Semantic Web Enabled Search Systems Based on the Domain Specific Ontology for Global Knowledge Management

Jae-Hun Joo  
University of Nebraska at Lincoln, givej@dongguk.ac.kr

Sang M. Lee  
University of Nebraska at Lincoln, slee1@unl.edu

Follow this and additional works at: http://aisel.aisnet.org/amcis2005

Recommended Citation
http://aisel.aisnet.org/amcis2005/470
Semantic Web Enabled Search Systems
Based on the Domain-Specific Ontology
for Global Knowledge Management

Jae-Hun Joo
Visiting Professor, Department of Management,
University of Nebraska-Lincoln
(Professor, Department of Electronic Commerce,
Dongguk University
707, Sukjnag-dong,Gyeongju city, 780-714,
South Korea)
Tel: (402) 472-1989, Fax: (402) 472-5855,
givej@dongguk.ac.kr

Sang M. Lee
University Eminent Professor, Department of
Management, College of Business Administration,
University of Nebraska-Lincoln
Lincoln, NE 68588-0491
Tel: (402) 472-3915, Fax: (402) 472-5855,
slee1@unl.edu

ABSTRACT

The Semantic Web is an extension of the current Web in which information is given a well-defined meaning, thus better enabling people to work more efficiently with computers. The Semantic Web handles machine-processable information. Its underlying technologies are Resource Description Framework (RDF), RDF Schema (RDFS), and ontology as the shared formal conceptualization of particular domains. The purpose of this study is to implement a Semantic Web enabled search system in the tourism business domain. The system exploits Web Ontology Language (OWL) as the standard ontology language proposed by W3C and Jena that is a Semantic Web toolkit, namely a Java framework for Semantic Web applications.

Keywords: semantic web, ontology, search method, knowledge management, tourism, RDF, RDF Schema, XML

INTRODUCTION

Although significant progress has been made in technologies for creating and disseminating information and knowledge on the Web, today’s Web still serves primarily as a global space to present information for human consumption (Wuwongse et al., 2001). There are two important limitations in exploiting the Web as the space for global knowledge management. Two important limitations are:

1. Lack of the mechanism for providing information and knowledge that computers or automatic agents can understand and process.
2. Information overload owing to keyword-based search modes.

First limitation results from Web documents represented in HTML. In most cases, computers including software agents cannot only understand and process data or information automatically on the Web, but also integrate them, in particular, in heterogeneous environments. The other occurs in lack of semantics in information retrieval. The more information resources exist on the Web, the more information overload arises out of search results.

The Semantic Web as an extension of the current Web, enables us to overcome the limitations. It makes the generation of new knowledge management tools possible (Davies et al., 2003). The mechanisms to realize the Semantic Web require eXtensible Markup Language (XML), RDF, RDFS, and ontology. The Semantic Web based on such tools provides the opportunity for global knowledge management by integrating the resources dispersed across the Internet Web as well as other resources including the existing knowledge base. There are many opportunities to apply the Semantic Web to a variety of areas, as its development is in its infancy.
Ontology is seen as a key enabler for the Semantic Web. Also, ontologies are useful for improving the accuracy of Web searches (Antoniou and Harmelen, 2004). Thus, ontology development in a variety of domains will enable us to realize the purpose of the Semantic Web. In addition, there is a need to develop ontology based on widely accepted or de facto standards such as RDF/RDFS and OWL in order to ensure the easiness of integration among ontologies.

The purpose of this paper is to present a domain-specific ontology, as well as a conceptual model for global knowledge management, based on Semantic Web and to implement a Semantic Web enabled search system for the tourism business domain. The system exploits OWL as the standard ontology language proposed by W3C and Jena tools, namely a Java framework for writing Semantic Web applications. Recently, there are many ontology editors that allow users to visually manipulate ontologies such as Protégé-2000, OntoEdit, and OilED (Haase and Sure, 2004). We adopt Jena tool because it is an open-source and supports OWL. In addition, we can directly write application programming interfaces (APIs) in Java, in which Jena provides flexibility of system development and extension for system builders.

The paper is organized as follows: after this introductory section, section 2 reviews related literatures and presents a conceptual design of the search system for knowledge management. Section 3 provides two scenarios of information retrieval in the tourism business domain. Section 4 designs the ontology providing answers to queries of the scenarios. Section 5 discusses the result of the implemented system. The final section presents conclusions.

RELEVANT LITERATURES AND CONCEPTUAL DESIGN

Figure 1 presents layers of the Semantic Web architecture (Berners-Lee et al., 2001). As the Figure 1 shows, RDF and RDFS play important roles in the Semantic Web.

RDF is a foundation for processing metadata, as it defines a simple model for describing relationships among resources. The goal of RDF is to define a mechanism for describing resources that makes no assumptions about a particular application domain (W3C, 1999). RDFS provides information about the interpretation of the statements given in an RDF data model (W3C, 2000). This allows software agents to process RDF vocabularies. An ontology is a formal explicit specification of a shared conceptualization (Gruninger and Lee, 2002). Ontologies provide higher semantics for a domain. In other words, they not only provide more expressive power for the domain, but also offer a reasoning support. The Web Ontology Language, OWL (W3C, 2004) is a semantic markup language for publishing and sharing ontologies on the Web. It is developed as a vocabulary extension of RDF and RDFS and is derived from the DAML+OIL (DARPA Agent Markup Language + Ontology Inference Layer) Web Ontology Language (Joint US/EU ad hoc Agent Markup Language Committee, 2001).

Many teams have been conducting projects and developing various tools as shown in Table 1.
<table>
<thead>
<tr>
<th>Tools</th>
<th>Overview</th>
<th>Site URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>KAON</td>
<td>An open-source ontology management infrastructure targeted for business applications. It includes a comprehensive tool suite allowing easy ontology creation and management and provides a framework for building ontology-based applications (Volz, 2003).</td>
<td><a href="http://kaon.semanticweb.org">http://kaon.semanticweb.org</a></td>
</tr>
<tr>
<td>Protégé-2000</td>
<td>An ontology and knowledge-base editor, and also an open-source Java tool, that provides an extensible architecture for the creation of customized knowledge-based applications (Noy, 2001).</td>
<td><a href="http://protege.semanticweb.org/">http://protege.semanticweb.org/</a></td>
</tr>
<tr>
<td>On-TO-Knowledge</td>
<td>Tools and methods for supporting knowledge management relying on sharable and reusable knowledge ontologies (Davies et al., 2003).</td>
<td><a href="http://ontoknowledge.semanticweb.org/">http://ontoknowledge.semanticweb.org/</a></td>
</tr>
</tbody>
</table>

Table 1. Ontology Tools and Projects

The Knowledge Systems Lab of Stanford University has been developing a Wine Agent. The Wine Agent receives a meal description and retrieves a selection of matching wines available on the Web by using DAML+OIL and OWL for representing a domain-ontology of foods and wines (KSL, 2003).

Figure 2 shows a conceptual model for global knowledge management systems exploiting the Semantic Web technology. The model is composed of four layers. The resource layer refers to a variety of knowledge resources including contents of the Internet Web. The Semantic Web enables us to manage Web contents as organizational knowledge as well as the existing internal knowledge such as database and knowledge base. Also it allows us to define a part of a document or one or more sentences as knowledge piece corresponding to a resource. Thus, the systems users can find more correct and personalized knowledge and seamlessly integrate knowledge extracted from various resources. The metadata layer includes RDFs as the foundation for processing metadata extracted from the resource layers. The system requires middleware modules or tools to support automatic knowledge extraction from unstructured and semi-structured data in heterogeneous resources. In this layer, we need RDF management subsystems including RDF repositories, XML/RDF parser, and RDF APIs which enable the interaction with ontology and resources layers. The ontology layer including RDFS and inference engine plays a critical role in the system. Ontology APIs enable Web services or existing knowledge management systems to be integrated in the system. Finally, users can find necessary knowledge through query and semantic search.

![Figure 2. A Conceptual Model of Knowledge Management Based on the Semantic Web](image-url)
SCENARIOS

The keyword search method being exploited in the Internet Web or knowledge management systems can retrieve irrelevant information that includes certain terms in different meanings. Thus, this search method can result in an information overload. It can miss the desired information when different terms with the same meaning are used.

Consider the situation where tourists are planning to visit Gyeongju (an old imperial city of Shilla Dynasty in Korea). The tourists need information based on their tour schedule, individual preferences, and budget. Let us consider the following two scenarios:

Scenario 1: Accommodation Recommendation for Tourists

“Recommend an appropriate accommodation for the knapsack tourist who plans to visit Gyeongju city”

Scenario 2: Search for Sightseeing Spots in the Nearby Regions

“Find attractive sightseeing sites near Kolon Hotel in Gyeongju for the tourist who has about four hours of tour time”

Consider the case we use Web search services such as Google or Yahoo. In each of the above scenarios, appropriate keywords are critical to extract good search results. Although desired keywords may be selected, limitations of a keyword-based search can not be overcome.

ONTOLOGY DESIGN FOR TOURISM BUSINESS

It is necessary to build ontologies to get answers for the tourist in the two scenarios. Figure 3 shows some of the major classes and properties representing the relations among them in the tourism domain. For example, both GroupTour and IndividualTour are subclasses of the class Tour and the class SchoolJourney is that of the class GroupTour. There are classes of concepts which constitute a hierarchy with multiple inheritances such as the class BusinessTour and the class KnapsackTour as depicted in Figure 3.

The property hasTour relates the class Person to the class Tour. In other words, resources have properties associated with them. For example, Person has Tour. In the relationship between the class Person and the class Tour, the reverse is true. Thus we can say “the Tour is provided to the Person”.

Figure 3. Tour Package Ontology for Tourism Business
Figure 4 presents the extended ontology to that of Figure 3 by adding the class Region and the class Distance. The relation between the class Attraction and the class Region makes it to infer that a region may have some tourism attraction sites. Inversely, we can infer that an attraction site is located in a region. To get the answer for query of the Scenario 2, we need an ontology enabling the system to identify the terminology, “nearby”. The region-distance ontology shown in figure 4 is used to represent knowledge about distance between two regions.

Figure 5 describes the ontology of Figure 4 as a semantic graph form. RDFS defines a property's domain - resources that can be subjects of the property, and a property's ranges – resources that can be objects of a property. For example, the property hasDistance may have a class Region as its domain and a class Distance as its range.
In OWL, there are two types of properties: object property and datatype property. The former relates objects to other objects. The latter relates objects to datatype values. OWL provides a powerful mechanism for enhanced reasoning about a property (W3C, 2004). Table 2 summarizes property characteristics for those depicted in Figure 4. For example, hasDistance and givenToRegion are ObjectProperty and there is an inverse relation between them. A region have distance, the distance is given to the region. The relation in either direction is true. The hasDistance property is functional, i.e., a region can have at most one distance. The givenToRegion property is inverse functional, i.e., a distance must be given to a region. Properties, “from”, “to”, and “distanceValue” are datatype. The property “distanceValue” takes positive integer.

<table>
<thead>
<tr>
<th>Object Property</th>
<th>HasDistance</th>
<th>FunctionalProperty</th>
<th>inverseOf</th>
</tr>
</thead>
<tbody>
<tr>
<td>GivenToRegion</td>
<td>hasAttraction</td>
<td>InverseFunctionalProperty</td>
<td>inverseOf</td>
</tr>
<tr>
<td></td>
<td>hasReataurant</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LocatedIn</td>
<td></td>
<td>TransitiveProperty</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Datatype Property</th>
<th>From</th>
<th>String type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To</td>
<td></td>
</tr>
<tr>
<td>DistanceValue</td>
<td></td>
<td>Integer Type</td>
</tr>
</tbody>
</table>

Table 2. Types of Property Elements in Region-Distance Ontology

We can define tourist as the person who has at least one tour package. In the RDF syntax, for example, “Tourist” represented in Person∩Tour would be written as

```xml
<owl:Class rdf:ID="Tourist">
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Person"/>
        <owl:Class rdf:ID="Tour"/>
      </owl:intersectionOf>
    </owl:Class>
    <rdfs:subClassOf/>
  </owl:Class>
</owl:Class>
```

The example defines the class Tourist to be the intersection of Person and Tour. OWL allows us to express the intersection of classes by using owl:intersectionOf.

In general, knapsack tourists don’t want to stay in expensive hotels. Following ontology represented in OWL requires only accommodations except hotels to be recommended to them.

```xml
<owl:Class rdf:about="#KnapsackTour">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:ID="#hasAccommodation"/>
      <owl:allValuesFrom>
        <owl:Class>
          <owl:oneOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Homestay"/>
            <owl:Class rdf:about="#Motel"/>
            <owl:Class rdf:about="#OtherAcc"/>
            <owl:Class rdf:about="#YouthHostel"/>
          </owl:oneOf>
        </owl:Class>
        <owl:allValuesFrom/>
      </owl:Restriction>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
```
As shown in Figure 3, the class KnapsackTour is a subclass of IndividualTour or GroupTour. Both classes, IndividualTour and GroupTour, are subclasses of Tour. The classes Hotel, Motel, Homestay, Youthhostel, and OtherAcc are all subclasses of Accommodation. The hasAccommodation property only has values of a class whose members are the individuals Homestay, Motel, YouthHostel, and OtherAcc. The property hasAccommodation has the domain of Tour and the range of Accommodation. That is, it relates to instances of subclasses of Tour to instances of subclasses of Accommodation.

The example states that the class KnapsackTour is a subclass of an anonymous OWL class that has as its extension a set of all individuals for whom the property hasAccommodation has values in only one class among Homestay, Model, OtherAcc, and Youthhostel. The example defines that Knapsack is subclass of an anonymous class with a property restriction. The element, owl:allValuesFrom is used to specify the class of possible values the property specified by owl:onProperty can take. In other words, the class KnapsackTour has a property called hasAccommodation restricted to have allValuesFrom only in one class among the collection of Homestay, Model, OtherAcc, and Youthhostel. Therefore, the class KnapsackTour never has a value from the instance of the class Hotel.

<owl:InverseFunctionalProperty rdf:about="#attractionLocatedIn">
  <rdfs:domain rdf:resource="#Attraction"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
  <rdfs:range rdf:resource="#Region"/>
</owl:InverseFunctionalProperty>

The owl:InverseFunctionalProperty defines a property for which two different objects cannot have the same value. If two instances of the class Attraction are respectively defined as “TombOfKingMunmu” and “DaewanamUnderwaterTomb,” we can infer that these two resources must refer to the same thing in the example.

<owl:FunctionalProperty rdf:ID="hasDistance">
  <rdfs:range rdf:resource="#Distance"/>
  <owl:inverseOf>
    <owl:InverseFunctionalProperty rdf:ID="givenToRegion"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#Region"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
</owl:FunctionalProperty>

The property hasDistance relates the class Region to the class Distance. The hasDistance defined as owl:FunctionalProperty has at most one unique value for each instance of the class Region.

As shown in Figure 4, the properties ‘from’ and ‘to’ defined as Datatype properties have string values and the property ‘distanceValue’ has a positive integer. We can infer the instances of the class Region having distance values less than a particular value for a given region.

IMPLEMENTATION

We use Jena toolkit to implement the ontology design for the tourism business. Jena is an open-source Semantic Web developer’s kit as a set of Java APIs for manipulating RDF models. It comprises a number of modules including RDF API, ARP RDF/XML parser, ontology API, RDQL, and storage modules (Jena2). We used J2SDK 1.4.2 to develop applications of the system.

Figure 6 presents a sequence diagram that displays object interactions arranged in a time sequence. The actor may set up environment variables by invoking the object Configurator. The object SearchOwl activates the inference engine to create a new inference model with the query data sets and returns search results of RDF triples by exploiting RDQL. The InferModel invokes the OwlReasoner to create a new inference model that is built in triple types (subject, predicate, object) as an extension of the Jena RDF triple. The Reasoner of Jena 2 supports OWL Lite and OWL DL reasoning (Jena2). The OwlReaner provides inference rules based on Description Logics. The OWL/RDF repository stores ontologies represented as OWL language and RDF/RDFS, and allows access and reuse of ontologies.
Figure 7 shows the hierarchy of classes created in the ontology buildings in the tourism business area. Figure 8 shows search results for the query of Scenario 2.

Figure 6. A Sequence Diagram of the Search System in Tourism Ontology

Figure 7. Treeview for the Tourism Ontology
CONCLUSIONS

Recently, a variety of studies on the Semantic Web are conducting for a few years. There are three approaches for realizing the Semantic Web. First approach is to add annotations to Web contents. Simple HTML Ontology Extensions (SHOE) is a typical example of the annotation method (Heflin, 2001). Second one is Topic Maps, in which knowledge is represented in topics, associations, and occurrences for resources, driven by ISO (Pepper, 2002). Finally, W3C has been propagating the Semantic Web based on RDF, RDFS, and OWL. RDF focuses on resources while Topic Maps are subject-oriented.

In this paper, we suggested a conceptual model of knowledge management systems based on the Semantic Web in terms of the third approach of w3C. The model consists of four layers such as resources, metadata, ontology, and user & query. In particular, we focused on ontology and query layers. Ontologies were designed to support semantic search and solve the problem two scenarios derived from the tourism domain. The ontology will be widely used as a fundamental tool in building a global knowledge management system. To develop a prototype of the ontology including RDF and RDFS, we used OWL and Jena toolkits. The results of the systems implementation indicate satisfactory search routines.

References


