AI Magazine Volume 26 Number 4 (2006) (© AAAI)

Knowledge Is Power: A View from the Semantic Web

James Hendler

■ The emerging Semantic Web focuses on bringing knowledge representationlike capabilities to Web applications in a Web-friendly way. The ability to put knowledge on the Web, share it, and reuse it through standard Web mechanisms provides new and interesting challenges to artificial intelligence. In this paper, I explore the similarities and differences between the Semantic Web and traditional AI knowledge representation systems, and see if I can validate the analogy "The Semantic Web is to KR as the Web is to hypertext."

Let me start with three quotations. The first comes from a tutorial on expert systems written by Robert Engelmore with Edward Feigenbaum in 1993. He wrote

> … but in knowledge resides the power. Because of the importance of knowledge in expert systems and because the current knowledge acquisition method is slow and tedious, much of the future of expert systems depends on breaking the knowledge acquisition bottleneck and in codifying and representing a large knowledge infrastructure.[1]

(Keep an eye on that last phrase "a large knowledge infrastructure"; we will return to it several times.)

The second quotation is from Tim Berners-Lee, the inventor of the World Wide Web, in his keynote talk at the first World Wide Web Conference held in 1994, around the same time as Engelmore penned the preceding quote. Berners-Lee stated:

> …documents on the Web describe real objects and imaginary concepts, and give particular relationships between them… The title document to a house describes a house and also the ownership relation with a person. This means that machines, as well as operating on the Web information, can do real things. For example, a program could search for a house and negotiate transfer of ownership of the house to a new owner. The land registry guarantees that the title actually represents reality.[2]

In Berners-Lee's view, the then young World Wide Web, could be seen as a Web not just of documents, but of concepts, where the information described, if liberated from the text, would provide information that machines could use for new, dare I say "knowledge based," applications.

The third quotation is again from Berners-Lee, this time joined by Ora Lassila of Nokia Corporation and me, writing in a *Scientific American* article titled "The Semantic Web" that appeared in 2001. In essence, we wrote about unifying the two earlier visions and said:

> Knowledge representation … is currently in a state comparable to that of hypertext before the advent of the Web: it is clearly a good idea, and some very nice demonstrations exist, but it has not yet changed the world. It contains the seeds of important applications, but to realize its full potential it must be linked into a single global system (Berners-Lee, Hendler, and Lassila 2001)

That is, we argued that a knowledge representation (KR) system that could live on the Web, and function in a weblike way, could provide the "large knowledge infrastructure" that Engelmore had said was needed to "break the knowledge acquisition bottleneck" that held back the use of AI techniques across a wide range of applications.

In this article, I revisit the Semantic Web vision that we espoused in a more AI-specific way. I will explore where we are with respect to that vision four years later, look at some of the techniques and tools we have available, and briefly present some of the challenges to AI inherent in this vision. I will examine some of the differ-

ences between what the Semantic Web brings and the AI tradition of knowledge representation languages, exploring the hypothesis inherent in our article, that the Semantic Web is to KR, what the Web was to hypertext.

Imagine If…

Here are three scenarios to challenge your thinking—imagine the knowledge-rich applications you could build if these things were out there now.

Scenario 1: Document Metadata Knowledge Bases

We're used to thinking of the Web as a web of documents, linked together in a multibillion page book that, unfortunately, is organized by few if any principles. It's the encyclopedia of everything, with no index except what can be found through keyword search. But what could you do if more of the pages in the book had even simple metadata? That is, what could you do with the knowledge base you could create if every PDF document out there on the Web had easily extractable meta-data in a standard machine-readable form? You could easily write a tool to tell who created each document, when, with what application, and what some of its key document features were.

That may not seem like much information per document, but there are hundreds of millions of PDF documents out there on the Web! With a few bits of information from each one, we could be looking at a knowledge base with hundreds of billions of facts about what was published when and by whom. It would be commonplace to know the history of a document you found on the Web, or to see who was authoring with whom, or what was the mean time for change of a Web publication, or a myriad of other ways that one could enhance the raw document sets available on the Web.

Scenario 2: Semantic Annotation of Nontext Media

Sharing photographs on the Web has become commonplace, with many sites allowing photos to be posted and friends to be notified in a tightly controlled way. Recently, however, a new phenomenom in photo sharing has been the advent of sites like flickr[3] which allow not just the posting of photographs, but the ability to post some simple text annotations about what is in them. Thus, a few keywords describing a photo can be attached, allowing flickr to use keyword approaches to index and search for photographs.

But imagine what you could do if instead of simple text annotation, you could mark up images, or parts thereof, with more precise information about the content of a photo. Instead of a sentence fragment such as "the crew of STS-100," you could use some KR about space missions to make it easy for you to state who was the commander of the mission (and where he or she was in the photo), what the payload of the mission was, who else was on the crew, what the date and time of the mission was, and the like.

Even better, assume that the very rich content metadata that you created in this way could be submitted to a Web portal and show up indexed under the appropriate terms. Submitting your photographs to a NASA site, using this enhanced markup, your photographs could be found by students looking for information about a particular shuttle mission, by scientists looking for photographs of a particular payload, or by fans looking for photos of their favorite astronaut.

And better still, imagine that the portals were not necessarily tied to a particular application vocabulary like space, but that you could mark up the same or different photographs against many different kinds of vocabularies. A search on the Web would find you vocabularies about space or biology or scuba diving or terrorist networks or whatever else you were working on. Your personal or group portal would expose what vocabulary was used, and related concepts between them. You could browse from the photograph of a conference to the conference Web site, or to other people in the photograph, or to the restaurant that the conference attendees were photographed in front of, and more. Think of flickr on steroids, powered by ontologies that were easy to find and use!

Scenario 3: Large Scale Knowledge Infrastructure

Now let's imagine that applications like the preceding are becoming more commonplace, and that the products of the metadata, and content fields, and annotations, and so on were becoming easier to find. Imagine there was a distributed knowledge base, accessible through any computer anywhere with a Web browser, that had information about millions of people, places, things, transactions, processes, services, and well, everything! Imagine the facts about all these things could be indexed against thousands of ontologies (in a standard and widely available KR language), and that there were dozens of open-source and freely available tools for parsing, serializing, browsing, editing, storing, searching, and inferencing all these things.

Imagine, in essence, what current AI re-

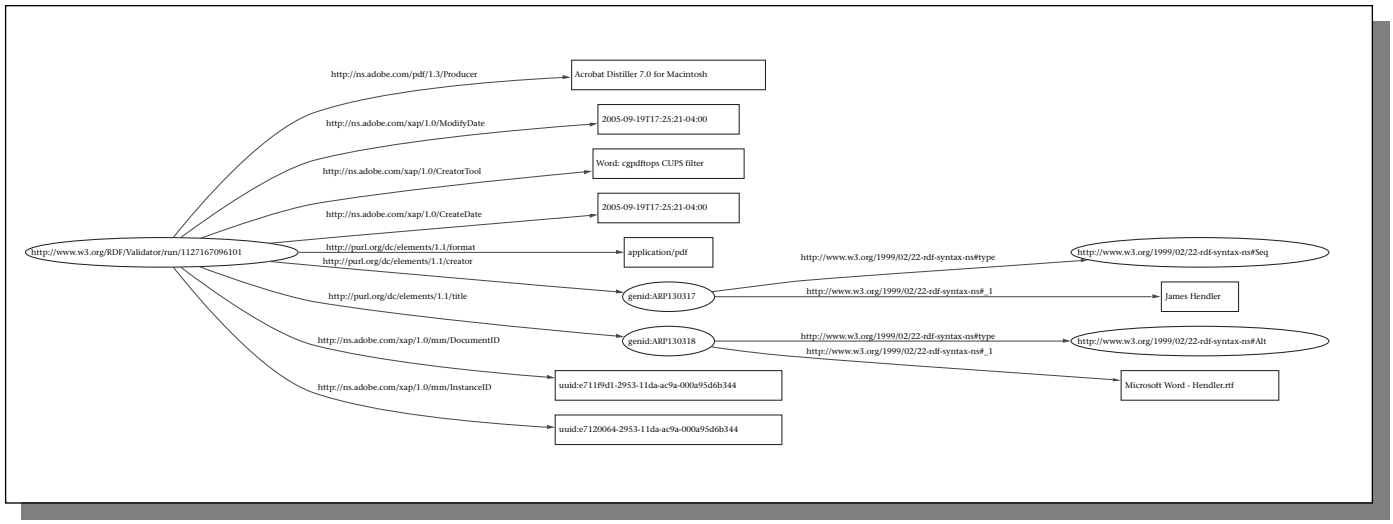

Figure 1. The Graph of the Metadata of This Article Saved in PDF.

Showing the author and the creation and modification dates and other simple metadata.

searchers could do with the very "large scale knowledge infrastructure" that Bob Engelmore described. Imagine what you could do with this enormous, Web-scale knowledge base. The possibilities are endless…

Stop Imagining…

The scenarios in the preceding section are not in the distant future, or even "just around the corner." Each of the scenarios described above is already in existence and growing at a rapid pace. Let's visit them each in turn.

Rich Metadata

Every PDF document created by Adobe Acrobat 6.0 (or, in fact, any recent tool in the Adobe creativity suite and many of the tools that do automatic saving of documents as PDF) is automatically tagged in a format called XMP[4] with some document metadata using tags from the Dublin Core Metadata Initiative.[5] XMP uses a subset of the Resource Description Framework (RDF) (I will discuss this in greater detail later on in this article) and embeds it in the document in an easily extractable way (with tools freely available from the Adobe Web site[6]). What all that means is that embedded in your document is an XML encoding of some simple metadata that is easily machine interpretable and not dissimilar from an AI "semantic network" graph. In fact, easily available tools can indeed turn this metadata into such a graph. For example, figure 1 shows the embedded metadata extracted from the PDF version of this article that I am sending to the publisher, which I have run through a tool that draws the graph encoded in the RDF.

Unique document IDs are assigned to each document and version, and these can be linked together in a straightforward way (essentially merging those nodes with the same ID). So a semantic network with the metadata from the huge number of documents already containing XMP, and from the thousands being created every day is now easily extractable. Couple such a Web with text mining, coindexing, learning, or other such AI approaches, and the potential is huge.

Content Annotation

Given the popularity of Web sites like flickr, it is not surprising that there is a great deal of research, and now productization, exploring this space. Several of the more advanced of these projects are exploring ontology-based annotation of images and portals that can index and provide search of annotated images and other Web resources.[7] One such project is the Photostuff tool developed by my research group,[8] which allows ontologies written in the Web-standard ontology language OWL (like RDF, this topic will be discussed later on in this article) to be used to annotate pictures.

Figure 2 shows an example of the use of Photostuff. In this case, the user of the tool has loaded an ontology about NASA crews from a Web site and is using it to mark up the photograph of a space shuttle crew from a particular mission (STS-100). The user can draw a box around part of the image and then drag a term from the ontology to that region to indicate that an instance of a particular class is being created. In this case, the user has drawn an image around a particular person and indicated that this is the "payload commander." The tool

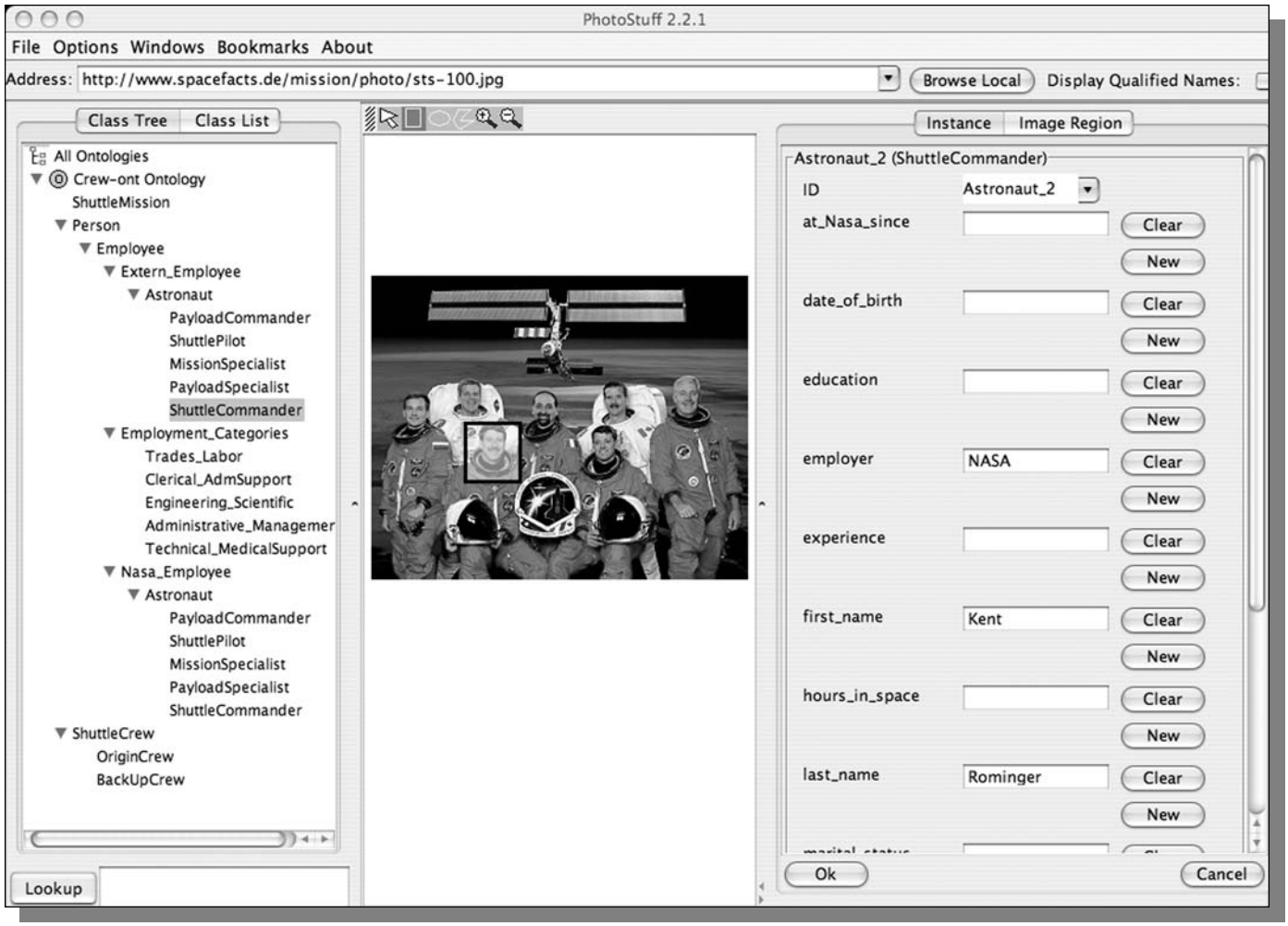

Figure 2. Photostuff Being Used to Mark Up Information About a NASA Mission Using an Ontology about NASA Crews.

The pane on the left shows the classes in the ontology as a tree, the middle pane shows the photograph being marked up, and the right hand pane is a form, generated by the tool from the ontology, allowing the user to fill in the information about this particular astronaut.

then extracts from the ontology the properties associated with that class and creates a simple form on which the user can enter the facts about the particular instance. In this case, the user is indicating that the payload commander is named "Kent Rominger" and that his employer is "NASA." The information about this instance is recorded in a Web document that can then be submitted to a Web site through a WebDav or other submission interface.

Photostuff was designed to work with Semantic Web portal technology that was also developed by our research group (Goldbeck, Alford, and Hendler 2003). We have created a number of specialized sites for viewing information about different domains, including space information, terrorism, biology, U.S. politicians, as well as a general portal for information about our research group and the work we do.[9] These Semantic Web portals can index information based on the ontologies to provide search based on these terms. Thus, for example, figure 3a shows the photographs available from our space portal[10] as they are indexed, and figure 3b shows the result of clicking on the STS-100 mission (one that Kent Rominger commanded).

Large-Scale Knowledge Infrastructure

The Semantic Web at the time I write this article is well past the starting gate. Earlier in this article I suggested we could imagine millions of facts, thousands of ontologies and dozens of tools all available through any machine on the Web. Those numbers are already here and are easy to validate.

On the Semantic Web, the facts are recorded using the RDF language alluded to earlier. RDF can be used to create, essentially, semantic networks that are encoded on Web documents, and linked to each other through the use of

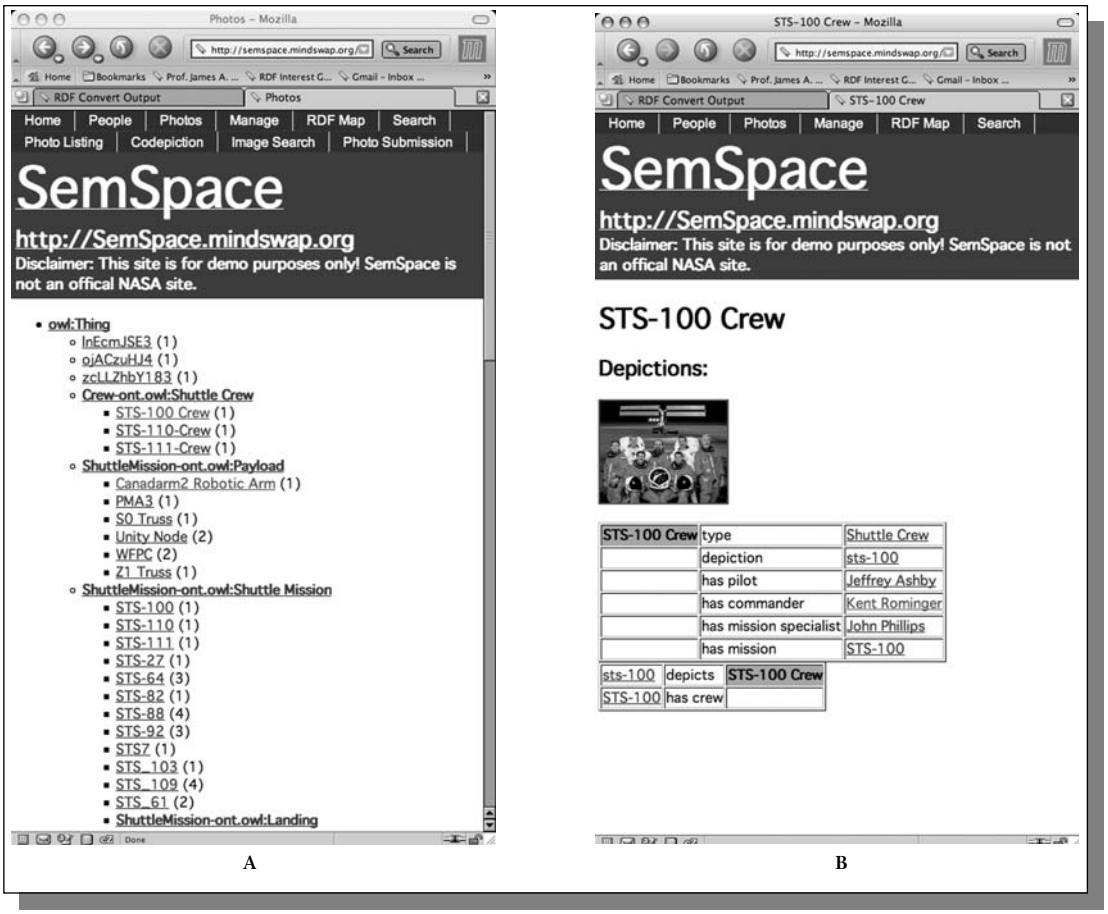

Figure 3. SemSpace Website.

A. The Web site displays photos indexed by the terms in the ontology with which they were indexed. B. Choosing a photograph displays, in table form, the annotated information about that image.

common Web designators (the URLs and URIs used to designate Web sites and other Web resources). Thus, for example, one could use my Web page URI—www.cs.umd.edu/~hendler—as a unique identifier on which to tie information about me. You could say that that site describes someone who is named "Hendler" and who works at the organization described in www.cs.umd.edu (the site that describes the computer science department where I work).

In fact, having files on the Web to describe oneself has become common, with the friend of a friend (Foaf) vocabulary[11] being a common one to use. Foaf files are regular Web files (for example, one can see what is in them simply by clicking on a link[12]) and that file can be used by anyone else anywhere on the Web to point to my designated Foaf file. It is estimated that there are several hundred thousands of Foaf files out there (for today's number, try Googling "foaf filetype:rdf" which will find a majority of, but by no means all, Foaf files). Indeed, a single file can designate many Foaf entries (for example all the people in a particular organization). A number of blogging and other sites automatically create Foaf entries for their users, and thus it is currently estimated there are well over five million individuals who are described in Foaf files.

Foaf is probably the most successful of the Semantic Web vocabularies in common use, but it is by no means the only one. There are vocabularies for describing open source licenses,[13] projects,[14] thesaurus vocabularies,[15] and lots of other common domains.[16] A Semantic Web crawler called Swoogle[17] has been built by Tim Finin and his students (Ding et al. 2004) using a targeted search (rather than a general search like Google). At the time of this writing, it finds more than 4,100 Web ontologies, 7.25 million defined individuals, and 47.5 million total RDF "statements" in the approximately 335,000 pages on which it discovers semantic information.

I also mentioned dozens of open source tools for using these Semantic Web languages, but I should have said dozens of good tools support-

Figure 4. Setting Up a Link Between Two URIs.

ed by reputable companies and large research organizations. If we included all the tools that have been released, no matter what quality, the number would surely be in the hundreds. The World Wide Web Consortium (W3C) maintains Web sites with pointers to a number of the better tools,[18] most of which are available open source or free.

Semantic Web Languages (An AI Perspective)

I've already mentioned two of the main Semantic Web languages (RDF and OWL), and there's actually a third—RDF Schema—which is used for many of the vocabulary definitions described earlier. In this section, I'll tell you a little bit about these languages and why they are interesting from an AI perspective. But I won't try to tell you much about the syntax of these languages or the details of their use. All three are Web recommendations from the W3C, and primers, guides, references, and technical documents (including model theories) are available through the W3C's Semantic Web activity page.[19] Instead, I want to draw analogies to some of the earlier AI work and discuss some of the similarities and, more importantly, the differences.

One of the key features that differentiates the hypertext markup language (HTML) on the Web from earlier hypertext languages is the ability to link a document to another across the Internet. In HTML, this is achieved through the "anchor" (<a>) tag in a document having the ability to uniquely name a Web resource, which can be a document, image, table, video, applet, or any other format transferable over the Web's protocols. The unique name is a universal resource indicator (URI), with the most common being the familiar "http" URIs we've all come to know and love. So if I have my Web page[20] and in it I have a pointer to another Web document, for example "<a href=http:// www. cs.umd.edu/>," then I am setting up a link between two URIs, as shown in figure 4.

Since each of the URIs in figure 4 can, in turn, be linked to other URIs, the famous Web graph is created—an unlabeled, directed (distributed) graph with billions of links.

If these links were labeled, rather than unlabeled, then Web applications could use the link types to do a better job of searching, indexing, and displaying pages. For example, if the link in figure 4 could be labeled something like "WorksAt", then a search engine could let a user look specifically for where I work without getting "confused" by all those other links that might or might not have anything to do with this. Even a simple labeling scheme (for example, designating whether the thing at the end of a link designates information about a person or not) could be used to help disambiguate things on the Web. For example, if I did an image search, I could look for only those images that were linked to through a "designatesPerson" link and thus tell photographs of people from photographs of inanimate objects, animals, and so on.

The problem, of course, is what link label set to use—much as AI researchers argued about what the "primitives" of knowledge representation systems for natural language should be, Web researchers have proposed various types of link sets. However, given the scope and generality of the Web, it became clear that no single, simple set could work for the myriad of domains of discourse being used every day.

The solution used in the Semantic Web, realized through the resource description framework (RDF), is that there exists on the Web a mechanism for creating an arbitrary set of link labels, with an essentially infinite set of unique tags—the URI mechanism itself. That is, if a group of my friends and I agreed that having a "designatesPerson" link would be beneficial to something we are doing, then we could agree to use the label http:/ /www.cs.umd.edu/ ~hendler/links# designatesPerson and build our applications around that. Thus, we reuse the naming scheme of the Web to get a unique, and unambiguous label to use among our group. We could now create a labeled link that would connect my home page to some image through the new link shown in figure 5.

And, if my home page had other named links to other resources, which might in turn link further, the Web could be extended, in essence, from an unlabeled to a labeled graph (with a potentially infinite set of labels). Since the links would have at least human meaning (that is, you as the reader can assign "designatesPerson" some sort of semantics), we could think of the labeled Web, from an AI point of view, a lot like the early AI KR systems that used "semantic networks" with labeling, but not with a lot of meaning in the links (Woods 1975). RDF as a language is made up of triples, each of three URIs (or a datatype, so we can link something to a string, or number) representing exactly these "Subject Property Object" graphs, where any two references to the same URI are assumed to refer to the same thing. A special property, called rdf:type is used to link a specific instance (say "book1") to a category ("book) allowing instance data to be

Figure 5. A Labeled Link Connecting My Home Page to an Image through a New Link.

created easily. There's a lot more details, but that's basically it—so think of RDF as the "semantic network" language for the Web.

Just as in AI where semantic networks are useful in themselves, but far more useful with extensions, the Semantic Web community has realized the need for more powerful representation. The first such is the resource description framework schema (RDFS), which is useful for creating vocabularies on the Web, extending RDF into something more like the frame representations used in AI. This is done by adding something akin to the ISA link (allowing the creation of a "class" and the expression of "subclassOf" relations between them in RDFS) and by being able to name the slots expected to be associated with a class (using "domain" and "range" statements). The rdf:type statement is used to link an individual to a class, thus allowing instances and classes to be distinguished (if desired).

One of the key advantages to the use of RDFS over earlier such framelike representations is that it is built to be compatible with the Web and Web architecture. First, all class names and properties respect the RDF convention of being expressed as URIs, thus making the linking of classes and properties across documents possible. Second, and more importantly, there is a convention in RDFS that the URI of the document in which the vocabulary is defined should be the same as the URI used in the relations. Thus, if I see a vocabulary term like the aforementioned "www.cs.umd.edu/~hendler/links#designatesPerson," then the document where the designatesPerson relation is defined would be assumed to be the one in this URI. By HTTP-GETting this document I, or my computer agent, can see the vocabulary definition associated with it. That is, in that file there you would see the XML corresponding to:

```
:designatesPerson a rdf:Property;
    rdfs:domain :Person;
    rdfs:range :Photograph.
```

which states that this property relates a Person to a Photograph (terms that are defined elsewhere on that page).

Representationally, RDFS is obviously not very expressive. Many KR languages developed by AI researchers had the ability to express more complex relationships between entities than simple domain and range constraints, and a language that could represent more powerful ontologies was needed for a number of emerging use cases on the Web.[21] Based on the progress of a number of research projects that were exploring how to use URIs for ontologies on the Web (Heflin and Hendler, 2000), how to represent knowledge in an XML syntax,[22] and how to do logic-based reasoning for such Web languages (Fensel et al. 2000), the U.S. Defense Advanced Research Projects Agency (DARPA) created a program aimed at funding the development of a standard in this area (called the DARPA Agent Markup Language, DAML). Interest in the use of DAML beyond the military grew, and the World Wide Web Consortium created the Web Ontology Working Group to create a Web standard based on DAML. The Working Group began in November 2001, and the OWL language was officially recognized as a W3C Recommendation in February of 2004.[23]

Although OWL contains expressivity that goes beyond that of description logic reasoners, the easiest approximation for the sake of this article is to think of OWL as a DL-like KR language for the Web. It adds a number of representational features to RDF Schema, including, among others, the following:

Terms for describing properties: allowing RDFS properties to be declared as transitive or symmetric, or to be known to be one-to-one, one-to-many, or many-to-one. It also allows for distinguishing properties that link classes to various datatypes instead of to other classes (compare with assigning the "text-name" property to link a person to a character string").

Terms for restrictions on property values of particular classes: in KR, it is important to be able to express that particular class/property pairings respect certain restrictions. This permits the expression of various kinds of quantification (universal vs. existential, cardinality) on OWL classes.

Terms for expressing similarity or differences between classes, properties, and/or individuals.
among others.

In doing this, OWL forms a fairly straightforward KR system, which is restricted in its scope, but a standard that is quite useful (and easy to extend).

The importance of the fact that these languages are Web recommendations is easy to underestimate, but really terribly important. These Semantic Web languages, by having gone through the often-painful process of standardization, have gained significant buy-in from the Web development and support communities. For example, the Firefox browser uses RDF extensively in support of bookmarks, history, configuration management, and interface design, and I already described the use of XMP by Adobe. IBM has an ontology management system called SNObase, which supports OWL, available on its Alphaworks portal. Oracle has recently announced support for RDF in Oracle DB10.2 and has publicly announced that it is considering

support for OWL in DB 11.0, and the list goes on with a number of companies both large and small, developing data stores, APIs, reasoners, and applications for these Semantic Web languages. In short, these languages are not just another AI "religion" but are the sort of consensus standards that have a shot at wide distribution and use. (Contrast this to one of the mistakes of the expert systems days, when AI companies competed to sell differing shells, trying to lock users into one rule-based representation over another, rather than developing some standard rule languages and competing to provide better support for the common representation.)

```
<owl:Class rdf:about="#Feline-Leukemia">
  <rdfs:subClassOf rdf:resource="NCI:Leukemia"/>
  <rdfs:subClassOf>
   <owl:Restriction>
     <owl:onProperty rdf:resource="#NCI: Organism_Affected"/>
     <owl:allValuesFrom rdf:resource="CYC:cat"/>
   </owl:Restriction>
  </rdfs:subClassOf>
 </owl:Class>
```

Figure 6. Feline Leukemia Ontology.

What's in a (Semantic Web) Link?

The real power of OWL and these other Semantic Web languages, however, is not so much in their abilities as traditional KR languages, or even in the standardization, but much more importantly in what Berners-Lee refers to as the "webizing" of these languages. He writes:

> The essential process in webizing is to take a system which is designed as a closed world, and then ask what happens when it is considered as part of an open world. Practically, this effect on a computer language is to replace the names/tokens/identifiers for URIs. Thus, where before reference could only be made to something in the same document/program/module one can with equal ease make reference to something in a different one somewhere in that abstract space which is the Web.[24]

So, by using RDF as the basis of OWL, we have webized KR, and this has profound implications that we are only starting to explore.

Consider the ontology defined in OWL in figure 6 (this is the file in its full, with the exception of some header information). This figure, in the XML form, states that this is defining a class called "Feline Leukemia", which is a subclass of the class Leukemia that is defined elsewhere by the National Cancer Institute ("NCI:Leukemia" is an abbreviation for the full URI.)[25] It goes on to say that there is a restriction on the property called "NCI:Organism_Affected" such that anything filling this property must be of type "CYC:cat" (that is, cat as defined in the OWL version of OpenCyc that has been released on the Web).

Why is this exciting? Consider, that the NCI ontology contains about 40,000 class definitions about cancers and cancer-related disease. The portion of the CYC ontology released in OWL to date has about 50,000 terms about the "commonsense" world. So this little ontology is linked to close to 90,000 class definitions. This means that if I create an instance of a cat with this disease, (figure 7) my reasoner can know (by HTTP-GETting the definitions from the linked files) that Puff is a cat, is a feline, is an animal, could be a pet owned by a person, is a member of the carnivore order (implying that her teeth have adapted for eating meat), and so on (from Cyc) and that poor Puff is inflicted with feline leukemia, which is a leukemia, which is a cancer disease, that there has been a clinical trial to explore whether Mercaptoethane Sulfonate is effective against this disease in people, and so on (from the NCI ontology). In short, the webized KR language lets us create a web of terminology and definitions that spans across multiple Web pages and has the potential to reuse knowledge across applications and domains.

But Wait, What About …

By this point, I suspect many a discerning reader is ready to cry "foul." So far, I've talked about the new capabilities of RDFS and OWL from the purely positive side. But there's a potential downside as well. After all, consider the following three issues:

First, OWL isn't very expressive—it certainly isn't a KIF, let alone even a full KL-ONE. The state of the art in AI languages long ago surpassed the expressivity of OWL.

Second, Web linking is not for free; while the "network effect" of ontologies and data and Web resources linked to other ontologies and data and Web resources is an exciting new playpen, there are challenges. What happens if the NCI adds some new information to the ontology I'm linked to and includes some information I don't agree with (after all, that document is out of my control). Consider what would happen to the reasoner trying to learn more about poor diseased Puff if the CYC server is down—what is an agent to do when faced with a semantic 404 error?

Third, the web of knowledge I've been describing cannot be guaranteed consistent and, in fact, is pretty much guaranteed not to be! On the Web there will be multiple causes of inconsistency including disagreement (when does life begin?), error (did I say "Cat"? oops, I meant "Car") and dishonesty (*my* site has just the term you're looking for, you just have to pay to use it…). KR in AI has worked off an assumption that KBs are largely consistent, and the approaches to reasoning in inconsistent knowledge bases are not yet scalable.

To me, however, this seems more of

```
:Puff a CYC:cat.
http://www.cs.umd.edu/~hendler/links#feline-leukemia NCI:Organism_Affected :Puff.
```

Figure 7. An Instance of a Cat with Leukemia.

a promise than a threat. Consider the analogies—HTML (and even XML) is much less expressive than the languages it derives from, and the Web "hypertext" is significantly less powerful than many of the hyperbooks that preceded it; the Web 404 error is crucial to the design of a scaleable open system, and just take a brief look at "blog space" if you want to see an open and thriving Web culture based on disagreement, misinterpretation errors, and yes, at times, calculated dishonesty.

In short, bringing AI to the amazingly huge, open and changing world that is the Web is one of the most exciting, and potentially high payoff, challenges to ever hit our field! We truly may be where hypertext was before the Web—sitting on top of an important technology that can, for the first time in fifty years, be linked into a "single global system" realizing its full potential, and building what Bob Engelmore was calling for when he stated that the full potential of the advanced sort of expert systems he was envisioning required the creation of a "knowledge infrastructure" that could help in "breaking the knowledge engineering bottleneck." I contend that the webized KR language that OWL brings to the Web (along with future extensions, which may include rule languages and more expressive logics), in allowing the linking of knowledge sources, in being standardized so that the use of these languages brings greater interoperability, and in building on the Web infrastructure that has become so pervasive in modern society, creates the very knowledge infrastructure that Bob prescribed more than a decade ago.

References

Berners-Lee, T.; Hendler, J.; Lassila, O. 2001. The Semantic Web. *Scientific American* 284(5): 34–43.

Ding, L.; Finin, T.; Joshi, A.; Pan, R.; Cost, R. S.; Sachs, J.; Doshi, V.; Reddivari, P.; and Peng, Y. 2004. Swoogle: A Search and Metadata Engine for the Semantic Web. In *Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management*. New York: Association for Computing Machinery.

Fensel, D.; Horrocks, I.; Van Harmelen, F.; Decker, S.; Erdmann, M.; and Klein, M. 2000. OIL in a Nutshell., In *Knowledge Acquisition, Modeling, and Management, Proceedings of the European Knowledge Acquisition Conference (EKAW-2000),* ed. R. Dieng and O. Corby. Lecture Notes in Artificial Intelligence, Volume 1937. Berlin: Springer-Verlag.

Golbeck, J.; Alford, A.; and Hendler, J. 2003. Organization and Structure of Information Using Semantic Web Technologies. In *Handbook of Human Factors in Web Design,* ed. R. Proctor and K. Vu. 2003. Mahweh, NJ: Lawrence Erlbaum.

Heflin, J., and Hendler, J. 2000. Dynamic Ontologies on the Web, In *Proceedings of the Seventeenth National Conference on Artificial Intelligence.* Menlo Park, CA: AAAI Press.

Woods, W. 1975. What's in a Link: Foundations for Semantic Networks. In *Representation and Understanding: Studies in Cognitive Science,* ed. D. Bobrow and A. Collins. New York: Academic Press.

Notes

1. See www.wtec.org/loyola/kb/c1_s1.htm.
2. See www.w3.org/Talks/WWW94Tim/.
3. www.flickr.com/.
4. www.adobe.com/products/xmp/ main.html.
5. dublincore.org/.
6. www.adobe.com/products/xmp/main.html.
7. A number of these are described at w3photo.org/semantic/.
8. www.mindswap.org/2003/PhotoStuff/.
9. www.mindswap.org.
10. semspace.mindswap.org.
11. www.foaf-project.org/.
12. Mine is at www.cs.umd.edu/~hendler/2003/foaf.rdf—click on it and see!
13. creativecommons.org/.
14. usefulinc.com/doap.
15. www.w3.org/2004/02/skos/.
16. See www.schemaweb.info/ for many more examples.
17. swoogle.umbc.edu.
18. Or RDF: www.w3.org/RDF; for OWL: www.w3. org/ 2004/OWL.
19. www.w3.org/2001/sw.
20. www.cs.umd.edu/users/hendler/index.html.
21. www.w3.org/TR/webont-req/ is the "use cases and requirements" document for Web ontologies, which summarizes a number of these.
22. www.ai.sri.com/pkarp/xol/.
23. The details of OWL, and of the compromises between expressivity, flexibility, decidability, and so on that were involved in its design, are beyond the scope of this article—www.w3.org/2004/OWL is the official OWL Web site, and contains not only the official documents but also documents and mail archives discussing the issues that were, and sometimes were not, resolved.
24. www.w3.org/DesignIssues/webize.
25. The full URI is www.ncibi.nih.gov/NCIT/NCIT.owl#leukemia.

Jim Hendler is a professor of computer science at the University of Maryland. Having been born about nine months after the Dartmouth Workshop, he is fond of pointing out that he and the field were conceived at about the same time. In the past few years, Hendler has been working on bringing AI to the Web and has devoted much of his effort to technical development and evangelism of and for the Semantic Web.