WEB METADATA: A Matter of Semantics

The sheer volume of information can make searching the Web frustrating. The Resource Description Framework, with its focus on machine-understandable semantics, has the potential for saving time and yielding more accurate search results.

ORA LASSILA
Nokia Research Center

The surge in popularity of the World Wide Web—and in the quantity of information it contains—is staggering. Although the Web is built on relatively simple principles, its growth has not been without substantial growing pains. The Web was built for human consumption, and although everything on the Web is machine-readable, it is not machine-understandable. This makes it very hard to automate anything on the Web and—because of the sheer volume of information—impossible to manage manually.

Coping with the volume of information on the Web is a real problem, as witnessed by anyone who has used one of the popular search services. Since Web documents are not designed to be understood by machines, the only real form of searching available to us is full-text search. Entering keywords into a search engine and receiving thousands of hits is not necessarily useful—the documents we seek may or may not be among those thousands. Mere words used as search keywords are subject to cross-disciplinary semantic drift. Keywords thus perform poorly in situations where a search index covers multiple subject areas, as is the case with the Web.[1]

Wouldn't it be useful if other means of searching were available to us, in addition to full-text search (matching strings)? For example, we might know who wrote the document, when it was published, and what specifically it discusses (although any particular word describing that subject might not be contained in the document desired). Machines (in this case, search engines) cannot understand a natural-language document and thus cannot always extract specific information from the document, such as author, publication date, or topic.

In a recent white paper, Tim Berners-Lee, director of the World Wide Web Consortium, wrote: "Currently there is not only a large industry in applications to put information from legacy information systems onto the Web, there is also an industry in applications which surf the Web and, programmed with some idea of how the Web pages were automatically generated, retrieve the information and reconvert it into hard, well-defined machine-processable data."[2]

 1089-7801/98/$10.00 ©1998 IEEE

It's clear that stronger, more precise means of describing documents are needed. Information such as author, publication dates, and so forth is often called metadata. Metadata is commonly defined as data about data. For example, a library catalog is metadata because it describes publications, or library data. Similarly, a file system maintains access control information about files; this information can also be seen as metadata. To maintain a library catalog you may also need an application that treats the catalog itself as data. Hence, one application's metadata is another application's data.

This article discusses Web metadata, which we define as "machine-understandable descriptions of Web resources." Web metadata has a number of uses, such as cataloging, software agents, and describing intellectual property rights. (For a description of Web metadata in applications, see the sidebar "Applications of Web Metadata.")

RDF: AN INTRODUCTION

The World Wide Web Consortium (W3C) recently published the Resource Description Framework,[2-5] a new standard for Web metadata. RDF, a foundation for processing metadata, provides interoperability between applications that exchange machine-understandable information on the Web. The design of RDF has been influenced by several sources, all of which have agreed on the basic principles of metadata representation and transport. Key influences have come, for example, from the Web development community itself, in the form of HTML metadata and the Platform for Internet Content Selection.[6] Other influences are the library community, the structured document community (in the form of SGML and, more importantly, XML, the Extensible Markup Language), and also the knowledge representation community. Framework design contributions have also come from object-oriented programming and modeling languages, and databases.

The framework's purpose is describing Web resources to facilitate automated processing of Web

APPLICATIONS OF WEB METADATA

Cataloging

Metadata can describe the contents of an individual Web resource, such as a page, an image, or the content of a collection—Web site, directory, and so forth. Metadata can also describe the relationships between members of a collection (for example, book, chapter, or table of contents). Descriptions of typically complex collections, especially those of Web sites, are sometimes referred to as site maps.

Software Agents and Resource Discovery

Search engines could take advantage of metadata, such as that used in cataloging, to perform more accurate searches. With the need for manual "weeding" of search results eliminated, we could better automate the search process. This also suggests that intelligent software agents could use metadata to exchange and share knowledge (agent to agent), to communicate (agent to service or agent to user), and to "understand" their environment (that is, to do resource discovery on their own).

Electronic Commerce

Metadata can encode information needed for electronic commerce. For example, with metadata we can locate a seller or buyer. We can find a product by searching the yellow pages, and we can agree on terms of sale (metadata can represent prices, terms of payment, and other contractual information).

Content Rating

The World Wide Web is a free medium, and balancing between free speech and protection of minors is difficult. Metadata can encode content rating labels that disclose the nature of a particular page's contents. This information, in turn, can be used in filtering content when "surfing" the Web. For example, parents can block their children's access to material deemed inappropriate.

Intellectual Property Rights

As with content rating, metadata could describe information about intellectual property rights of a document: the contractual terms related to the document's use and distribution.

Digital Signatures

Metadata can encode digital signatures, which, in turn, can help users decide which information and documents to trust.

Privacy

Metadata can describe users' preferences regarding privacy—that is, what information users are willing to disclose about themselves when visiting a Web site. Metadata can also describe a Web site's information-gathering policy regarding visiting users. This capability may dissuade users' suspicions about privacy on the Web and the perceived need for anonymity.

```
<?xml:namespace ns="http://www.w3.org/TR/WD-rdf-syntax" prefix="RDF"?>
<?xml:namespace ns="http://purl.org/metadata/dublin_core" prefix="DC"?>

<RDF:RDF>
    <RDF:Description about="http://www.some.org/smith">
        <DC:Creator>John Smith</DC:Creator>
    </RDF:Description>
</RDF:RDF>
```

Figure 1. This RDF instance describes a Web resource with a given URL and states that "John Smith" is the creator of this particular resource. The Web page is a node with one property—DC:Creator—whose value is the string "John Smith."

Figure 2. A graph generated from the example in Figure 1.

information. The resources RDF describes are generally anything that can be named with a Uniform Resource Identifier (URI), the class of Web identifier that includes the common URL. Designed as domain-neutral, RDF makes no assumptions about any particular application domain, nor defines a priori the semantics of any domain. Despite this, the mechanism is suitable for describing information about any domain.

RDF is a data model of metadata instances. The RDF Model and Syntax Specification[3] describes the model and one possible syntax for encoding and transporting RDF instances. To give RDF an object-oriented nature, the RDF Schema Specification[5] defines an extensible type system using the basic RDF model as building blocks.

Model and Syntax

RDF data consists of nodes and attached attribute/value pairs. Nodes can be any Web resources (in fact, anything to which you can give a URI), including other metadata instances. Attributes are named properties of nodes, and their values are either atomic (text strings) or other nodes (Web resources or metadata instances). The essence of RDF is this model of nodes, attributes (or properties), and their values.

In addition to the node-centric view—an object-oriented view of the RDF model reminiscent of frame-based representation systems—the RDF model can be seen as directed, labeled graphs (DLGs). The nodes are the vertices of a graph, and the properties name the edges. Therefore, if X has a property Y with the value Z, we can think of X and Z linked by an edge labeled Y, pointing from X to Z.

To store instances of this model in files, or to communicate these instances from one agent to another, we need a graph serialization syntax. XML is the language the designers chose for use in the RDF specification.[7] RDF and XML are complementary. RDF leverages XML; however, XML needs RDF for defining what instances of metadata mean, and for allowing agents to agree on a common meaning. XML is only one syntactic representation for the RDF model; other syntaxes are possible.

In the following example we will use terminology from **Dublin Core**, a metadata schema for building digital library catalogs. Figure 1 is an example of a simple RDF instance. This metadata fragment describes a Web resource with a given URL and states that "John Smith" is the creator—that is, author in Dublin Core library metadata terms—of this particular resource. In the model, the Web page is a node and it has one property, namely DC:Creator, whose value is the string "John Smith." RDF relies on the XML namespace mechanism[8] to uniquely qualify element names, hence two XML processing instructions precede the example. The element name prefix "RDF:" is used by all RDF core names, and in this example an XML processing instruction associates the prefix "DC:" with a Dublin Core schema URI. RDF designers anticipate that RDF metadata will typically consist of instances and attributes from many different sources. The probability of name conflicts is high, but the namespace mechanism solves this problem.

Figure 2 is a graphical representation of the RDF instance shown in Figure 1.

The RDF designers also discussed alternate syntaxes based on S-expressions.[6] S-expressions are an efficient, compact way of encoding structured data. The RDF instance in Figure 1 could have been expressed as follows:

```
(rdf:description about "http://www.some.org/smith"
        dc:creator "John Smith")
```

The designers chose XML in the RDF specification on the basis of its perceived prevalence in

Web software, rather than its technical merit.

In RDF, property values can be complex objects. In Figure 3, the creator property from Figure 1 now has a value with more structure. Here, the value of the DC:Creator property is an instance with two properties: Name and EMail. Using the RDF instance in Figure 3, we could produce the graph shown in Figure 4.

```
<?xml:namespace ns="http://www.w3.org/TR/WD-rdf-syntax" prefix="RDF"?>
<?xml:namespace ns="http://purl.org/metadata/dublin_core" prefix="DC"?>
<?xml:namespace ns="http://some.org/schemata/people" prefix="P"?>

<RDF:RDF>
  <RDF:Description about="http://www.some.org/smith">
    <DC:Creator>
      <RDF:Description>
        <P:Name>John Smith</P:Name>
        <P:EMail>mailto:smith@some.org</P:EMail>
      </RDF:Description>
    </DC:Creator>
  </RDF:Description>
</RDF:RDF>
```

Figure 3. An RDF instance where the value of the creator property from Figure 1 has more structure.

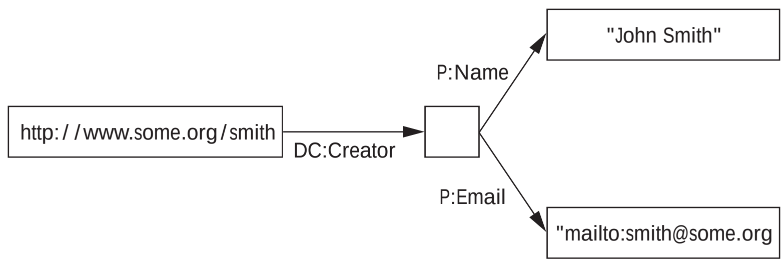

Figure 4. Graph generated from example in Figure 3.

Metadata on Metadata

As is often the case, metadata authors and processors need to make statements about other statements expressed in RDF (we refer to these as higher-order statements). This possibility requires careful consideration. For example, if we make the natural-language statement "The Web contains one billion documents," RDF would regard this as true. On the other hand, the statement "John estimates that the Web contains one billion documents" makes a statement about the relationship between John and his view of the Web, but it does not express any facts about the Web per se. Both kinds of statements are possible in RDF.

When we create a statement in RDF that consists of a node X, property Y, and value Z, we think of a triple [Y, X, Z] having been asserted (placed in RDF's internal database). Statements that exist in this database in the form of triples are considered true. This has nothing to do with epistemological, absolute truth; it merely says that the RDF system, when queried, will know that these statements have been asserted. To make statements about [Y, X, Z], we must build a model of this statement. In RDF we do this by asserting three new statements:

[RDF:PropObj, P, X]
[RDF:PropName, P, Y]
[RDF:Value, P, Z]

This modeling process is often called reification. We now have a new node, P, representing the statement. We can make statements involving P, as in

[Believes, mailto:smith@some.org, P]
("smith believes P")

or

[CreatedOn, P, "1998-05-01"]
(the assertion P was created on 1 May 1998)

Whether the original triple is still in the database determines whether the RDF system considers it true, but we could have asserted the above three reifying triples without ever asserting [Y, X, Z].

The RDF syntax has a shorthand for expressing statements about other statements. If we wanted to augment the first example by saying that "Jane Smith" is the author (= DC:Creator) of the statement about John's home page, it could be written as shown in Figure 5.

The ability to make statements about other statements is important. We originally included it in RDF to make it possible to digitally sign RDF statements. Because a typical use of RDF is to manipulate metadata from many different sources, however, it makes sense to have a mechanism for expressing beliefs and other modalities.

```
<?xml:namespace ns="http://www.w3.org/TR/WD-rdf-syntax" prefix="RDF"?>
<?xml:namespace ns="http://purl.org/metadata/dublin_core" prefix="DC"?>

<RDF:RDF>
  <RDF:Description about="http://www.some.org/smith" bagID="foo">
    <DC:Creator>John Smith</DC:Creator>
  </RDF:Description>
  <RDF:Description aboutEach="#foo">
    <DC:Creator>Jane Smith</DC:Creator>
  </RDF:Description>
</RDF:RDF>
```

Figure 5. An example of a higher-order statement.

Schemata

RDF does not contain any predefined vocabularies for authoring metadata. However, standard vocabularies, or schemata as they are called in RDF, will emerge. They will do so either by specialized communities cooperating in the design, or by natural selection. (Some schemata are selected simply because they are used more frequently than others in the same domain.) The existence of standard, or de facto standard, schemata is a core requirement for large-scale interoperability.

Anticipated schemata include a PICS-like content-rating architecture, a digital library vocabulary (currently the "Dublin Core"), and a schema for expressing digital signatures. Anyone can design a new schema; the only requirement is that a designating URI be included in the metadata instances. The use of URIs to name vocabularies is an important RDF design feature: some metadata standardization efforts have stumbled on the issue of establishing a central attribute registry. RDF permits, but does not require, a central registry.

The RDF schema mechanism defines the root of the RDF type hierarchy. It does so in the form of basic classes such as Resource, Class, and so on. The classes include the necessary meta-object types for defining new classes: for example, PropertyType. The RDF schema specification's class definition facilities let metadata authors place restrictions on property values and define classes in terms of existing classes (subclassing). Property value restrictions are in the form of cardinality and type constraints; that is, restrictions on the number of values a property can have, and the classes of objects that can be values of a particular property. The W3C's schema work is ongoing. For specific instances, see the sidebar "Sample Schema Projects."

WHY RDF AND NOT JUST XML?

What are the RDF framework's major benefits? After all, XML offers structured data that could be used to encode and transport attribute/value pairs. As I stated earlier, RDF and XML are complementary. RDF is a model of metadata, and it only superficially addresses many encoding issues that transportation and file storage require, such as internationalization and character sets. For these issues, RDF relies on XML. But RDF also has several advantages over XML.

One design goal for RDF was to enable metadata authors to specify semantics for data based on XML in a standardized, interoperable manner. RDF also offers features like collection containers and higher-order statements. RDF's main advantage, however, is that it requires metadata authors to designate at least one underlying schema, and that the schemata are sharable and extensible. RDF is based on an object-oriented mindset, and schemata correspond to classes in an object-oriented programming system. Organized in a hierarchy, schemata offer extensibility through subclass refinement. To create a schema slightly different from an existing one, therefore, requires only that you provide incremental modifications to the base schema. XML document type descriptions (DTDs) do not offer this capability. Through schemata sharability, RDF supports the reusability of definitions resulting from the metadata work by individuals and specialized communities.

Due to RDF's incremental extensibility, agents processing metadata will be able to trace the origins of schemata with which they are unfamiliar to known schemata. They will be able to perform meaningful actions on metadata they weren't originally designed

WEB RESOURCES FOR RDF

W3C Metadata and RDF Information
http://www.w3.org/RDF/
http://www.w3.org/TR/NOTE-rdfarch

Introductory Articles
http://www.w3.org/TR/NOTE-rdf-simple-intro
http://www.dlib.org/dlib/may98/miller/05miller.html

Resources for Programmers
http://www.mozilla.org/rdf/doc/
http://www.alphaWorks.ibm.com/formula/rdfxml

to process. For example, suppose you were to design an extension to the Dublin Core schema to leverage work done by the library community and also to allow organization-specific document metadata. To do so, you could simply use standard tools designed for plain Dublin Core. Because of the self-describing nature of RDF schemata, a well-designed tool would be able to do meaningful processing for the extended properties as well.

RDF's sharability and extensibility will also lead to a mix-and-match use of metadata and metadata schema descriptions. Metadata authors will be able to use multiple inheritance to provide multiple views to their data, leveraging work done by others. Moreover, it's possible to create RDF instance data based on multiple schemata from multiple sources—that is, interleaving different types of metadata; XML DTDs do not support this feature. This will lead to exciting possibilities when agents process metadata. For example, a processing agent may know how to process several types of RDF instances individually, but it will later also be able to reason about the combination. Essentially, the combination is more powerful than the sum of its parts.

From an implementation standpoint, RDF offers a clean, simple object model independent of the transport syntax of metadata. An API for processing RDF is likely to appear. It is also important to remember that although the RDF specification defines an encoding syntax for RDF based on XML, RDF itself is not dependent on XML: it could also use other syntaxes (for example, S-expressions). It is conceivable that various "translators" will emerge, allowing data in various formats (corresponding internally to the RDF data model) to be filtered and used by RDF processors.

ORIGINS OF WEB METADATA AND RDF

From the standpoint of the Web, the history of standardized metadata mechanisms begins with the HTML <META> and <LINK> tags. These let a Web page author record metainformation about a page and also indicate that page's relationship to other relevant pages, such as a table of contents.

The <META> tag can specify the author of a Web page as follows:

```
<META name="Author" content="John Smith">
```

Although both <META> and <LINK> tags are useful, they have certain shortcomings. What does the name "Author" really mean? It could be the name of the person who created the page, or the person who wrote the page contents, or even the Webmaster who maintains the page. In other words, the meaning of "Author" on one Web page might be different from its meaning on another page. In short, the namespace of attribute names is uncontrolled (at least prior to the new HTML 4.0 specification). The structure of attribute values is also not specified (for example, is it "John Smith" or "Smith, John"?). Furthermore, it is very difficult to use <META> for higher-order statements such as those described in the section "Metadata on Metadata."

SAMPLE SCHEMA PROJECTS

Dublin Core. The library community is designing a schema for building digital library catalogs (http://purl.org/metadata/dublin_core/).

P3P. This is the W3C's project to allow privacy preferences and policies to be expressed (http://www.w3.org/P3P/).

IMS. This Instructional Management Systems project is building a metadata schema for managing online learning resources (http://www.imsproject.org/metadata).

Content Rating

Content rating is a hot topic in the standardization community. Attempts to balance free speech and protection of minors resulted in PICS, the W3C's content-rating architecture.[9] PICS is a simple metadata mechanism well suited to content rating; however, because attribute values can be chosen only from controlled vocabularies (actually, they are all numeric), it has limited use as a general metadata architecture. On the positive side, PICS introduced the notion of machine-interpretable schemata for metadata. It also defined various ways in which metadata can be associated with Web resources. Metadata can be

- embedded in an HTML <META> tag in the document head;
- transported in HTTP headers—this is also possible with the <META> tag by using the attribute "http-equiv" instead of "name," and
- stored in and retrieved from a third-party metadata bureau.

Figure 6 shows an example of a PICS label.

```
(PICS-1.1 "http://www.gcf.org/v2.5"
   by "John Doe"
   labels on "1994.11.05T08:15-0500"
      until "1995.12.31T23:59-0000"
      for "http://w3.org/PICS/Overview.html"
      ratings (suds 0.5 density 0 color/hue 1)
      for "http://w3.org/PICS/Underview.html"
      by "Jane Doe"
      ratings (subject 2 density 1 color/hue 1))
```

Figure 6. A PICS label (that is, an instance of PICS metadata) provides information about the content of a Web page.

The desire to develop PICS into a general metadata mechanism led the W3C to work on "PICS-NG," RDF's predecessor.[6] With the advent of RDF, W3C plans to transition PICS to use RDF. RDF will also allow content-rating information to be mixed with privacy information. The W3C's project on privacy technologies, **P3P**, builds directly on top of RDF.

Support for the RDF Standard

Numerous recent projects build metadata mechanisms and standards for narrow domains. Examples include the Internet Mail Consortium's vCard, an electronic business card formalism,[10] and Microsoft's **Channel Definition Format** for describing pushed Web content. When RDF is widely deployed, many of the special metadata standards can be cast as RDF applications.

The library community has invested considerable effort in the development of electronic cataloging standards (for example, MARC, which stands for "Machine-Readable Catalog."[11]) Unfortunately, some of these standards are not useful in the Web's context. These efforts are important, however, because they led to the Dublin Core metadata element. W3C considers the support of the Dublin Core paramount in the current metadata standardization efforts. The digital library community has very strongly advocated RDF throughout the development of this standard.

Metadata is a form of structured data transmitted on the Web. The structured document, or SGML, community has influenced the metadata standardization through the introduction of XML.[7] XML is often billed as a type of universal syntax to solve the lack of interoperability between various Web-based software systems. This language is only a way of "serializing a tree" or, more generally, a way of encoding structured data for transport on the Internet. It has no inherent semantics, nor does it offer a way for agents to exchange descriptions of semantics. Provided that mechanisms to define semantics are built atop XML, it is a natural choice for metadata syntax. This is because the ability to parse XML syntax is (or will be) prevalent in numerous Web-related software products. Providing the semantic machinery is exactly what the W3C's RDF project has done. Without RDF, everybody would have to reinvent a mechanism for communicating semantics between interoperating software systems.

Microsoft's XML-Data[12] is another framework that simplifies the definition of data written in XML. Early versions of XML-Data were studied by the RDF design team. Current focus in XML-Data seems to be how to map legacy data into XML.

RDF has also been influenced by knowledge representation research. The KR community has spent a great deal of effort on the crucial problem of how to represent knowledge in a way machines can understand. RDF design was influenced, for one, by associational representations such as semantic networks developed by Ross Quillian[13] and William Woods.[14] Equal influence came from frame-based representation systems: for example, those by Marvin Minsky,[15] and by Richard Fikes and Tom Kehler.[16] Another direct influence came from Meta Content Framework (MCF), a metadata framework reminiscent of semantic networks.[17]

RDF should not be confused with more powerful knowledge representation formalisms. In knowledge interchange, KIF[18] is a de facto standard in the research community. Description logics, such as **CLASSIC**, is another area attracting much attention recently. RDF lacks certain mechanisms, such as negation and quantification. Designers deliberately excluded these features—first-order predicate logic—for fear that such complex features would discourage RDF's acceptance and deployment within the Web community.

FUTURE OF WEB METADATA

Standardized metadata is a solution to the lack of machine-understandable semantics, one of the World Wide Web's big problems. To build on the strengths of RDF's shared schemata, the types (classes) designers introduce should be organized as concept ontologies. These can then be used in describing data, Web resources, and services. In addition to RDF, ongoing research is addressing the lack of semantics and helping to build ontologies on the Web. Examples of these include the **Conceptual Knowledge Markup Language** and SHOE ("Simple HTML Ontology Extensions").[19]

Social and cultural issues, such as trust, that are associated with the large-scale deployment of shared ontologies, need study. However, by using metadata now for digital signatures, privacy preferences, and intellectual property contracts, we can start building a "Web of trust" and start leveraging economies of scale. Through automation, perhaps in the form of intelligent agents, we may turn the Web into a usable repository where we can manage the vast amounts of information that now, so often, seem to escape our reach. ■

URLs used in this feature

CLASSIC • www.research.att.com/sw/tools/classic/

Channel Definition Format • www.microsoft.com/standards/cdf-f.htm

Conceptual Knowledge Markup Language • wave.eecs.wsu.edu/WAVE/Ontologies/CKML/CKML-DTD.html

Dublin Core • purl.org/metadata/dublin_core/

P3P • www.w3.org/P3P

ACKNOWLEDGMENTS

The author would like to thank R.V. Guha, Thomas Hubbard, Marcia Lassila, Eve Maler, Eric Miller, and Ralph Swick for their help and comments.

REFERENCES

1. S. Weibel, "Metadata: The Foundations of Resource Description," *D-Lib Magazine*, July 1995; http://www.dlib.org/dlib/July95/07weibel.html.
2. T. Berners-Lee, "W3C Data Formats," W3C Note, World Wide Web Consortium, 1997; http://www.w3.org/TR/NOTE-rdfarch.
3. O. Lassila and R.R. Swick, "Resource Description Framework (RDF) Model and Syntax," W3C Working Draft, World Wide Web Consortium, 1998; http://www.w3.org/TR/WD-rdf-syntax/.
4. O. Lassila, "Introduction to RDF Metadata," W3C Note, World Wide Web Consortium, 1997; http://www.w3.org/TR/NOTE-rdf-simple-intro.
5. D. Brickley, R.V. Guha, and A. Layman, "Resource Description Framework (RDF) Schemas," W3C Working Draft, World Wide Web Consortium, 1998; http://www.w3.org/TR/WD-rdf-schema/.
6. O. Lassila, "PICS-NG Metadata Model and Label Syntax," W3C Note, World Wide Web Consortium, 1997; http://www.w3.org/TR/NOTE-pics-ng-metadata.
7. T. Bray, J. Paoli, and C.M. Sperberg-McQueen, "Extensible Markup Language (XML) 1.0," W3C Recommendation, World Wide Web Consortium, 1998; http://www.w3.org/TR/REC-xml.
8. T. Bray, D. Hollander, and A. Layman, "Namespaces in XML," W3C Working Draft, World Wide Web Consortium, 1998; http://www.w3.org/TR/WD-xml-names.
9. T. Krauskopf et al., "PICS Label Distribution Label Syntax and Communication Protocols—Version 1.1," W3C Recommendation, World Wide Web Consortium, 1996; http://www.w3.org/TR/REC-PICS-labels.
10. "vCard—The Electronic Business Card, Version 2.1," http://www.imc.org/pdi/pdiproddev.html.
11. *Information Interchange Format, ANSI Z39.2-1994*, American Nat'l Stds. Inst., New York, 1994.
12. A. Layman et al., "XML-Data," W3C Note, World Wide Web Consortium, 1998; http://www.w3.org/TR/1998/NOTE-XML-data-0105/Overview.html.
13. M.R. Quillian, "Word Concepts: A Theory and Simulation of Some Basic Semantic Capabilities," *Behavioral Science 12*, 1967, pp. 410-430.
14. W.A. Woods, "What's in a Link: Foundations of Semantic Networks," in *Representation and Understanding: Studies in Cognitive Science*, D.G. Bobrow and A.M. Collins, eds., Academic Press, New York, 1975.
15. M. Minsky, "A Framework for Representing Knowledge," in *Psychology of Computer Vision*, P. Winston, ed., McGraw-Hill, New York, 1975.
16. R. Fikes and T. Kehler, "The Role of Frame-Based Representation in Reasoning," *Comm. ACM*, Vol. 28, No. 9, 1985, pp. 904-920.
17. R.V. Guha and T. Bray, "Meta Content Framework Using XML," W3C Note, World Wide Web Consortium, 1997; http://www.w3.org/TR/NOTE-MCF-XML/.
18. M.R. Genesereth, "Knowledge Interchange Format Specification," ANSI X3T2 working draft, 1995; http://logic.stanford.edu/kif/specification.html.
19. S. Luke et al., "Ontology-based Web Agents," *Proc. First Int'l Conf. Autonomous Agents*, ACM Press, New York, 1997.

Ora Lassila is a project manager at the Nokia Research Center in Burlington, Mass. Previously he was a project manager at Carnegie Mellon University and a research scientist at the Helsinki University of Technology. His research interests include intelligent agents, artificial intelligence, mixed-initiative decision support, activity planning and scheduling, the Web, object-oriented programming, and ubiquitous computing. He is currently a member of the advisory board of the World Wide Web Consortium. Lassila received an MSc in computer science and telecommunications from the Helsinki University of Technology.

Contact the author at Nokia Research Center, 3 Burlington Woods Dr., Ste. 260, Burlington, MA 01803; ora.lassila@research.nokia.com.