Applying Ontology to the Web: A Case Study

Jeff Heflin, James Hendler, and Sean Luke

Department of Computer Science
University of Maryland
College Park, MD 20742
{heflin, hendler, seanl}@cs.umd.edu

Abstract

This paper describes the use of Simple HTML Ontology Extensions (SHOE) in a real world internet application. SHOE allows authors to add semantic content to web pages and to relate this content to common ontologies that provide contextual information about the domain. Using this information, query systems can provide more accurate responses than are possible with the search engines available on the Web. We have applied these techniques to the domain of Transmissible Spongiform Encephalopathies (TSEs), a class of diseases that include "Mad Cow Disease". We discuss our experiences and provides lessons learned from the process.

1. Introduction

The "Mad Cow Disease" epidemic in Great Britain and the apparent link to Creutzfeldt-Jakob disease (CJD) in humans generated an international interest in these diseases. Bovine Spongiform Encephalopathy (BSE), the technical name for "Mad Cow Disease", and CJD are both Transmissible Spongiform Encephalopathies (TSEs), brain diseases that cause sponge-like abnormalities in brain cells. Concern about the risks of BSE to humans continues to spawn a number of websites on the topic; some of these sites provide valuable information, while others are simply sources of rumors. The reliable sites range in content from epidemiology of the diseases, to scientific studies on inactivation, to regulations by various agencies. It is difficult for users to locate relevant information with the standard web search engines because these tools match on individual words instead of their meanings. As such, they cannot take the relationship between words into account, map between the terminology of different communities, or use any contextual information to differentiate between terms with many meanings.

The Joint Institute for Food Safety and Nutrition (JIFSAN), a partnership between the Food and Drug Administration (FDA) and the University of Maryland, is attempting to rectify this situation. They wish to provide a clearinghouse for information on TSEs. This site must be able to serve a diverse group of users, including the general public, researchers, risk assessors, and policy makers. However, the diversity of data, the constant appearance of new information, and the distribution of ownership make it difficult to manually maintain an accurate index. Additionally, the nature of the target user community means the retrieval tools must be able to respond to general queries and very specialized queries with the appropriate level of detail to inform the user.

We have built a suite of tools to address these problems, with the basis for these tools being an internet compatible knowledge representation language called Simple HTML Ontology Extensions (SHOE). The underlying philosophy of SHOE is that intelligent agents will be able to better perform tasks on the Internet if the most useful information on web pages is provided in a structured manner. To this end, SHOE extends HTML with a set of knowledge oriented tags that, unlike HTML tags, provide structure for knowledge acquisition as opposed to information presentation. In addition to providing explicit knowledge, SHOE sanctions the discovery of implicit knowledge through the use of taxonomies and inference rules available in reusable ontologies that are referenced by SHOE web pages. This allows information providers to encode only the necessary information on their web pages, and to use the level of detail that is appropriate to the context. SHOE-enabled web tools can then process this information in novel ways to provide more intelligent access to the information on the Internet.

This paper describes the first application of SHOE to a large-scale, real world domain. In Section 2, we lay out the architecture of the system and detail the efforts to put each piece in place. Section 3 discusses what we have learned from the process. Sections 4 and 5 discuss related and future work, respectively. Finally, Section 6 presents our conclusions.

2. Building the System

This section describes the procedural and technical aspects of the TSE application. We also explain our design choices based on the features of the TSE problem domain. The system architecture can be summarized as follows:

- A single, comprehensive ontology is available on the TSE Risk Website.
- Knowledge providers who wish to make material available to the TSE Risk Website use a tool called the Knowledge Annotator to mark-up their pages with SHOE. The instances within these pages are described using elements from the TSE Ontology.
- The knowledge providers then place the pages on the Web and notify JIFSAN.
- JIFSAN reviews the site and if it meets their standards, adds it to the list of sites that Exposé, the SHOE web crawler, is allowed to visit.
- Exposé crawls along the selected sites, searching for more SHOE annotated pages with relevant TSE information. It will also look for updates to pages.
- SHOE knowledge discovered by Exposé is loaded into a Parka knowledge base.
- Java applets on the TSE Risk Website access the knowledge base to respond to users' queries or update displays. These applets include the TSE Path Analyzer and the Parka Interface.

The following subsections describe how we created our ontology, how SHOE tags were added to web pages, how new SHOE information is discovered, and how users access information that is relevant to them.

2.1 Ontology Design

The fundamental component of SHOE is the ontology. In SHOE, an ontology can extend one or more existing ontologies by adding its own category hierarchies, relations, and inference rules. The excerpts from the TSE ontology shown in Figure 1 give a sample of the SHOE syntax. A complete description of the syntax can be found in the SHOE Specification (Luke and Heflin 1997).

An important problem when designing an ontology is setting an appropriate scope. We asked the following questions to set an initial scope for the TSE ontology:

- What kinds of pages will be annotated?
- What sorts of queries can the pages be used to answer?
- Who will be the users of the pages?
- What kinds of objects are of interest to these users?
- What are the interesting relationships between these objects?

Note that the motivation for web ontologies is slightly different from that of traditional ontologies. People rarely query the web searching for abstract concepts or similarities between very disparate concepts, and as such, complex upper ontologies are not necessary. Since most pages with SHOE annotations will tend to have tags that categorize the concepts, there is no need for complex inference rules to perform automatic classification. In many cases, rules that identify the symmetric, inverse, and transitive relationships will provide sufficient inference.

The initial TSE ontology was fleshed out in a series of meetings that included members of the FDA and the Maryland Veterinarian School. Since one of the key goals was to help risk assessors gather information, the ontology focused on the three main concerns for TSE Risks: source material, processing, and end-product use. Source materials are described using the concepts of *Animal*, *Tissue*, and *DiseaseAgent*. Processing focused on the types of *Processes*, and relations to describe inputs, outputs, duration, etc. Finally, end-product use categorized the types of *Products* and dealt with the

```
…
<BODY>
<ONTOLOGY ID="TSE Ontology" VERSION="1.0">
<USE-ONTOLOGY ID="Base Ontology" VERSION="1.0" PREFIX="base">
…
<DEF-CATEGORY NAME="Disease_Agent" ISA="base.SHOEEntity">
<DEF-CATEGORY NAME="BSE" ISA="Disease_Agent">
<DEF-CATEGORY NAME="CJD" ISA="Disease_Agent">
<DEF-CATEGORY NAME="NV-CJD" ISA="Disease_Agent">
…
<RELATION NAME="hasInput">
  <ARG POS=1 TYPE="Process">
  <ARG POS=2 TYPE="Material">
</RELATION>
<RELATION NAME="hasOutput">
  <ARG POS=1 TYPE="Process">
  <ARG POS=2 TYPE="Material">
</RELATION>
…
</ONTOLOGY>
</BODY>
…
```

Figure 1. Excerpts from the TSE Ontology

RouteOfExposure. We also defined number of general concepts such as *People*, *Organizations*, *Events*, and *Locations*.

Currently, the ontology has 73 categories and 88 relations. It is stored as a file on a web server with an HTML section that presents a human-readable description and a machine-readable section with SHOE syntax. In this way, the file can serve the purpose of educating users in addition to being understandable to machines.

2.2 Annotation

Annotation is the process of adding SHOE semantic markup to a web page. A SHOE web page describes one or more instances, each representing an entity or concept. An instance is uniquely identified by a key, which is usually formed from the URL of the web page. The description of an instance consists of ontologies that it references, categories that classify it, and relations that describe it. A sample instance is shown in Figure 2.

Determining what concepts in a page to annotate can be complicated. First, if the document represents or describes a real world object, then an instance whose key is the document's URL should be created. Second, hyperlinks are often signs that there is some relation between the object in the document and another object represented by the hyperlinked URL. If a hyperlinked document does not have SHOE annotations, it may also be useful to make claims about its object. Third, one can create an instance for every proper noun, although in large documents this may be excessive. If these concepts have a web presence, then that URL should be used as the key, otherwise, unique keys can be created by appending a "#" and a unique string to the end of the document's URL.

Since manually annotating a page can be time consuming and prone to error, we have developed the Knowledge Annotator, a tool that makes it easy to add SHOE knowledge to web pages by making selections and filling in forms. As can be seen in Figure 3, the tool has an interface that displays instances, ontologies, and claims. Users can add, edit or remove any of these objects. When creating a new object, users are prompted for the necessary information. In the case of claims, a user can choose the source ontology from a list, and then choose categories or relations from a corresponding list. The available relations will automatically filter based upon whether the instances entered can fill the argument positions. A variety of methods can be used to view the knowledge in the document. These include a view of the source HTML, a logical notation view, and a view

```
<HTML>
<BODY>
…
<INSTANCE KEY="http://www.cs.umd.edu/projects/plus/SHOE/tse/rendering.html">
<USE-ONTOLOGY ID="TSE-Ontology" VERSION="1.0" PREFIX="tse"
              URL="http://www.cs.umd.edu/projects/plus/SHOE/tse/tseont.html">
<CATEGORY NAME="tse.Process">
<RELATION NAME="tse.name">
  <ARG POS="TO" VALUE="Rendering">
</RELATION>
<RELATION NAME="tse.hasInput">
  <ARG POS="TO" VALUE="http://www.cs.umd.edu/projects/plus/SHOE/tse/offal.html">
</RELATION>
<RELATION NAME="tse.hasInput">
  <ARG POS="TO" VALUE="http://www.cs.umd.edu/projects/plus/SHOE/tse/bones.html">
</RELATION>
<RELATION NAME="tse.hasOutput">
  <ARG POS="TO" VALUE="http://www.cs.umd.edu/projects/plus/SHOE/tse/mbm.html">
</RELATION>
<RELATION NAME="tse.hasOutput">
  <ARG POS="TO" VALUE="http://www.cs.umd.edu/projects/plus/SHOE/tse/tallow.html">
</RELATION>
<RELATION NAME="tse.hasOutput">
  <ARG POS="TO" VALUE="http://www.cs.umd.edu/projects/plus/SHOE/tse/gellatin.html">
</RELATION>
</INSTANCE>
</BODY>
</HTML>
```

Figure 2. Sample Instance

that organizes claims by subject and describes them using simple English. In addition to prompting the user for inputs, the tool performs error checking to ensure correctness[1] and converts the inputs into legal SHOE syntax. For these reasons, only a rudimentary understanding of SHOE is necessary to markup web pages.

We selected pages to annotate with two goals in mind: provide information on the processing of animal-based products and provide access to existing documents related to TSEs. We were unable to locate web pages relevant to the first goal, and therefore had to create a set of pages describing many important source materials, processes and products. To achieve the second goal we selected relevant pages from sites provided by the FDA, United States Department of Agriculture (USDA), the World Health Organization and others. For the pages that we created, we added the SHOE tags inline. Since we did not have the authority to modify the other pages, we created summary pages that basically consisted of the SHOE information and pointers to the originals.

2.3 Information Gathering

The vastness of the Internet and bandwidth limitations make it difficult for a system to perform direct queries on it efficiently. However, if the relevant data is already stored in a knowledge base, then it is possible to respond to queries very quickly. For this reason, we have designed Exposé, a softbot that searches for web pages with SHOE markup and interns the knowledge. However, since a web-crawler can only process information so quickly, there is a tradeoff between coverage of the Web and freshness of the data: if the system revisits pages frequently, then there is less time for discovering new pages. Since we are only concerned with information on TSEs for this project, we chose to limit the sites Exposé may visit, so that it does not waste time exploring pages where there is no relevant information.

In order to use Exposé, we had to choose a knowledge base system for storing the information. The selection of such a system depends on a number of criteria. First, many knowledge base systems cannot handle the volume of data that would be discovered by the web-crawler. Second, the knowledge base system must support the kinds of inference that will be needed by the application. Third, since SHOE allows for n-ary relations, it is useful, though not absolutely necessary, to choose a knowledge base that can support

[1] Here correctness is in respect to SHOE's syntax and semantics. The Knowledge Annotator cannot verify if the user's inputs properly describe the page.

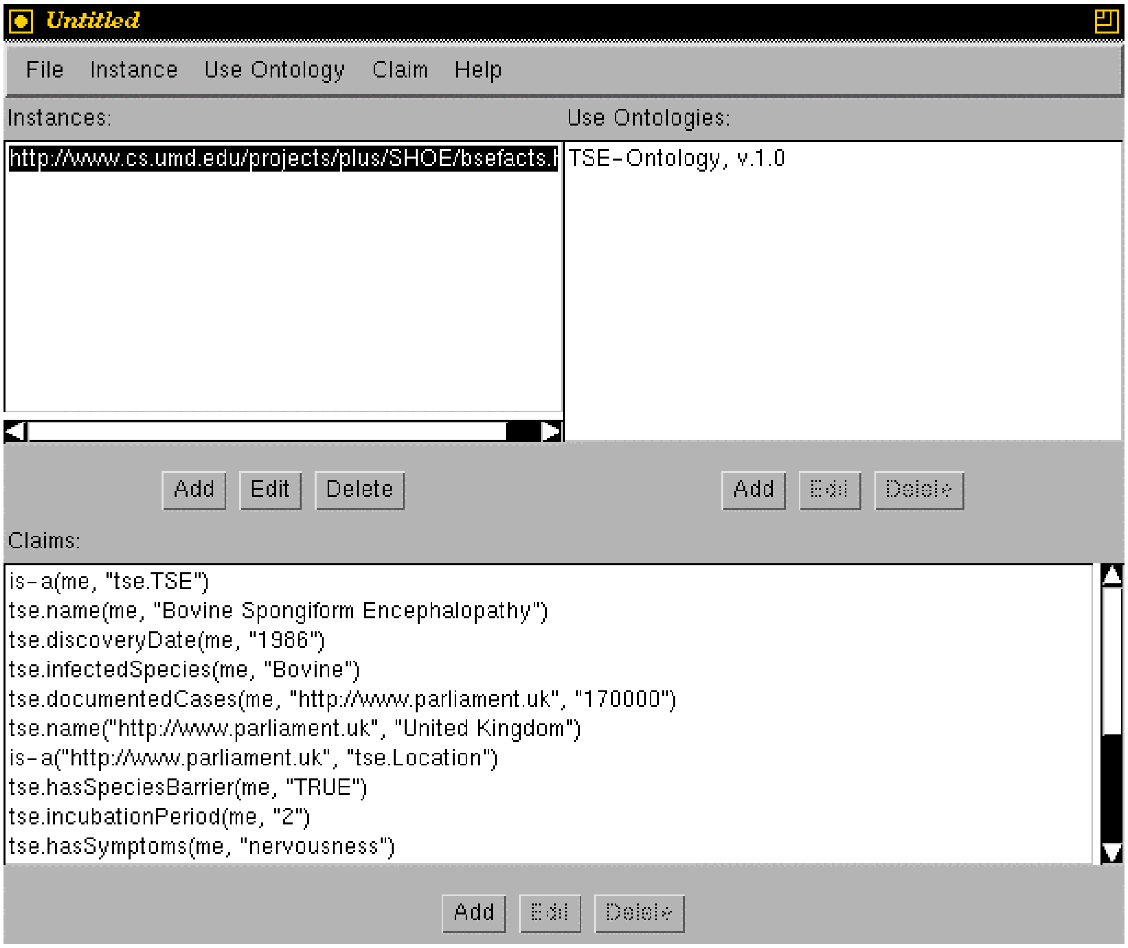

Figure 3. The Knowledge Annotator

them[2]. We chose Parka (Evett, Andersen, and Hendler 1993; Stoffel, Taylor, and Hendler 1997) as our knowledge base because evaluations have shown it to be very scalable, there is an n-ary version, and parallel processing can be used to improve query execution time. Since we were not interested in performing complex inferences on the data at the time, the fact that Parka's only inference mechanism is inheritance was of no consequence.

An important aspect of the Internet is that its distributed nature means that all information discovered must be treated as claims rather than facts. Parka, as well as most other knowledge base systems, does not provide a mechanism for attaching sources to assertions or facilities for treating these assertions as claims. To represent such information, one must create an extra layer of structure using the existing representation. Parka uses categories, instances and n-ary predicates to represent the world. A natural representation of SHOE information would be to treat each declaration of a SHOE relation as an assertion where the relation name is the predicate, and each category declaration as an assertion where *instanceof* is the predicate. To represent the source of the information, we could add an extra term to each predicate. Thus, an n-ary predicate would become an (n+1)-ary predicate. However, the structural links (i.e., *isa* and *instanceof*) are default binary predicates in Parka. Thus, this approach could not be used without changing the internal workings of the knowledge base. We opted for a simpler approach, and instead made two assertions for each claim. The first assertion ignores the claimant, and can be used normally in Parka. The second assertion uses a *claims* predicate to link the source to

---

[2] A binary knowledge base can represent the same data as an n-ary knowledge base, but requires an intermediate processing step to convert an n-ary relation into a set of binary relations. This is inefficient in terms of storage and execution time.

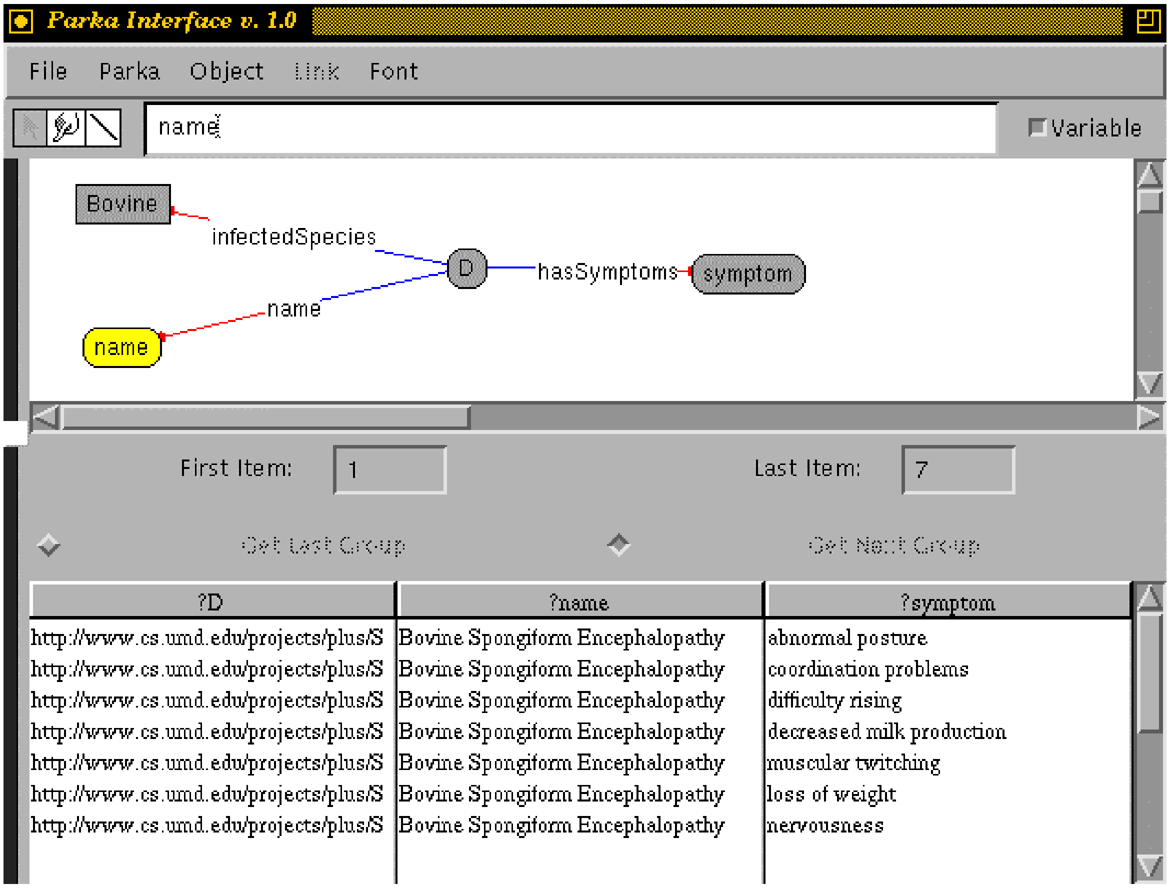

Figure 4. The Parka Interface for Queries (PIQ)

the first assertion. When the source of information is important, it can be retrieved through the *claims* predicate. Although this results in twice as many assertions being made to the knowledge base, it preserves classifcation while keeping queries straightforward.

As designed, the agent will only visit websites that have registered with JIFSAN. This allows JIFSAN to review the sites so that Exposé will only be directed to search sites that meet a certain level of quality. Note that this does not restrict the ability of approved sites to get current information indexed. Once a site is registered, it is considered trusted and Exposé will revisit it periodically.

2.4 User Interfaces

The most important aspect of the system is the ability to provide users with the information they need. Since we are dealing with an internet environment, it is important that users can access this information through their web browsers. For this reason, the tools we have created are Java applets that are available from the TSE website. We currently provide a general purpose query tool and a custom tool built to meet the needs of TSE community.

The Java Parka Interface for Queries (PIQ), as shown in Figure 4, is a graphical tool that can be used to query any Parka knowledge base. This interface gives users a new way to browse the web by allowing them to submit complex queries and open documents by clicking on the URLs in the results. A user inputs a query by drawing frames and the relations between them. This specifies a conjunctive query in which the frames are either constants or variables and the relations can be a string matching function, a numerical comparison or a relation defined in an ontology. The answers to the query are displayed as a table of the possible variable bindings. If the user double-clicks on a binding that is a URL, then the corresponding web page will be opened in a new window of the user's web browser.

It is widely believed that the outbreak of BSE in Great Britain was the result of changes in rendering practices. Since processing can lead to the inactivation or spread of a

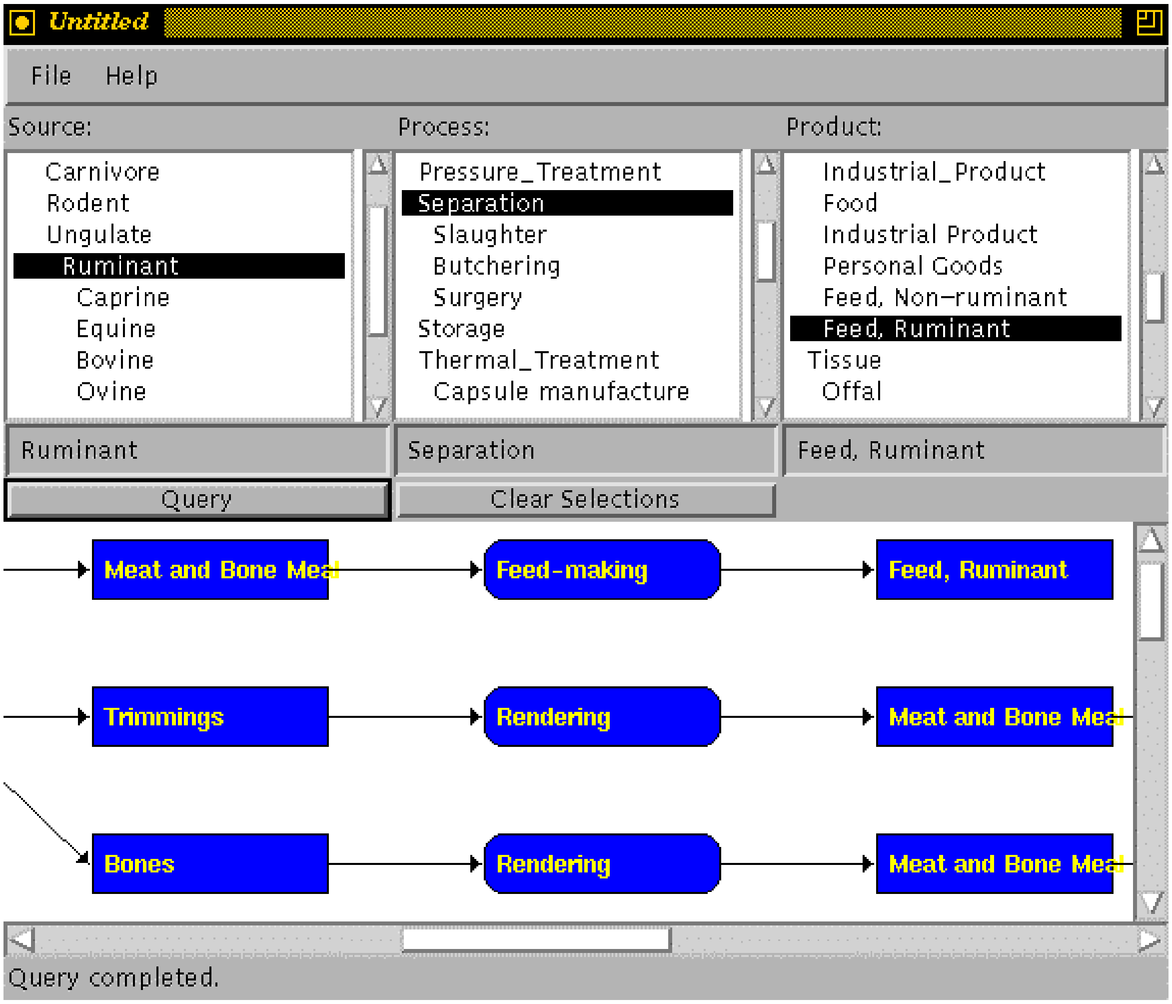

Figure 5. The Path Analyzer

disease, JIFSAN expressed a desire to be able to visualize and understand the processing of animal materials from source to end-product. To accommodate this, we built the TSE Path Analyzer, a graphical tool which allows the user to pick a source, process and/or end product and view all possible pathways that match their queries. The input choices are derived from the taxonomies of the ontology, allowing the user to specify the query at the level of generality that they wish. This display, which can be seen in Figure 5, is created dynamically based on the semantic information in the SHOE web pages. As such, it is automatically updated as new information becomes available, including information that has been made available elsewhere on the web.

Since both these interfaces are applets, they are executed on the machine of each user who opens it. This client application communicates with the central Parka knowledge base through a Parka server that is located on the JIFSAN website. When a user starts one of these applets on their machine, the applet sends a message to the Parka server. The server responds by creating a new process and establishing a socket for communication with the applet.

3. Lessons Learned

This research has given us many insights into the use of ontologies in providing access to internet information. The first insight is that it is worthwhile to spend time getting the ontology "right". By "right", we mean that it must cover the concepts in the types of pages that are to be used and the ways in which these pages will be accessed. We often had to extend our ontology to accommodate concepts in pages that we were annotating, and this slowed the annotation process.

Second, real world web pages often refer to shared entities such as BSE or the North American continent. Such concepts may be described in many web pages, none of which should have the authority to assign a key to them. In such cases, we revise the appropriate ontologies to include a constant for the shared object. However, this may result in frequent updates if the ontology is used extensively.

Third, ordinary web-users do not have the time or desire to learn to use complex tools. Although the PIQ is easy to use once one has gained a little experience with it, it can be intimidating to the occasional user. On the other hand, users liked the Path Analyzer, even though it can only be used to answer a restricted set of queries, because it presents the results in a way that makes it easy to explore the problem. It seems web users are often willing to sacrifice power for simplicity.

Finally, the knowledge base must be able to perform certain complex operations as a single unit. For example, the Path Analyzer needs to display certain descendant hierarchies. Although such lists can be built by recursively asking for the immediate children of the categories retrieved in the last step, this requires many separate queries. In a client-server situation this is expensive, since each query requires its own communication overhead and internet transmission delays can be significant. To improve performance, we implemented a special server request that returns the complete set of parent-child pairs that form a hierarchy. Although this requires the same amount of processing by the knowledge base, it results in a significant speedup of the client application.

4. Related Work

The World-Wide Web Consortium (W3C) has proposed the Extensible Markup Language (XML) (Bray, Paoli, and Sperberg-McQueen 1998) as a standard that is a simplified version of SGML (ISO 1986) intended for the Internet. XML allows web authors to create customized sets of tags for their documents. Style sheets can then be used to display this information in whatever format is appropriate. SHOE is a natural fit with XML: XML allows SHOE to be added to web pages without creating an HTML variant, while SHOE adds to XML a standard way of expressing semantics within a specified context. The Resource Description Framework (RDF) (Lassila and Swick 1998) is another work in progress by the W3C. RDF uses XML to specify semantic networks of information on web pages, but has no inferential capabilities and is limited to binary relations.

There are many other projects that are using ontologies with the Web. The World Wide Knowledge Base (WebKB) project (Craven et al. 1998) is using ontologies and machine learning to attempt automatic classification of web pages. The Ontobroker (Fensel et al. 1998) project has resulted in a language which, like SHOE, is embedded in HTML. Although the syntax of this language is more compact, it is not as easy to understand as SHOE. Also, Ontobroker does not have a mechanism for pages to use multiple ontologies and those who are not members of the community have no way of discovering the ontology information.

5. Future Work

The JIFSAN TSE Website is a work in progress, and we will continue to annotate pages, refine the ontology, and improve the tool set. When we have accumulated a significantly large and diverse set of annotated pages, we will systematically evaluate the performance of SHOE relative to other methods. We also plan to develop a set of reusable ontologies for concepts that appear commonly on the Web, so that future ontologies may be constructed more quickly and will have a commonality that allows for queries across subject areas when appropriate.

To gain acceptance by the web community, a new language must have intuitive tools. We plan to create an ontology design tool that simplifies the ontology development process. We also plan to improve the Knowledge Annotator so that more pages can be annotated more quickly. We are particularly interested in including lightweight natural language processing techniques that suggest annotations to the users. Finally, we are investigating other query tools with the goal of reducing the learning curve while still providing the full capabilities of the underlying knowledge base.

6. Conclusion

The TSE Risk Website is the first step in developing a clearinghouse on food safety risks that serves both the general public and individuals who assess risk. SHOE allows this information to be accessed and processed in powerful ways without constraining the distributed nature of the sources. Since SHOE does not depend on keyword matching, it prevents the false hits that occur with ordinary search engines and finds other matches that they cannot. Additionally, the structure of SHOE allows intelligent agents to process the information from many sources and combine or present it in novel ways.

We have demonstrated that SHOE can be used in large domains without clear boundaries. The methodology and tools we have described in this paper can be applied to other subject areas with little or no modifications. We have determined that the hardest part of using SHOE in new domains is creating the ontology, but we are convinced that as high quality ontology components are made available, this process will be simplified. We are encouraged by the interest that our initial efforts have generated in the TSE community, and believe that improvements in our tools and the availability of basic ontologies will lead to an internet where the right data is always available at the right time.

Acknowledgments

This work was supported by the Army Research Laboratory under contract number DAAL01-97-K0135.

References

Bray, T., J. Paoli and C.M. Sperberg-McQueen. 1998. *Extensible Markup Language (XML)*. W3C (World-Wide Web Consortium). (At http://www.w3.org/TR/1998/REC-xml-19980210.html)

Craven, M., D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigram and S. Slattery. 1998. Learning to Extract Symbolic Knowledge from the World Wide Web. In *Proceedings of the AAAI-98 Conference on Artificial Intelligence*. AAAI/MIT Press.

Evett, M.P., W.A. Andersen and J.A. Hendler. 1993. Providing Computational Effective Knowledge Representation via Massive Parallelism. *In Parallel Processing for Artificial Intelligence*. L. Kanal, V. Kumar, H. Kitano, and C. Suttner, Eds. Amsterdam: Elsevier Science Publishers.

Fensel, D., S. Decker, M. Erdmann, and R. Studer. 1998. Ontobroker: How to enable intelligent access to the WWW. In *AAAI-98 Workshop on AI and Information Integration*. Madison, WI.

ISO (International Organization for Standardization). 1986. ISO 8879:1986(E). *Information processing -- Text and Office Systems -- Standard Generalized Markup Language (SGML)*. First edition -- 1986-10-15. [Geneva]: International Organization for Standardization.

Lassila, O. and R.R. Swick. 1998. *Resource Description Framework (RDF) Model and Syntax*. W3C (World-Wide Web Consortium). At http://www.w3.org/TR/WD-rdf-syntax-19980216.html.

Luke, S. and J. Heflin. 1997. *SHOE 1.0, Proposed Specification*. At http://www.cs.umd.edu/projects/plus/SHOE/spec.html

Stoffel, K., M. Taylor and J. Hendler. 1997. Efficient Management of Very Large Ontologies. In *Proceedings of American Association for Artificial Intelligence Conference (AAAI-97)*. AAAI/MIT Press.