From the Editor-in-Chief . . .

The Service Web

Munindar P. Singh • singh@ncsu.edu

The Web can seem so uniformly accessible that it is easy to begin thinking of it as a single large system, its distinguishing purpose being the exchange of marked-up documents and its distinguishing characteristic being the hyperlinks among those documents.

But the Web is actually different things to different people and understanding its main variants can help us program accordingly.

Webs of Joy

In his keynote presentation at this year's JavaOne conference,[1] Bill Joy, chief scientist of Sun Microsystems, offered a classification of four main kinds of "webs," which he distinguished based on the modalities of the interface as experienced by the user:

- The *near web* involves conventional mouse-keyboard-monitor interaction with a personal computer, typically for purposes of information, such as surfing the Web.
- The *far web* entails interaction with a computer from across a room, as with a TV remote control, typically for entertainment, such as viewing a movie.
- The *here web* features interaction with a mobile device that you carry with you, typically involving narrow bandwidths for input and output.
- The *weird web* extends interaction through emerging interface technologies, such as voice and wearable computing.

If we want to engineer new applications, we will need new kinds of services as well—services that would live in the Web.

Joy defined two additional webs where there are no direct user interactions: the e-commerce web (dealing with the supply chain network of business transactions) and the pervasive web (dealing with the interactions of embedded devices).

The Medium Is the Service

Joy's classification provides an interesting historical perspective on how Web applications are evolving; the near and far webs, for example, were concepts he said originated in the early work on Java. His classification scheme is confined, however, to the interface for a single user and doesn't handle the varieties of interaction that we see in practice. For example, he makes no distinction between a single user surfing the Web and two users engaged in a chat. Moreover, the distinction between the near and far webs may disappear as proximity to a keyboard or screen matters less. Consider Internet telephony: Is it a near or a far application? As interface technologies improve, in effect the weird web will subsume both the near and the far webs. The constraining modalities of the here web, too, will relax over time, thereby reducing the effects of constraints defined along the lines that Joy emphasizes.

But another feature of the here web will remain: People will continue to want—and increasingly to expect—the Internet to be where they are. I have argued that this expectation will mean not just physical connectivity, but also something more conceptual and more personal—specifically, an Internet that is not only portable but also "knowledgeable" about what you have been doing, where you are, and what you wish to do.[2]

So let's take Joy's classification further by building a taxonomy of Web applications and relating it to the architectural components and programming abstractions of a service.

Right from the beginning, the Web has supported not just document exchange but services (for example, through CGI). However, early services were first used only by people and later by simple aggregating programs such as shopbots. A typical shopbot would find the products and prices from several merchant sites and present the results on a single page to a user. A shopbot would invoke operations mainly by "screen scraping" sites that were intended for humans, submitting the same forms that a browser would submit and extracting data from pages intended to be shown to users. This architecture has been practical for a while, but it essentially limits the shopbots to doing what a single human user would do. True, they can access a bunch of sites much faster than you or I can, but they fundamentally add nothing to the picture.

What we need are better services. There are proposals in the works that would offer programmatic interfaces to the services provided by a site or that would otherwise standardize the interaction. The W3C's simple object access protocol (SOAP) is a case in point.[3]

But if we want to engineer new applications in new ways, we will need new kinds of services as well—services that would live in the Web and for the Web. They might not be ends in themselves, but would exist expressly to support new applications. A typical application might comprise several services to deliver added value to a user.

Webs Revisited

Some of the generic, building-block services that come to mind are those for security, finance, transactions, location discovery, personalization, and socialization. Each service would aggregate key functionalities for its intended purpose, and each would form its own web within the Web. We would have a web of security functions, a web of finance functions, and so on.

The interface modalities that underlie Joy's classification would not relate to these service webs directly; such interfaces exist on the fringes of the Web, anyway. The prototypical applications he describes, however, *do* map to these services. Surfing the Web would skim over the personalization web. Shopping would involve part of the finance web. Entertainment, going solo, is again just a matter of personalization; entertainment in a group, such as a chat, also runs through the socialization web. Mobile situations would place the user in the location and personalization webs. B2B applications would have a larger footprint in the finance and transactions webs.

Services offer a structure for creating a Web that is more manageable in size, more knowledgeable about your needs, and generally more friendly and useful. ■

REFERENCES

1. B. Joy, JavaOne keynote address, June 2000; available online at http://java.sun.com/javaone/javaone00/transcripts/keynote2.html.
2. M.P. Singh, "In the Thick of Things," *IEEE Internet Computing*, vol. 4, no. 1, Jan. 2000, pp. 6-7.
3. Simple Object Access Protocol (SOAP) 1.1, W3C note; available online at http://www.w3.org/TR/SOAP/.

IEEE INTERNET COMPUTING

IEEE Computer Society Publications Office
10662 Los Vaqueros Circle, PO Box 3014
Los Alamitos, CA 90720-1314