May 17, 2001

The Semantic Web

A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities

By Tim Berners-Lee, James Hendler and Ora Lassila

The entertainment system was belting out the Beatles' "We Can Work It Out" when the phone rang. When Pete answered, his phone turned the sound down by sending a message to all the other *local* devices that had a *volume control*. His sister, Lucy, was on the line from the doctor's office: "Mom needs to see a specialist and then has to have a series of physical therapy sessions. Biweekly or something. I'm going to have my agent set up the appointments." Pete immediately agreed to share the chauffeuring.

BY MIGUEL SALMERON

At the doctor's office, Lucy instructed her Semantic Web agent through her handheld Web browser. The agent promptly retrieved information about Mom's *prescribed treatment* from the doctor's agent, looked up several lists of *providers*, and checked for the ones *in-plan* for Mom's insurance within a *20-mile radius* of her *home* and with a *rating* of *excellent* or *very good* on trusted rating services. It then began trying to find a match between available *appointment times* (supplied by the agents of individual providers through their Web sites) and Pete's and Lucy's busy schedules. (The emphasized keywords indicate terms whose semantics, or meaning, were defined for the agent through the Semantic Web.)

In a few minutes the agent presented them with a plan. Pete didn't like it—University Hospital was all the way across town from Mom's place, and he'd be driving back in the middle of rush hour. He set his own agent to redo the search with stricter preferences about *location* and *time*. Lucy's agent, having *complete trust* in Pete's agent in the context of the present task, automatically assisted by supplying access certificates and shortcuts to the data it had already sorted through.

Almost instantly the new plan was presented: a much closer clinic and earlier times—but there were two warning notes. First, Pete would have to reschedule a couple of his *less important* appointments. He checked what they were—not a problem. The other was something about the insurance company's list failing to include this provider under *physical therapists*: "Service type and insurance plan status securely verified by other means," the agent reassured him. "(Details?)"

Lucy registered her assent at about the same moment Pete was muttering, "Spare me the details," and it was all set. (Of course, Pete couldn't resist the details and later that night had his agent explain how it had found that provider even though it wasn't on the proper list.)

Expressing Meaning

Pete and Lucy could use their agents to carry out all these tasks thanks not to the World Wide Web of today but rather the Semantic Web that it will evolve into tomorrow. Most of the Web's content today is designed for humans to read, not for computer programs to manipulate meaningfully. Computers can adeptly parse Web pages for layout and routine processing—here a header, there a link to another page—but in general, computers have no reliable way to process the semantics: this is the home page of the Hartman and Strauss Physio Clinic, this link goes to Dr. Hartman's curriculum vitae.

The Semantic Web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users. Such an agent coming to the clinic's Web page will know not just that the page has keywords such as "treatment, medicine, physical, therapy" (as might be encoded today) but also that Dr. Hartman *works* at this *clinic* on *Mondays*, *Wednesdays* and *Fridays* and that the script takes a *date range* in *yyyy-mm-dd format* and returns *appointment times*. And it will "know" all this without needing artificial intelligence on the scale of 2001's Hal or Star Wars's C-3PO. Instead these semantics were encoded into the Web page when the clinic's office manager (who never took Comp Sci 101) massaged it into shape using off-the-shelf software for writing Semantic Web pages along with resources listed on the Physical Therapy Association's site.

The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation. The first steps in weaving the Semantic Web into the structure of the existing Web are already under way. In the near future, these developments will usher in significant new functionality as machines become much better able to process and "understand" the data that they merely display at present.

The essential property of the World Wide Web is its universality. The power of a hypertext link is that "anything can link to anything." Web technology, therefore, must not discriminate between the scribbled draft and the polished performance, between commercial and academic information, or among cultures, languages, media and so on. Information varies along many axes. One of these is the difference between information produced primarily for human consumption and that produced mainly for machines. At one end of the scale we have everything from the five-second TV commercial to poetry. At the other end we have databases, programs and sensor output. To date, the Web has developed most rapidly as a medium of documents for people rather than for data and information that can be processed automatically. The Semantic Web aims to make up for this.

Like the Internet, the Semantic Web will be as decentralized as possible. Such Web-like systems generate a lot of excitement at every level, from major corporation to individual user, and provide benefits that are hard or impossible to predict in advance. Decentralization requires compromises: the Web had to throw away the ideal of total consistency of all of its interconnections, ushering in the infamous message "Error 404: Not Found" but allowing unchecked exponential growth.

Knowledge Representation

For the semantic web to function, computers must have access to structured collections of information and sets of inference rules that they can use to conduct automated reasoning. Artificial-intelligence researchers have studied such systems since long before the Web was developed. Knowledge representation, as this technology is often called, 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.

BY MIGUEL SALMERON

WEB SEARCHES TODAY

Traditional knowledge-representation systems typically have been centralized, requiring everyone to share exactly the same definition of common concepts such as "parent" or "vehicle." But central control is stifling, and increasing the size and scope of such a system rapidly becomes unmanageable.

Moreover, these systems usually carefully limit the questions that can be asked so that the computer can answer reliably—or answer at all. The problem is reminiscent of Gödel's theorem from mathematics:

any system that is complex enough to be useful also encompasses unanswerable questions, much like sophisticated versions of the basic paradox "This sentence is false." To avoid such problems, traditional knowledge-representation systems generally each had their own narrow and idiosyncratic set of rules for making inferences about their data. For example, a genealogy system, acting on a database of family trees, might include the rule "a wife of an uncle is an aunt." Even if the data could be transferred from one system to another, the rules, existing in a completely different form, usually could not.

Semantic Web researchers, in contrast, accept that paradoxes and unanswerable questions are a price that must be paid to achieve versatility. We make the language for the rules as expressive as needed to allow the Web to reason as widely as desired. This philosophy is similar to that of the conventional Web: early in the Web's development, detractors pointed out that it could never be a well-organized library; without a central database and tree structure, one would never be sure of finding everything. They were right. But the expressive power of the system made vast amounts of information available, and search engines (which would have seemed quite impractical a decade ago) now produce remarkably complete indices of a lot of the material out there. The challenge of the Semantic Web, therefore, is to provide a language that expresses both data and rules for reasoning about the data and that allows rules from any existing knowledge-representation system to be exported onto the Web.

Adding logic to the Web—the means to use rules to make inferences, choose courses of action and answer questions—is the task before the Semantic

Web community at the moment. A mixture of mathematical and engineering decisions complicate this task. The logic must be powerful enough to describe complex properties of objects but not so powerful that agents can be tricked by being asked to consider a paradox. Fortunately, a large majority of the information we want to express is along the lines of "a hex-head bolt is a type of machine bolt," which is readily written in existing languages with a little extra vocabulary.

Two important technologies for developing the Semantic Web are already in place: eXtensible Markup Language (XML) and the Resource Description Framework (RDF). XML lets everyone create their own tags—hidden labels such as or that annotate Web pages or sections of text on a page. Scripts, or programs, can make use of these tags in sophisticated ways, but the script writer has to know what the page writer uses each tag for. In short, XML allows users to add arbitrary structure to their documents but says nothing about what the structures mean.

The Semantic Web will enable machines to COMPREHEND semantic documents and data, not human speech and writings.

Meaning is expressed by RDF, which encodes it in sets of triples, each triple being rather like the subject, verb and object of an elementary sentence. These triples can be written using XML tags. In RDF, a document makes assertions that particular things (people, Web pages or whatever) have properties (such as "is a sister of," "is the author of") with certain values (another person, another Web page). This structure turns out to be a natural way to describe the vast majority of the data processed by machines. Subject and object are each identified by a Universal Resource Identifier (URI), just as used in a link on a Web page. (URLs, Uniform Resource Locators, are the most common type of URI.) The verbs are also identified by URIs, which enables anyone to define a new concept, a new verb, just by defining a URI for it somewhere on the Web.

Human language thrives when using the same term to mean somewhat different things, but automation does not. Imagine that I hire a clown messenger service to deliver balloons to my customers on their birthdays. Unfortunately, the service transfers the addresses from my database to its database, not knowing that the "addresses" in mine are where bills are sent and that many of them are post office boxes. My hired clowns end up entertaining a number of postal workers—not necessarily a bad thing but certainly not the intended effect. Using a different URI for each specific concept solves that problem. An address that is a mailing address can be distinguished from one that is a street address, and both can be distinguished from an address that is a speech.

The triples of RDF form webs of information about related things. Because RDF uses URIs to encode this information in a document, the URIs ensure that concepts are not just words in a document but are tied to a unique definition that everyone can find on the Web. For example, imagine that we have access to a variety of databases with information about people, including their addresses. If we want to find people living in a specific zip code, we need to know which fields in each database represent names and which represent zip codes. RDF can specify that "(field 5 in database A) (is a field of type) (zip code)," using URIs rather than phrases for each term.

Ontologies

Of course, this is not the end of the story, because two databases may use different identifiers for what is in fact the same concept, such as *zip code*. A program that wants to compare or combine information across the two databases has to know that these two terms are being used to mean the same thing. Ideally, the program must have a way to discover such common meanings for whatever databases it encounters.

A solution to this problem is provided by the third basic component of the Semantic Web, collections of information called ontologies. In philosophy, an ontology is a theory about the nature of existence, of what types of things exist; ontology as a discipline studies such theories. Artificial-intelligence and Web researchers have co-opted the term for their own jargon, and for them an ontology is a document or file that formally defines the relations among terms. The most typical kind of ontology for the Web has a taxonomy and a set of inference rules.

The taxonomy defines classes of objects and relations among them. For example, an *address* may be defined as a type of *location*, and *city codes* may be defined to apply only to *locations*, and so on. Classes, subclasses and relations among entities are a very powerful tool for Web use. We can express a large number of relations among entities by assigning properties to classes and allowing subclasses to inherit such properties. If *city codes* must be of type *city* and cities generally have Web sites, we can discuss the Web site associated with a *city code* even if no database links a city code directly to a Web site.

Inference rules in ontologies supply further power. An ontology may express the rule "If a city code is associated with a state code, and an address uses that city code, then that address has the associated state code." A program could then readily deduce, for instance, that a Cornell University address, being in Ithaca, must be in New York State, which is in the U.S., and therefore should be formatted to U.S. standards. The computer doesn't truly "understand" any of this information, but it can now manipulate the terms much more effectively in ways that are useful and meaningful to the human user.

With ontology pages on the Web, solutions to terminology (and other) problems begin to emerge. The meaning of terms or XML codes used on a Web page can be defined by pointers from the page to an ontology. Of course, the same problems as before now arise if I point to an ontology that defines *addresses* as containing a *zip code* and you point to one that uses *postal code*. This kind of confusion can be resolved if ontologies (or other Web services) provide equivalence relations: one or both of our ontologies may contain the information that my zip code is equivalent to your postal code.

Our scheme for sending in the clowns to entertain my customers is partially solved when the two databases point to different definitions of *address*. The program, using distinct URIs for different concepts of address, will not confuse them and in fact will need to discover that the concepts are related at all. The program could then use a service that takes a list of *postal addresses* (defined in the first ontology) and converts it into a list of physical *addresses* (the second ontology) by recognizing and removing post office boxes and other unsuitable addresses. The structure and semantics provided by ontologies make it easier for an entrepreneur to provide such a service and can make its use completely transparent.

Ontologies can enhance the functioning of the Web in many ways. They can be used in a simple fashion to improve the accuracy of Web searches—the search program can look for only those pages that refer to a precise concept instead of all the ones using ambiguous keywords. More advanced applications will use ontologies to relate the information on a page to the associated knowledge structures and inference rules. An example of a page marked up for such use is online at http://www.cs.umd.edu/~hendler. If you send your Web browser to that page, you will see the normal Web page entitled "Dr. James A. Hendler." As a human, you can readily find the link to a short biographical note and read there that Hendler received his Ph.D. from Brown University. A computer program trying to find such information, however, would have to be very complex to guess that this information might be in a biography and to understand the English language used there.

For computers, the page is linked to an ontology page that defines information about computer science departments. For instance, professors work at universities and they generally have doctorates. Further markup on the page (not displayed by the typical Web browser) uses the ontology's concepts to specify that Hendler received his Ph.D. from the entity described at the URI http://www. brown.edu — the Web page for Brown. Computers can also find that Hendler is a member of a particular research project, has a particular e-mail address, and so on. All that information is readily processed by a computer and could be used to answer queries (such as where Dr. Hendler received his degree) that currently would require a human to sift through the content of various pages turned up by a search engine.

In addition, this markup makes it much easier to develop programs that can tackle complicated questions whose answers do not reside on a single Web page. Suppose you wish to find the Ms. Cook you met at a trade conference last year. You don't remember her first name, but you remember that she worked for one of your clients and that her son was a student at your alma mater. An intelligent search program can sift through all the pages of people whose name is "Cook" (sidestepping all the pages relating to cooks, cooking, the Cook Islands and so forth), find the ones that mention working for a company that's on your list of clients and follow links to Web pages of their children to track down if any are in school at the right place.

Agents


BY MIGUEL SALMERON

AGENTS

The real power of the Semantic Web will be realized when people create many programs that collect Web content from diverse sources, process the information and exchange the results with other programs. The effectiveness of such software agents will increase exponentially as more machine-readable Web content and automated services (including other agents) become available. The Semantic Web promotes this synergy: even agents that were not expressly designed to work together can transfer data among themselves when the data come with semantics.

An important facet of agents' functioning will be the exchange of "proofs" written in the Semantic Web's unifying language (the language that expresses logical inferences made using rules and information such as those specified by ontologies). For example, suppose Ms. Cook's contact information has been located by an online service, and to your great surprise it places her in Johannesburg. Naturally, you want to check this, so your computer asks the service for a proof of its answer, which it promptly provides by translating its internal reasoning into the Semantic Web's unifying language. An inference engine in your computer readily verifies that this Ms. Cook indeed matches the one you were seeking, and it can show you the relevant Web pages if you still have doubts. Although they are still far from plumbing the depths of the Semantic Web's potential, some programs can already exchange proofs in this way, using the current preliminary versions of the unifying language.

Another vital feature will be digital signatures, which are encrypted blocks of data that computers and agents can use to verify that the attached information has been provided by a specific trusted source. You want to be quite sure that a statement sent to your accounting program that you owe money to an online retailer is not a forgery generated by the computer-savvy teenager next door. Agents should be skeptical of assertions that they read on the Semantic Web until they have checked the sources of information. (We wish more *people* would learn to do this on the Web as it is!)

Many automated Web-based services already exist without semantics, but other programs such as agents have no way to locate one that will perform a specific function. This process, called service discovery, can happen only when there is a common language to describe a service in a way that lets other agents "understand" both the function offered and how to take advantage of it. Services and agents can advertise their function by, for example, depositing such descriptions in directories analogous to the Yellow Pages.

Some low-level service-discovery schemes are currently available, such as Microsoft's Universal Plug and Play, which focuses on connecting different types of devices, and Sun Microsystems's Jini, which aims to connect services. These initiatives, however, attack the problem at a structural or syntactic level and rely heavily on standardization of a predetermined set of functionality descriptions. Standardization can only go so far, because we can't anticipate all possible future needs.

> Properly designed, the Semantic Web can assist the evolution of human knowledge as a whole.

The Semantic Web, in contrast, is more flexible. The consumer and producer agents can reach a shared understanding by exchanging ontologies, which provide the vocabulary needed for discussion. Agents can even "bootstrap" new reasoning capabilities when they discover new ontologies. Semantics also makes it easier to take advantage of a service that only partially matches a request.

A typical process will involve the creation of a "value chain" in which subassemblies of information are passed from one agent to another, each one "adding value," to construct the final product requested by the end user. Make no mistake: to create complicated value chains automatically on demand, some agents will exploit artificial-intelligence technologies in addition to the Semantic Web. But the Semantic Web will provide the foundations and the framework to make such technologies more feasible.

Putting all these features together results in the abilities exhibited by Pete's and Lucy's agents in the scenario that opened this article. Their agents would have delegated the task in piecemeal fashion to other services and agents discovered through service advertisements. For example, they could have used a *trusted* service to take a list of *providers* and determine which of them are *in-plan* for a specified *insurance plan* and *course of treatment*. The list of providers would have been supplied by another search service, et cetera. These activities formed chains in which a large amount of data distributed across the Web (and almost worthless in that form) was progressively reduced to the small amount of data of high value to Pete and Lucy—a plan of appointments to fit their schedules and other requirements.

In the next step, the Semantic Web will break out of the virtual realm and extend into our physical world. URIs can point to anything, including physical entities, which means we can use the RDF language to describe devices such as cell phones and TVs. Such devices can advertise their functionality—what they can do and how they are controlled—much like software agents. Being much more flexible than low-level schemes such as Universal Plug and Play, such a semantic approach opens up a world of exciting possibilities.

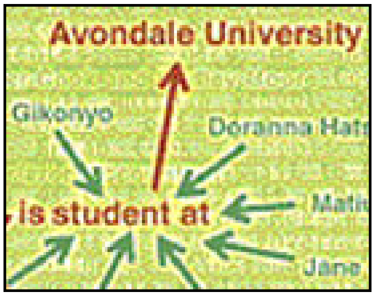
For instance, what today is called home automation requires careful configuration for appliances to work together. Semantic descriptions of device capabilities and functionality will let us achieve such automation with minimal human intervention. A trivial example occurs when Pete answers his phone and the stereo sound is turned down. Instead of having to program each specific appliance, he could program such a function once and for all to cover every *local* device that advertises having a *volume control* —the TV, the DVD player and even the media players on the laptop that he brought home from work this one evening.

The first concrete steps have already been taken in this area, with work on developing a standard for describing functional capabilities of devices (such as screen sizes) and user preferences. Built on RDF, this standard is called Composite Capability/Preference Profile (CC/PP). Initially it will let cell phones and other nonstandard Web clients describe their characteristics so that Web content can be tailored for them on the fly. Later, when we add the full versatility of languages for handling ontologies and logic, devices could automatically seek out and employ services and other devices for added information or functionality. It is not hard to imagine your Web-enabled microwave oven consulting the frozen-food manufacturer's Web site for optimal cooking parameters. >

Evolution of Knowledge

The semantic web is not "merely" the tool for conducting individual tasks that we have discussed so far. In addition, if properly designed, the Semantic Web can assist the evolution of human knowledge as a whole.

Human endeavor is caught in an eternal tension between the effectiveness of small groups acting



BY MIGUEL SALMERON

ELABORATE, PRECISE SEARCHES

independently and the need to mesh with the wider community. A small group can innovate rapidly and efficiently, but this produces a subculture whose concepts are not understood by others. Coordinating actions across a large group, however, is painfully slow and takes an enormous amount of communication. The world works across the spectrum between these extremes, with a tendency to start small—from the personal idea—and move toward a wider understanding over time.

An essential process is the joining together of subcultures when a wider common language is needed. Often two groups independently develop very similar concepts, and describing the relation between them brings great benefits. Like a Finnish-English dictionary, or a weights-and-measures conversion table, the relations allow communication and collaboration even when the commonality of concept has not (yet) led to a commonality of terms.

The Semantic Web, in naming every concept simply by a URI, lets anyone express new concepts that they invent with minimal effort. Its unifying logical language will enable these concepts to be progressively linked into a universal Web. This structure will open up the knowledge and workings of humankind to meaningful analysis by software agents, providing a new class of tools by which we can live, work and learn together.

Further Information:

Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor.
Tim Berners-Lee, with Mark Fischetti. Harper San Francisco, 1999.
An enhanced version of this article is on the Scientific American Web site, with additional material and links.

World Wide Web Consortium (W3C): **www.w3.org/**

W3C Semantic Web Activity: **www.w3.org/2001/sw/**

An introduction to ontologies:
www.SemanticWeb.org/knowmarkup.html

Simple HTML Ontology Extensions Frequently Asked Questions (SHOE FAQ):
www.cs.umd.edu/projects/plus/SHOE/faq.html

DARPA Agent Markup Language (DAML) home page: **www.daml.org/**

New papers Past issues Index of papers

Journal of Digital information, volume 3 issue 1
Themes: Hypermedia systems, Information discovery
2002-05-17
Peer reviewed paper

Hypermedia and the Semantic Web: A Research Agenda

Jacco van Ossenbruggen, Lynda Hardman and Lloyd Rutledge
CWI Amsterdam, Kruislaan 413, P.O. Box 94079,
1090 GB Amsterdam, The Netherlands
Email: {Jacco.van.Ossenbruggen, Lynda.Hardman, Lloyd.Rutledge}@cwi.nl

Key features: References; Figure 1

Contents

Abstract

Until recently, the Semantic Web was little more than a name for the next-generation Web infrastructure as envisioned by its inventor, Tim Berners-Lee. With the introduction of XML and RDF, and new developments such as RDF Schema and DAML+OIL, the Semantic Web is rapidly taking shape. This paper gives an overview of the state-of-the-art in Semantic Web technology, the key relationships with traditional hypermedia research, and a comprehensive reference list to various sets of literature (hypertext, Web and Semantic Web). A research agenda describes the open research issues in the development of the Semantic Web from the perspective of hypermedia research.

1 Introduction

The bulk of the content currently available on the Web is notoriously hard to process automatically: "...data transmitted across the Web is largely throw-away data that looks good but has little structure" [19]. Markup languages such as (X)HTML [69], SVG [32] and

SMIL [66] are primarily geared to documents whose content should be interpretable by *human* interpreters, and hence tend to focus primarily on document structure and document presentation. Little or no attention is given to the representation of the semantics of the content itself, i.e. the (domain-specific) representation of the subject of the document.

In contrast, knowledge representation techniques developed within the Artificial Intelligence (AI) community have a strong tradition in describing domain-specific knowledge in a machine-processable manner. In addition, the digital library community has studied issues related to more persistent ways of storing and cataloging digital content [14,46]. Recently, initiatives within and outside the World Wide Web Consortium (W3C) have been building upon the expertise of these communities by developing knowledge representation and annotation languages on top of the current Web infrastructure. This not only allows newly encoded knowledge to be easily disseminated over the Web, but also provides a convenient syntax for annotating existing content, such as (X)HTML or SMIL content. This combination is a key enabler for the main objective of the Semantic Web [6]: documents with content that is processable by both humans and machines.

While the Semantic Web appears at first sight to be far from the current research trends of the hypertext community, much earlier work in the field lay extremely close to the borders of knowledge representation, for example [17,18,48,52,60]. These authors were attempting to bridge the gap between knowledge representation and information presentation in a technological context that lacked support for this integration. The Web today provides a sound technological basis for document processing and already supports the first layers of the Semantic Web. This paper briefly sketches current developments of the Semantic Web, compares these with the issues long ago fielded in the hypertext literature, and highlights those that should form the basis of a research agenda for a universal information repository.

2 Current Semantic Web Infrastructure

Figure 1 provides an overview of both the document and knowledge representation languages on the Web. Following current document languages such as XHTML, SVG and SMIL (in the left half of the figure), the various layers of the Semantic Web are all built on top of XML [8], as shown in the right half of the figure. This makes generic XML-based software and languages such as XML parsers, transformation engines (XSLT [15]), path and pointer engines (XPath, XPointer [16,27]), style engines and formatters (CSS, XSL [7,70]), etc., directly available on the Semantic Web.

Stacked model of document and knowledge representationlanguages on the Web.

Figure 1. Document and knowledge representation languages on the Web

2.1 RDF and RDF Schema

The second layer of the Semantic Web infrastructure is the Resource Description Framework (RDF [67]). RDF provides a simple data model for expressing statements using *(subject, predicate, value)* triples, and an associated serialization syntax in XML. The *subject* and *value* of the triple can be defined within the current document or refer to another resource on the Web. The predicate can be any (namespace qualified) XML name. To make statements about a collection of resources, RDF specifies a simple container model, modeling sequences (ordered), bags (unordered) and lists of alternatives. RDF also supports *reification*, that is, statements about other RDF statements.

A set of RDF statements uses a particular vocabulary that defines the properties and data types that are meaningful for the application at hand. Such an RDF vocabulary can be defined by using RDF Schema (RDF-S [68]). As part of its schema language, RDF-S also defines some predefined concepts, including primitives to model a class/subclass hierarchy, relationships between classes ("properties"), and domain/range restrictions on such properties. Note that while the RDF model by itself merely provides a set of triples, RDF-S is already sufficiently expressive to describe a class hierarchy which allows some useful querying and reasoning support. For example, one could query an RDF-S system as to whether a given instance belongs to a specific class, what (inherited) properties it has, etc. [44]

2.2 DAML+OIL

While several applications are built directly on the RDF and RDF-S layers, another layer (currently under development) is the ontology layer defined by DAML+OIL [25,64]. RDF-S is missing some features that are commonly found in systems developed within the AI community (e.g. frame-based systems, description logics), while it also contains some features (most notably reification) that make it hard to provide a formal semantics for RDF-S and to provide fully automated and efficient inference engines.

DAML+OIL addresses these issues by removing support for reification, and extending RDF-S with concepts commonly found in frame-based languages and description logics. The result is a language that is compliant with RDF and RDF-S, has a sound formal semantics and an efficiently implemented inference engine. This allows not only more advanced querying, but the inference engine can also be used to detect contradictions and other errors in a DAML+OIL specification. DAML+OIL is currently used by the W3C Web-Ontology (WebOnt) Working Group as a starting point for a W3C Ontology Web Language (OWL) [65].

2.3 Applications: PICS, P3P, Dublin Core

Examples of applications that use the infrastructure sketched above include W3C's Platform for Internet Content Selection (PICS [53]), Platform for Privacy Preferences Project (P3P [21]) and the Dublin Core [14]. While PICS was defined before its more generic successor RDF, a mapping to RDF has been developed [9]. Dublin Core also predates RDF, but now also has an RDF-based serialization syntax.

3 Relation with Hypermedia Research

While the Semantic Web aims primarily at providing a generic infrastructure for machine-processable Web content, it has direct relevance to hypermedia research. To capture the breadth of relevance of the Semantic Web to hypermedia research, we have analyzed the visionary articles of Malcolm *et al*. [50], Engelbart [31] and Halasz [33]. A large proportion of these features relate directly to the Semantic Web. On the one hand, the Semantic Web infrastructure should enable several features commonly found in systems developed within the hypermedia community that are currently missing on the Web. On the other hand, the development of the currently emerging Semantic Web infrastructure could directly benefit from the models, systems and lessons learned within the hypermedia community.

Based on the articles mentioned above, we identified around 30 features that have been grouped into the eight categories discussed below:

1. **Basic node, link and anchor data model** -- Many hypermedia systems feature a model that is similar to the typical data model of nodes, links and anchors defined by the Dexter Hypertext Reference Model [34]. This model is directly applicable to the Semantic Web. To be able to annotate a specific portion of a Web resource, it needs an anchoring mechanism, and to establish a relationship between the annotation and the target resource, a linking model is necessary. The remaining features discussed below can be seen as variations on, or applications of, this basic model.

2. **Typed nodes, links and anchors** -- Many hypertext systems base a large part of their functionality on their ability to assign *types* to nodes [42], links [61], and to a lesser extent, anchors [54]. Argumentation systems such as gIBIS [18], for example, use link types to label "response-to" or "object-to" relationships (note that such relationships may, but need not be, represented by a navigational hyperlink in the user interface). RDF allows embedded and external annotation of links and anchors, and with schema languages such as RDF-S and DAML+OIL, one can easily define an (extensible) type system for links and anchors. For example, RDF-S allows "object-to" to be defined as a subtype of "response-to", and in DAML+OIL one could define "is-criticized-by" as the inverse of an "object-to" relation.
3. **Conceptual hypertext** -- Conceptual hypertext systems introduced a layered hypermedia model, adding a hyperlinked network of related index terms (or concepts) on top of a hyperlinked document base. Additional links up and down between the two levels relate the information in the documents to the concepts in the hyperindex [11,17]. More recent approaches, such as COHSE [13], go even further and use the full power of *ontologies* to improve hypertext linking based on the semantic relations among the associated concepts. The emergence of Semantic Web languages -- along with comparable approaches such as ISO's Topic Maps standard [40] -- has the potential to allow conceptual hypertext to outgrow the research labs and become a common feature of the next generation Web.
4. **Virtual links and anchors** -- Systems such as Microcosm [23,38] feature virtual (or "dynamic") links and anchors. That is, they support run-time computation of links and anchors in addition to statically defined links and anchors that are defined at authoring time. While the current Semantic Web developments tend to be mainly language-oriented (standard interfaces for generic RDF(S)-based services are yet to be defined), an RDF(S) query/inferences engine could provide an excellent basis for semantically driven hyperlink services. Related areas include ontology-driven linking as discussed in [13,20] and agent-based navigation assistance as discussed in [29].
5. **Searching and querying** -- The need to support good search and query interfaces was recognized by the hypermedia community long before the appearance of the first search engines on the Web. In one of his famous "Seven Issues", Halasz explained the need for both content-based and structure-based retrieval on hypertexts [33]. In addition, the digital library community has always stressed the use of cataloging techniques and metadata-based search [63]. While this has still to be proven in practice, RDF-enabled search engines have the potential to provide a significant improvement over the current keyword-based engines, especially when it comes to metadata and structure-based searching. An example of such a system, albeit not using RDF for encoding its semantic annotation, is the Ontobroker system discussed in [24].

6. **Versioning and authentication features** -- While features such as versioning, concurrency and authentication are not commonly recognized as fundamental hypermedia features, they have frequently been topics of hypermedia research because they are essential for one of the most important hypermedia application domains: Computer Supported Collaborative Work (CSCW). CSCW has been, for example, the driving force for most of Engelbart's work on NLS/Augment [30,31] and is listed as one of Halasz's seven issues [33]. Research on CSCW has also been carried out in the context of hypermedia systems such as NoteCards [62] and CHIPS [71]. Because early generations of hypermedia systems were designed as stand-alone systems or as part of an organization's local network, these features are even more important in Web-based collaboration. It is only because of the Web's initial focus on "read-only" browsing that these features hardly received any attention. A notable exception is the joint IETF/W3C work on WebDAV [28]. While WebDAV predates RDF, it has a similar property-based model for Web resources.
7. **Annotation** -- The ability to annotate the work of others has traditionally been an important feature of many hypertext systems, and it is another key feature of collaborative hypermedia systems. The ability to annotate Web resources was a feature in early Web browsers and servers such as NCSA's Mosaic [55,47] and Standford's ComMentor [58]. These features were not standardized and soon disappeared because the annotations could not be shared across applications in the same way as other Web resources (see [12] for a short overview of other early Web hypertext features that have now disappeared). Note that the HTML embedded link syntax by itself does not provide an appropriate, interoperable foundation for Web annotations. This syntax requires a user to have write access to the original page to be able to annotate it, which is hardly a realistic requirement on the Web. RDF and its relatives are designed to make statements about any resource on the Web (that is, anything that has a URI), without the need to modify the resource itself. This allows for rich annotations and encoding of semantic relationships among resources on the Web.
8. **User interface design: beyond navigational hypermedia** -- Early hypertext research was firmly rooted in human computer interaction, and user interface design has always been an important issue. Navigational hypermedia models such as the Dexter model, however, abstracted away from user interface details. Within Open Hypermedia Research, the user interface is part of the application's functionality and is usually more or less ignored. Within other hypermedia application domains, such as temporal hypermedia [36,37], spatial hypermedia [51] and taxonomic hypermedia [57], the presentation and interactive behavior of hypermedia structures is more complex than the typical button-like behavior of navigational links, and is often tightly intertwined with the underlying semantics of these structures. This also applies to adaptive hypermedia [10,45] and, to a lesser extent, the conceptual hypermedia systems discussed above. The

ability of the Semantic Web to model the semantics of hypermedia structures explicitly, combined with the rich functionality the Web already has in terms of presentation (e.g. by standardizing style sheets) and user interaction (e.g. by standardizing forms and link behavior), provides new opportunities to improve the hypermedia user interface by bridging the gap between hypermedia semantics and hypermedia presentation and interaction.

Despite the many relations between the Semantic Web and previous hypermedia research, many new research questions arise. The following section addresses these questions from two perspectives:

1. it investigates the issues that need to be taken into account when hypermedia features are implemented in the emerging Semantic Web infrastructure;
2. it also investigates the lessons learned from (open) hypermedia system design that need to be taken into account in the design of the Semantic Web itself.

4 Open Research Questions

Before the true potential of the Semantic Web can be fully exploited, a number of key issues need to be resolved. This section identifies open issues related to links and relationships, open hypermedia, time-based hypermedia and computer-supported collaborative work.

4.1 Links versus Relationships

While current Semantic Web languages are strong in representing (semantic) relationships between Web resources, this is insufficient for full hyperlink support. First, in addition to the currently defined *languages*, hypermedia applications also need to be able to access the associated *services*. For example, given an RDF annotation, finding the resources this annotation is about is simply a matter of dereferencing the URIs used. The other way round, however, is a lot harder. This requires intranet or even Internet crawlers that collect and index RDF annotations so that, given a particular Web resource, one can find the relevant annotations associated with that resource (the issues related to the software architecture of such services are discussed in section 4.2).

Another issue is the fact that the Web uses different approaches for modeling and encoding links and relationships across Web resources. In addition to the RDF family discussed above and the embedded links commonly found in Web languages such as HTML, WML and SMIL, W3C is also developing the XML Linking Language (XLink [26]) as a common syntax for encoding embedded and non-embedded links in XML documents. When compared to RDF, XLink provides some extra built-in link functionality (some basic traversal behavior, for example). The ability of XLink to encode semantic relationships, however, is far less than that of RDF, and XLink's hyperlink syntax is not backward-compatible with that of HTML, WML or SMIL. Whether the extra link functionality of XLink is sufficient to justify widespread adoption is still a matter of debate. Sticking to HTML for simple, embedded links while adopting the full power of the RDF family for encoding extended and external links seems to be a viable alternative. For example, taxonomic hypertext systems might benefit more from ontology-oriented languages such as DAML+OIL than from languages oriented towards navigational hyperlinks such as XLink.

A third, and more complex, issue is not related to linking across documents, but to linking across knowledge sources. Traditionally, knowledge bases, expert systems, ontologies, etc., as developed within the AI community, have focussed on representing centralized, consistent and trustworthy knowledge. On the Web, knowledge is typically decentralized, inconsistent and not always to be trusted. These differences raise new, fundamental problems, most of which remain to be solved. For example, most of the problems that arise when linking in fragments from one ontology into another are still unsolved. On the Web, an application has to be able to deal with distributed, cross-linked, incompatible or even inconsistent pieces of knowledge. A related issue is the requirement to be able to use terms from different ontology fragments. For example, Hunter *et al*. [39] describe the issues that arise when multiple metadata ontologies need to be used within a single application profile.

4.2 Open hypermedia and the Semantic Web

Open hypermedia systems (OHS) aim at adding hypermedia functionality to existing applications with minimal impact on the original application and its native file format [56]. These goals explain two fundamental differences between the OHS approach and the Web. First, while the majority of the links on the Web are *embedded links*, OHS focus on encoding links externally from the documents being linked, in order to preserve the application's native file format. Second, while Web browsers implement linking functionality within the browser, OHS architectures require minimal extra functionality of the client application because most of the link services are realized by a dedicated link server.

While the reduced complexity of embedded links on the Web has many advantages [72], for the Semantic Web the OHS approach seems more realistic. First of all, the traditional "to embed or not to embed" discussion [22] also applies to the Semantic Web. Semantic relationships are, even on the Web, expected to be significantly more complex than simple HTML uni-directional links. Embedded encoding of such information will increase the complexity of authoring Web content and increase maintenance costs when keeping Web pages up-to-date. In addition, bulky annotations will increase download times for all applications, even those that do not need to (or cannot) process the semantic annotations. The processing of (domain-specific) semantic annotations is likely to be domain specific in itself, and will thus vary from site to site. Implementing specific reasoning and inference services makes sense only at the server-side and not in a generic Web-client. The picture sketched above, with a focus on externally encoded semantic relationships and dedicated server applications to maintain and process these semantics, is very similar to the OHS approach. It suggests that many of the lessons learned in OHS modeling, software architecture and the design of interoperable protocols will be directly applicable to the Semantic Web. In the context of the current, mainly language-driven, developments on the Web, open hypermedia systems may very well provide a blueprint for an emerging Semantic Web infrastructure.

Such an infrastructure should provide interoperable interfaces and protocols to a variety of annotation services. Examples of such services include the common storage, maintenance and retrieval of semantic annotations on the Web, and the (domain) specific reasoning and inference engines that use these data effectively. A good example of the first type of service is provided by the Annotea project [43]. Annotea provides an OHS-like annotation service, based on external metadata stored by annotation servers. By deploying RDF to encode annotations, XPointer and XLink to associate metadata with the applicable portions of the document, and HTTP as the access protocol, Annotea provides a level of interoperability that many earlier attempts lacked.

4.3 Time-based hypermedia and the Semantic Web

Time-based hypermedia systems integrate hyperlink navigation with synchronized multimedia presentation [35,36]. They bring problems of timing and synchronization, inclusion of different media and streaming of data-intensive media such as video and audio. Time often plays an important role, on multiple levels, in the modeling of the semantics, narrative and document structure of hypermedia content [37,49,59]. The special role of time, and also space [51], in describing hypermedia content and hypermedia structure seems to justify the representation of these concepts as primitives of standardized hypermedia annotation vocabularies that could be built on top of languages such as RDF-S and DAML+OIL.

To integrate time-based hypermedia into the Semantic Web, a requirement is that we are able to annotate multimedia content as easily as text-based (XML) content. Existing pointing languages such as XPath and XPointer are limited to XML content, so new languages need to be developed to be able to point into the time-variant, binary encoded and compressed data formats that are common in the multimedia domain. To optimize both the quality of the presentation as well as the interactive response times, streamed delivery of media content is currently the norm in distributed multimedia environments such as the Web. Downloading bulky metadata in today's non-streamable formats is a major threat to both presentation quality and interactive response time. Instead, we need to investigate streamable versions of the RDF family of languages, and -- probably even harder -- the associated (incremental) reasoning and inference algorithms.

4.4 CSCW and the Semantic Web

Even with the current Semantic Web infrastructure and distributed Web authoring protocols such as WebDAV, many of the features related to authentication, access control, concurrency control and version control as discussed by [31,33,50] are not yet fully integrated in the Web's infrastructure. Part of this problem could be addressed by providing interoperable realizations of these features in the form of extensions to and layers upon the currently available protocols. This would, however, only solve the technical part and neglect the social and dynamic aspects of collaboration. Addressing this part of the problem requires integration of the Semantic Web infrastructure into collaborative tools that support typical groupware features related to awareness, synchronous and asynchronous communication and workflow-oriented systems that explicitly support dependencies between user tasks and other coordination mechanisms.

5 Conclusion

This paper has given an overview of the developing Semantic Web infrastructure, showed how this relates to typical hypermedia research topics and given comprehensive pointers to the relevant literature. Four important areas of research that need to be addressed to allow the Semantic Web to realise its full potential have been described.

Originally, hypertext research aimed to bring user interaction with digitally stored information closer to the semantic relations implicit within the information. Much of the more "hypertext-specific" research, however, turned to system and application-oriented topics, possibly through the lack of an available infrastructure to support more explicit semantics. The introduction of the Web, as a highly distributed, but relatively simple, hypermedia system has also influenced the character of hypermedia research. The existence of XML and RDF, along with developments such as RDF Schema and DAML+OIL, provide the impetus for realizing the Semantic Web. During these early stages of its development, we want to ensure that the many hypertext lessons learned in the past will not be lost, and that future research tackles the most urgent issues of the Semantic Web.

Acknowledgements

Frank van Harmelen provided many insightful comments on a preliminary draft of this paper. Part of the research described here has been carried out in the context of the Eureka/ITEA project RTIPA and two national projects funded by the Dutch government: Dynamo and Token2000.

References

1 *Proceedings of Hypertext '87*, Chapel Hill, North Carolina, November 1987 (ACM Press)

2 *Proceedings of the Third ACM Conference on Hypertext (Hypertext'91)*, San Antonio, TX, December, 1991 (ACM Press)

3 *Proceedings of the ACM European Conference on Hypermedia Technology (ECHT'94)*, Edinburgh, September 1994 (ACM Press)

4 GRØNBÆCK, K., MYLONAS, E. and SHIPMAN, F. M. III (eds) (1998) *Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia*, Pittsburgh, PA, June (ACM Press)

5 TOCHTERMAN, K., WESTBOMKE, J., WILL, U. K. and LEGGETT, J. J. (eds) (1999) *Proceedings of the tenth ACM conference on Hypertext and Hypermedia*, Darmstadt, Germany, February

6 BERNERS-LEE, T., HENDLER, J., AND LASSILA, O. (2001) "The Semantic Web". *Scientific American*, May http://www.sciam.com/2001/0501issue/0501berners-lee.html

7 BOS, B., LIE, H. W., LILLEY, C., AND JACOBS, I. (1998) *Cascading Style Sheets, level 2 CSS2 Specification*, 12 May http://www.w3.org/TR/REC-CSS2/

8 BRAY, T., PAOLI, J., AND SPERBERG-MCQUEEN, C. M. (1998) *Extensible Markup Language (XML) 1.0 Specification*, 10 February http://www.w3.org/TR/1998/REC-xml-19980210

9 BRICKLEY, D., AND SWICK, R. R. (2000) *PICS Rating Vocabularies in XML/RDF*, 27 March. W3C Note for discussion only, available at http://www.w3.org/TR/rdf-pics

10 BRUSILOVSKY, P. (1997) "Efficient techniques for adaptive hypermedia". *Intelligent Hypertext: Advanced Techniques for the World Wide Web*, Lecture Notes in Computer Science, Vol. 1326, edited by C. Nicholas and J. Mayfield, pp. 12-30

11 BRUZA, P. D. (1990) "Hyperindices: A Novel Aid For Searching in Hypermedia". In

Hypertext: Concepts, Systems and Applications, Proceedings of the European Conference on Hypertext, INRIA, France, November (Cambridge University Press), pp. 109-122

12 CAILLIAU, R., AND ASHMAN, H. (1999) "Hypertext in the Web - a History". *ACM Computing Surveys*, Vol. 31, No. 4, December http://www.cs.brown.edu/memex/ACM_HypertextTestbed/papers/62.html

13 CARR, L., BECHHOFER, S., GOBLE, C., AND HALL, W. (2001) "Conceptual Linking: Ontology-based Open Hypermedia". In *The Tenth International World Wide Web Conference*, Hong Kong, May, pp. 334-342 http://www10.org/cdrom/papers/246/index.html

14 CATHRO, W. (1997) "Metadata: An Overview". In *Standards Australia Seminar: Matching Discovery and Recovery*, August http://www.nla.gov.au/nla/staffpaper/cathro3.html

15 CLARK, J. (1999) *XSL Transformations (XSLT) Version 1.0*, 16 November http://www.w3.org/TR/xslt

16 CLARK, J., AND DEROSE, S. (1999) *XML Path Language (XPath) Version 1.0*, 16 November http://www.w3.org/TR/xpath

17 COLLIER, G. H. (1987) "Thoth-II: Hypertext with Explicit Semantics". In *Proceedings of Hypertext '87*, Chapel Hill, North Carolina, November 1987 (ACM Press), pp. 269-289

18 CONKLIN, J., AND BEGEMAN, M. L. (1987) "gIBIS: A Hypertext Tool for Team Design Deliberation". In *Proceedings of Hypertext '87*, Chapel Hill, North Carolina, November 1987 (ACM Press), pp. 247-251

19 CONNOLLY, D. W. (1998) *An Evaluation of the World Wide Web with respect to Engelbart's Requirements*, February http://www.w3.org/Architecture/NOTE-ioh-arch

20 CRAMPES, M., AND RANWEZ, S. (2000) "Ontology-supported and ontology-driven conceptual navigation on the World Wide Web". In *Proceedings of the 11th ACM conference on Hypertext and Hypermedia*, San Antonio, TX, USA, May - June (ACM Press), pp. 191-199

21 CRANOR, L., LANGHEINRICH, M., MARCHIORI, M., PRESLER-MARSHALL, M., AND REAGLE, J. (2000) *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification*, 15

December http://www.w3.org/TR/2000/CR-P3P-20001215/

22 DAVIS, H. (1995) "To Embed or Not to Embed...". *Communications of the ACM, Vol.* 38, No. 8, August, 108-109

23 DAVIS, H., HALL, W., HEATH, I., HILL, G., AND WILKINS, R. (1992) "Towards an Integrated Information Environment with Open Hypermedia Systems". In *Proceedings of the ACM European Conference on Hypertext (ECHT'92)*, Milan, Italy, November - December (ACM Press), pp. 181-190 http://www.mmrg.ecs.soton.ac.uk/publications/papers/conference92/echt92abstract.html

24 DECKER, S., ERDMANN, M., FENSEL, D., AND STUDER, R. (1999) "Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information". In *Semantic Issues in Multimedia Systems, Proceedings of DS-8* (Kluwer Academic Publisher), pp. 351-369

25 DECKER, S., MELNIK, S., HARMELEN, F. V., FENSEL, D., KLEIN, M., BROEKSTRA, J., ERDMANN, M., AND HORROCKS, I. (2000) "The Semantic Web: The roles of XML and RDF". *IEEE Internet Computing*, Vol. 15, No. 3, October, 63-74

26 DEROSE, S., MALER, E., AND ORCHARD, D. (2000) *XML Linking Language (XLink)*, 20 December http://www.w3.org/TR/2000/PR-xlink-20001220/

27 DEROSE, S., MALER, E., AND RON DANIEL, J. (2001) *XML Pointer Language (XPointer) Version 1.0*, 8 January http://www.w3.org/TR/2001/WD-xptr-20010108/

28 E. JAMES WHITEHEAD, J., AND WIGGINS, M. (1998) "WEBDAV: IETF Standard for Collaborative Authoring on the Web". *IEEE Internet Computing,* September/October, 34-40

29 EL-BELTAGY, S., DEROURE, D. C., AND HALL, W. (1999) "A Multiagent system for Navigation Assistance and Information Finding". In *The Fourth International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology,* April, pp. 281-295 http://www.bib.ecs.soton.ac.uk/data/1459/html/html/

30 ENGELBART, D. C. (1963) "A Conceptual Framework for the Augmentation of Man's Intellect". *Vistas in Information Handling* (Spartan Books: Washington, D. C.), pp. 1-29

31 ENGELBART, D. C. (1990) "Knowledge-Domain Interoperability and an Open Hyperdocument System". In *Proceedings of the Conference on Computer-Supported Collaborative Work* (Los Angeles, CA, October, 7-10, 1990), pp. 143-156

http://www.bootstrap.org/augment-132082.htm

32 FERRAIOLO, J. (2001) *Scalable Vector Graphics (SVG) 1.0 Specification*, 4 September http://www.w3.org/TR/SVG/

33 HALASZ, F. (1988) "Reflections on NoteCards: Seven Issues for the Next Generation of Hypermedia Systems".
Communications of the ACM, Vol. 31, No. 7, July, 836-852

34 HALASZ, F., AND SCHWARZ, M. (1994) "The Dexter Hypertext Reference Model". *Communications of the ACM*, Vol. 37, No. 2, February, 30-39

35 HARDMAN, L. (1998) *Modelling and Authoring Hypermedia Documents*. PhD thesis, University of Amsterdam http://www.cwi.nl/~lynda/thesis/

36 HARDMAN, L., BULTERMAN, D. C. A., AND ROSSUM, G. (1994) "The Amsterdam Hypermedia Model: Adding Time and Context to the Dexter Model". *Communications of the ACM*, Vol. 37, No. 2, February, 50-62

37 HARDMAN, L., VAN OSSENBRUGGEN, J., RUTLEDGE, L., MULLENDER, K. S., AND BULTERMAN, D. C. A. (1999) "Do You Have the Time? Composition and Linking in Time-based Hypermedia". In *Proceedings of the tenth ACM conference on Hypertext and Hypermedia*, Darmstadt, Germany, February, edited by Klaus Tochterman, Jorg Westbomke, Uffe K. Will and John J. Leggett, pp. 189-196

38 HILL, G., AND HALL, W. (1994) "Extending the Microcosm Model to a Distributed Environment". In *Proceedings of the ACM European Conference on Hypermedia Technology (ECHT'94)*, Edinburgh, September (ACM Press), pp. 32-40
http://www.bib.ecs.soton.ac.uk/data/1425/html/html/

39 HUNTER, J., AND LAGOZE, C. (2001) "Combining RDF and XML Schemas to Enhance Interoperability Between Metadata Application Profiles". In *The Tenth International World Wide Web Conference*, Hong Kong, May, pp. 457-466
http://www10.org/cdrom/papers/572/index.html

40 INTERNATIONAL ORGANIZATION FOR STANDARDIZATION/INTERNATIONAL ELECTROTECHNICAL COMMISSION (1999)
Information Technology -- Document Description and Processing Languages: ISO/IEC FCD 13250:1999 - Topic Maps http://www.ornl.gov/sgml/sc34/document/0058.htm

41 *The Tenth International World Wide Web Conference*, Hong Kong, May 2001

http://www10.org/

42 JORDAN, D. S., RUSSELL, D. M., JENSEN, A.-M. S., AND ROGERS, R. A. (1989) "Facilitating the development of representations in hypertext with IDE". In *Hypertext '89 Proceedings*, Pittsburgh, PA, November (ACM Press), pp. 93-104

43 KAHAN, J., KOIVUNEN, M.-R., PRUD'HOMMEAUX, E., AND SWICK, R. R. (2001) "Annotea: An Open RDF Infrastructure for Shared Web Annotations". In *The Tenth International World Wide Web Conference*, Hong Kong, May, pp. 623-632 http://www10.org/cdrom/papers/488/index.html

44 KARVOUNARAKIS, G., CHRISTOPHIDES, V., PLEXOUSAKIS, D., AND ALEXAKI, S. Querying Community Web Portals http://www.ics.forth.gr/proj/isst/RDF/RQL/rql.html

45 KOBSA, A., KOENEMANN, J., AND POHL, W. (2001) "Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships". In *The Knowledge Engineering Review*, Vol. 16, pp. 111-155

46 KOCH, T. (2002) "Introduction to a Special Issue on Metadata:: Selected papers from the Dublin Core 2001 Conference". *Journal of Digital Information*, Vol. 2, No. 2 http://jodi.ecs.soton.ac.uk/Articles/v02/i02/editorial/

47 LALIBERTE, D., AND BRAVERMAN, A. (1995) "A Protocol for Scalable Group and Public Annotations". In *The Third International World-Wide Web Conference: Technology, Tools and Applications*, Darmstadt, Germany, April http://www.igd.fhg.de/archive/1995_www95/papers/100/scalable-annotations.html

48 LANDOW, G. P. (1987) "Relationally encoded links and the rhetoric of hypertext". In *Hypertext '87 Proceedings*, Chapel Hill, North Carolina, November, pp. 331-343

49 LUESEBRINK, M. C. The moment in hypertext: a brief lexicon of time. In *The Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia*, Pittsburgh, PA, June, edited by Grønbæck, Kaj, Elli Mylonas and Frank M. Shipman III (ACM Press), pp. 106-112

50 MALCOLM, K. C., POLTROCK, S. E., AND SCHULER, D. (1991) "Industrial Strength Hypermedia: Requirements for a Large Engineering Enterprise". In *Third ACM Conference on Hypertext Proceedings (Hypertext'91)*, San Antonio, TX, December, pp. 13-24

51 MARSHALL, C., SHIPMAN, F., AND COOMBS, J. (1994) "VIKI: Spatial Hypertext

Supporting Emergent Structure". In *Proceedings of the ACM European Conference on Hypermedia Technology (ECHT'94),* Edinburgh, September (ACM Press), pp. 13-23

52 MARSHALL, C. C. (1987) "Exploring representation problems using hypertext". In *Hypertext '87 Proceedings* Chapel Hill, North Carolina, November (ACM Press), pp. 253-268

53 MILLER, J., RESNICK, P., AND SINGER, D. (1996) *Rating Services and Rating Systems (and Their Machine Readable Descriptions), Version 1.1*, 31 October http://www.w3.org/TR/REC-PICS-services

54 NANARD, J., AND NANARD, M. (1993) "Should Anchors Be Typed Too -- An Experiment with MacWeb". In *Fifth ACM Conference on Hypertext Proceedings (Hypertext'93),* Seattle, Washington, November (ACM Press), pp. 51-62

55 NATIONAL CENTER FOR SUPERCOMPUTING APPLICATIONS, UNIVERSITY OF ILLINOIS. *NCSA Mosaic for the X Window System* http://www.ncsa.uiuc.edu/SDG/Software/XMosaic/

56 NÜRNBERG, P. J., LEGGETT, J. J., AND WILL, U. K. (1998) "An Agenda for Open Hypermedia Research". In *The Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia*, Pittsburgh, PA, June, edited by Kaj Grønbæck, Elli Mylonas and Frank M. Shipman III (ACM Press)

57 PARUNAK, H. V. D. (1991) "Don't Link Me In: Set-Based Hypermedia for Taxonomic Reasoning". In *Third ACM Conference on Hypertext Proceedings (Hypertext'91),* San Antonio, TX, December (ACM Press), pp. 33-242

58 RÖSCHEISEN, M., MOGENSEN, C., AND WINOGRAD, T. (1995) "Beyond browsing: Shared comments, SOAPS, trails and on-line communities". In *The Third International World-Wide Web Conference: Technology, Tools and Applications,* Darmstadt, Germany, April http://www.igd.fhg.de/archive/1995_www95/papers/88/TR/WWW95.html

59 RUTLEDGE, L., VAN OSSENBRUGGEN, J., HARDMAN, L., AND BULTERMAN, D. C. (1998) "Structural Distinctions Between Hypermedia Storage and Presentation". In *Proceedings of ACM Multimedia* (ACM Press), pp. 145-150

60 SMITH, J. B., WEISS, S. F., AND FERGUSON, G. J. (1987) "A hypertext writing environment and its cognitive basis". In *Hypertext '87 Proceedings,* Chapel Hill, North Carolina, November (ACM Press), pp. 195-214

61 TRIGG, R. (1983) *A Network-Based Approach to Text Handling for the Online Scientific Community*. PhD thesis, University of Maryland, Technical Report TR-1346, November

62 TRIGG, R., SUCHMAN, L., AND HALASZ, F. (1986) "Supporting Collaboration in NoteCards". In *Proceedings of the Conference on Computer-Supported Collaborative Work,* Austin, TX, December, pp. 147-153

63 TUDHOPE, D., AND CUNLIFFE, D. (1999) "Semantically Indexed Hypermedia: Linking Information Disciplines". *ACM Computing Surveys*, Vol. 31, 4es, December http://www.cs.brown.edu/memex/ACMCSHT/6/6.html

64 VAN HARMELEN, F., PATEL-SCHNEIDER, P. F., AND HORROCKS, I. (eds) (2001) *Reference description of the DAML+OIL ontology markup language*, March http://www.daml.org/2001/03/reference.html

65 W3C (2001) Web-Ontology (WebOnt) Working Group, see http://www.w3.org/2001/sw/WebOnt/

66 HOSCHKA, P. (ed.) (1998) *Synchronized Multimedia Integration Language (SMIL) 1.0 Specification,* 15 June http://www.w3.org/TR/REC-smil/

67 LASSILA, O. and SWICK, R. R. (eds) (1999) *Resource Description Framework (RDF) Model and Syntax Specification*, 22 February http://www.w3.org/TR/REC-rdf-syntax/

68 BRICKLEY, D. and GUHA, R.V. (eds) (2000) Resource Description Framework (RDF) Schema Specification 1.0, 27 March http://www.w3.org/TR/rdf-schema/

69 PEMBERTON, S., *et al*. (2000) *XHTML 1.0: The Extensible HyperText Markup Language: A Reformulation of HTML 4.0 in XML 1.0*, 26 January http://www.w3.org/TR/xhtml1/

70 ADLER, S., *et al*. (2001) *Extensible Stylesheet Language (XSL) Version 1.0*, 15 October http://www.w3.org/TR/xsl/

71 WANG, W., AND HAAKE, J. M. (1998) "Flexible Coordination with Cooperative Hypermedia". In *The Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia*, Pittsburgh, PA, June (ACM Press), pp. 245-255

Edited by Kaj Grønbæck, Elli Mylonas and Frank M. Shipman III

72 WHITEHEAD, E. J. (1999) "Control choices and network effects in hypertext systems". In *Proceedings of the 10th ACM conference on Hypertext and Hypermedia*, Darmstadt, Germany, February, edited by Klaus Tochterman, Jorg Westbomke, Uffe K. Will and John J. Leggett (ACM Press), pp. 75-82

Top of page New papers Past issues Index of papers