IMSC Technical Report 05-002
Working Draft
Ontology-based Semantic Information Management for Seismology and Geoscience

Dennis McLeod
Department of Computer Science
Integrated Media System Center
University of Southern California
Los Angeles, California 90086
mcleod@usc.edu

Sang Soo Sung
Department of Computer Science
Integrated Media System Center
University of Southern California
Los Angeles, California 90086
sangsung@usc.edu

Abstract

Ontologies can play a key role in managing ever expanding earthquake data. Seismology and geoscience domain ontologies have been designed on top of semantic model theory and are written in ontology languages such as RDF, DAML+OIL, and OWL. It turns out that traditional model theory causes syntactical and semantic conflicts with contemporary ontology languages. This paper addresses a new semantic model for ontology languages and introduces an ontology-based metadata management system. The proposed system supports an effective ontology development and management methodology based on our model theory. The benefits of the model are here presented and analyzed. In addition, the system is evaluated in the seismology and geoscience domains.

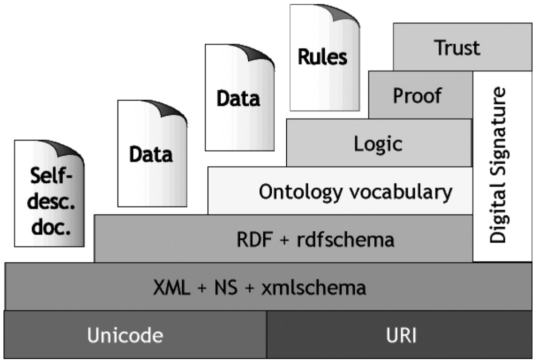


Figure 1. The Semantic Web Architecture from http://www.w3c.org/2000/Talks/1206-xml2k-tbl/

1. Introduction

QuakeSim[1], a joint project that we defined with Jet Propulsion Laboratory (JPL) for NASA, provides a solid Earth science framework to better understand active tectonic and earthquake processes. In the part of the project, several tools and systems employ enormous amount of earthquake data provided by U.S. Government agencies including the Southern California Earthquake Data Center [2] and the Southern California Seismic Network [3].

Seismologists typically have their own interpretations and analysis of raw data from different sources (observations, simulations, etc) [4, 8, 30]. Thus, the process of managing earthquake data faces an emerging problem that can be characterized as semantic conflicts among various information sources [28]. Individual databases, which contain earthquake data, are distributed and organized in different manners and diverse terminologies from earthquake information providers bring misunderstanding of semantics to people and computers. Therefore, a new generation of technology is necessary to manage earthquake information that exchanges information between distributed online resource catalogs using knowledge concepts and relationships from seismology and geoscience.

One of the key solutions for accessing the vast number of the growing heterogeneous geoscience data is ontology-based metadata management technologies. Metadata is in-

---

[1] http://quakesim.jpl.nasa.gov
[2] http://www.data.scec.org
[3] http://www.trinet.org/scsn/scsn.html

formation that represents the characteristics of data units [5]. In order to reuse and understand metadata easily, the Semantic Web suggests a common framework that allows computer systems.

Ontologies provide an explicit model for structuring concepts, together with their definitions and interrelationships [1, 14, 19, 26]. Thus, ontology-based metadata management technologies provide scientists ways to conduct the structuring, transforming, retrieving meaning and quality of the information [17, 18]. Many ontology languages have been developed such as Ontology Interchange Language (OIL) [7], DAML+OIL [13, 14], Ontology Web Language (OWL) [22], Exchange Ontology Language [16], and Simple HTML Ontology Extension [21]. Ontology languages are placed at the lower layers of the Semantic Web architecture in Figure 1. Some of them lie on top of Resource Description Framework (RDF) [3] such as OIL, DAML+OIL and OWL, while others are based on XML or HTML [9]. Based on XML, RDF provide a lightweight ontology framework to support the exchange of information on the Web.

Recent studies have indicated the semantic and syntactic conflicts with contemporary ontology languages, especially focusing on OWL and DAML+OIL [15, 27]. In order to layer two languages on top of RDF, the semantics of these language should be an extension of RDF. However, new semantics are not compatible with these languages model theory, since the semantics are normally provided by a classical Description Logic (DL) model theory. The straightforward way of the semantic layering is problematic that can lead to the semantic paradox [27]. Therefore, one of the key issues in ontology development is providing a general model theory that supports an extension of the RDF syntax without conflicting with the semantic extension.

In this paper, we focus on an ontology-based metadata management methodology combined with a new model theory for constructing ontologies. Such a model must provide the explicit representation of a conceptualization, which is preferably shared and agreed. In addition, the model should be written in the current ontology languages. Furthermore, the model is required to include the designer's knowledge in the largest portion possible. In other word, it should employ sufficient mechanism to express the meaning of interrelationships among concepts.

Toward this end, we propose a novel model for developing ontologies, which is named Classified Interrelated Object Model (CIOM). CIOM is the design of a higher-level ontology model that will enable the ontology developers to naturally and directly incorporate the semantics of ontologies into its meanings. In order to apply CIOM in applications that utilize domain-specific knowledge, we introduce Ontronic; This provides general functionality for the engineering, discovery, management, and presentation of ontology-based metadata incorperated with CIOM. In addition, Ontronic establishes a platform that is necessary to support the Semantic Web technologies for geoscience and seismology applications such as deformation fault information, GPS movement data, and simulation code results.

The remainder of this paper is organized as follows: In Section 2, we introduce and define CIOM. In Section 3, we present the overview and the architecture of Ontronic. Finally, in Section 4, we conclude the paper and provide our future work.

2. Design and Specification of CIOM

In this section, we first present the current status and limitation of the Semantic Web technology. Section 2.1 defines a a specification of our semantic model. Section 2.2 analyzes our model by comparing it with the models of other languages. Finally, Section 2.3 shows an example of ontology using CIOM.

The most popular ontology languages (such as OWL and DAML+OIL), which are submitted to W3C, have four ways of layering on the top of RDF(S) [27].

1. Syntax: OWL and DAML+OIL can be the same as RDF
   Semantics: OWL and DAML+OIL can be an extension of RDFS

2. Syntax: OWL and DAML+OIL can be an extension of RDF
   Semantics: OWL and DAML+OIL can be an extension of RDFS

3. Syntax: OWL and DAML+OIL can be a subset of RDF
   Semantics: OWL and DAML+OIL can differ from RDFS

4. Syntax: OWL and DAML+OIL can differ from RDF
   Semantics: OWL and DAML+OIL can differ from RDFS

In the first layering, it is possible to occur that the collection of all the sets that do not contain themselves. We call this problem as a Russell's paradox [27]. The semantic or syntactic conflicts can occurr in other layering methods. OWL and DAML+OIL are closely associated to expressive Description Logic while the semantics of RDFS are given by the traditional Triple model theory [15]. Figure 2 describes a semantic layering structure of RDF, DAML+OIL, and OWL. In order to reduce the semantic gap between them, we suggest CIOM for the contemporary ontology languages.

Our approach is to incorporate an object-based classified database model to structure ontologies. CIOM inherits the core primitives of the Semantic Database Model (SDM)

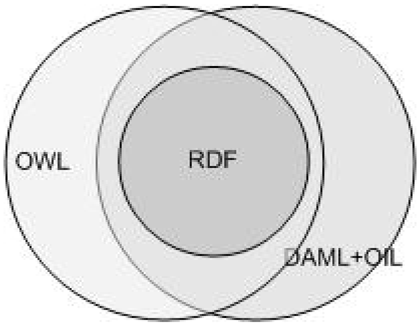


Figure 2. The Semantic Layering Structure of RDF, DAML+OIL, and OWL

[12]. SDM is a design of a higher-level database model that enables the designer to naturally and directly utilize more of the semantics of schema into its meanings. Therefore, by extending the basic structures and constraints of SDM, CIOM intends to model the real world and capture the semantics of applications.

2.1 A Specification of CIOM

This section describes fundamental semantic primitives in CIOM to define concepts and interrelationships. The semantic primitives can be categorized into concept and interconnections.

2.1.1. Concept

Ontology defines a set of representational terms that we call concepts. Concepts are constituted into *classes* in CIOM. A class is defined as a collection of objects. This collection is a logically homogeneous set of one type of objects. The objects of a class are said to be the instances of the class. The name of a class is used to identify the class from others and is written inside of an oval to represent the class. There are several built-in classes predefined in CIOM. They are often used as the value classes for basic attributes such as strings and numbers.

2.1.2. Inter-Connections

In ontologies, concepts are interconnected by means of interrelationships among concepts. The types of these relationship connections are detailed as follow.

- Attributes Connection

  Attributes are the common aspects of instances of a class. An attribute describes each instance of the class. Each attribute has a name that uniquely identifies itself within a class. Similar to a class name, an attribute name can be any string of symbols. The value class of an attribute is the set of all possible values that applies to the attribute. This set can be any built-in or user-defined classes.

  Furthermore, attributes always have inverses, and are usually shown in pairs. In each pair, each attribute is said to be the inverse of the other. This relation is specified symmetrically. The *inverse attribute* of attribute A is denoted as A^{-1}. In addition, three kinds of *cardinality* are used to describe the type of relationship between attributes such as one-to-one $(1:1)$, one-to-many $(1:m)$ and many-to-many $(n:m)$. In CIOM, an attribute can be also specified as non-null (nn) which requires a value. Moreover, CIOM supports aggregate functions such as maximum, minimum, average, and sum by means of *class attributes*.

- Instance Connection

  This is used to show membership. An *instance* is a member of class. The inter-relationship between instances and classes corresponds to an instance connection.

- Group Connection

  A *grouping class* defines a class of classes. In other words, the instances of a grouping class can be a *Part-Of* grouping class [17,18]. It is a second order class in that its instances are of higher-order object type than those of the underlying classes. The instances of a grouping class are viewed as classes themselves.

- Subclasses Connection

Semantic Primitives	*CIOM*	*DAML+OIL*	*OWL*
class	+	+	+
attribute	+	+	+
inverse attribute	+	+	+
cadinality	+	+	+
not null	+	-	-
subclass	+	+	+
predicate-defined subclass	+	-	-
operationally-defined subclass	+	-	-
mutual exclusiveness	+	-	-
collective exhaustiveness	+	-	-
grouping class	+	-	-

Table 1. The Comparison among CIOM, DAML+OIL, and OWL (+ indicates a supported feature; - for unsupported features)

Semantic Primitives in CIOM	*Axioms of OWL*	*Axioms of DAML+OIL*
predicate-defined subclass	owl:Restriction, owl:onProperty, owl:hasValue	daml:Restriction, daml:onProperty, daml:hasValue
operationally-defined subclass (intersection)	owl:subclassOf, owl:intersectionOf	daml:subclassOf, daml:intersectionOf
operationally-defined subclass (union)	owl:subclassOf, owl:unionOf	daml:subclassOf, daml:unionOf
operationally-defined subclass (difference)	owl:subclassOf, owl:intersectionOf, owl:complementOf	daml:subclassOf, daml:intersectionOf, daml:omplementOf
mutual exclusiveness group	owl:equivalentClass, owl:unionOf	daml:sameIndividualAs, daml:unionOf
collective exhaustiveness group	owl:subClassOf , owl:disjointWith	daml:subClassOf , daml:disjointWith
grouping class	owl:equivalentClass, owl:onProperty	daml:equivalentClass, daml:onProperty

Table 2. The Matches Between the Semantic Primitives in CIOM and the Axioms of OWL and DAML+OIL

This connection is used to represent concept inclusion. A concept represented by *subclass* is said to be a specialization of the concept represented by super class, like a child is a kind of the parent. A subclass automatically inherits all attributes from its parent class. These attributes need not to be shown on the subclass but they exist implicitly.

This connection can be further categorized into four types: *mutually exclusive* group, *collectively exhaustive* group, *predicate-defined subclasses* and *operationally-defined subclasses*. For two or more subclasses, mutual exclusiveness means that there is no instance of belonging to more than one of the subclasses, while collective exhaustiveness means that it must belong to at least one of the subclasses for any instance of the parent class.

For a predicate-defined subclass, its membership is determined by specific conditions (predicates) upon one or more attributes. If an instance's attribute values satisfy all conditions, then the instance is automatically added to the subclass. Their membership is decided by three operations; *intersection*, *union*, and *difference* for operationally-defined subclasses.

An intersection subclass contains the instances in both of the parent classes involved in the intersection. For type compatibility, classes involved in this operation must both be subclasses of a common parent, directly or indirectly. A union subclass contains the instances in either of the parent classes involved in the union operation. A difference subclass contains the instances of one parent class that are not in the other class. In addition, a subclass by parent's intersection inherits all attributes of both parents. A subclass by parents union inherits only those attributes common to both parents. A subclass generated by parent's difference inherits all attributes of the parent at the left side of the operator.

2.2. An Analysis of CIOM

In essence, CIOM is comprehensively inclusive of the semantics of DAML+OIL and OWL, as well as DL. Table 1 summarizes the major features of CIOM compared with the syntax of DAML+OIL, OWL, and DL.

A single axiom in DAML+OIL or OWL cannot express some of features in CIOM. However, most of the semantic primitives in CIOM can be implemented in DAML+OIL and OWL using combinations of axioms. Table 2 describes the matches between the semantic primitives in CIOM and the axioms of OWL and DAML+OIL.

For instance, mutual exclusiveness in CIOM can be expressed in OWL. For example, if a person must be either male or female, then the subclasses male and female are collectively exhaustive.

```
<owl:Class rdf:about="#Male">
  <rdfs:subClassOf
   rdf:resource="#Person"/>
</owl:Class>

<owl:Class rdf:about="#Female">
  <rdfs:subClassOf
   rdf:resource="#Person"/>
  <owl:disjointWith
```

```
   rdf:resource="#Male"/>
</owl:Class>
```

Since the interpretations of information in seismology are full of variety depending on scientists and systems [4, 8, 11, 30], a generic modelling methodology is necessary to manage the interoperability of heterogeneous data. As illustrated in Figure 3, CIOM plays a pivotal role in providing a general model for contemporary ontology languages.

2.3. A Seismology-domain Ontology

This section exemplifies a seismology-domain ontology. This ontology is obtained from domain experts and generic seismology terminology of the QuakeTables [4] system, which is a subproject of QuakeSim [11, 30].

Figure 4 shows the example ontology of the seismology domain. This ontology is constructed by domain experts using Ontronic, an ontology-based metadata management system. It has been designed to conceptualize several types of fault data and data sets, as well as the simulated or hypothetical data.

In the example, there are two roots: event and seismology. seismology can be specialized as either geo-phenomenon, seismology research or geological feature. In particular, geo-phenomenon is a subclass of the intersection of the event and seismology (operationally-defined subclass). Seismology paper is a kind of seismology research. Geological features are further classified into fault and segment. In addition, event can be sub-categorized as disaster, conference or geo-phenomenon. Earthquake and volcano are types of both natural (predicate) disaster and geo-phenomenon while nuclear test is a kind of both man-made (predicate) disaster and geo-phenomenon. These relationships are subclass connections.

Earthquakes make one or many faults that have name and strand name. Faults are divided into characteristic segments that are expected to rupture as a unit. Segment has name, depth, friction, strike, and etc. As well, seismology papers are published in conferences and they referred by each segment. Seismology papers includes author, title, published year, and etc. These relationships are attribute connections.

The given ontology also presents the characterization of dynamically-defined earthquake faults it also includes Material Rectilinear Layer parameters for 3-dimensional tectonic deformation modelling. In addition, the instances of the fault in the ontology contain data from California faults, but there is no geographic restriction on future data entries. Furthermore, the instances of the seismology paper in the ontology are extracted from refereed journal articles, professional papers, professional reports, and conference abstracts.

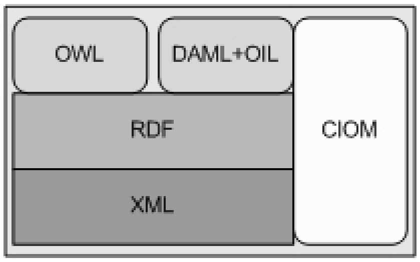


Figure 3. The Semantic Structure of CIOM on the Semantic Web

In CIOM, ontologies are represented by a directed acyclic graph (DAG). Here, each node in the DAG indicates a concept. A thin arrow is used to denote attributes while a subclass is drawn in a thick arrow. An inverse attribute is drawn in a pair of inter-linked arrows. We denote grouping classes with triple arrows. Whereas, an Instance-Of is denoted with a narrow line.

3. Ontronic: A General Purpose Ontology-based Metadata Management System

This section introduces Ontronic, which provides general functionalities to manage ontology-based metadata. In order to access a large-scale and growing amount of heterogeneous geoscience data, an ontology-based semantic metadata management system is required to support effective information retrieval and web-based search for data of interest to scientists and the ability to interoperate such diverse datasets with various models and simulation codes.

To address the aforementioned requirements, the current ontology management tools such as OilEd [2], Protégé-2000 [20], Ontolingua [6], Chimaera [23–25], and KAON [31] have been developed. Most of them typically use the semantic models of the current ontology languages. Differentiated from the contemporary tools, we approach the construction of ontologies by incorporating CIOM.

As shown in Section 2.2, generated ontologies based on CIOM can be translated into various kinds of the current ontology languages such as OWL and DAML+OIL. Thus, by deploying CIOM, Ontronic is capable of increasing the level of semantic interoperability. As well, Ontronic provides the high accessibility to users by allowing cooperative multi-author to develop and share ontologies in a web-based environment. The following sections present two viewpoints of the process of ontology design and the conceptual architecture.

---

[4] http://infogroup.usc.edu:8080/public.html

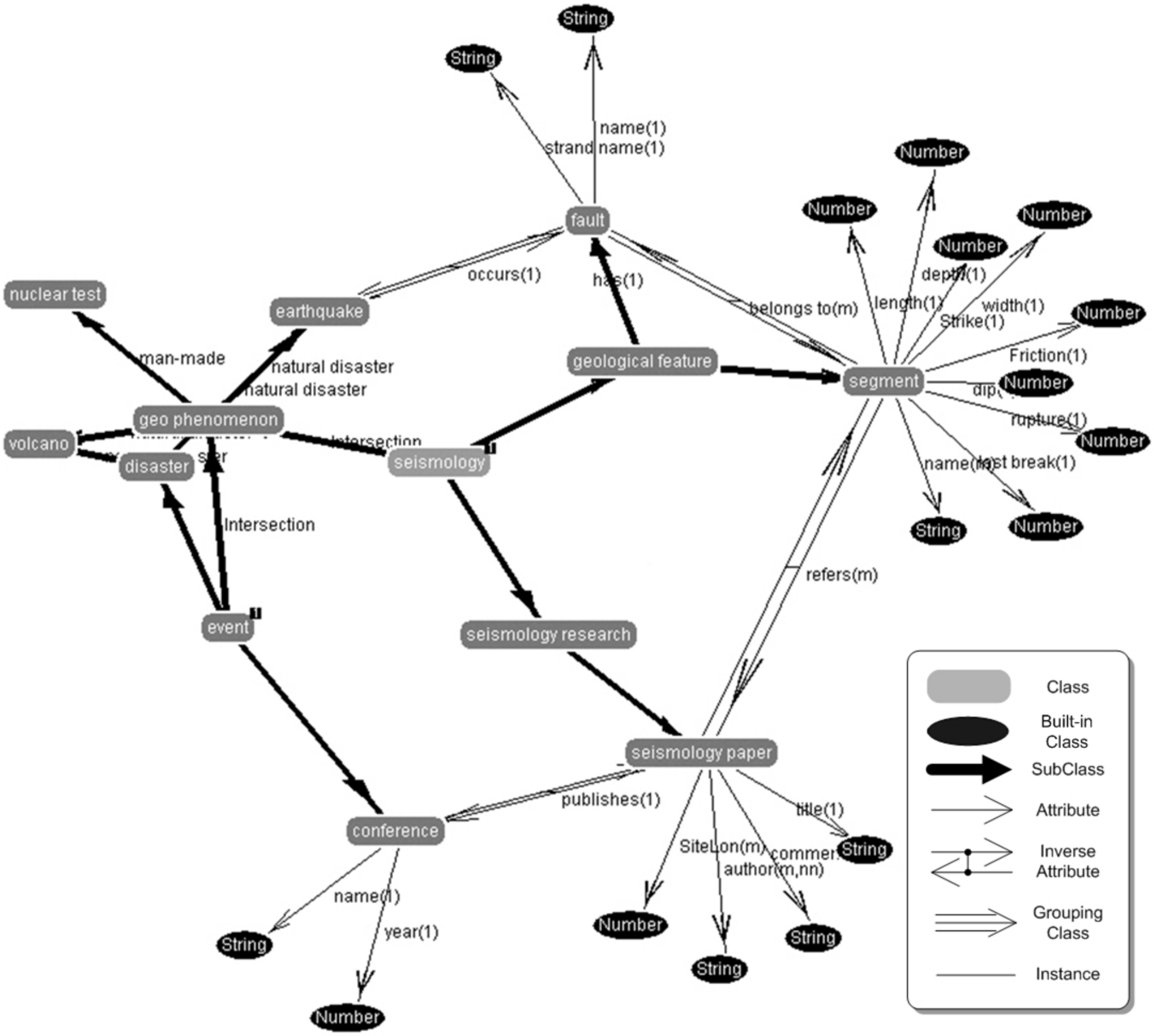


Figure 4. The Seismology-Domain Ontology (Created by Ontronic)

3.1. The Process of Ontology Design

The process of ontology design, which describes a fundamental method of constructing ontologies in Ontronic, consists mainly of three layers as follows:

- Domain layer: A conceptualization that captures a shared knowledge of the given domain.
- Semantic layer: An explicit ontology model based on CIOM is generated from the conceptualization of the given domain. Such a model has a collection of concepts, interrelationships and the constraints.
- Metadata layer: A formal description for the above model is produced to be machine understandable.

As shown in Figure 5, in the domain layer, ontology developers usually conduct an information requirements analysis and express the results of their analysis in terms of the semantic model. The gap between the semantic level of the domain and ontologies can be bridged by CIOM in the semantic layer. In other words, in comparison with the model theory of the contemporary ontology languages, CIOM can be used as a higher-level semantic model in which the ontology developers design ontologies. Consequently, a collection of metadata, which is represented as various kinds of ontology languages such as DAML+OIL and OWL, can be generated from ontologies that are already produced in the previous layer.

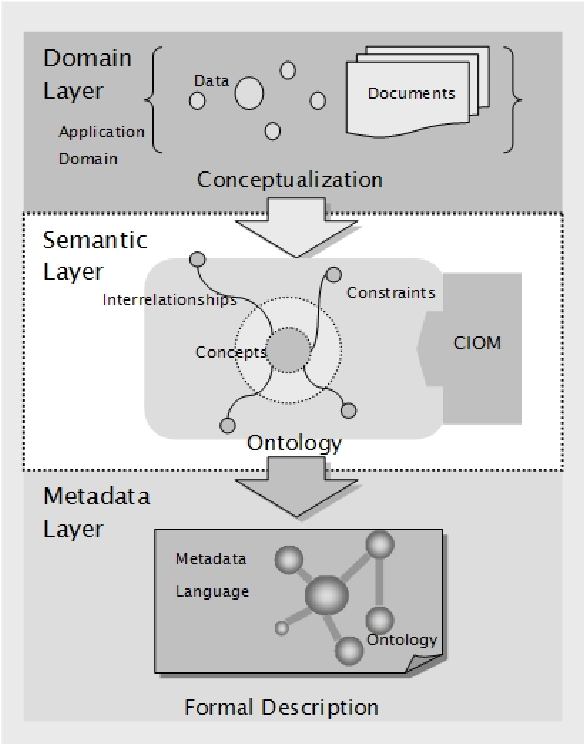


Figure 5. The Process of Ontology Design

3.2. Conceptual Architecture

Given the process of ontology design in Section 3.1, we can identify key functionalities that must be provided by Ontronic:

1. We present an interface to communicate graphical representation between users (ontology developers) and Ontronic.
2. Ontronic contains functions to facilitate ontology creation with the semantic primitives in CIOM.
3. Ontronic provides a metadata generation mechanism from the ontologies created with CIOM. The produced metadata can be exported to permanent storages and also can be imported from them.

As illustrated in Figure 6, the architecture of Ontronic is clearly separated into a user-interface, a model, and a storage component to meet the above key requirements.

Ontronic provides the visual ontology manager for a graphical representation. The primary role of the visual ontology manager is to provide a graphical user interface (GUI) to view, analyze, and compose ontologies. It contains an ontology DAG, ontology tree, ontology visualization API, query processor, and metadata viewer. Ontology DAG illustrates ontologies with DAGs based on the denotations of the semantic primitives in CIOM. The ontology tree also presents a hierarchical structure of concepts. The query processor parses and executes a query. In addition, generated metadata written in the ontology languages such as DAML+OIL and OWL is displayed in metadata viewer. Figure 7 presents a snapshot of the visual ontology manager in Ontronic.

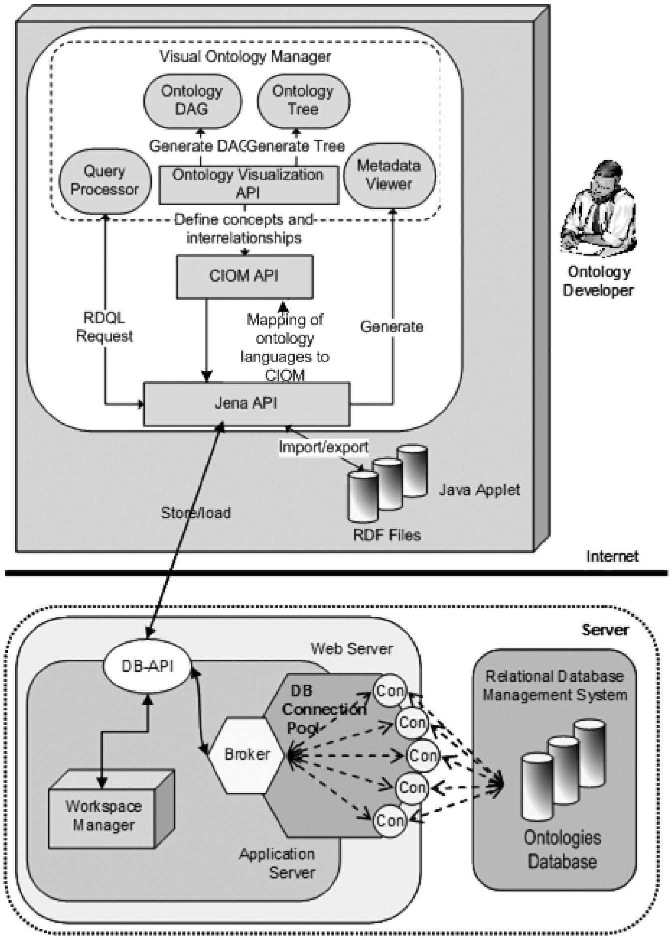
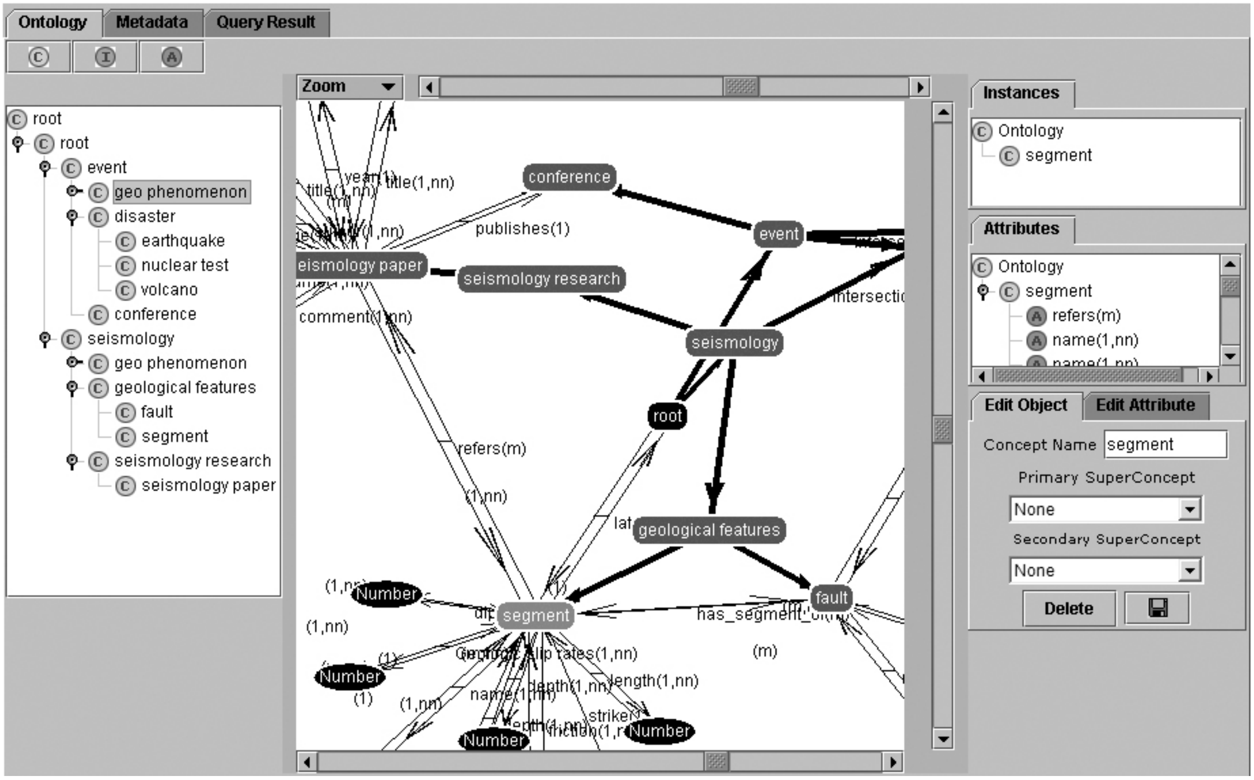


Figure 6. The Conceptual Architecture of Ontronic

Ontronic provides APIs to retrieve and manipulate ontologies. Ontology visualization API facilitates creating, updating, and deleting graphical units for an ontology presentation, interacting with CIOM API. CIOM API can construct meta-models for ontologies consisting of the semantic primitives in CIOM. CIOM API also translates queries for CIOM model to RDQL [29]. Additionally, CIOM API provides a comprehensive transformation from CIOM meta-model into RDF by utilizing Jena[5] API which pro-

---

[5] http://jena.sourceforge.net/



Concept Name segment

Primary SuperConcept None

Secondary SuperConcept None

Figure 7. The Snapshot of Visual Ontology Manager

vides RDF-based metadata management infrastructure designed for semantics-driven applications.

Finally, Ontronic supports the storage of ontologies into both a database and a file system. This framework supports the definition, storage, access, and control of collections of structured data. The relational database management system (DBMS) physically stores the metadata in the Ontronic server. Furthermore, Ontronic provides a web-based cooperative workspace in which multiple authors can build and share same ontologies simultaneously. In order to endure heavy requests to the DBMS, Ontronic deploys connection pool. Additionally, the clients can export and import the RDF, OWL, and DAML+OIL files.

3.3. Contribution of Ontronic

Ontronic currently provides the capacity to access and manage ontologies populated with paleoseismic data from major faults, and with three structured data sets containing summary fault attributes. These ontologies provide geographic coordinates, geometry, and summary attributes for many active faults and fault segments in California.

4. Conclusion

In order to resolve a semantic conflicts among various information sources in seismology and geoscience, we proposed the ontology-based semantic information management methodology. In this paper, we first defined CIOM as a design of a higher-level ontology model. We then introduced Ontronic, a system that provides the general functionalities to manage ontology-based metadata. A basic idea of the process of ontology creation has been presented.

The major contributions of this paper are

- defining seismology-domain meta-ontology: a federation of semantic specifications for the various types of geophysical data.

- providing a tool to manage ontology-based metamodel and metadata with the capability of supporting a general ontology modeling methodology with CIOM.

Our next work will focus on incorporating ontologies into the data integration processes in Ontronic with the federated database service.

5. Acknowledgements

This research was supported in part by the Advanced Information System Technology program of NASA's Earth Science Technology Office, and in part by the USC Integrated Media Systems Center, a National Science Foundation Engineering Center, cooperative agreement No. EEC-9529152.

References

[1] A. Anjomshoaa et al., "The Design and Implementation of Grid Database Services in OGSA-DAI", In *Proceedings of Concurrency and Computation: Practice and Experience*, 2004.

[2] S. Bechhofer, I. Horrocks, C. Goble, and R. Stevens, "OilEd: a Reason-able Ontology Editor for the Semantic Web", In *Proceedings of KI2001, Joint German/Austrian Conference on Artificial Intelligence*, 2001.

[3] D. Beckett, RDF/XML Syntax Specification (Revised), W3C Recommendation, 2004, URL: http://www.w3.org/TR/rdf-syntax-grammar/ (current Feb. 2005).

[4] A. Chen, A. Donnellan, D. McLeod, G. Fox, J. Parker, J. Rundle, L. Grant, M. Pierce, M. Gould, S. Chung, and S. Gao, "Interoperability and Semantics for Heterogeneous Earthquake Science Data", In *Proceedings in International Workshop on Semantic Web Technologies for Searching and Retrieving Scientific Data*, 2003.

[5] A. Chervenak, E. Deelman, C. Kesselman, L. Pearlman, and G. Singh, "A Metadata Catalog Service for Data Intensive Applications", In *Proceedings of SC03*, 2003.

[6] A. Farquhar, R. Fikes, and J. Rice, "The Ontolingua Server: A Tool for Collaborative Ontology Construction", In *Proceedings of International Journal of Human-Computer Studies*, 1996.

[7] D. Fensel, I. Horrocks, F.V. Harmelen, S. Decker, M. Erdmann, and M. Klein, "OIL in a Nutshell", In *Proceedings of Workshop on Application of Ontologies and PSMs*, 2000.

[8] G.C. Fox, K. Hurst, A. Donnellan, and J. Parker, *GeoComplexity and the Physics of Earthquakes*, AGU, 2000, pp 219-245.

[9] A. Goméz-Pérez and O. Corcho, "Ontology Languages for the Semantic Web", *IEEE Intelligent Systems*, January 2002, 17(1):54-60.

[10] L.B. Grant and M.M. Gould, "Assimilation of Paleoseismic Data for Earthquake Simulation", In *Proceedings of Pure and Applied Geophysics*, 2003.

[11] L. Grant, A. Donnellan, D. McLeod, M. Pierce, G. Fox, A. Chen, M. Gould, and S.S. Sung, "A Web Service Based Universal Approach to Heterogeneous Fault Databases", *Submitted to Computing in Science and Engineering*, 2005.

[12] M. Hammer, D. McLeod, "Database Description with SDM: A Semantic Database Model", *ACM Transactions on Database systems*, September 1981, 6(3):351-386.

[13] F.V. Harmelen, P.F. Patel-Schneider, and I. Horrocks. Reference Description of the DAML+ OIL (March 2001) Ontology Markup Langauge, 2001 URL: http://www.daml.org/2001/03/daml+oil-index.html (current Feb. 2005).

[14] I. Horrocks, P.F. Patel-Schneider, F.V. Harmelen, Reviewing the Design of DAML+OIL: An Ontology Language for the Semantic Web, In *Proceedings of National Conference on Artificial Intelligence*, 2002.

[15] I. Horrocks and P.F. Patel-Schneider, "Three Theses of Representation in the Semantic Web", In *Proceedings of the Twelfth International Conference on World Wide Web*, 2003.

[16] R. Karp, V. Chaudhri, and J. Thomere, XOL: An XML-Based Ontology Exchange Language, 1999, URL: http://http://xml.coverpages.org/xol-03.html (current Feb. 2005).

[17] L. Khan and D. McLeod, "Disambiguation of Annotated Text of Audio Using Onologies", In *Proceedings of ACM SIGKDD Workshop on Text Mining*, 2000.

[18] L. Khan, D. McLeod, and E.H. Hovy, "Retrieval Effectiveness of an Ontology-based Model for Information Selection", *The VLDB Journal*, 2004, 13(1):71-85.

[19] H. Kim, "Ontology Applications and Design: Predicting How Ontologies for the Semantic Web Will Evolve", *Communications of the ACM*, 2002, 45(2):48-74.

[20] Q. Li, P. Shilane, N. F. Noy, and M. Musen, "Ontology Acquisition from On-line Knowledge Sources", In *Proceedings of AMIA Symposium*, 2000.

[21] S. Luke and J. Heflin, SHOE 1.01 Proposed Specification, SHOE Project, 2000, URL: http://www.cs.umd.edu/projects/plus/SHOE/spec.html (current Feb. 2005).

[22] D.L. McGuinness and F.V. Harmelen, OWL Web Ontology Language Overview, 2004, URL: http://www.w3.org/TR/owl-features/ (current Feb. 2005).

[23] D.L. McGuinness, "Conceptual Modeling for Distributed Ontology Environments", In *Proceedings of the Eighth International Conference on Conceptual Structures Logical, Linguistic, and Computational Issues*, 2000.

[24] D.L. McGuinness, R. Fikes, J. Rice, and S. Wilder. "The Chimaera Ontology Environment", In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, 2000.

[25] D.L. McGuinness, R. Fikes, J. Rice, and S. Wilder. An Environment for Merging and Testing Large Ontologies, In *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning*, 2000.

[26] J.Z. Pan, I. Horrocks, "RDFS(FA) and RDF MT: Two Semantics for RDFS", In *Proceedings of International Symposium on Wearable Computers*, 2003.

[27] P.F. Patel-Schneider and D. Fensel. "Layering the Semantic Web: Problems and Directions", In *Proceedings of the 1st International Semantic Web Conference*, 2002.

[28] R. Raskin and M. Pan, "Semantic Web for Earth and Environmental Terminology (SWEET)", In *Proceedings of Earth Science Technology Conference*, 2003.

[29] A. Seaborne, RDQL - A Query Language for RDF, W3C Member Submission, 2004, URL: http://www.w3.org/Submission/2004/SUBM-RDQL-20040109/ (current Feb. 2005).

[30] S. Thacker, A. Sheth and S. Patel, *Creating the Semantic Web*, MIT Press, 2001.

[31] R. Volz, D. Oberle, S. Staab, and B. Motik, "KAON SERVER - A Semantic Web Management System", In *Proceedings of the Twelfth International Conference on World Wide Web*, 2003.