Ontologies for Semantically Interoperable Systems

Leo Obrst
The MITRE Corporation
7515 Colshire Drive, M/S H305
McLean, VA 22102-7508
1-703-883-6770
lobrst@mitre.org

ABSTRACT

In this paper, we discuss the use of ontologies for semantic interoperability and integration. We argue that information technology has evolved into a world of largely loosely coupled systems and as such, needs increasingly more explicit, machine-interpretable semantics. Ontologies in the form of logical domain theories and their knowledge bases offer the richest representations of machine-interpretable semantics for systems and databases in the loosely coupled world, thus ensuring greater semantic interoperability and integration. Finally, we discuss how ontologies support semantic interoperability in the real, commercial and governmental world.

Categories & Subject Descriptors: I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods – *Predicate Logic, Representation Languages, Representations (procedural and rule-based), Semantic networks.* D.1.12 [**Software Engineering**]: Interoperability – *data mapping.*

General Terms: Management, Performance, Design, Standardization, Languages, Theory.

Keywords: Ontologies, ontological engineering, semantic interoperability, semantic integration.

1. INTRODUCTION

We increasingly live and interoperate in a loosely coupled information world. The evolution from tightly coupled systems (and databases) has been inexorable, as information technology has grown from focusing on local systems and databases to encompassing the more global interaction and integration of multiple systems in enterprises and communities. Necessarily correlated with that evolution to loosely coupled systems is the need for increasingly more explicit, machine-interpretable semantics. Ontologies in the form of logical domain theories and their knowledge bases offer the richest representations of machine-interpretable semantics for systems and databases in the loosely coupled world.

This paper discusses the use of ontologies for semantic interoperability and integration. Because the notion of interoperability usually is used in the context of systems and the notion of integration is used in the context of data, we consider both concepts with respect to the use of ontologies. In addition, we discuss how ontologies support semantic interoperability in the real, commercial and governmental world.

CIKM'03, November 3–8, 2003, New Orleans, Louisiana, USA.

2. ONTOLOGIES

An ontology defines the common words and concepts (meanings) used to describe and represent an area of knowledge. Ontologies are used by people, databases, and applications that need to share domain information (a *domain* is just a specific subject area or area of knowledge, like medicine, counterterrorism, imagery, automobile repair, etc.) Ontologies include computer-usable definitions of basic concepts in the domain and the relationships among them. Ontologies encode knowledge in a domain and also knowledge that spans domains. So, they make that knowledge reusable.[1]

An ontology includes the following kinds of concepts: *classes* (general things) in the many domains of interest; *instances* (particular things); the *relationships* among those things; the *properties* (and property values) of those things; the *functions* of and *processes* involving those things; *constraints* on and *rules* involving those things.

Ontologies are usually expressed in a logic-based language, so that fine, accurate, consistent, sound, and meaningful distinctions can be made among the classes, instances, properties, attributes, and relations. Some ontology tools can perform automated reasoning using the ontologies, and thus provide advanced services to intelligent applications such as: conceptual/semantic search and retrieval (non-keyword based), software agents, decision support, speech and natural language understanding, knowledge management, intelligent databases, and electronic commerce. Because what are known as ontologies include a range of models of varying degree of semantic richness and complexity, we've created an *Ontology Spectrum*, a framework relating the various information models in terms of increasing semantic richness.[2]

---

[1] See [1], section 1.1. This section of [1] represents an adaptation of our original definition, as given in this paper (these first three paragraphs of Section 2), and one of our contributions to the OWL set of documents.

[2] See [2], p. 156-157, and also [3]. See also [4] for an independently created but comparable view of the ontology spectrum. A third view is given in [10], p. v.

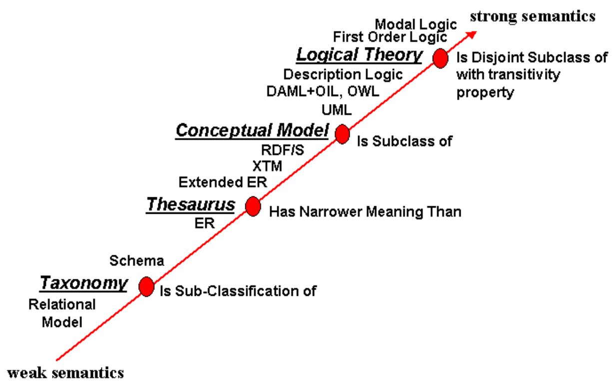

Figure 1. The Ontology Spectrum

As one moves up the spectrum from lower left to upper right, the semantic richness increases. We characterize the poles of the spectrum as "weak semantics" and "strong semantics", i.e., the richness of the expressible or characterizable semantics increases from weak to strong. At the "weaker" side, one can only express very simple meaning; at the "stronger" side, one can express arbitrarily complex meaning.

For the loosely coupled world, in order for the semantics to be both explicit and machine-interpretable (rather than simply machine-processable), the model formalism in which the domain semantics is represented must be based on a logic, so that the machine can make valid inferences and enforce sound semantic constraints. Hence, the domain model is a *logical theory*, the highest and rightmost region in the ontology spectrum.

Ontologies and knowledge bases are distinguished by their types of assertions. Ontologies assert generic or class-level assertions, about entities, their properties, and relations. Knowledge bases are instance or equivalently fact bases, and assert instance-level assertions based on class-level properties and relations inherited from the ontological assertions. At the ontological level, a Person Lives at a Location Having an Address. At the knowledge base level, John Quincy Public is an instance of a Person and Lives at a Cape Cod-style House on 223 Lincoln Blvd., Nachahasset, MA.

3. LOOSELY COUPLED SYSTEMS

In general, information technology has evolved from the view and reality of systems and data as locally realized, i.e., as tightly coupled through some predefined API exemplified by the software procedure call. The software procedure call typically made explicit only the types and values of the parameterized arguments passed to the procedure and the type of the expected result. The lower portion of Figure 2 illustrates this case, a simple notion of *composition*, i.e., a tightly coupled connection between systems.

Figure 2 depicts integration/composition (tight to loose coupling) as it has and is evolving over time, towards greater complexity.

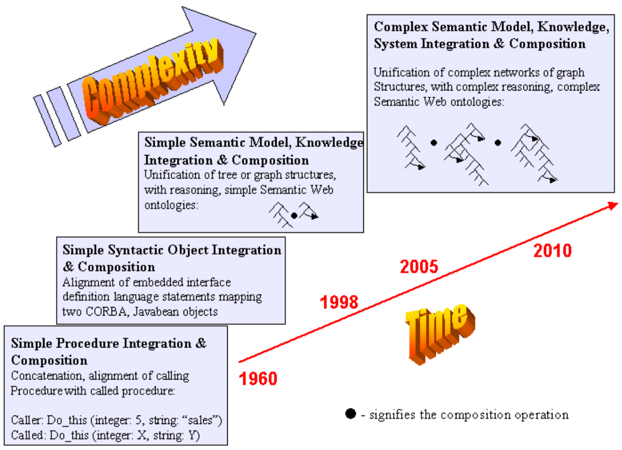

Figure 2. Evolution of Integration/Composition

There has been evolution. From tightly coupled systems (and databases), information technology has evolved from a focus on local systems and databases to more global interaction and integration in enterprises and communities. Concomitantly, the need for increasingly more explicit, machine-interpretable semantics has increased. No longer can a small group of developers nod their heads in agreement over the implicit semantics of their two databases and three systems that use those databases. There are many databases and many systems, systems of systems, cross-enterprise and cross-community integration, the entire Internet. Implicit semantics in a necessarily loosely coupled world (many programming languages, many CPUs, many operating systems, many networks, etc.) does not work. Three billion users of the Internet cannot all nod their heads in agreement over the implicit semantics of all their databases, systems, enterprises, and communities. Increasingly more explicit semantics is needed.

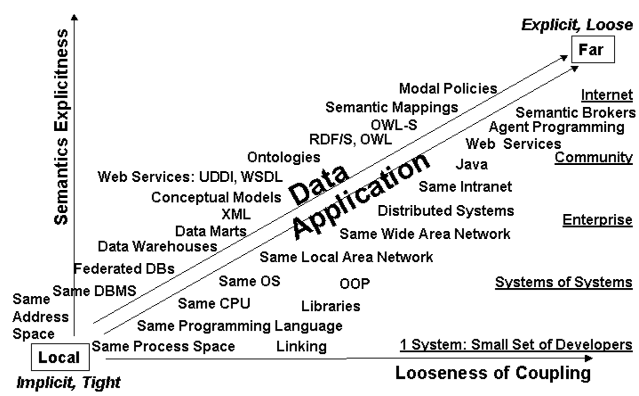

Figure 3. Looseness of Coupling and Semantic Explicitness

Figure 3 depicts the correlation between increased looseness of coupling and the increased explicitness of semantics necessary, from both the data and application perspectives. On the far right of the figure (underlined words) are the way stations of information technology, from bottom to top: system, systems of systems, enterprise, community, and Internet. Above the double diagonal arrow in the center of the figure are the data constructs that have evolved to adapt to the increasingly loose-coupled world; below the double arrow are the corresponding application constructs. As the verticality increases, the need for explicit semantics increases. In the past we could count on implicit semantics in the form of informal agreements among developers and users because our software and data foci were local: possibly the same operating system, the same CPU, the same programming language, even the same address and process space. But in the new loosely-coupled world, we cannot. We need increasingly more explicit semantics, so that machines can assist us with our semantic interoperability and integration requirements.

4. SEMANTIC INTEROPERABILITY AND INTEGRATION

Although the notion of *interoperability* usually is used in the context of systems and the notion of *integration* is used in the context of data, we will distinguish these in slightly different terms. Data, because it is typically considered in terms of its representation and usually its declarative representation, can be considered as having the properties of that representation, i.e., it can be considered syntactically, structurally, and semantically.[3] So *integration* is describable in terms of the degree of syntactic, structural, and semantic correspondence between two data sources [9].

Interoperability, however, implies interoperability between software constructs, i.e., typically in terms of the aggregative level of software constructs, from data to community levels of aggregation. So, in Figure 3, we display the six levels of interoperability against the three kinds of integration, with the approximate contribution each makes to each level. Our focus lies in semantic interoperability and integration, so, though important, we do not discuss syntactic and structural integration except peripherally.

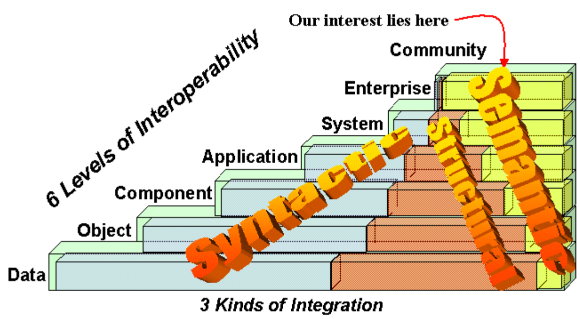

Figure 3. Dimensions of Interoperability and Integration[4]

Increasingly, our software aggregations need to interoperate with more precise semantics, and are forced to do so in a largely heterogeneous environment of loosely coupled systems and distributed or federated databases, data warehouses, and data marts. These latter, in particular, are attempting to push syntactic and structural interoperability to its limits, by continuing to employ the methods of the former tightly coupled world with its implicit semantics. They are meeting with limited success.

5. ONTOLOGIES AND SEMANTIC INTEROPERABILITY

Because ontologies explicitly represent domain semantics, i.e., the entities in a domain, and their properties and relationships in the information world as approximately characterizing their real world equivalents, they are used to span heterogeneously structured databases and multiple systems that nevertheless have comparable semantics. Ontologies support semantic interoperability and integration in both the commercial and governmental worlds. For example, explicit product and service ontologies have been created for Business-to-Business (B2B) applications in multiple market sectors and domains [6, 7, 8]. These electronic commerce companies have used ontologies to map vendor-specific catalogs and databases together semantically, so that buyers have a seamless interface via an application employing the domain ontologies for property-parameterized product search and buying. These same ontologies have been used with some modification for sellers in the categorization of their catalog products and services. The important issue is the preservation of semantics across databases and applications. B2B systems must support buyer product search and catalog navigation, seller product classification, payment processing, order and inventory management, distribution, delivery, planning and forecasting, financing, analysis, and security, all of which processes have distinct application and data requirements but core shared semantics.

Government agencies are also beginning to employ ontologies to support their systems' semantic interoperability at the enterprise and community levels. These efforts largely begin as metadata initiatives, defining enterprise and community standards for the exchange of information, initially based on XML and XML Schema. They may invoke the use of thesauri, as in [11], as an intermediate solution to assist in the mapping of distinct vocabularies. Typically, these metadata efforts will create community-wide metadata registries and repositories to register their schemas, taxonomies, and ontologies (all classification schemes) [12]. Two standards are widely observed: the 11179 Metadata Registries Standards [13], and ebXML [14]. Other efforts are more focused on web services. One new effort that involves B2B electronic commerce and may include the extension of ebXML registries to incorporate ontologies is the open source Ontolog effort [15], which is attempting to build ontologies to support the Universal Business Language (UBL) standard [16].

Another effort is the e-Government Federal Enterprise Architecture initiative [17], currently based on XML technologies but increasingly moving toward the adoption of ontologies and Semantic Web technologies.

6. CONCLUSION

This paper has briefly discussed the use of ontologies for semantic interoperability. The need for the semantic explicitness that ontologies can provide is largely driven by the continuing evolution of information technology into a loosely coupled world, where systems cannot count on homogeneous environments but still need to interoperate with shared semantics and increasing semantic precision.

[3] Data can also be considered pragmatically, by which we mean in terms of its formal pragmatics, i.e., its semantics with regard to the use and intent of the semantics in context. We will not address issues of pragmatics in this paper.

[4] This figure was originally rendered in [5], based on a study of semantic interoperability for objects undertaken for DARPA.

7. ACKNOWLEDGMENTS

I'd like to thank my colleagues Len Seligman, Greg Whittaker, Alex Meng, David Ferrell, Peter Yim, Kurt Conrad, and Mike Uschold for their encouragement, discussions, and past collaboration on some of the core ideas of this paper. The views expressed here are mine alone and do not reflect the official policy or position of The MITRE Corporation or any other company or individual.

8. REFERENCES

[1] OWL Web Ontology Language Use Cases and Requirements. 2003, W3C Candidate Recommendation, August 18, 2003. http://ww1w.w3.org/TR/webont-req/.

[2] Daconta, M., L. Obrst, K. Smith. 2003. The Semantic Web: The Future of XML, Web Services, and Knowledge Management. John Wiley, Inc., June, 2003.

[3] Obrst, L., H. Liu. 2002. Knowledge Representation, Ontological Engineering, and Topic Maps. Chapter in XML Topic Maps: Creating and Using Topic Maps for the Web. Jack Park, ed. Addison-Wesley.

[4] D. L. McGuinness. 2002. Ontologies Come of Age. In Dieter Fensel, Jim Hendler, Henry Lieberman, and Wolfgang Wahlster, editors. Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential. MIT Press, 2002.

[5] Obrst, L., G. Whittaker, A. Meng. 1999. Semantic Context for Object Exchange, AAAI Workshop on Context in AI Applications, Orlando, FL, July 19, 1999.

[6] Obrst, L., H. Liu, R. Wray. 2003. Ontologies for Corporate Web Applications. Artificial Intelligence Magazine, special issue on Ontologies, American Association for Artificial Intelligence, Chris Welty, ed., forthcoming, 2003

[7] Obrst, L., H. Liu, R. Wray, L. Wilson. 2002. Ontologies for Semantically Interoperable Electronic Commerce. In the Proceedings of ICEIMT'02, International Conference on Enterprise Modelling and Enterprise Integration Technologies (ICEIMT), and the Conference of the EI3-IC Initiative (Enterprise Inter- and Intra-Organisational Integration – International Consensus), Valencia, Spain, April 24-26, 2002, Kluwer Academic Publishers, 2002.

[8] Obrst, L., R. Wray, H. Liu. 2001. Ontological Engineering for B2B E-Commerce. In: Formal Ontology in Information Systems: Collected Papers from the Second International Conference, October 17-19, 2001, Ogunquit, ME. Chris Welty, Barry Smith, eds. ACM Publishing, http://www.fois.org/fois-2001/index.html.

[9] Smith, K., L. Obrst. 1999. Unpacking The Semantics of Source and Usage To Achieve Semantic Reconciliation In Large-Scale Information Systems, SIGMOD special issue on Semantic Interoperability, March, 1999, A. Sheth & A. Ouksel, eds.

[10] Smith, B.; C. Welty. 2001. Ontology: Towards a New Synthesis. Introduction to Formal Ontology in Information Systems: Collected Papers from the Second International Conference, October 17-19, Ogunquit, Maine, C. Welty and B. Smith, eds.

[11] Hunter, J. 2001. MetaNet: A Metadata Term Thesaurus to Enable Semantic Interoperability Between Metadata Domains. Journal of Digital information, volume 1, issue 8, Themes: Digital libraries, Information Discovery 2001-02-08.

[12] Department of Defense Metadata Registry. http://diides.ncr.disa.mil/xmlreg/user/index.cfm

[13] ISO/IEC JTC1 SC32 WG2 Development/Maintenance, ISO/IEC 11179, Information Technology – Metadata Registries (MDR). http://metadata-stds.org/11179/.

[14] OASIS ebXML Registry Technical Committee. http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=regrep.

[15] Ontolog UBL Ontology Project. http://ontolog.cim3.net/cgi-bin/wiki.pl?UblOntology.

[16] OASIS Universal Business Language Technical Committee. http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ubl.

[17] Federal Enterprise Architecture. http://www.feapmo.gov/.