



*From the Editor in Chief...*

# Semantic Services

**Robert E. Filman** • *Research Institute for Advanced Computer Science* • [filman@computer.org](mailto:filman@computer.org)

I confess I was surprised by the success of the World Wide Web. I'm a longtime member of the techno-elite. Despite the proliferation of personal computers, I didn't imagine the hoi polloi mastering networking. I didn't expect that grandmothers would relish the art of posting their own home pages, viewing pictures of their friends' grandchildren, and ordering their birthday presents from e-toy companies. I didn't foresee URLs on billboards.

A prime reason for the Web's success was that it is such an essentially human activity. Because Web site developers know that their sites will be interpreted by people, they can expend less energy at figuring out exactly what needs to happen. The natural language description of a particular application is likely to be precisely the best way to describe that action to a person, and generating the natural language description is, well, natural. You don't need a Masters in computer science to create a Web page. (It might even get in the way.) This contrasts with building computer systems where, by and large, you have to get everything correctly coordinated.

The other prime reasons for the Web's success were the cleverness of its design and search engines. The Web's design was shrewd in being both open and expansive. By not imposing very much structure on "universal" resource locators or the meanings of messages, developers could adapt the Web protocols to their needs. Fortunately, the system converged to diversity without chaos, but this result was not a foregone conclusion.

Borges's library contained all the truths of the universe, but you would be hard pressed to find any of them. Search engines — first-generation engines like Altavista, which performed statistical analysis on lexical text, and second-generation engines like Google, which extend that with analysis of the hypertext graph structure — are key to the successful Web. It wouldn't attract grandmothers if

they had to be initiated into the secrets of the software elite to actually find things.

## Web Services

The Web, in its original incarnation, was a mechanism for distributing static data results — a grandchild's photo or a scientific paper. However, it didn't take long before more complex processes were superimposed on the underlying Web structure. These included both information-providing services such as product prices, stock quotes, and flight arrival times, and services that actually modified the external world, producing delivered products, stock sales, and ticket reservations. Even the early Web had mechanisms for processing, but that Web was centered on interfaces for serving people. The idea behind Web services is to use HTTP-like protocols to make network processing available to programs. These can include both "one shot" services that just perform some action or return some information, on through the entire issue of choreographing multiple services through a series of steps, including contingent actions and the ability to monitor the figures of the dance.

Before Web services, programmers used programming-language-centric, distributed object technology like Corba and Java RMI to build network applications. Web services have two critical advantages over such architectures. First, programming-language-based mechanisms demand considerable shared knowledge between the programmer of the service and the programmer of the client (usually realized in an interface definition language or shared class files). Technically, modifying a service demands modifying all its clients — even if the modification doesn't change anything the client uses. Second, HTTP has become ubiquitous, able to tunnel through protective firewalls in a single gulp. System administrators who wouldn't dream of allowing an RMI service into their network are smugly complacent about anything coming in on port 80 — even when

## EDITOR IN CHIEF

Robert E. Filman • [filman@computer.org](mailto:filman@computer.org)

## ASSOCIATE EDITOR IN CHIEF

Li Gong • [li.gong@sun.com](mailto:li.gong@sun.com)

## EDITORIAL BOARD

Jean Bacon • [jean.bacon@cl.cam.ac.uk](mailto:jean.bacon@cl.cam.ac.uk)Miroslav Benda • [miro@amazon.com](mailto:miro@amazon.com)Elisa Bertino • [bertino@dsi.unimi.it](mailto:bertino@dsi.unimi.it)Scott Bradner • [sob@harvard.edu](mailto:sob@harvard.edu)Siobhán Clarke • [siobhan.clarke@cs.tcd.ie](mailto:siobhan.clarke@cs.tcd.ie)Fred Douglass • [f.douglass@computer.org](mailto:f.douglass@computer.org)Stuart I. Feldman • [sif@us.ibm.com](mailto:sif@us.ibm.com)Ian Foster • [foster@cs.uchicago.edu](mailto:foster@cs.uchicago.edu)Michael N. Huhns • [huhns@sc.edu](mailto:huhns@sc.edu)Leonard Kleinrock • [lk@cs.ucla.edu](mailto:lk@cs.ucla.edu)G.S. Kuo • [gskuo@ieee.nccu.edu.tw](mailto:gskuo@ieee.nccu.edu.tw)

(IEEE Communications Society Liaison)

Doug Lea • [dl@altair.cs.oswego.edu](mailto:dl@altair.cs.oswego.edu)Frank Maurer • [maurer@cpsc.ucalgary.ca](mailto:maurer@cpsc.ucalgary.ca)Daniel A. Menascé • [menasce@cs.gmu.edu](mailto:menasce@cs.gmu.edu)Chris Metz • [chmetz@cisco.com](mailto:chmetz@cisco.com)Peter Norvig • [peter@norvig.com](mailto:peter@norvig.com)Charles J. Petrie • [petrie@nrc.stanford.edu](mailto:petrie@nrc.stanford.edu)  
(EIC emeritus)Krithi Ramamritham • [krithi@cse.iitb.ac.in](mailto:krithi@cse.iitb.ac.in)Ravi Sandhu • [sandhu@gmu.edu](mailto:sandhu@gmu.edu)Munindar P. Singh • [singh@ncsu.edu](mailto:singh@ncsu.edu)

(EIC emeritus)

## STAFF

Lead Editor: Steve Woods

[swoods@computer.org](mailto:swoods@computer.org)

Group Managing Editor: Gene Smarte

Staff Editors: Scott L. Andresen,

Kathy Clark-Fisher, and Jenny Ferrero

Production Assistant: Monette Velasco

Magazine Assistant: Hazel Kosky

[internet@computer.org](mailto:internet@computer.org)

Graphic Artists: Larry Bauer, Alex Torres

Contributing Editors: David Clark,

Greg Goth, Keri Schreiner, Joan Taylor

Publisher: Angela Burgess

Membership/Circulation Marketing Manager:

Georgann Carter

Business Development Manager: Sandy Brown

Advertising Supervisor: Marian Anderson

## CS Magazine Operations Committee

Jean Bacon (chair), Thomas J. Bergin, Pradip Bose,  
Doris L. Carver, George Cybenko, John C. Dill,  
Frank E. Ferrante, Robert E. Filman, Forouzan Golshani,  
Rajesh Gupta, Warren Harrison, Mahadev  
Satyanarayanan, Nigel Shadbolt, Francis Sullivan

## CS Publications Board

Rangachar Kasturi (chair), Jean Bacon,  
Laxmi Bhuyan, Mark Christensen,  
Thomas Keefe, Deepandra Moitra,  
Steven L. Tanimoto, Anand Tripathi

its connecting to something that can wreak equivalent havoc. (On the other hand, the lack of the shared knowledge of IDLs and such might prove to be a future impediment to reliability and maintainability of Web service systems, but that's fodder for a different discussion.)

## Form and Meaning

As I've noted, keys to the Web's success have been search engines and the human ability to interpret Web pages. The latter supports the former: were it not for people looking at the little paragraphs generated by the search engines, search engines wouldn't work too well, either. How can these properties be realized in an autonomous system?

A hot topic at this year's World Wide Web conference was the convergence of Web services and the Semantic Web. The idea behind the Semantic Web is that generators of Web pages or services will create formal declarative descriptions of these services that programs can use to find the appropriate service and use it correctly.

Declarative descriptions have two parts: form and content. Form delimits what can be said and how to say it. This includes not only the low-level lexical syntax of commands (for example, XML of the following format...) but also the quality of statements you can say about a service. For example, a particular descriptive system might allow you to assert only simple ground-level facts about a service ("the cost of this service is \$1"), quantified statements ("the cost of every service at this site is \$1"), or more complex statements with richer logic ("\$1 buys an hour of this service," or "users can try this service for free for an hour, if they haven't used the service in the last month."). Ideally, Semantic Web services demand the ability to make not just these assertions but also more complex ones, with the appropriate inferences about elements such as temporal durations and quantities.

What kind of formats can we reasonably expect? We want two things from our descriptive logic, fast computability and rich expressiveness. That is, we want an autonomous process to be able to quickly determine the consequences of a set of assertions, and that the lan-

## How to Reach IC

### Articles

We welcome submissions about Internet application technologies. For detailed instructions and information on peer review, see *IEEE Internet Computing's* author guidelines are available online at [www.computer.org/internet/author.htm](http://www.computer.org/internet/author.htm).

### Letters to the Editor

Please send letters, including reference to the article in question, via email to [internet@computer.org](mailto:internet@computer.org).

### Reuse Permission

For permission to reprint an article published in *IC*, contact William J. Hagen, IEEE Copyrights and Trademarks Manager, IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08855-1331; [w.hagen@ieee.org](mailto:w.hagen@ieee.org). Complete information is available at [computer.org/permission.htm](http://computer.org/permission.htm). To purchase reprints, see [www.computer.org/author/reprint.htm](http://www.computer.org/author/reprint.htm).

IEEE  
**distributed systems**  
ONLINE  
Expert-authored articles and resources

**IEEE Distributed Systems Online**  
brings you peer-reviewed features, tutorials, and expert-moderated pages covering a growing spectrum of important topics, including

- **Grid Computing**
- **Mobile and Wireless**
- **Distributed Agents**
- **Security**
- **Middleware**
- **and more!**

**IEEE Distributed Systems Online**  
supplements the coverage in *IEEE Internet Computing* and *IEEE Pervasive Computing*. Each monthly issue includes magazine content and issue addenda such as interviews and tutorial examples.

**To receive regular updates, email [dsonline@computer.org](mailto:dsonline@computer.org)**

dsonline.computer.org

guage of assertions be able to express complex descriptions. Unfortunately, these are incompatible goals; richer logics are the royal road to exponential complexity and noncomputability. Thus, the developers of descriptive systems usually settle for something with only primitive semantics like ground clauses or, moving a small step up the expressivity hierarchy, Horn-clause logics like Prolog.

Form dictates how to say things, but it doesn't tell us what the symbols mean. Humans can have trouble with word meaning, and the intersection of precise computers and ambiguous humans can cause even more misunderstanding. I'm reminded of two stories that illustrate this point. One was related by a US Undersecretary of Defense, who said that when he asks the Army to secure a building, they send over a troop of soldiers with automatic weapons. When he asks the Navy to secure a building, they send a yeoman to lock the doors and turn off the lights. And when he asks the Air Force to secure a building, they get a three-year lease with option to renew.

The other was from a friend who had worked at Hewlett Packard on the first handheld calculator. When he asked the information system accounting folks how much profit HP had made on that invention, they responded with a list of 43 questions to clarify what he meant by "profit." (For example, "How should earnings and losses on foreign currency futures transactions be treated?")

### The Only Telephone in the World

The simplest way of ascribing meaning to symbols, the one adopted by the classical Web, is to direct those symbols to people, who will apply their natural interpretations to the symbols — often correctly. However, Web services must act (mostly) autonomously.

It is possible to infer the meaning of complex statements through statistical or artificial-intelligence techniques. After all, statistical processes have performed surprisingly well for Web search, though not without a bit of human help.

Ideally, we would like Web services to be annotated by formal, computable descriptions of what they do. There's currently a lot of activity on generating semantic description languages for Web pages. The Resource Description Framework (RDF)<sup>1</sup> is based on the idea of describing resources in terms of subject-predicate-object triples. Several groups are furiously pursuing the ability to write more complex languages — most notably the combination of the DARPA Agent Markup Language (DAML; [www.daml.org](http://www.daml.org)) with the Ontology Inference Layer (OIL; [www.ontoknowledge.org/oil/](http://www.ontoknowledge.org/oil/)) — to produce a richer but computable language for describing resources. However, creating the formal representation structure is considerably easier than actually modeling the complexities of the real world. It's one thing to be able to provide a language for "security" or "profit," and quite another to completely express the ramifications in automatic weapons and exchange rates needed to autonomously decide which services to invoke.

Ultimately, semantic descriptions of Web services will arise through human processes, either because committees get together and create standards for particular subdomains or because a particular style used by some influential service providers comes to be commonly adapted. I can't predict whether this will happen. No one wants to be the only person in the world with a telephone. Telephones acquire more value as more people have them. People will not go to the trouble of doing something (annotating their Web services with semantic information) unless some benefit happens from this markup (the service search engine will find them). The service search engines won't be written unless there is markup to be found.

No matter what happens, I'm sure I'll be surprised — whatever that means.<sup>2</sup> □

#### References

1. S. Decker et al., "The Semantic Web: The Roles of XML and RDF," *IEEE Internet Computing*, vol. 4, no. 5, Sept. 2000, pp. 63–74.
2. M. Gardner, *The Unexpected Hanging and Other Mathematical Diversions*, Chicago University Press, 1991, pp. 11–23.