects for Embedded Systems.* Addison-Wesley, USA, 2001.

[38] Derridder, D., Wouters, B. The use of an Ontology to Support a Coupling between Software Models and Implementation. *European Conference on Object-Oriented Programming (ECOOP'00), International Workshop on Model Engineering,* 2000

[39] Dameron, O., Noy, N.F., Knublauch, H. and Musen, M.A. Accessing and Manipulating Ontologies Using Web Services. *Workshop on Semantic Web Services: Preparing to Meet the World of Business Applications at the Third International Conference on the Semantic Web (ISWC-2004),* Hiroshima, Japan, 2004

[40] Djuric, D., Gasevic, D. and Devedzic, V. Ontology Modeling and MDA. *Journal of Object Technology,* Vol 4, No 1, pp. 109-128, January-February 2005

[41] Djuric, D., Gasevic, D., Devedzic, V., Damjanovic, V. A UML Profile for OWL Ontologies. *Workshop on Model Driven Architecture: Foundations and Applications,* Linköping University, Sweden, 2004.

[42] Devedzic, V. Understanding Ontological Engineering. *Communications of the ACM,* Volume 45, No.4, pp. 136-144, April 2002.

[43] Das, A., Wu, W. and McGuiness, D.L., Industrial Strength Ontology Management, *Proceedings of the Semantic Web Working Symposium*, Standard, CA, 2001

[44] Eklund, P., Roberts, N., Gree, S. OntoRama: Browsing RDF Ontologies using a Hyperbolic-style Browser. *Proceedings of Proceedings of the First International Symposium on Cyber Worlds (CW.02),* IEEE Computer Society, 2002.

[45] Ells, R. Hypermedia, Help and How-To, *Proceedings of the 16th annual ACM SIGUCCS Conference on User Services*, pp.363-368, Long Beach, California, United States, October 30-November 02, 1988.

[46] Evernden, R. The Information Framework, *IBM Systems Journal*, v.35 n.1, pp.37-68, January 1996

[47] Evernden, R,. Evernden, E. Third-generation information architecture, *Communications of the ACM*, v.46 n.3, p.95-98, March 2003

[48] Estival, D., Nowak, C., Zschorn, A. Towards Ontology-Based Natural Language Processing. *RDF/RDFS and OWL in Language Technology: 4th Workshop on NLP and XML (NLPXML-2004), ACL 2004*, Barcelona, Spain, 2004.

[49] Fensel, D. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Management.* Spinger, Berlin, 2001

[50] Fensel, D., Van Harmelen, F., Horrocks, I., McGuiness, D.L. and Patel-Schneider, P.F. OIL: Ontology Infrastructure to Enable the Semantic Web. *IEEE Intelligent Systems,* 16(2), pp. 38-45, 2001

[51] Falkovych, K., Sabou, M. and Stuckenschmidt, H. *UML for the Semantic Web: Transformation-Based Approaches.* In Omelayenko, B. and Klein, M. eds. Knowledge Transformation for the Semantic Web, Frontiers in Artificial Intelligence and Applications, Vol. 95, IOS Press, pp. 92-106, 2003.

[52] Fonseca, F., Egenhofer, M. Ontology-Driven Geographic Information Systems. *In the 7th ACM Symposium on Advances in Geographic Information Systems,* Kansas City, MO: ACM Press, N.Y, 1999.

[53] Frank, A. *Spatial Ontology: A Geographical Point of View.* In: Stock, O. (eds): Spatial and Temporal Reasoning, Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 135-153, 1997.

[54] Falasconi, S., Lanzola, G., Stefanelli, M. Using Ontologies in Multi-Agent Systems. *Proceedings of 10th Knowledge Acquisition for Knowledge-Based Systems Workshop (KAW'96),* 1996.

[55] Farquhar, A., Fikes, R. and Rice, J. The Ontolingua Server: A Tool for Collaborative Ontology Construction. *International Journal of Human-Computer Studies,* 46:707-728, 1997

[56] Frehiwot, F., Hagedorn, K., Keizer, J. and Stephen, K. Creating the Semantic Web: the Role of an Agricultural Ontology Server (AOS). *International Semantic Web Workshop (Infrastructure and Applications for the Semantic Web),* 2001.

[57] Foundation for Intelligent Physical Agents (FIPA), FIPA Ontology Service Specification,

http://www.fipa.org/specs/fipa00086/index.html, Specification of 2001, (12 June 2006)

[58] Gruber T.R. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2): 199-220, 1993

[59] Gomaa, H. *Designing Concurrent, Distributed, and Real-Time Applications with UML*. Addison-Wesley, USA, 2000.

[60] Guarino, N. Formal Ontology in Informations Systems. *Proceeding of the 1st International Conference on Formal Ontology in Information Systems (FOIS'98)*, Trento, Italy, 6-8 June 1998

[61] Guarino, N., Masolo, C., Vetere, G. OntoSeek: Content-Based Access to the Web. *IEEE Intelligent Systems*, pp. 70-80, June 1999.

[62] Guarino, N. Ontology-Driven Conceptual Modelling. *Tutorial at 21st International Conference on Conceptual Modeling (ER'02)*, Tempere, Finland, 2002.

[63] Guarino, N., Persidis, A. *Evaluation Framework for Content Standards*. Deliverable 3.5, OntoWeb EU Project (IST-2000-29243), 2003.

[64] Glaser, H., Alani, H., Carr, L., Chapman, S., Ciravegna, F., Dingli, A., Gibbins, N., Harris, S., Schraefel, M.L., Shadbolt, N. *CSAKTive Space: Building a Semantic Web*. In: Bussler, C., Davies, J., Fensel, D. and Struder, R. (Eds). 1st European Web Symposium (ESWS'04), Springer-Verlag, pp. 417-432, 2004.

[65] Gasevic, D.V., Djuric, D.O. and Devedzic, V.B. Bridging MDA and OWL Ontologies. *Journal of Web Engineering*, Vol. 4, No. 2, pp. 18-143, 2005

[66] Gasevic, D., Djuric, D, Devedzic, V., Damjanovic, V. UML for Read-To-Use OWL Ontologies. *Proceedings of the 2ndIEEE International Conference Intelligent Systems*, Vol. II, Varna, Bulgaria, pp. 485-490, 2004.

[67] Guizzardi, G., Wagner, G. and Herre, H. On the Foundations of UML as an Ontology Representation Language. *EKAW 2004*, pp. 47-62, 2004.

[68] Guizzardi, G. *Ontological Foundations For Structural Conceptual Models*. Telematica Instituut, Netherlands, 2005.

[69] Hart, L., Emery, P., Colomb, R. M., Raymond, K., Chang, D., Ye, Y., Kendall, E. and Dutra, M. Usage Scenarios and Goals for Ontology Definition Metamodel. *Lecture Notes in Computer Science*, Vol. 3306, pp 596, Springer Berlin, 2004

[70] Haase, P., Völker, J., Sure, Y. Management of Dynamic Knowledge. *Journal of Knowledge Management*, Vol. 9, No. 5, pp. 97-107, October 2005.

[71] Harrison, R and Chan, C.W. Distributed Ontology Management System, *Proceedings of the 18th Annual Canadian Conference on Electrical and Computer Engineering, (CCECE'05)*, IEEE, Saskatoon, 2005.

[72] Hennicker, R and Koch, N. Systematic Design of Web Applications with UML. In Keng Siau and Terry Halpin, editors, *Unified Modeling Language: Systems Analysis, Design and Development Issues*, IDEA Group Publishing, 1-20, 2001.

[73] Inmon, W.H., Zachman, J.A., Geiger, J.G. *Data Stores, Data Warehousing, and the Zachman Framework: Managing Enterprise Knowledge*. McGraw-Hill, USA, 1997.

[74] Jannik, J., Srinivasan, P., Verheijen, D. and Wiederhoid, G. Encapsulation and Composition of Ontologies. *Proceedings of AAAI Summer Conference*, 1998.

[75] Jong, G. A UML-Based Design Methodology for Real Time and Embedded Systems. *Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE.02)*, IEEE Computer Society, 2002.

[76] Jasper, R. and Uschold, M. A Framework For Understanding and Classifying Ontology Applications. *Proceeding of the IJCAI-99 Workshop on Ontologies and Problem-Solving Method (KRR5)*, Stockholm, Sweeden, August 1999.

[77] Jarrar, M and Meersman, R. Formal Ontology Engineering in the DOGMA Approach, *Proceeding of ODBASE'02*, LNCS 2519, pp. 1238-1254, 2002

[78] Jarrar, M. *PhD Thesis: Towards Methodological Principles for Ontology Engineering*. Faculty of Science, Vrije Universiteit Brussel, 2005.

[79] Jarrar M., Verlinden R. and Meersman R. *Ontology-Based Consumer Complaint Management*. In Jarrar J., Salaun A., (eds): Proceedings of the Workshop on Regulatory Ontologies and the Modeling of Complaint Regulations (WORM CoRe 2003), Springer LNCS, Sicily, Italy. November 2003

[80] Jones, S. Toward an Acceptable Definition of Service, *IEEE Software*, v.22 n.3, pp.87-93, May 2005

[81] Jarzabek, S. Modeling Multiple Domains in Software Reuse. *ACM SIGSOFT Software Engineering Notes*, v.22 n.3, p.65-74, May 1997

[82] Khosla, R. Multi-Layered Distributed Agent Ontology for Soft Computing Systems. *Proceedings of the 17th International Conference of Knowledge-Based Intelligent Information and Engineering Systems*, Oxford, UK, pp. 445-452, Sept 2003.

[83] Kang, D., Xu, B., Xu., Lu1, J., C.Chu, W. A Complexity Measure for Ontology Based on UML. *Proceedings of the 10th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS'04)*, 2004.

[84] Karp, P.D., Chaundhri, V.K and Thomere, J. XOL: An XML-based Ontology Exchange Language. Version 0.3, July 3, 1999

[85] Karp, P.D., Riley, M., Saier, M., Paulsen, I.T., Paley, S., Pellegrini-Toole, A. The Ecocyc Database. *Nucleic Acids Research*, 30(1):56, 2002.

[86] Klyne, G., Carroll, J.J and McBride, B. RDF Resource Description Framework Reference. W3C Recommendation, http://www.w3.org/RDF/, 2004

[87] Kogut, P., Cranfield, S., Hart, L., Dutra, M., Baclawski, K., Kokar, M. and Smith, J. UML for Ontology Development. *The Knowledge Engineering Review*, 17(1), 2002, 61-64

[88] Koch, N., Baumeister, H., Hennicker, R. and Mandel, L. Extending UML for Modeling Navigation and Presentation in Web Applications. In Geri Winters and Jason Winters, editors, *In Modeling Web Applications in the UML Workshop, UML2000*, York, England, October 2000.

[89] Klein, M., Fensel, D., Van Harmelan, F., Horrocks, I. The Relation between Ontologies and Schema Languages: Translating OIL-Specifications to XML schema In: *Proceedings of the Workshop on Applications of Ontologies and Problem-Solving Methods, 14th European Conference on Artificial Intelligence ECAI-00,* Berlin, Germany, August 20-25, 2000.

[90] Liu, W., Liu, Z-T., Shao, K. UML-Based Domain Ontology Modeling For Multi-Agent System. *Proceedings of the Second International Conference on Machine Learning and Cybernetics*, Wan, 2-5 November 2003

[91] Lenat, D.B. CYC: A Large-scale investment in knowledge infrastructure. Communication of ACM, Vol. 38, No. 11, pp. 33-38, November 1995.

[92] Li, Y., Thompson, S., Tan, Z., Giles, N. and Gharib, H. Beyond Ontology Construction: Ontology Services as Online Knowledge Sharing Communities. *Lecture Notes in Computer Science*, Volume 2870, pp. 469 – 483, Springer-Verlag GmbH, 2003

[93] Lopez, M.F., Gomez-Perez, A., Sierra, J.P., Sierra, A.P. Building a Chemical Ontology using Methontology and the Ontology Design Environment. *IEEE Intelligent Systems*, Vol. 14, No. 1, pp. 37-46, Jan-Feb 1999

[94] Lu, S., Halang, W.A., Zhang, L. A Component-Based UML Profile to Model Embedded Real-Time Systems Designed by the MDA Approach. *RTCSA'05*, pp. 563-566, 2005

[95] Marcos, C. and Pryor, J. UML Profile for Agent-Oriented Software Engineering. En co-autoria con Jane Pryor. *In Proceedings of the Fourth Argentine Symposium on Artificial Inteligence (ASAI´2005),* 32 JAIIO (Jornadas Argentinas de Informática e Investigación Operativa), pp. 1-11, Rosario Agosto 2005.

[96] Mika, P. Social Networks and the Semantic Web. *International Conference on Web Intelligence (WI'04), IEEE Computer Society*, Beijing, China, pp. 285-291, 2004

[97] Muller, R.J. *Database Design for Smarties: Using UML for Data Modeling (The Morgan Kaufmann Series in Data Management Systems).* Morgan Kaufmann Publisher, USA, 1999.

[98] Mizoguchi, R. Tutorial on Ontological Engineering: Part 2: Ontology Development, Tools and Languages. *New Generation Comput.* 22(1): 2003

[99] Musen, M. Domain Ontologies in Software Engineering: Use of the Protégé with the EON Architecture Methods of Information in Medicine. No 37, pp. 540-550, 1998.

[100] Mauger, B. *Undergraduate Thesis: Tools to Help an Agent Commit to an Ontology*. Computer Science Department, University of Queensland, 2005

[101] Mohd Syazwan Abdullah, Evans, A., Benest, I., Kimble, C. Developing UML Profile for Modelling Knowledge-Based Systems, *Model Driven Architecture®: Foundations and Applications (MDAFA 2004),* Linköping University, Sweden, June 10-11, 2004.

[102] McGuinness, D.L., Fikes, R., Rice, J. and Wilder, S. The Chimaera Ontology Environment. *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI 2000),* Austin, TX, USA, July 30- August 3, 2000.

[103] Maedche, A., Motik, B., Stojanovic, L. Managing Multiple and Distributed Ontologies on the Semantic Web. *The VLDB Journal*, Vol. 12, Issue 4, pp. 286-302, 2003

[104] MySQL. http://www.mysql.org (12 June 2006)

[105] Noy, N.F. and Musen, M.A. Ontology Versioning in an Ontology-Management Framework**.** *IEEE Intelligent Systems*, 19(4): 6-13, 2004

[106] Noy, N.F. and Musen, M. PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000),* Austin, TX, AAAI/MIT Press, 2000

[107] O'Rourke, C., Fishman, N., Selkow, W. *Enterprise Architecture Using the Zachman Framework.* Course Technology, USA, 2003

[108] Oberle, D., Volz, R., Motik, B., Staab, S. *KAON Server Prototype.* Technical Report, IST Project 2001-33052 WonderWeb, Department of Computer Science, The Victoria University of Manchester, 2002.

[109] Pan, J., Cranefield, S. and Carter, D. A Lightweight Ontology Repository. *AAMAS'03,* ACM, Melbourne Australia, 2003

[110] Pisanelli, D. M., Gangemi, A., Steve, G. Ontologies and Information Systems: the Marriage of the Century? *In the International Workshop on Governmental Methodology for Software Providence (LYEE02)*, 2002.

[111] Pretorius, A.J. Visual Analysis for Ontology Engineering, *Journal of Visual Languages and Computing*, Vol. 16, pp. 359-381, 2005

[112] Roberto, C., Smith, B., Varzi, A. *Ontological Tools for Geographic Representation.* In: N.

Guarino (eds): Formal Ontology In Information Systems, In the Proceedings of the 1st International Conference (FOIS'98), Amsterdam IOS Press, Trento, Italy, pp. 77-85, 1998.

[113] Rothenburger, B. and Galarreta, D. Facing Knowledge Evolution in Space Project: a Multi-Viewpoint Approach. *Journal of Knowledge Management*, Vol. 10, No.2, pp. 52-65, 2006.

[114] [114], D. and Paquette, G. Managing Ontology Changes on the Semantic Web. *Proceedings of the International Conference on Web Intelligence (WI'05)*, IEEE, 2005

[115] Shum, S., Motta, E., Domingue, J. SchoolOnto: an Ontology-Based Digital Library Server for Research Document and Discourse. *International Journal on Digital Libraries*, 3(3), pp. 237-248, 2000.

[116] Sampson, D., Karagiannidis, C., Cardinali, F. An Architecture for Web-Based e-Learning Promoting Re-usable Adaptive Educational e-Content, Educational Technology & Society *Journal of International Forum of Educational Technology & Society and IEEE Computer Society Learning Technology Task Force*, ISSN 1436-4522, Special Issue on Innovations in Learning Technologies, 5(4), August 2002.

[117] Silva, V.T., Choren, R. and Lucena, C.J.P. A UML Based Approach for Modeling and Implementing Multi-Agent Systems. *AAMAS'04*, July 19-23, 2004, ACM, New York, USA.

[118] Sure, Y., Erdmann, M., Angele, J., Staab, S., Studer, R. and Wenke, D. OntoEdit: Collaborative Ontology Engineering for the Semantic Web. In I. Horrock and J.Hendler (eds.), The Semantic Web – ISWC 2002, *Proceedings of the 1st International Semantic Web Conference*, Sardinia, Italy, June 9-12, 2002, Lecture Notes in Computer Science (LNCS 2342), Springer, 2002.

[119] Swartout, B., Patil, R., Knight, K. and Russ, T. Towards Distributed use of large-scale ontologies. In B.R. Gaines and M.A. Musen (eds), *Proceedings of the 10th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff, Canada, 1996.

[120] Spyns, P., Meersman, R. and Jarrar, M. Data Modeling versus Ontology Engineering, *SIGMOD Record*, Vol 31, No 4, December 2002

[121] Sekiuchi, R., Aoki, C., Kurematsu, M., Yamaguchi, T. DODDLE: A Domain Ontology Rapid Development Environment. *Proceedings of PRICAI'98*, 1998

[122] Tamma, V., Bench-Capton, T. A Conceptual Model to Facilitate Knowledge Sharing in Multi-Agent Systems. *Proceedings of the Autonomous Agents 2001 Workshop on Ontologies in Agent Systems (OAS'01)*, Montreal, pp. 69-76, May 2001.

[123] Uitermark, H. *PhD Thesis: Ontology Based Geographic Data Set Integration*, Twente University, 2001.

[124] Verbert, K., Klerkx, J., Meire, M., Najjar, J., Duval, E. Towards a Global Component Architecture for Learning Objects: An Ontology Based Approach. *Proceedings of on the Move to Meaningful Internet Systems 2004, OTM 2004, Workshops*, LNCS 3292, Cyprus, pp. 713-722, 2004.

[125] Valo, A. and Hyvönen, A and Komulainen, V. Tool for Collaborative Ontology Development for the Semantic Web. *Proceedings of International Conference on Dublin Core and Metadata Applications (DC 2005)*, Nov, 2005.

[126] Valente, A., Russ, T., MacGregor, R., Swartout, W. Building and Reusing an Ontology of Air Campaign Planning. *IEEE Intelligent Systems*, Vol. 14, No. 1, pp. 27-36, Jan-Feb 1999

[127] V.U.B. Star Lab. Ontology Server Research. http://www.starlab.vub.ac.be/research/dogma/OntologyServer.htm (10 June 2006)

[128] Waterson, A. and Preece, A. Verifying Ontological Commitment in Knowledge-Based Systems. *Knowledge-Based Systems*, 12, pp. 45-54, 1999

[129] Weinstein, P.C. Ontology-Based Metadata: Transforming the MARC Legacy. *ACM Digital Libraries*, Pittsburgh, USA, 1998.

[130] Welty, C. Ontology-Driven Conceptual Modeling. Invited Talk at the *14th international Conference on Advanced Information Software Engineering (CAiSE)*, Toronto, Canada, 2002.

[131] Welty, C., Ferrucci, D. A Formal Ontology for Reuse of Software Architecture Documents. *Proceedings of the International Conference on Automated Software Engineering*, IEEE Computer Society Press, pp. 259-262, October 1999.

[132] Wand, Y and Weber, R. An Ontological Analysis of the Relationship Construct in Conceptual Modeling, *ACM Transactions on Database Systems*, Vol. 24, No. 4, pp. 494-528, December 1999.

[133] Whitten, J.L., Bentley, L.D., Dittman, K. *Analysis and Design Method.* McGrawHill, USA, 2004.

[134] Yang, X. *Ontologies and How to Build Them.* Technical Report, Department of Computer Science, University of California, 2001.

[135] Zachman, J.A. (1987). A Framework for Information Systems Architecture. *IBM Systems Journal*, (26:3), pp. 276-292.

[136] Zhang, B., Chen, Y. Enhancing UML Conceptual Modeling through the Use of Virtual Reality. *Proceedings of the 38th Hawaii International Conference on System Sciences*, Hawaii, 2005.

[137] Zhang, C., Tamma, V A M. and Bellifemine, F. Ontology Management in Enterprises, *BT Technology Journal*, Vol. 17, No. 4, 1999

[138] Zhao, G., Zheng, J. and Meersman, R. An Architecture Framework of Ontology Development and Deployment. *Proceedings of the E-Society IADIS International Conference*, 2003.

Appendix

Table 1. A Comparison of Ontology Servers

Servers / **General Features**	**Ontolingua [55]**	**ACOS [92]**	**Starlab** [127]	**KAON** [108]	**OntoRama** [44]	**OWS** [39]	**FIPA Ontology Service Specification** [57]	**Adapted Ontology Server** [27]
Main Goal	Focus on Collaborative Ontology Development	Focus on Collaborative Ontology Development	Focus on Ontology Development	Focus on Ontology Development	Focus on Storing, Browsing Ontology. Limited Support to Ontology Development	Focus on Access and Manipulating Ontologies	Focus on Ontology Development and Ontology-Based Agent Communication	Focus on Ontology Development Specifically in Electronic Commerce
Ontology Representation Languages	Ontolingua and KIF	DAML+OIL	XML	RDF	RDF	Open (unspecified)	Open (unspecified)	Not Mentioned
Knowledge Representation	Complex	Simple	Medium	Complex	Medium	Open (unspecified)	Complex	Not Mentioned
Software Architecture	Client-Server	Client-Server	Client-Server	Client-Server	Client-Server	Client-Server	Client-Server	Client-Server
Application/User Remote Support to Ontologies	OKBC/HTTP	HTTP	JDBC-ODBC	HTTP	HTTP	HTTP	OKBC	JDBC-ODBC
Main Usage of Ontology Server	Design Time	Design Time	Design Time	Design Time	Design Time	Runtime	Design Time, Runtime	Design Time
Ontology Repository Platform	Ontolingua-Based Repository	Jena + DAML + OIL	RDBMS	RDBMS	Jena + RDF	Open (unspecified)	Open (unspecified)	RDBMS
Main Deliverables	Server Functions Mainly for Supporting Ontology Development	Server Functions Mainly for Supporting Ontology Development	Server Functions Mainly for Supporting Ontology Development	Server Functions Mainly for Supporting Ontology Development	Server Functions Mainly for Supporting Ontology Development	Server Functions Mainly for Accessing and Manipulating Ontologies	Server functions Mainly for Ontology Development and Agent Communication	Server Functions Mainly for Ontology Development

Server Functions Implementation	APIs + Tool Development	APIs	APIs + Tool Development	APIs	Tool Development	Semantic Web Services	Tool Development	APIs
Using Standard Methodological Development	No	No	No	No	No	No	No	No
Management Provision	Central	Community	Central	Central	Central	Central – (tasks)	Not Available	Central
Benefit from Graphical Language	No	No	Yes – ORM	No	No	No	No	No
To Support the Semantic Web Vision or Web	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Import Mechanism	Yes	Yes	No	Yes	No	No	Yes	No
Manner of Consistency Checking	Human approval, simple data type checking	Auto	Not Mentioned	Yes	Not Mentioned	Partly	Open (unspecified)	No
Ontology Representation Consider to Supporting Information Systems Interoperating	Not Mentioned	Yes	Yes	Yes	Not Mentioned	Yes	Not Mentioned	Not Mentioned
Main Ontology Engineering Challenging Tasks	Reusability	Reusability, Usability	Reusability, Usability	Reusability	Viewing	Usability	Reusability	Extracting
Ontology Server Standard Design Artifacts	No	No	No	No	No	No	No	No