



From the Editor in Chief...

# The Pragmatic Web

Munindar P. Singh • [singh@ncsu.edu](mailto:singh@ncsu.edu)

People use the Web to share information. For machines to exploit information on the Web, however, presupposes well-structured, meaningful markups, which is what the World Wide Web Consortium's Semantic Web activities seek to develop ([www.w3.org/2001/sw/](http://www.w3.org/2001/sw/)). Current approaches are limited, however. They provide meaning only to the extent that it can be captured statically. I claim that successful approaches to creating the Semantic Web lie within the scope of *pragmatics*, which I introduce below.

The Web is a symbolic system, and its symbols are Web page content and markups. Normally, browsers and human users interpret these symbols. For machines to exploit information on the Web, we must consider the meanings of symbols explicitly. *Semiotics*, the study of symbolic systems, has three parts:<sup>1</sup>

- *Syntax*, or structure
- *Semantics*, or structure-based meaning
- *Pragmatics*, or context-based meaning

We can apply semiotics to the Web because it helps us think systematically about the symbols that constitute it. On the Web, syntax refers to tags (such as HTML or XML tags); semantics refers to what those tags denote (parts inventories, for example), and pragmatics refers to the context-sensitive aspects of meaning (for instance, inventory dates and times or processes affecting inventory size).

We have a good understanding of Web structure, and Web semantics is drawing attention. This column focuses on pragmatics. Pragmatics adds to but doesn't replace the semantics. Moreover, advances in semantic representations such as ontologies will facilitate pragmatics capture.

### Service Composition

Web services enable program-to-program interactions over the Web ([www.w3.org/2002/ws/](http://www.w3.org/2002/ws/)). Current services approaches concentrate on syntax, but semantic Web services are obviously desirable.

An explicit semantics will enable service discovery and appropriate use without human intervention. The key challenge, however, is in composing Web services automatically.

For businesses, composing multiple services, rather than accessing a single service, is essential. Human travel agents, for example, provide a great user interface – friendly and with a nice touch. When travel agents interface to a single service type, say, airlines, they are vulnerable to the providers (the airlines) removing them from the equation by offering services directly to consumers. When travel agents interface to multiple services to offer complex products (package tours, for example), they are much harder to work around. In general, it seems that the more subtle their composition, and thus the greater their complexity, the more easily intermediaries can flourish.

### New Challenges

Semantic Web services present some major challenges, especially in terms of composition, that require a pragmatic approach.

- *Service description*. Current approaches assume that service providers can describe services simply by listing their methods. Even using standardized languages, however, these descriptions might be incomplete because specialized communities of practice use services in novel ways. In addition, rather than merely invoking a method, the consumer and service provider negotiate about whether to interact with each other and, if so, how.
- *Service discovery and location*. Current approaches assume that a logically central services registry can make the right matches for all comers. A registry cannot be implicitly trusted in real life, however, where prospective service consumers need to find trustworthy providers. The registry has no interactions of its own by which to judge the quality or trustworthiness of a service pro-

vider or a prospective consumer, and both parties must be able to trust each other. This mutual trust depends on the participants' specific interactions and contexts – something that a registry cannot model for all (this is related to service description, above). A proper solution requires social mechanisms for evaluating and discovering trustworthy parties, both providers and consumers.

- **Interaction.** Current service-invocation approaches are geared for method invocation, not for enabling service composition. Any higher-level constraints among methods, such as for long-lived interactions, are ignored. (Such an e-commerce interaction might involve changing an order because of unexpected conditions, for example, or getting a refund for a faulty product.) Even short-lived settings involve protocols – to check a service provider's authentication and authorization before accepting its order, for example. Interesting service compositions will often involve subtle interactions among participating services. In other words, these protocols describe multistep interactions, potentially among equal, proactive participants, not single-shot method invocations between client and server.

These challenges involve the key functionalities of a services architecture: describing, discovering, and engaging services. For each functionality, we need to apply an appropriate context.

## Design Principles

What are some of the principles involved in designing and linking pragmatically sound Web services?

- **User before provider.** Modeling service consumers' context-dependent needs is equally or more important than modeling the service providers' capabilities. Service description, discovery, and invocation must be tied to the context of the intended compositions.
- **Process before data.** The context in which data is produced or consumed is important, though difficult to capture. But anyone who models the processes behind the data will have a better understanding of the data's meaning than someone who views data without the associated processes.
- **Interaction before representation.** Just as functional interfaces to data structures hide their implementation details, so too will service interaction models hide the “excess” data semantics that might otherwise be revealed.

These principles involve models that service providers, consumers, or other composers might apply. Accommodating these principles means focusing not on the data but on the processes and contexts in which it is used.

Comedian George Carlin once asked, “If man evolved from monkeys and apes, how come we still have monkeys and apes?” The Pragmatic Web augments and refines the Semantic Web, just as the Semantic Web refines the “syntactic” Web. So contra Carlin, when the Pragmatic Web arises, the Semantic Web will still be around. ☐

## Reference

1. Charles Morris, *Foundations of the Theory of Signs*, Univ. of Chicago Press, Chicago, 1938.

## EDITOR IN CHIEF

Munindar P. Singh • [singh@ncsu.edu](mailto:singh@ncsu.edu)

## ASSOCIATE EDITOR IN CHIEF

Robert Filman • [rfilman@arc.nasa.gov](mailto:rfilman@arc.nasa.gov)

## EDITORIAL BOARD

Salah Aidarous • [saidarous@necam.com](mailto:saidarous@necam.com)  
(IEEE Communications Society Liaison)

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)

Fred Douglass • [f.douglass@computer.org](mailto:f.douglass@computer.org)

Stuart I. Feldman • [sif@us.ibm.com](mailto:sif@us.ibm.com)

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

Michael N. Huhns • [huhns@sc.edu](mailto:huhns@sc.edu)

Leonard Kleinrock • [lk@cs.ucla.edu](mailto:lk@cs.ucla.edu)

Doug Lea • [dl@altair.cs.oswego.edu](mailto:dl@altair.cs.oswego.edu)

Frank Maurer • [maurer@cpsc.ucalgary.ca](mailto:maurer@cpsc.ucalgary.ca)

Chris Metz • [chmetz@cisco.com](mailto:chmetz@cisco.com)

Charles E. Perkins • [charliep@iprg.nokia.com](mailto:charliep@iprg.nokia.com)

Charles J. Petrie • [petrie@nrc.stanford.edu](mailto:petrie@nrc.stanford.edu)  
(EIC emeritus)

Agostino Poggi • [poggi@ce.unipr.it](mailto:poggi@ce.unipr.it)

Anthony Michael Rutkowski • [amr@chaos.com](mailto:amr@chaos.com)

Ravi Sandhu • [sandhu@gmu.edu](mailto:sandhu@gmu.edu)

## STAFF

Assistant Publisher: Dick Price  
[dprice@computer.org](mailto:dprice@computer.org)

Lead Editor: Steve Woods  
[swoods@computer.org](mailto:swoods@computer.org)

Production Assistant: Monette Velasco  
[mvelasco@computer.org](mailto:mvelasco@computer.org)

Magazine Assistant: Hazel Kosky  
[internet@computer.org](mailto:internet@computer.org)

Graphic Artists: Larry Bauer, Alex Torres

Contributing Editors: David Clark, Tricia Dunn,  
Greg Goth, Keri Schreiner, Linda World, Martin Zacks

Publisher: Angela Burgess

Membership/Circulation Marketing Manager:  
Georgann Carter

Business Development Manager: Sandy Brown

Advertising Supervisor: Marian Anderson

## CS Magazine Operations Committee

George Cybenko (chair), James H. Aylor,  
Thomas J. Bergin, Frank E. Ferrante, Forouzan Golshani,  
Rajesh Gupta, Steven C. McConnell, Ken Sakamura,  
Mahadev Satyanarayanan, Nigel Shadbolt,  
Munindar P. Singh, Francis Sullivan, James J. Thomas

## CS Publications Board

Rangachar Kasturi (chair), Mark Christensen,  
George Cybenko, Thomas Keefe,  
Richard A. Kemmerer, Gabriella Sanniti di Baja,  
Steven L. Tanimoto, Anand Tripathi

