WEB 2.0: IS IT REALLY DIFFERENT?

Ten years ago (has it really been that long?), we were well into the technology hype of the World Wide Web. New technologies were going to revolutionize how we live, how businesses work—just about everything. Enormous amounts of money were invested in all kinds of Web-based ideas, from optical fiber in the ground to telecommunications equipment to online banking to selling pet food on the Internet using a sock puppet mascot.

A few years later, the bubble burst. Both technologies and business models had been oversold, and the market collapsed. From a business (and investing) point of view, novel Web technologies were something to be avoided.

What is often missed about the post-bubble time is that Internet usage didn't stop, or even pause very much. Even electronic commerce, the big hype-driver of the late '90s, continued to grow in actual use even as company valuations crashed. Similarly, technology innovation and development continued, although it was much harder for a time to raise capital for a startup company.

Today, hype is back, there's a lot of startup activity, and there are new claims about how Web technology will revolutionize this or that. Much of the excitement is described as "Web 2.0," suggesting the second major release of Web technology. This begs at least two questions: first, what exactly is "Web 2.0?" Second, what does it really mean?

Pinning down Web 2.0 is like trying to scoop up water with your hands. You can't really hold onto all of it, but after most of the water runs through your fingers, there's still something left. The common threads seem to

Win Treese

Putting it Together

Because of the lull after the dot-com bubble burst, Web 3.0 looks like a bigger technology shift than perhaps it really is.

fall into several main categories, encompassing both technologies and structures for using technology. The main ones seem to be interactivity, social networking, tagging, and some aspects of Web services. In addition, the widespread popularity of Web publishing tools such as blogs can be folded in. Let's look briefly at each of these.

Interactivity is clearly a dominant Web 2.0 theme. Most often, people seem to talk about "AJAX" as the Web 2.0 technology. AJAX is said to stand for "Asynchronous JavaScript and XML," and the best-known example of an AJAX application is probably Google Maps. What distinguishes Google Maps from previous Web-based map programs is that the map is interactive. You can slide the map around, and usually you don't have to wait for parts of the map from outside your view to be filled in. After the Web browser displays the original map, JavaScript code executing in the browser gets busy fetching other parts of the map so they will be available when you want them. That's the "asynchronous" part: It happens in the background, without a specific request from the user. And because most browsers with JavaScript have included code for handling XML, it is convenient to use XML for downloading other data needed for the applications.

Another technology of interactivity is Flash, developed by Macromedia (which was acquired by Adobe Systems). Flash has been around for about ten years, but it is becoming more common. Since it is not built in to Web browsers, Web users have had to download a plug-in for it, which slows the adoption.

However, it is nearly ubiquitous today, which makes it more plausible as a tool for Web developers. As a tool for interactivity, Flash is somewhat different from AJAX. Flash is typically used for making an active application within the Web browser, such as a small game or interactive data exploration, whereas AJAX is commonly used to provide more interaction with an application running on a Web server, as well as to change the interactive nature of almost anything displayed on a Web page.

The second main area of Web 2.0 applications is "social networking." This is not a technology so much as the Web facilitating the tying together of social groups. As a simple example, imagine a diagram of nodes with yourself at the center, and links to nodes representing your friends, family, or business colleagues. Now imagine expanding the diagram to include their friends, families, and business colleagues. Social networking software is, more or less, programs that capture such information and make it usable in various ways. For example, if you wanted to get a job at company X, the social network software might be able to tell you that you know someone who knows someone who works there. Some well-known social network systems including Friendster (one of the earliest) and LinkedIn, but there are many, many others.

A third main area is "tagging." Like interactivity and social networking, tagging is a fairly simple idea at its core. The key idea is that you want to associate a keyword, or "tag" with a data object. The object might be a picture, Web site, an email message, or almost anything else. Later, when you want to find it, you search on the tags, rather than trying to keep everything organized in (say) a hierarchy of folders. In other words, searching becomes the fundamental method of organization, and tagging is a way of adding some

additional information (sometimes called "metadata") to data objects.

Tagging has some interesting properties. First, the tags are personal, but not always private. What is important to you may be different than what is important to someone else, but that's fine—you each have your own tags for the objects. Many systems allow tags to be shared, and the result may be a fairly common set of tags that effectively group together objects by some relationship. For example, the photo Web site Flickr lets users tag photos, both those they post and those posted by others; the result is a searchable database of photos with a collected view of what the photos are about. Similarly, the bookmarking site del.icio.us (that's the domain name) lets you associate tags with bookmarks for Web sites, and it makes available the tags that other people have used.

Finally, it is worth noting the use of Web services in Web 2.0. The term "Web services" can be somewhat fuzzy in its usage, but generally means the use of the Web by computer programs, rather than just as a way to display pages in a Web browser. In a sense, an AJAX program is a client of Web services, because it is using the Web to fetch additional XML data. One doesn't hear so much about Web services directly in Web 2.0, but it's a critical part of the infrastructure used by Web 2.0 applications.

From a business point of view, the model for most companies seems to be both simple and old: Aggregate an audience and sell advertising. Whereas Web 2.0 technologies may be further developed, the business models haven't changed much. Here the lessons of the 1990s are extremely relevant: When it comes to building an audience, some Web sites will be extraordinarily successful, and they can generate real revenue. However, what makes such a site successful is a combination of fashion and of some real value for the users (in some cases, being fashionable may be sufficiently valuable by itself, but it's an especially risky investment). The result of this in the marketplace is a very small number of big winners, and a very large number of also-rans. And it does sound like a rerun.

But that shouldn't mask the reality of how the technology is progressing. Rather than "Web 2.0," I would say what we are seeing is a continuous evolution of ideas and exploratory innovations. Occasionally even a small change can transform how we look at problems or opportunities, but most changes in technology are incremental. Rather, in the hype, what we are seeing is a resurgence of interest in applying the technologies. Because of the lull following the bursting of the dot-com bubble, it looks like a bigger technology shift than perhaps it really is.

And, at the moment, the technology still seems to be in the early stages. So far, there seems to be more AJAX hype than reality, in the sense that one doesn't run into AJAX applications all the time. Although the components have been available for some time, the Web development community as a whole is still working to figure out how best to use the technology.

On the other hand, anyone developing applications for the Web, or improving the ones they have, should be paying close attention. Whether or not Web 2.0 is the second coming of the Web or just a set of incremental improvements, the technologies and the ideas are becoming part of the common Web experience. The history of the Web shows that new ideas and interactions often make the jump from novel new technology to expected behavior, and the Web 2.0 technologies are certainly no exception. Think about how to make the Web experience for your applications as excellent as the leading applications, and take advantage of leading-edge technology to do so. ~

Win Treese is co-author of *Designing Systems for Internet Commerce*. He can be reached at treese@acm.org.