Consensus Ontologies

Reconciling the Semantics of Web Pages and Agents

Larry M. Stephens • *University of South Carolina • stephens@sc.edu*
Michael N. Huhns • *University of South Carolina • huhns@sc.edu*

I n an old joke, a drunk is on his hands and knees searching for his keys underneath a lamppost. "Is this where you dropped them?" he is asked. "No, I dropped them over there, but the light is better here."

As you build a Web site, it is worthwhile to ask a similar question: "Should you put your information where it belongs or where people are most likely to look for it?" Our recent research to improve search through ontologies is providing some interesting results for answering this question.

Reconciling Web Semantics

Web searches typically yield pointers to a large number of Web sites — only some of which are relevant. Search engines might rank the sites, but the results are otherwise unorganized and too numerous for users to investigate manually. Many solutions have been proposed to this familiar problem, including constructing more intelligent search engines, requiring users to specify more precise search criteria, or requiring Web sites to describe their contents more precisely.

These approaches all use ontologies to describe both requirements and sources. Unfortunately, the comprehensive ontology that could solve the problem of information retrieval does not yet exist. Moreover, the Web's dynamic and eclectic nature makes it unlikely that everyone would adhere to such an ontology if it did.

To overcome these limitations, Web developers could choose among three possible approaches to associate, organize, and merge information semantically from large numbers of independently developed sources:

- All Web sites could use the same terminology with agreed-upon semantics — a method considered improbable.
- Each Web site could use its own terminology and provide translations to a global ontology — a method considered difficult, and thus unlikely.
- Each Web site could use small, local ontologies that can be related indirectly with the assistance of agents — a method we describe here.

Our methodology is consistent with the envisaged semantic Web,[1] which presumes that Web sources will be annotated with ontological information.[2,3] We also presume that the independently developed sources and ontologies returned from a Web search are for similar domains — there would be no interesting relationships among them otherwise — but that they will undoubtedly have dissimilar formulations and terminologies.

Our hypothesis is that a multiplicity of ontology fragments, representing the semantics of the independent sources, can be related to each other automatically without using a global ontology. That is, even when there is no way to determine a direct relationship between a pair of ontologies, they can be related indirectly through a *semantic bridge* consisting of other previously unrelated ontologies. Rather than scale causing a problem, additional ontologies can make it easier — or even possible — to relate two ontologies. The resultant merged ontologies provide a semantic characterization of the set of sources and their domains, and effectively create a single large ontology to serve as a global hub for interactions. This methodology establishes a means for agents and

other information system components to interoperate.

Reconciling Separately Developed Ontologies

In agent-assisted information retrieval, a user will describe a need to an agent, which will use terms from the user's local ontology to translate the description into a set of requests. The agent will contact online brokers and request help in locating sources that can satisfy the requests. The agents must reconcile their semantics to communicate about the request, which seems impossible if their ontologies share no concepts. If they share concepts with a third ontology, however, that one might provide a semantic bridge to relate all three. Note that the agents need to relate only the portions of their ontologies that are necessary for responding to the request.

The difficulty in establishing a bridge will depend on the semantic distance between the concepts, and on the number of ontologies that constitute the bridge. The methodology we are investigating is appropriate with large numbers of small ontologies — the situation we expect to find in complex information environments. A small ontology is like one piece in a jigsaw puzzle: it is difficult to relate two random puzzle pieces until they are constrained by others. We expect the same to be true for ontologies.

Two concepts can have seven mutually exclusive relationships between them: *subclass*, *superclass*, *equivalence*, *partOf*, *hasPart*, *sibling*, or *other*. If a request contains three concepts, for example, and the request must be related to an ontology containing 10 concepts, then there are 7 x 3 x 10 = 210 possible relationships among them. Only 30 of these will be correct because each of the three concepts in the request will have exactly one relationship with each of the 10 concepts in the source's ontology. The correct relationships will be determined automatically by applying constraints among the concepts within each ontology as well

as constraints discovered among multiple ontologies. The relationships of major interest are *equivalence* and *sibling*. Where those do not exist, we are interested in the most specific *superclass* or most specific *partOf*.

Consider the example in Figure 1a. The ontology fragment on the left would be represented as *partOf(Wheel, Truck)* and the one on the right would be *partOf(Tire, APC)*. There are no obvious relationships between these two fragments. The concept *Truck* could be related to *APC* by *equivalence*, *partOf*, *hasPart*, *subclass*, *superclass*, or *other*, and there is no way to decide which is correct. Now consider the addition of the middle ontology fragment *partOf(Wheel, APC)* in Figure 1b. With this added information, there is evidence that we could link the concepts *Truck* and *APC* as equivalent as well as the concepts *Wheel* and *Tire*.

This example exploits the *partOf* relation, which is common to all three ontologies. Other domain-independent relations, such as *subclassOf*, *instanceOf*, and *subrelationOf*, will be necessary for the reconciliation process. Moreover, the following six properties of relations can help in relating occurrences of the relations to each other: reflexivity, symmetry, asymmetry, transitivity, irreflexivity, and antisymmetry.[6] Domain concepts and relations can be related to each other by converse/inverse, composition, (exhaustive) partition, part-whole (with six subtypes), and temporal attitude. All local ontologies and information system components must understand and use some minimum set of these fundamental relations.

In attempting to relate two ontologies, a system might not be able to find correspondences between concepts because not enough constraints and similarities exist among terms.[5] Trying to locate correspondences with other ontologies, however, might yield
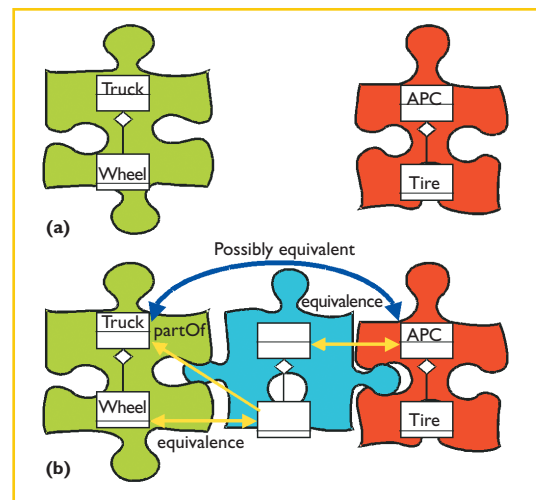

Figure 1. Ontology relationships. Two ontology fragments with no obvious relationships (a) can be related by introducing a third ontology to reveal equivalences between components of the original two fragments (b).

enough constraints to relate the original pair. As more ontologies are related, additional constraints arise among the terms of any pair of ontologies. In this way, the presence of many small ontologies becomes an advantage. It is also a disadvantage in that some constraints might conflict, but we use the preponderance of evidence to resolve these statistically.

Our Experiments

We asked 55 graduate students in computer science and engineering to construct small ontologies in DAML (DARPA Agent Markup Language, http://www.daml.org/) for the given domain of People. Figure 2 on the next page shows a typical example of one of these ontologies.

The 55 component ontologies described 864 classes. Using a string-matching algorithm and other heuristics, we constructed one merged ontology from these (shown in Figure 3) that contained 281 classes in a single graph with the root node #Thing. This graph related all of the concepts from the ontologies with no orphans — that is, there was some relationship (path) between every pair of concepts.

We constructed a *consensus ontology* during the merge operation by counting the number of times classes and
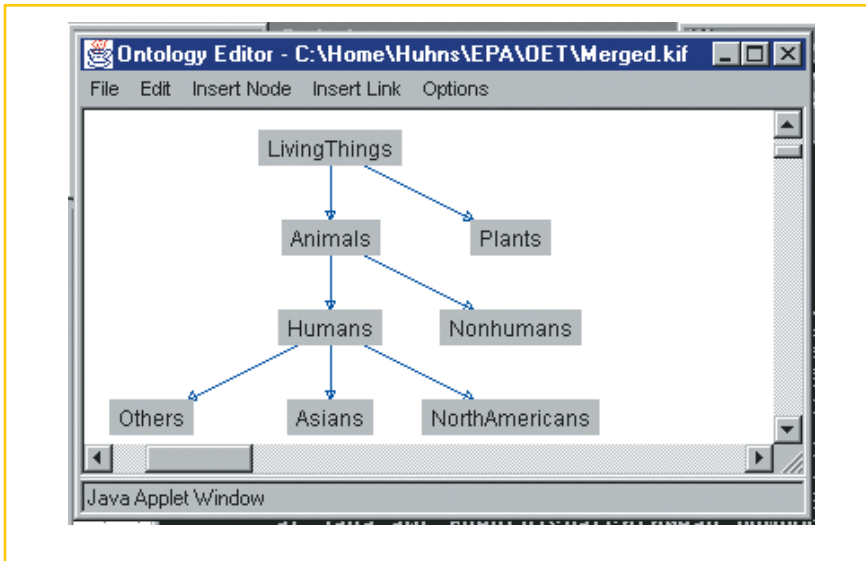
Figure 2. Typical ontology. Students used DAML to create small ontologies like this one to characterize a Web site about People. All links denote subclasses.
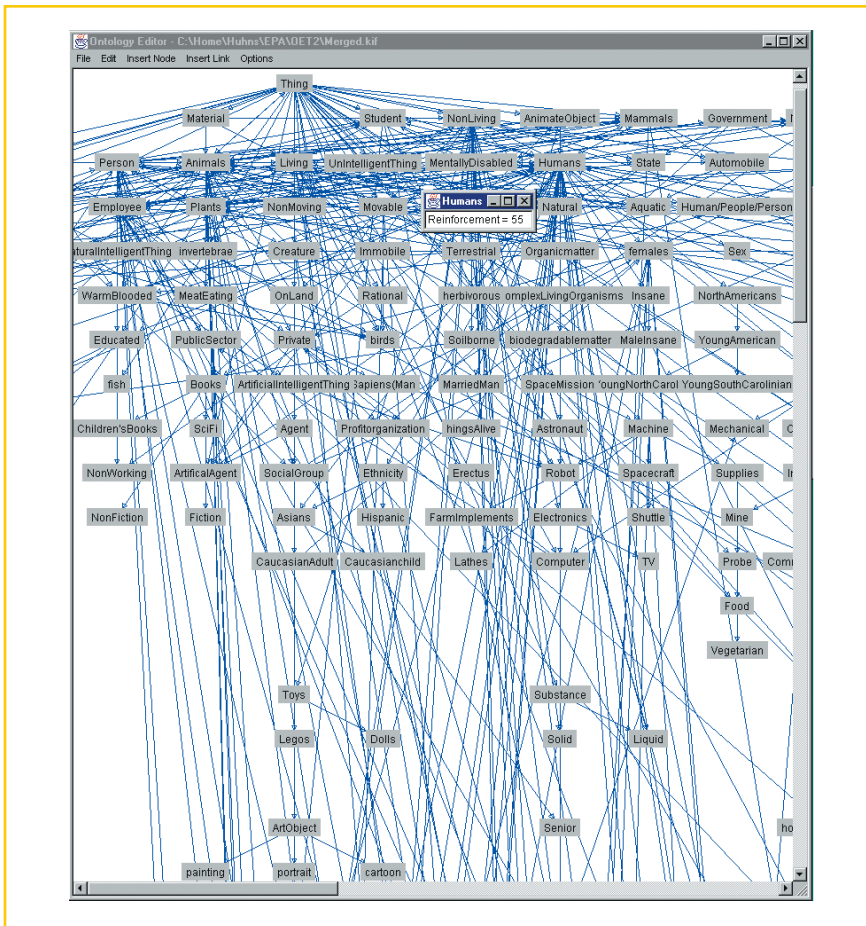

Figure 3. Merged ontology from 55 independently constructed ontologies for the People domain. Of 281 concepts, 38 with 71 subclass links appear more than once.

subclass links appeared in the component ontologies. The class Person, and all similar classes such as Persons and Personnel whose names matched using our simple string-matching algorithm, appeared 14 times, for example. The subclass link from Mammals (and its matches) to Humans (and its matches) appeared 9 times. We termed these values the *reinforcement* of a concept.

Redundant subclass links were removed and the corresponding transitive closure links were reinforced. That is, if C had subclass A with reinforcement 2, C had subclass B, and B had subclass A, then the link from C directly to A was removed and the remaining link reinforcements from C to B and B to A were each increased by 2. We then removed any classes or links that were not reinforced by appearing multiple times in the merged ontology. The result represents an implicit consensus among the ontology writers about which concepts should appear in the domain and how they should be related.

Finally, we applied an *equivalence heuristic* for collapsing classes with common reinforced superclasses and subclasses. The merged ontology contains both Human and Person, for example; The equivalence heuristic found that all reinforced subclasses of person are also reinforced subclasses of human, and all reinforced superclasses of person are also reinforced superclasses of humans. It thus deemed that human and person were the same concept. This heuristic is similar to an inexact graph matching technique such as the one used by Manocha et al. Figure 4 shows the collapsed consensus ontology, now containing 36 classes related by 62 subclass links.

Discussion of Results

In analyzing the 55 ontologies, we noted immediately that each student had a different way of describing and organizing the domain — even for a domain as familiar and simple as people. It was also apparent that the descriptions were inaccurate and contradictory. Mammals, for example, were described as both a subclass and a superclass of animals.

A consensus ontology is perhaps the most useful for information retrieval by humans because it represents the way most people view the world and

its information. If most people wrongly believe that crocodiles are a kind of mammal, for example, then most users would find it easier to locate information about crocodiles located in a mammals grouping, rather than in reptiles where it factually belongs.

The information retrieval measures of *precision* and *recall* are based on some degree of match between a request and a response. The length of a semantic bridge between two concepts can provide an alternative measure of conceptual distance and an improved notion of information relevance.[3] Previous measures relied on the number of properties shared by, or the number of links separating, two concepts within the same ontology. These measures not only require a common ontology, but also fail to account for the density or paucity of information about a concept. Our suggested measure does not require a common ontology and is sensitive to the amount of information available in the domain.[6]

Conclusion

Imagine again that in response to a request for information, a user receives pointers to more than 1,000 documents. The techniques developed by our research would bring organization to the information received and would reconcile the semantics of each document. Our goal is to help users retrieve dynamically generated information that is tailored to their individual needs and preferences.

We believe that it is easier for individuals or small groups to develop their own ontologies, regardless of whether global ones are available, and that these can be automatically and ex post facto related. We are working to determine the efficacy of local annotation for Web sources, as well as performing reconciliation that is qualified by measures of semantic distance. If successful, this research will enable software agents to resolve the semantic misconceptions that inhibit successful interoperation with other agents and that limit the effectiveness of searching distributed information sources.
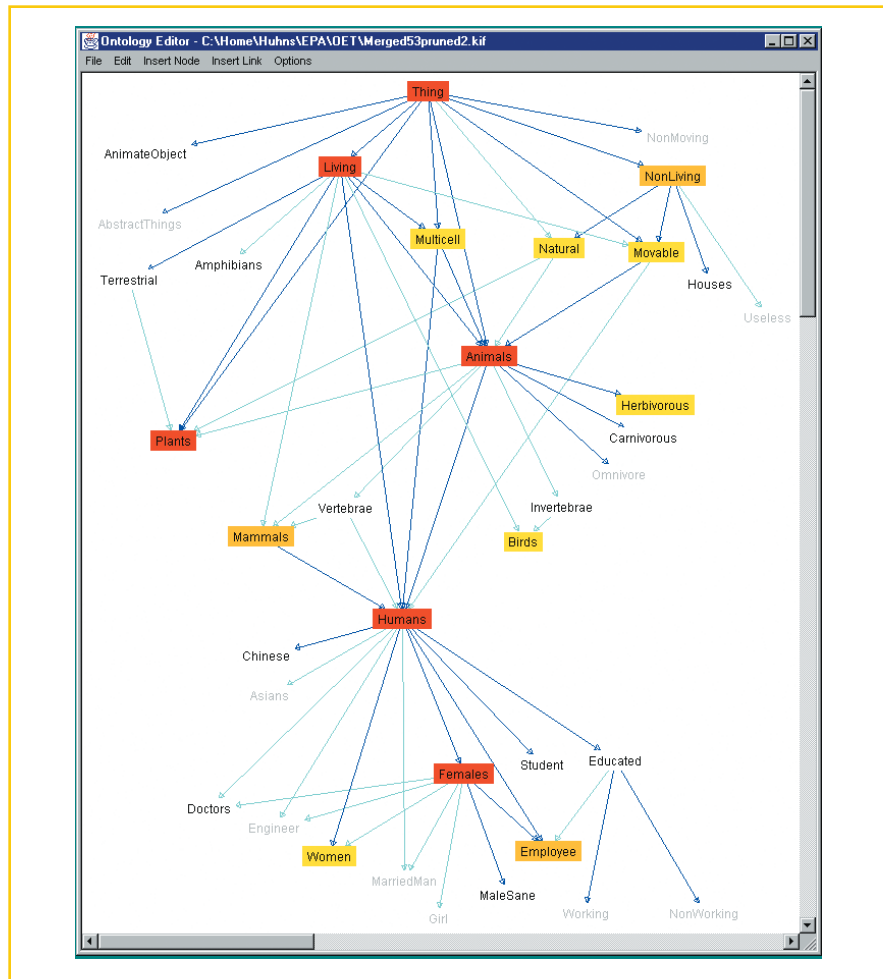

Figure 4. Consensus ontology. Weakly reinforced concepts were removed and concepts with common subclasses and superclasses were merged to produce this graph containing 36 concepts related by 62 subclass links.

References
1. T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Scientific American*, May 2001; available online at http://www.sciam.com/2001/0501issue/.
2. J. Heflin and J. Hendler, "Dynamic Ontologies on the Web," *Proc. 17th Nat'l Conf. Artificial Intelligence (AAAI-2000)*, AAAI Press, 2000, pp. 443-449.
3. J.M. Pierre, "Practical Issues for Automated Categorization of Web Sites," *Electronic Proc. ECDL 2000 Workshop on Semantic Web*, Sept. 2000; available at http://www.ics.forth.gr/proj/isst/SemWeb/program.html.
4. L.M. Stephens and Y.F. Chen. "Principles for Organizing Semantic Relations in Large Knowledge Bases," *IEEE Trans. Knowledge and Data Engineering*, vol. 8, no. 3, June 1996, pp. 492-496.
5. G. Wiederhold, "An Algebra for Ontology Composition," *Proc. 1994 Monterey Workshop on Formal Methods*, U.S. Naval Postgraduate School, 1994, pp. 56-62.
6. H.S. Delugach, "An Exploration Into Semantic Distance," *Lecture Notes in Artificial Intelligence*, No.754, Springer-Verlag, Berlin, 1993, pp. 119-124.

Michael N. Huhns is a professor of computer science and engineering at the University of South Carolina, where he also directs the Center for Information Technology.

Larry M. Stephens is a professor of computer science and engineering at the University of South Carolina, where his primary research interests are agents and ontologies.