



Enterprise Architecture

Vol. 9, No. 7

**CUTTER**  
**CONSORTIUM**

## Web 2.0

by Tom Welsh, Senior Consultant,  
Cutter Consortium

The blogosphere and, increasingly, the media are humming with excitement about Web 2.0. But what lies behind this catchy new buzzword, and does it really represent a step forward from the “old” Web? Even enthusiasts do not agree on just what defines a “Web 2.0 site,” and when they brainstorm the question, the resulting feature lists can grow to remarkable lengths. This report examines Web 2.0 from all sides, including its potentially revolutionary social implications.

Executive  
Report

# Cutter Business Technology Council



Rob Austin



Tom DeMarco



Christine Davis



Lynne Ellyn



Jim Highsmith



Tim Lister



Ken Orr



Lou Mazzucchelli



Ed Yourdon



## About Cutter Consortium

Cutter Consortium is a unique IT advisory firm, comprising a group of more than 150 internationally recognized experts who have come together to offer content, consulting, and training to our clients. These experts are committed to delivering top-level, critical, and objective advice. They have done, and are doing, groundbreaking work in organizations worldwide, helping companies deal with issues in the core areas of software development and agile project management, enterprise architecture, business technology trends and strategies, enterprise risk management, business intelligence, metrics, and sourcing.

Cutter delivers what no other IT research firm can: We give you Access to the Experts. You get practitioners' points of view, derived from hands-on experience with the same critical issues you are facing, not the perspective of a desk-bound analyst who can only make predictions and observations on what's happening in the marketplace. With Cutter Consortium, you get the best practices and lessons learned from the world's leading experts, experts who are implementing these techniques at companies like yours right now.

Cutter's clients are able to tap into its expertise in a variety of formats including print and online advisory services and journals, mentoring, workshops, training, and consulting. And by customizing our information products and training/consulting services, you get the solutions you need, while staying within your budget.

Cutter Consortium's philosophy is that there is no single right solution for all enterprises, or all departments within one enterprise, or even all projects within a department. Cutter believes that the complexity of the business technology issues confronting corporations today demands multiple detailed perspectives from which a company can view its opportunities and risks in order to make the right strategic and tactical decisions. The simplistic pronouncements other analyst firms make do not take into account the unique situation of each organization. This is another reason to present the several sides to each issue: to enable clients to determine the course of action that best fits their unique situation.

**For more information, contact Cutter Consortium at +1 781 648 8700 or [sales@cutter.com](mailto:sales@cutter.com).**

# Web 2.0

## ENTERPRISE ARCHITECTURE ADVISORY SERVICE

Executive Report, Vol. 9, No. 7

---

by Tom Welsh, Senior Consultant, Cutter Consortium

In the time-honored tradition of earlier buzzwords such as “dot-com,” “Web services,” and “business process management,” the media and the blogosphere are currently humming with discussion of a phenomenon called “Web 2.0.” The implication of this name is that the original Web, “Web 1.0,” has somehow become worn out or obsolete, and that it is fast being replaced with a newer, better model.

This *Executive Report* examines the evidence for and against this proposition and digs a little way into the tangled mass of technology that has become associated with Web 2.0.

### WHAT IS “WEB 2.0”?

The “2.0” meme is in fashion (along with the often-misused

“meme” meme): we also have “SOA 2.0,” “Library 2.0,” “Identity 2.0,” “Democracy 2.0,” and “Marketing 2.0,” and there is even a magazine called *Business 2.0*. Esther Dyson may have started the fad with her 1997 book *Release 2.0* (about which, the less said the better). It’s like the notoriously silly formula, “X is the new Y.”

There is some confusion about the relationship between SOA and Web 2.0, which has been further compounded by Oracle’s recent coinage of the term “SOA 2.0.” Fortunately for us, this has nothing at all to do with Web 2.0, since Oracle defines SOA 2.0 as a combination of SOA and event-driven architecture.

While the definition of Web 2.0 is not easily pinned down, Wikipedia lists the following as

general characteristics of Web 2.0, in the sense that they are often ascribed to Web 2.0 sites (such as MySpace.com):

- The site should not act as a “walled garden” — it should be easy to get data in and out of the system.
- Users usually own their data on the site and can modify it at their convenience.
- Mainly web-based — most successful Web 2.0 applications can be used almost entirely through a web browser: this is commonly referred to by the phrase “network as platform.”
- Data returns should be dynamic, not static, changing depending on variables associated with the user’s query (e.g. keywords, location).



del.icio.us (more on these sites below).

One straightforward interpretation is that “Web 2.0” is a grab-bag term for any and all techniques that are not deemed part of the original “Web 1.0.” If accepted, this view more or less rules out any attempt to find a single unifying theme behind the name, because the Web has been extended in so many ways for various reasons.

According to a related idea, the essence of Web 1.0 lay in creator of the World Wide Web Tim Berners-Lee’s original mission to create a virtual, hypertext-linked library. The task given to him at CERN was to find a better way for researchers to refer to one another’s papers, and it must be admitted that the Web, as Berners-Lee defined it, met that goal very thoroughly. In this view, Web 1.0 was a network of essentially static HTML documents, consisting mainly of text, which were rarely if ever updated. In contrast, Web 2.0 sites are highly dynamic, graphical, and interactive. The use of dynamic HTML (DHTML) to serve content from databases has been labeled “Web 1.5.”

It must be noted at this point, however, that Berners-Lee’s original vision of the Web went a good deal further, in some respects, than even Web 2.0 (as it has manifested so far). For instance, the first Web browser that he wrote doubled as an

editor, like W3C’s present-day Amaya, and allowed users to create or change site content as well as read it.

Over a year ago, *The Economist* pointed out the dangers of a possible “Bubble 2.0.”<sup>4</sup> Other prominent critics of the Web 2.0 concept include Nicholas Carr and John Dvorak. In a blog entry entitled “The amorality of Web 2.0,” Carr associates it with those to whom the Web has always been “a vessel of quasi-religious longing” and writes scathingly about “the cult of the amateur.”<sup>5</sup> His main point, that the Web cannot be said to be either good or bad in itself, is of course quite obvious. However, most of the blog entry and Carr’s followup responses are subjective, and his argument is larded with value judgments about the content of Wikipedia and other Web 2.0 sites.

For his part, Dvorak grumbles that “little has changed. Bad ideas of the past have been renamed and spiffed up. We’re watching a classic example of ‘old wine in new bottles’: Changing the label doesn’t make the wine any better, but it does get us to buy more wine.”<sup>6</sup>

When online magazine *The Register* asked its readers for feedback about the Web 2.0 concept, they complied with a will. As Andrew Orlovski wrote, “[W]hat we didn’t expect was the level of vitriol you threw at this latest nebulous piece of metaphorical

marketingese.”<sup>7</sup> While the cited article features as powerful and extensive a collection of negative opinions on Web 2.0 as you are likely to find, some of the language employed is not for the faint-hearted. Suffice it to quote the very first piece of feedback: “Web 2.0 is made of ... 600 million unwanted opinions in realtime.”

### WEB 2.0 EXAMPLE SITES

Among the Web sites often cited as typical of Web 2.0 are Flickr, digg, del.icio.us, MySpace, Netvibes, Techmeme, TechCrunch, Writely, YouTube, and Pageflakes. Dion Hinchcliffe gives a good introductory summary in his article “The Best Web 2.0 Software of 2005.”<sup>8</sup> But it is not immediately clear exactly what common factors link these sites, other than the immanent perception of “coolness.” There are so many of them springing up that the ironic acronym YASN (Yet Another Social Network) has been coined to describe them.

MySpace’s 80 million registered users are said to make it the world’s third most popular Web site. They have also opened the door to a whole new dimension of trouble. A Texas woman is currently suing MySpace for US \$30 million, alleging that her 14-year-old daughter was attacked by a man whom she met through its pages.<sup>9</sup> This kind of legal repercussion seems inevitable once millions of ordinary people start using the Web as a social

vehicle, bringing their hopes, fears, and other emotions into play — and sometimes bringing them into contact with criminals.

Another type of dispute is illustrated by the lawsuit recently filed by Los Angeles News Service against YouTube, in which the TV station complains that YouTube breached its copyright by posting copyrighted video footage.<sup>10</sup>

Critics may maintain that the old Internet and its predecessor ARPAnet were “ivory towers,” artificially shielded from the pulse of everyday human life. But their users enjoyed the luxury of engaging in relatively rigorous, factual discussions in an environment of enlightened tolerance. Apart from the occasional “flaming session,” objective dialog was the ideal of most participants. And even the most ardent flammers rarely held any personal grudges against those who disagreed with them. Much the same was also true of the Web in its first few years, when it adhered most closely to the “library” model.

Even a relatively old, well-established site like Yahoo! now encourages users to treat it as a dynamic portal. Pages, blogs, and the like can easily be added to My Yahoo! and updated on schedules defined by the user. Yahoo! has also moved in the direction of Web 2.0 by enhancing its page customization options under the slogans “Jazz things up a bit”

and “Let your page reflect your personality.”

One of the most fashionable Web 2.0 applications is Flock, a new browser based on Mozilla Firefox. As *BusinessWeek* put it:

Unlike plain-vanilla browsers such as Microsoft’s Internet Explorer, Flock’s browser is built specifically for a new, emerging generation of Web users, one that isn’t satisfied passively browsing media online.

Flock hopes to turn the browser into a dashboard for collaborating, blogging, sharing photos, reveling in a raft of other group activities that have recently caught fire online.<sup>11</sup>

Here is how Flock’s creators summarize its value proposition, as distinct from conventional browsers:

For this release, and for at least the next year or so, we are primarily focused on supporting the social dimension of the web, and on bringing information closer to the user. Right now this includes a heavy emphasis on the photo experience (from upload through discovery to notification), RSS support, dramatic improvements to the search box (including a new take on favorites), and blogging.

In each of these areas, our goal is to offer an integrated experience that’s easy to use and that leverages modern web standards. We are trying to build a web browser for

the tens of millions of users for whom social engagement (sharing pictures, talking to existing friends and discovering new ones) is what they enjoy doing online. That includes people on MySpace, LiveJournal, FaceBook and YouTube, for example.<sup>12</sup>

Building on the robust, efficient foundation of Firefox, Flock developers have added features like the ability to upload photos to MySpace, Flickr, or Photobucket by dragging and dropping; a built-in RSS (Really Simple Syndication; more on the other names for this below) aggregator for reading and managing blogs; as well as direct support for shared bookmarks (e.g., del.icio.us). Most of these features can be added to Firefox itself, through its burgeoning library of extensions, but Flock is aimed at people who just want to download a browser and start using it.

## THE O’REILLY CONNECTION

It immediately becomes apparent that there is no divorcing Web 2.0 from Tim O’Reilly, founder and CEO of O’Reilly Media, Inc. He helped to publicize the term and the ideas with which it has become identified, and his company’s series of annual conferences has driven the publicity and the discussion to fever pitch.

### *How the Name Was Coined*

The term “Web 2.0” was identified as a potentially useful brand by Dale Dougherty at O’Reilly Media way back in 2003 — practically in

the 20th century. It may have been floating around for a while before that, but no one had recognized its potential as a money-making vehicle. O'Reilly Media and its partner CMP Media organized popular *Web 2.0* conferences in 2004 and 2005, and the 2006 event is scheduled for November.

### "Web 2.0" as a Service Mark

Meanwhile, CMP Media has — perhaps unwisely — applied for a US service mark (a type of trademark) on the term. The media scented a story and hastened to write it up — see, for example, *Newsweek's* 3 April cover story.<sup>13</sup> While O'Reilly himself was away on vacation, O'Reilly Media officials sent a cease-and-desist letter to the organizers of what they saw as a rival conference, *it@cork*. The latter, a small, Irish nonprofit group, had the temerity to name its half-day conference "Web 2.0," a decision that O'Reilly Media's partner CMP Media protested as a "flagrant violation" of its rights.

The Web erupted in protest, and O'Reilly Media apologized for its tactlessness, offering *it@cork* permission to use the Web 2.0 name for this year's event — which it did, while rejecting the implication that it needed permission. As one commentator suggested, *it@cork* could have used the name "Web 2.1" instead and avoided any argument.

However this legal storm in a teacup is eventually resolved, the damage has already been done. O'Reilly, his company, and the

Web 2.0 meme have all come in for heavy and sustained criticism in the aftermath. Many serious developers noticed the ambiguity between "Web 2.0" as a technology revolution, and "Web 2.0" as a proprietary service mark. Their conclusion was the obvious one — that it could reasonably be one or the other, but not both.

### O'Reilly's Definition of "Web 2.0"

According to O'Reilly:

There's a set of "Web 2.0 design patterns" — architecting systems so that they get smarter the more people use them, monetizing the long tail via a combination of customer self-service and algorithmic management, lightweight business models made possible by cooperating internet services and

data syndication, data as the "Intel inside", and so on."<sup>14</sup>

O'Reilly notes that the key idea behind Web 2.0 is that the Web itself becomes the central computing platform — he considered "Web operating system" as an alternative term.

In a very readable paper entitled "What Is Web 2.0,"<sup>15</sup> O'Reilly introduced his famous Web 2.0 meme map (see Figure 2).

This graphic can be interpreted in various ways. Most straightforwardly, it is just a random gathering of ideas that are associated — more or less loosely — with Web 2.0. In addition, its very format somehow symbolizes the ad hoc, "make it up as we go" spirit of the Web 2.0 movement.

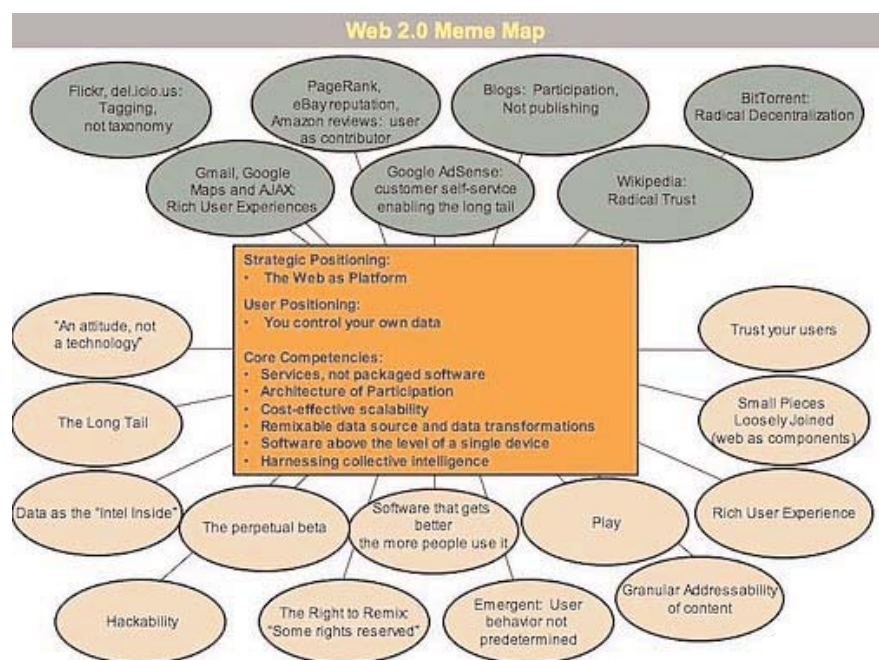


Figure 2 — Tim O'Reilly's Web 2.0 meme map. (Source: O'Reilly.<sup>15</sup>)

To be fair, it must be acknowledged that the meme map was the outcome of a brainstorming session; but it is interesting that O'Reilly saw no need to tidy it up for publication. (Skeptics might wonder whether that would even be possible.)

One thing that pops right out of Figure 2 is that Web 2.0 is not a technology movement. Hardly any of the ideas that it documents have any bearing on how the Web actually works, let alone the specifications that control its operation. The salient common factor, in fact, is that they are mostly “human factors,” concerned with users, how they participate in the Web, and what they get from it in turn.

“What Is Web 2.0” also includes Table 1, which is meant to illustrate the difference between Web

1.0 and Web 2.0. (O'Reilly recalls that when it was first captured at a brainstorming session, “the list went on and on,” implying that Table 1 is just the tip of the iceberg.)

To give even a brief explanation of all the 28 items listed in Table 1 would require a whole series of reports. Some of these items will be familiar, while (except to Web 2.0 experts) others will not. But one thing leaps to the eye: the table is very eclectic. To take one of the most obvious examples, is it really true that Web 2.0 is characterized by cost per click, while page views went out with Web 1.0? That would come as a surprise and a disappointment to millions of Web users, whose reluctance to pay for information is almost religious. Besides, O'Reilly's own paper consists

of plain, old-fashioned page views. If it were provided on a cost-per-click basis, far fewer people would have read it. All in all, the list looks like a hodgepodge of heterogeneous improvements (or, at least, additions — some of them might be seen as steps backward). They have no obvious common factor other than a general feeling of “Gee, what a long way we have come since 1995!”

It is unclear how much Web 2.0 has in common with one of O'Reilly's earlier epiphanies, the concept of “Infoware.” This is expounded in his 1999 piece “Hardware, Software, and Infoware,” which forms one chapter of the successful book *Open Sources: Voices from the Open Source Revolution*.<sup>16</sup> Noting that some friends had bought a PC purely to buy books and music from Amazon.com, O'Reilly points out that this is the classic definition of a “killer application.” By that definition, Amazon.com must be an application, even though it looks superficially like a mass of data. “Traditional software embeds small amounts of information in a lot of software,” explains O'Reilly. “[I]nfware embeds small amounts of software in a lot of information.”

### THE IMPACT OF WEB 2.0

As early as 1997, Berners-Lee pointed out one of the strongest influences that the Web was going to have on the world of commerce. He noted that, “Marketing on the Web is going to be a lot

**Table 1 — Web 1.0 Versus Web 2.0 (Source: O'Reilly<sup>15</sup>)**

Web 1.0	Web 2.0
DoubleClick	→ Google AdSense
Ofoto	→ Flickr
Akamai	→ BitTorrent
mp3.com	→ Napster
Britannica Online	→ Wikipedia
personal websites	→ blogging
evite	→ upcoming.org and EVDB
domain name speculation	→ search engine optimization
page views	→ cost per click
screen scraping	→ web services
publishing	→ participation
content management systems	→ wikis
directories (taxonomy)	→ tagging (“folksonomy”)
stickiness	→ syndication

more humane than marketing in traditional mass media because it's possible to treat people individually."

Nine years on, it looks as if "the Long Tail" — one of the main Web 2.0 ideas — may be one of the key marketing themes of 2006. In its capitalized form, the term is attributed to Chris Anderson, who has been writing and speaking about its implications since 2004. His book on the subject<sup>17</sup> is currently definitive. When printed in lower case, "long tail" is a standard term used by statisticians.

Many of us recall, with mixed feelings, earlier world-changing ideas such as "mass customization," which have succeeded to a much lesser extent — if at all — than was predicted at their introduction. However, it may turn out that the Long Tail is a less ambitious and more attainable goal than mass customization.

The long tail is a common enough phenomenon: it turns up whenever a statistical distribution has a roughly exponential shape, with a few frequent events and a lot of infrequent ones (see Figure 3). It is of particular interest when the distribution in question represents the sales of some kind of goods or services. Figure 3 actually illustrates the distribution of popularity (in terms of page hits) for all Wikipedia pages. Amusingly enough, the page about the Long Tail is itself part of Wikipedia's long tail of less popular pages (shown in the lighter color).

The point is that companies like Amazon do more business in — and make more money from — low-popularity items than they do in high-popularity ones. Contrast this with a conventional brick-and-mortar bookshop, whose staff go to great pains selecting a few best sellers and stacking them high to catch the eye of the public. The difference, of course, is that the bookstore has very limited space and can only afford to stock a relatively small number of books.

Amazon, on the other hand, has built its whole business model around stocking as many books (and other products) as possible, thus allowing it to reap the benefits of the Long Tail. As one wag put it, "Niches are riches." Even Amazon could take its model one long step further. Quite often, it lists books that are shown as "out of print." If Amazon could record queries about these books and feed back demand to the publishers, they would know when to issue a reprint — and even how many copies to produce.

The Long Tail has numerous potential implications, many of

which are thoroughly discussed by Anderson. Its effect on popular culture could be quite profound and far-reaching; instead of all having to compromise on a few "least unpopular" books, pieces of music, movies, TV shows, sports events, and so on, everyone can select what they personally enjoy most.

Computer dating is another good example of the Long Tail in action. Without it, anyone wishing to meet suitable romantic partners has to participate in a collective social roundabout: finding out where the meeting places are, getting there, trying to look eligible — and always within the confines of a single village, town, or city. When well implemented, computer dating skips over all of that and allows people to select one another on the basis of desirable (and undesirable) characteristics, wherever they happen to live — and even if they never socialize at all.

The Long Tail promises to remove a harmful blockage from the distribution system that links producers to consumers, allowing far



Figure 3 — The Long Tail. (Source: Wikipedia.<sup>18</sup>)

more people to get paid for doing what they choose, as well as giving everyone a chance to buy what they really want. In the long run, producers and consumers will become far more tightly (and responsively) coupled.

Wikipedia has a good, detailed study of the Long Tail, including a bibliography.<sup>18</sup>

### WEB 1.0

For purposes of contrast, a few brief remarks about the original Web — Web 1.0 — may be useful at this point.

#### *How the Web Was Won*

Tim (now Sir Tim) Berners-Lee invented the World Wide Web in late 1990 while working at CERN, the European Particle Physics Laboratory in Geneva, Switzerland. He wrote the first Web client (a browser-editor running under NeXTStep) and the first Web server along with most of the communications software, defining URLs, HTTP, and HTML. Most of this apparatus is still with us. URLs are recognizably the same, and HTTP has been revised slightly.

The Web was one of those unexpected, and unusual, breakthroughs that often lead to a period of continuous rapid progress. The preconditions for the Web had been building up for more than a decade: we can identify at least the following:

- The ubiquity of TCP/IP and the associated Internet application protocols (FTP, NNTP, SMTP, and so on)
- The emergence of a worldwide community accustomed to communicating across the Internet
- The accumulation of large amounts of valuable information in various Internet archives
- The gradual development of production-quality, user-friendly hypertext software
- More than 50 million personal computers and Unix workstations sitting on people's desks, all capable of displaying color graphics, and an increasing number capable of audio and full motion video as well
- The availability of comprehensive, affordable solutions for connecting standalone PCs running Windows to the Internet

Berners-Lee had the flash of insight that led to the first Web server and browser. Once the idea caught on, it spread like wildfire because it was so useful and cost-effective. The Web meets all the criteria for a successful open systems concept: it adds considerable end-user value, in return for little expense, and the same technology is available from a large pool of competing suppliers. This is because the Web is built on a foundation of Internet technology, which is about as open as

anything in this imperfect world. The Web did for the Internet what Windows had previously done for DOS, transforming a humdrum and rather difficult user interface into one that is a genuine pleasure to interact with.

Berners-Lee is now director of the World Wide Web Consortium (W3C), a group that is trying to shape policy for the Web. He has consistently emphasized the importance of keeping the Web open and has warned vendors not to make the mistake of differentiating their browsers to keep out the competition.

Under the circumstances, it is amazing that the original Web protocols have worn so well. After all, they were designed to permit a limited number of academics and researchers to compare notes and read each other's papers. Using them in today's conditions is rather like trying to provide a commuter train service with half a dozen bicycles. Yet they continue to function, with only a little creaking and a few subdued oaths from developers who find it unduly difficult to deliver the systems their paymasters demand.

#### *The Semantic Web*

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.

That is how Berners-Lee summed up his next project, the Semantic Web, in an article published in the May 2001 issue of *Scientific American*.<sup>19</sup> Berners-Lee is well qualified to design a more powerful, intelligent, second-generation Web. In the article, he and his colleagues painted a picture so futuristic that it sounds like science fiction. They foretold such things as household “operating systems” that automatically hush entertainment systems when the phone is picked up and software agents that find providers of specific medical treatment and book appointments. These scenarios seem quite far-fetched, but the authors went on to show how they follow logically from a set of emerging software innovations.

Specifically, these are:

- Shared meaning (semantics), provided by XML and Resource Description Framework (RDF)
- Ontologies, or shared vocabularies, that ensure different applications are able to communicate effectively
- Software agents that collect information from the Web, process the information, and exchange the results with other programs
- Service discovery — a kind of Yellow Pages for agents

Semantic Web research and development has been proceeding steadily, but unobtrusively, for about a decade. It is not a heavily

funded, high-profile project, relying instead on the efforts of an ad hoc global community of software experts. W3C provides coordination, overall direction, and the occasional touch of the rudder.

The possibilities of the Semantic Web, when it finally becomes a reality, are enormous. In a word, all devices that can be connected to the Internet will be able to read data from Web sites and make inferences based on it, using something that looks like (but is not) intelligence. Precisely because the stakes are so large, it is vital to get the underlying technology and specifications right, and that is what is taking so long.

## THE TECHNOLOGY OF WEB 2.0

From a developer’s point of view, Web 2.0 looks like the intersection of a rich, eclectic set of software techniques that have arisen over the last 10 years or so. This section addresses some of these techniques.

### **Broadband**

When broadband was first mooted as the standard connection method for millions of people, pundits reacted by trying to think of uses for all that extra bandwidth. Surely it would be many times more than sufficient for e-mail, Web browsing, and such everyday tasks. So they came up with downloading videos on demand, watching sports and listening to music in real time, and other “labors of Hercules.” Most of

these proposed applications, however, would have the effect of turning the Internet into a consumer channel and minimizing its interactivity, which many users find its most valuable quality.

Concern about how all that bandwidth would be used turned out, as usual, to be misplaced. In just the same way, 15 years ago experts wondered why anyone would ever need more than a 1MHz processor and 1MB of RAM on their desktop. Today, of course, the cheapest PCs available from Dell and other manufacturers run at over 1GHz and have 256MB or 512MB of RAM — yet few users complain that their response times are too fast!

Most Webmasters today make sure that their pages are kept to under 100KB; Microsoft’s home page is around 30KB, and IBM’s is less than 25KB. At broadband speeds, all of these should load in well under a second (not counting the delays due to latency). But with a 56 kbit/sec dialup link, even IBM’s home page would take at least five seconds, while Amazon could take 20 seconds or so to load. That is well outside the bounds of acceptable response time.

The fact that a Web site features pages approaching 100KB in size suggests that the Webmaster expects most visitors to have broadband links, with a bandwidth of at least 512 kbit/sec. If that can safely be assumed, then

quite a lot of attractive and interesting effects become possible — especially with efficient programming techniques.

### **“Network as Platform”**

Many Web 2.0 characteristics — notably fast broadband access, dynamic client software, and interactivity — contribute to realizing the long-sought vision of “the network as platform.” In this paradigm, which dates back to (and even perhaps beyond) Sun Microsystems’ slogan “The Network is the Computer,” individual computer platforms lose their importance as both data and processing become freely available over the Internet.

“Software as a Service” (SaaS) is the commercial face of this relatively novel concept, which threatens to move the emphasis from rich-client platforms like Windows and Mac OS to platform-neutral distributed software. Pioneers like Salesforce.com and NetSuite have vigorously grabbed the opportunity of getting in on the ground floor of what they see as a gigantic new market.

The pre-Web Internet, and even Web 1.0, posed no great threats to rich-client platforms. Before the Web was introduced, most Internet applications were limited to command-line interfaces. The early Web was heavily document-oriented and did not lend itself to interactive applications other than its characteristic cycle of “read document; click hyperlink; read

document ....” Even with the help of Java, JavaScript, and ActiveX, Web interfaces remained uglier, less user-friendly, and above all less responsive than rich-client interfaces.

Since 2000, however, technical advances have allowed Web applications to close the usability gap on their rich-client rivals. Techniques like Ajax make it possible to enjoy the benefits of a thin client without sacrificing response time, GUI quality, or the range of features that can be supplied.

So we have seen the arrival of a new wave of Web applications that seek to replace traditional rich-client (most Windows) software. This was Netscape’s grand strategic vision almost 10 years ago, but the technology was not yet in place to support the ambitious plans of founders Jim Clark and Marc Andreessen. Besides, Microsoft quickly identified Netscape as a potential threat and lost no time in zeroing in on the impudent company’s Achilles’ heel — its precarious cash flow position. Today it is not a single, brash startup that is challenging Microsoft’s grip on the PC client market, it is the whole industry.

E-mail was the first mass-market client application to take to the Web. An early package called, simply, Webmail, is said to have been available as early as 1995. Gradually the idea caught on, and now millions of people around the world use Microsoft Hotmail,

Yahoo! Mail, Google Gmail, and many other Web e-mail services. The advantages are considerable:

- With no need for a local e-mail client (a Web browser is used instead), the service can be accessed from any computer anywhere.
- Saved e-mail, contacts, and profiles are stored on a professionally managed server, so they cannot be lost through a local hard disk crash or careless mistake.
- If the account name is suitably anonymous, there is no easy way to trace the owner through his or her e-mail.
- There is no need to worry about whether to send e-mail in text format or HTML; in the world of browsers, everything is HTML.
- Although the need for security is not eliminated, it is somewhat simplified.

The other well-known functions of client “productivity” suites soon began to follow e-mail onto the Web. Today it is possible to edit, do word processing, work with spreadsheets, build presentations, coordinate multiple calendars, and manage projects with nothing but a browser.

Even Microsoft has joined the movement, perhaps recollecting the old saying that it is better to risk cannibalizing your own business than to wait until your

competitors do so. Apart from Hotmail and one or two abortive research programs, the desktop giant's first major foray into "network as platform" territory was the announcement by Bill Gates himself, in May, of the forthcoming Windows Live and Office Live products.<sup>20</sup>

Perhaps the most interesting aspect of the announcement was how little the new products will impinge on Microsoft's existing portfolio. That is the advantage of adopting new technology promptly — with enough skill and a little luck, you may not have to do any cannibalizing at all!

Windows Live will be a set of Web-based personal services, such as e-mail, blogging, and instant messaging. It will be positioned as a more flexible and personalized alternative to MSN, Microsoft's older subscription service, as well as offering a wider range of features. In addition to e-mail, IM, blog feeds, and podcasts, Windows Live will let users add extraneous items like SharePoint servers and recently opened documents. There will also be a free set of security tools called Windows Live Safety Center.

Office Live is to be aimed mostly at small businesses; according to Microsoft, 28 million of the world's 42 million companies have fewer than 10 employees. Users will be able to set up their own Web sites

with custom domain names and multiple e-mail addresses free of charge (initially at least). As many as 22 collaborative tools will be offered within Office Live, including customer management software, spreadsheets, and other productivity applications.

Both Windows Live and Office Live are likely to carry advertising, although customers will be able to avoid it by paying fees or subscriptions. Microsoft sees online advertising as a relatively untapped source of revenue and expects the market to grow from US \$15 billion today to \$150 billion in 2015.

#### ***Pervasive User-Friendly Presentation***

Web 2.0 sites generally give, as Wikipedia puts it, "priority to ... visual design and aesthetics." While obviously important, this might not seem to be a technical matter. The point is that quite sophisticated technology is required before any serious pursuit of visual design and aesthetics is even worth undertaking. Netscape, for example, just did not have a rich enough palette of user interface features to present its users with anything more than a simple, fairly spartan visual environment.

Today's Web 2.0 sites, in contrast, go to great lengths to provide a pleasant "user experience" in the hopes of winning the commitment of people who have thousands of other sites to choose from.

#### ***Rich Internet Application Techniques***

The best things about the Web have always been its ubiquity, uniformity, and simplicity:

- **Ubiquity** — wherever you happen to be, whether at work, at home, traveling, or abroad, you can almost always get a Web connection.
- **Uniformity** — wherever you are, the Web always looks very much the same (subject to the vagaries of browsers and the speed or slowness of the local ISP).
- **Simplicity** — the Web sticks to an admirably simple model of operation. The client sends a request; the server sends back a page of HTML.

Hardly anyone has any objections to ubiquity and uniformity; but while simplicity is obviously desirable, it militates against some other desirable features. Take response time, for instance. Sometimes it would be nice if the Web browser could show a little more initiative and update or validate a field on its own, rather than sending a long request string back to the server, which then sends back another whole page of HTML.

DHTML certainly helps in this respect, by making it possible to send snippets of JavaScript or other active objects to the client; these can satisfy some types

of user requirements without downloading any more HTML pages. But DHTML has very definite limitations, and sometimes nothing less than a “rich” client will do. That, of course, is where Ajax comes in. Whereas DHTML with JavaScript is limited to sending down one or more script objects along with an HTML page, Ajax sets up a continuous background exchange of information between client and server.

The contrast between pure REST (Representational State Transfer) principles and techniques like Ajax — which, in REST terms, “cheat” by sending data out-of-band — is a typical engineering tradeoff. REST has the great advantage of being simple and transparent, but for these very reasons it sacrifices some efficiency. Ajax breaks the pure REST model, but often delivers the pragmatic benefit of faster response time.

A more immediate and practical drawback to Ajax and similar techniques is that they completely exclude users who have turned off JavaScript in their browsers. Unfortunately, in the present dubious state of Internet security, many security experts recommend exactly that step.

### **CSS**

Cascading Style Sheets (CSS) is the name of a W3C standard that is widely used, in conjunction with HTML or Extensible HTML (XHTML), to give a consistent look and feel to a set of pages or a

whole Web site. Its great strength is that the look and feel is wholly divorced from the creation or editing of content. Thus a user interface specialist can prepare a site’s CSS, which will then be rolled out across all that site’s pages. Meanwhile, the content providers just type in their material without worrying too much about presentation or layout.

### ***Semantically Valid XHTML Markup***

The great bulk of the Web is still written in HTML, but since 2000, XHTML has provided a far superior alternative. XHTML is a W3C Recommendation that does the same job as HTML, but in a much better way. It is actually XML, but an XML application that has the same vocabulary as HTML. This allows it to be validated and checked in many ways that HTML cannot be, as well as opening up all sorts of possibilities for semantic markup. In this latter respect, XHTML shares some of the good points of microformats.

### ***Microformats***

Microformats are used in many Web 2.0 sites; for example, Technorati tags employ them. The basic idea is very simple: microformats are HTML or XML markup that adds metadata tags to the “plain old data.” This enables search engines to pick out the specific kind of data that a particular user desires to locate, rather than just presenting a huge mass of hits that match the input data string. Dedicated “microformat

bots” can do a lot better than generic search engines like Google and AltaVista, as they have the advantage of “understanding” the appropriate microformats.

Microformats, then, are a tactical stopgap that fulfill much the same purpose as the Semantic Web’s RDF. Unlike RDF, they are not universal; but they do work and deliver useful results, here and now.

### ***Syndication and Aggregation Through Atom, RSS, and So On***

Syndication is the provision of Web feeds to “export” the latest updates to other sites or individual subscribers. It has much in common with the “push technology” that made such a stir in the 1990s but eventually became discredited. The main difference is that today’s Web syndication is largely divorced from the profit motive: most feeds are free of charge.

Many people feel that content syndication, using Atom, RSS, or any of the other protocols that have sprung up of late, is the most important single aspect of Web 2.0. Like so many other Web 2.0 techniques, syndication allows users to take advantage of Web content without being tied down to the traditional interactive browser interface. Thus, instead of having to visit every Web site of interest individually, a syndication user can just click open a specialized reader and browse a whole catalog of feeds.

RSS, the most popular Web feed format, has stood for “Really Simple Syndication,” “Rich Site Summary,” “RDF Site Summary,” and quite possibly other things as well. But people who use it know what it does, and that is what matters. RSS delivers information, such as blog entries, as XML data, which is called an RSS feed, Web feed, RSS stream, or RSS channel. As well as enabling syndication, Web feeds let users track and manage their favorite blogs and Web sites using a special tool called an aggregator. Atom is another popular Web feed format; many happy hours have been spent arguing about its relative strengths and weaknesses compared to RSS.

#### **Weblogs, Wikis, Blikis, and More**

One of the most commonly cited aspects of Web 2.0 is what might be called “interactive mass publishing.” Unfortunately there is little agreement as to exactly what this means. Its anthropological and social implications, however, are somewhat clearer. Many global virtual communities are forming like Internet “town meetings.” Unlike physical meetings, the participants are usually self-selecting. In a way, this could be seen as yet another manifestation of the Long Tail.

The most distinctive feature of wikis is that any user is free (in principle) to add new material or indeed to change existing content. This anarchic openness has

sometimes had to be restrained or limited — as in the highly publicized case of Wikipedia, for instance — when someone was deemed to be abusing it.

Blogs have a different set of special characteristics, perhaps the most salient of which is the persistence of entries through time. A typical blog takes the form of a diary, with the present day at the top of the page and previous days following in line. There are usually separate navigation controls, including a graphical calendar for picking out previous dates of interest. Individual entries can often be clicked on to see them packaged as a separate HTML page, which can be e-mailed, printed, or filed.

Ideally, the need to do any of those things should be minimized if the blog author maintains a complete archive of all entries. After all, the cardinal principle of the Web is that each URL should point to one, and only one, resource; that resources should never change or disappear; and that URLs should never be reallocated. Like most principles, this admits of some practical exceptions — if a company goes out of business, for instance, there will eventually be no money to pay for its pages to be served. But the propensity of blogs to maintain their content over time is a welcome contrast to the kaleidoscopic changes some Web sites undergo every time there is a reorganization or a new Webmaster.

Although wikis, blogs, and their many hybrids (such as “blikis”) are Web pages, they nevertheless form an identifiable subset of the greater Web. This means that it makes sense to program search engines to index only blogs, wikis, or certain well-defined categories of either. Indexing and cross-referencing are just as important, when it comes to the exploitation of any really big body of knowledge, as the information itself. Without the ability to find the right knowledge, the blogosphere, for instance, would be nothing more than a huge “data haystack” that might, or might not, have a needle of useful information somewhere inside it.

#### **Mashups**

Ever since Web services were invented, people have been speculating about the benefits of being able to cascade them. Once there was a critical number and variety of Web services, the argument ran, developers would find it cost-effective to build applications that collected data from several Web services, mixed them together creatively, and delivered the results within a tasteful user interface.

Well, those applications have arrived, and they are called mashups. Perhaps the most popular so far is Google Maps, one of the many Google services that have been widely overlooked because the name “Google” is overwhelmingly identified with

“Web search.” Accessible through the “more” link on the main Google page, Google Maps superimposes maps on aerial photographs of the same piece of terrain. But that is not all, by any means. You can also search for a street, post code, or ZIP code, businesses in chosen categories, restaurants, or hotels. Google Maps will even plan your trip, displaying directions between any two places, both on a map and as a written itinerary. Microsoft and Yahoo! offer similar functionality, although Yahoo!’s is limited to North America.

So far, so impressive, but that is only the beginning. Microsoft, always alert to commercial opportunities, has various upgrades available — including a GPS interface — for paying customers. On the other hand, Microsoft Research’s popular MapCruncher data visualization tool is free. The virtual map application Google Earth is free of charge but tempts users to upgrade to Google Earth Plus (\$20) and Google Earth Pro (\$400).

#### **REST or XML Web Service APIs**

Web services are finally beginning to be used on a mass scale, but not in the way that most prognosticators expected. Instead of heavy-duty corporate applications tooled up with the full panoply of WS-\* features, the great majority of today’s Web service invocations are handled by simple, lightweight interfaces. Leading players

like Amazon, eBay, Google, and Yahoo! have opened up many of their customer-facing (and partner-facing) operations through Web services.

When the Web services concept was first worked out, at Microsoft in 1999 and subsequently with the participation of IBM and others, it was seen above all as a way for programs to communicate across the Internet. The Web was decidedly, and unapologetically, interactive: it had an HTTP server at one end and a human being, sitting in front of a Web browser, at the other end.

With Web services, the same infrastructure (the Internet protocols, HTTP, and HTML) could be repurposed to let unattended applications exchange data at full machine speed. Moreover, by substituting software for a human reader, it became possible to “cascade” Web services. An application could read data from several Web services, perform some kind of computation, and offer the result as a new “added value” service.

It was soon realized, however, that these exciting new possibilities came at a price. One of the most surprising things about the interactive Web is that it works at all, considering its lack of rigor, which sometimes extends to sheer sloppiness. Most Web browsers are designed to accept, interpret, or work around incorrect HTML.

This seems to conform to Postel’s Prescription: “Be liberal in what you accept, and conservative in what you send.”<sup>21</sup> This famous principle was laid down by Jon Postel, one of the Internet’s original architects, but many experts feel that it came to be taken out of context. In his book *The Art of UNIX Programming*, Eric Raymond quotes Doug McIlroy’s warning on the subject:

The original HTML documents recommended “be generous in what you accept,” and it has bedeviled us ever since because each browser accepts a different superset of the specifications. It is the *specifications* that should be generous, not their interpretation.<sup>22</sup>

Whatever the rights and wrongs of the argument about correct HTML (or XHTML, as it really should be nowadays), it is undeniable that the interactive Web depends critically upon the tolerance of users. If unexpected results appear, if the information displayed for a page is incomplete or corrupted in some way, or even if the page is not there at all, in which case the famous Error 404 is displayed, a human user can be relied upon to react philosophically, make the most of a bad job, and work around the deficiency.

Web services, on the other hand, cannot hope for such latitude. If a single byte is missing or out of place, the calling application will fail. Worse still, if the data appears to be in the right format but is not

the expected data, the calling application may carry on “fat, dumb, and happy” and deliver incorrect output. Precisely this kind of metadata error brought about the loss of NASA’s \$125 million Mars Climate Orbiter spacecraft in 1998. Lockheed Martin’s development team used imperial measurement units (pounds) in its software, while NASA’s Jet Propulsion Laboratory had adopted metric units (newtons). As a result, wrong data was fed into a critical computation, causing the spacecraft to miss its orbit around Mars.

Whatever the reason, the fact remains that Web services have rarely been used to create powerful corporate and interorganizational distributed systems. Instead, the market has drawn them back toward the interactive arena, thus making them one of the principal (though often unseen) ingredients of the Web 2.0 mix.

Amazon’s designers considered two distinct types of Web service:

**1. SOAP** — conventional “corporate style” Web services, using SOAP as the application protocol, and WSDL to describe the services in detail. This approach is likely to be more efficient if the services are called frequently and may even be more convenient for experienced Java and .NET programmers. Its main drawback is that there is quite a lot to be learned before making even

the simplest request, so non-programmers might be put off.

**2. REST or XML over HTTP** — REST is a set of design principles for the Web, documented by Roy Thomas Fielding in his thesis “Architectural Styles and the Design of Network-based Software Architectures.”<sup>23</sup> According to Fielding, REST is “intended to evoke an image of how a well-designed Web application behaves: a network of [Web] pages (a virtual state-machine), where the user progresses through an application by selecting links (state transitions), resulting in the next page (representing the next state of the application) being transferred to the user and rendered for their use.” XML over HTTP is a less formal style that dispenses with many of the powerful features of SOAP and WSDL, merely exchanging XML data over HTTP as application protocol.

Since experts were divided as to which approach was better, Amazon decided to offer both and let the developers choose. Interestingly enough, at least 80% of the calls that Amazon processes are of the REST type.

Jeff Barr, Amazon’s head of Web services evangelism and developer relations, explained this observed preference as follows:

First, I think, it’s an easier and more obvious model for developers. It’s very easy to

literally show them a REST query. You can just give them a URL, and say put this in your browser. They put it in the browser. They look at the XML result and it’s an easy jump from seeing that data — you see the returned XML data — for them to then figure, “Okay, I can process that. I can either be very formal and I can use an XML parser, or I can be a little bit more informal and could use like PERL regular expressions or just simple string processing even to process the returns.” And it’s just one less piece of technology that they need to know. Like if you are going to go with SOAP — and there is a lot of good reasons to go with SOAP — you need to, like, learn yet another piece.<sup>24</sup>

Interestingly enough, Ajax and similar techniques are quite “unRESTful” in that they break with REST’s requirement that all operations should be done through HTTP requests such as GET, PUT, POST, and DELETE.

## A TECHNICAL OR A SOCIAL PHENOMENON?

The chaotic, experimental nature of Web 2.0 implies a state of continuous flux that has been described as “permanent beta.” Like many of its aspects, that sounds attractive to the adventurous but implies some awkward practical difficulties. Surely, today’s software needs to become more reliable, secure, and efficient — whereas “permanent beta” implies quite the opposite.

There are at least three different conversations going on about Web 2.0 simultaneously — and, often enough, in the same discussions. Ostensibly, it is all about new technology. But nobody who really matters cares about technology for its own sake. Some want to know how to make lots of money, while others are looking for better ways of using the Web to further their business. In other words, Web 2.0 is a big confused colloquium in which users, vendors, pundits, and developers are all talking about slightly different things.

Sometimes the conversation gets a bit out of hand. For example, I do not know whether venture capitalist Peter Rip was joking when he wrote in his blog that, “Web 2.0 is a lighter weight version of SOA. RSS/REST is the new EAI.”<sup>25</sup> Inasmuch as it has any technical meaning, Web 2.0 is poles apart from SOA. RSS and REST are entirely different kinds of things, and neither of them has anything to do with EAI. Rip winds up that particular blog entry with a purple passage typical of much Web 2.0 advocacy: “To paraphrase Marx and Engels, there is a spectre haunting the Enterprise. It is the spectre of Web 2.0.”

The more you look into the whole Web 2.0 affair, the clearer it becomes that this is overwhelmingly a social, rather than technical, phenomenon. There is a huge amount of comment, most of it highly emotional and speculative, resulting in a very low signal-to-

noise ratio. That makes it very hard to extract a precise technical definition. But most contributors to the debate seem to agree on one proposition: “Web 2.0 is an attitude, not a technology.”<sup>26</sup> Reactions to it are strongly polarized. Down-to-earth people who like to know where they stand, and who think in concrete terms, are likely to reject it as unhelpful (and annoying) hype. Those who like the idea see themselves as open-minded, creative, and visionary, but risk being criticized for woolly thinking.

One reason for all the excitement may be that the Web has made it possible for anyone at all to become rich and successful, almost overnight. You do not need a PhD in engineering, deep pockets, or even exceptional creativity. All it takes is a good idea and plenty of self-confidence, and you too could be a millionaire by this time next year. So it is hardly surprising that tens of thousands of people should be chattering away online, hoping for recognition, wealth and fame, or even just a plum job. In the words of one blog responder, “Web 2.0 is all about new opportunities”; and large numbers of people feel that opportunity might very well knock on their door. In their own minds, they have become players rather than onlookers.

Some aficionados paint glowing pictures of the surge in productivity that would result if people outside the IT community, like

doctors, lawyers, and office workers, could harness the power of Web 2.0. But even the most enthusiastic fans have to admit that they spend hours, almost every day, just keeping up with the ever-changing kaleidoscope of tools, services, and communities. The only way it could all be “harnessed” by outsiders would be if there was a single interface, a single set of tools, and a single way of working. But that would be antithetical to the roiling, evolving chaos that is Web 2.0. By definition, it is a way of life rather than a static set of tools: frontier territory, not settled farmland.

If there is any consensus about what Web 2.0 actually is, it probably involves collaboration between arbitrary numbers of people in different places. It is easy to conjure up utopian visions along these lines, but at the same time one has to ask, exactly how is this better than what Netscape was talking about back in 1996? In which case, might Web 2.0 not be better named “dot-com 2.0”?

#### **WHY “WEB 2.0” DOES NOT SUPERSEDE THE WEB**

In an industry that is notoriously obscure to outsiders, it is risky to introduce a grandiose new term that seems to promise something quite illusory. Yet that is exactly what Web 2.0 is. To the tens of millions of people who know very little about the Web, the name is bound to suggest a radical change to something entirely new — a new major release of the Web.

That is a thoroughly misleading implication. Since the standards that underlie the Web are controlled by W3C, no one else is entitled to talk about “Web 2.0”; and no one at W3C does. Web 2.0 certainly has nothing to do with the Semantic Web, the next stage of Berners-Lee’s vision, which is being driven forward through a number of official W3C standards.

For anyone who knows about Berners-Lee and how he invented the Web, the excitement about Web 2.0 means that people are finally beginning to glimpse the potential that he saw 16 years ago. There is nothing new in an architectural sense, although a lot of novel implementation techniques have been added to the mix — things like XML, JavaScript, and syndication protocols.

Hal Stern’s definition of Web 2.0 is the shortest of all: pointing out that all O’Reilly’s Web 2.0 examples involve users writing data to the network, he dubs it “the read-write Web.” This is undoubtedly a valuable insight, but it merely underlines how little is really new, because Berners-Lee meant the Web to be read-write all along.

Tim Bray, director of Web Technologies at Sun and one of the architects of XML, makes a similar point. “Web 2.0 ... [is] really only used seriously by VCs and marketers and prognosticators ... not by the actual hands-on developers who build whatever it is we’re talking about; and for that, I still

prefer *read-write* Web.”<sup>27</sup> (Emphasis added.)

Some farsighted researchers saw the social potential of the Internet long before Berners-Lee. As early as 1960, for instance, J.C.R. Licklider was writing about how networked computers could boost people’s knowledge by letting them exchange ideas. Indeed, we are still far from attaining the human-computer partnership that he described in his paper “Man-Machine Symbiosis” nearly half a century ago:

The hope is that in not too many years, human brains and computing machines will be coupled ... tightly, and that the resulting partnership will think as no human brain has ever thought and process data in a way not approached by the information-handling machines we know today.<sup>28</sup>

One possibly important trend is the expansion of the Web user population to include a large majority of consumers, in the sense of “people who are not interested in IT for its own sake.” It was always inevitable — and entirely predictable — that this would happen at some point, but some of the early Web evangelists seem to have imagined that the Web would somehow transform these new users into computer enthusiasts.

That has not happened. The tens of millions — and soon, hundreds of millions — who are piling onto

the Web do not, broadly speaking, care about how it works. What they want is to communicate with one another, buy things, get better information sooner about things to buy, see, and do; enjoy games, music, and movies; and generally to go on living their lives as before. Instead of the Web changing them, they are beginning to change the Web — to mold it in their own image.

That is why Web 2.0 strikes so many people as a radically new phenomenon, although technically speaking, it has not yet reached as far as Berners-Lee foresaw in 1990. The main difference is not one of technology, but of the purposes for which technology is used. Web 2.0, then, can be summed up as the consumerization of the Web.

Incidentally, this trend is not unconnected with the current lively debate over “network neutrality.” The remarkable stability of the Internet, which has lasted since long before it began to be widely popular in the early 1990s, rests largely on the predominance of TCP traffic. This, in turn, had a lot to do with the fact that the Internet was mostly being used for purposes that could be lumped together as “research” and, subsequently, “business.”

In some ways, Web 2.0 is closer to a set of requirements than a piece of technology. So far it has nothing equivalent to the Semantic Web, for instance — although such a

capability would be immensely useful. One of the classic Web 2.0 activities is gathering together data from heterogeneous sources and mashing them up together to produce something distinctively new. But this trend has been sharply limited, to date, by the lack of any way to reconcile different semantic models.

So we would not be too far wrong in describing Web 2.0 as a movement that takes the new technology coming out of centers like W3C and the leading vendors and finds exciting new uses for it — uses that appeal to the millions of “nontechnical” people out there, as well as the much smaller core of Web enthusiasts.

As *The Register*'s Orlowski points out, some of Web 2.0's attributes clash rather badly with the fundamental purpose of the Web — to convey information efficiently. Deploring AOL's redesign of its Netscape.com portal, he calls it:

[A] faddish redesign ... [that] appears to have achieved the worst of all worlds.

A core of users “vote” for their favorite headlines, which are wrapped in the pastel shaded rounded rectangles that's the “Web 2.0” style. Many users are furious and it's not hard to see why the exercise has attracted comparisons to Coca Cola's catastrophic New Coke launch.

When something isn't broken, you don't need to fix it, and the old Netscape.com

was actually a model of effective web design: it delivered lots of information very concisely. You can fit a dozen headlines, or teasers, in one of the new site's pastel-coloured lozenges.<sup>29</sup>

Some people seem to find it irresistible to compare and contrast Web 2.0 and SOA — as we have already seen, Oracle has seen fit to coin the term “SOA 2.0.” Extreme curmudgeons tend to call down “a plague on both your houses,” while extreme enthusiasts believe that both are valid and useful; their problem then becomes how to fit the two trends together in a meaningful way. Sandwiched in between are two partisan camps: one that cheers for Web 2.0 and consigns SOA to the dustbin of history; and another that sees SOA as the next corporate software infrastructure, and Web 2.0 as the unsound, jerry-built offspring of pseudo-intellectuals and technology dabblers.

The dichotomy between Web 2.0 and the Semantic Web was summed up by a dialog that took place at a recent conference sponsored by the American Association for Artificial Intelligence (AAAI). Berners-Lee delivered a keynote speech focusing on artificial intelligence and the potential of the Semantic Web. His talk followed the usual lines: how Web data must be made more amenable to automatic processing without human intervention, thus rendering it far more reusable and valuable. He also

stressed the importance of using persistent URIs and RDF to identify information in order to guarantee compatibility with existing Web sites.

Responding to Berners-Lee's presentation, Google's Director of Search Peter Norvig questioned the fundamental practicality of the Semantic Web, and hence its desirability:

“What I get a lot is: ‘Why are you against the Semantic Web?’ I am not against the Semantic Web. But from Google's point of view, there are a few things you need to overcome, incompetence being the first,” Norvig said. Norvig clarified that it was not Berners-Lee or his group that he was referring to as incompetent, but the general user.

“We deal with millions of Web masters who can't configure a server, can't write HTML. It's hard for them to go to the next step. The second problem is competition. Some commercial providers say, ‘I'm the leader. Why should I standardize?’ The third problem is one of deception. We deal every day with people who try to rank higher in the results and then try to sell someone Viagra when that's not what they are looking for. With less human oversight with the Semantic Web, we are worried about it being easier to be deceptive,” Norvig said.<sup>30</sup>

Berners-Lee and Norvig are both highly intelligent, creative, and

extremely knowledgeable. The divergence in their points of view stems, no doubt, from their different backgrounds and predominant concerns. Berners-Lee heads up W3C and is generally regarded as the senior technical decision maker and evangelist of the Web. While he meets plenty of ordinary users, there is no doubt a filter effect that ensures he meets mostly the best and brightest — and then in a relatively theoretical context.

Norvig and his colleagues at Google, on the other hand, are in the business of making money from advertising. In the process, they have learned a lot about what makes the average Web user — and Web site owner — tick. In the words of *The Inquirer's* somewhat reductionist headline, “Google says web users are too stupid for semantics.”<sup>31</sup> Blunt, but perhaps it captures the essence of the controversy.

It is tempting to think of Berners-Lee as a beaver who has built a strong, well-engineered dam, creating an artificial lake within which many small creatures frolic. Most of them have no comprehension of the dam's construction — indeed, they may be blissfully unaware of its existence. To them, the pool in which they live is the whole universe. Naturally, they are preoccupied with everyday concerns: foraging for food, drinking the abundant water, building homes, seeking mates, and forever doing their best to rise within

their particular species' pecking order.

These everyday concerns are, to a great extent, exactly what Web 2.0 is about. It lets ordinary people — consumers — exchange their thoughts, display their photos, videos, and music, and generally helps to amplify their lifestyle. All this is well and good — very good, indeed — but it is a benefit, not a field of technology. The technology resides in the dam that permits this microcosm to exist. When he was engaged in putting it together, the beaver probably did not think in any very concrete terms about the advantages it would bring to the many species of inhabitant. He was driven, as beavers are, by a general conviction that a lake would be a very good thing, and that a lake could be brought about by building a dam.

This analogy, while perhaps amusing, is of course quite limited. The key proposition that it illustrates is that Berners-Lee saw very far ahead when he designed the Web architecture. But even he could not possibly foresee all the uses to which his invention might be put in the following years and decades. Now he hovers on the sidelines, occasionally pointing out perverse practices that should be avoided — such as URLs that come and go mysteriously, or insistence on legal agreements before linking to someone else's pages.

Meanwhile Berners-Lee continues looking for ways to improve the

Web and enable it to become even more valuable to individuals and society. He does this mostly by identifying limitations in the existing protocols and looking for the simplest and most transparent set of additions that would remove those limitations. This is a quintessentially architectural occupation — indeed, it would serve as a definitive example of what “architecture” means.

Berners-Lee and his colleagues are engaged in a highly abstract and tremendously important activity: improving and extending the Web without sacrificing any of its existing benefits. This is an immensely delicate business, which brings to mind the words of Lao Tzu: “Governing a great nation is like cooking a small fish — too much handling will spoil it.” True, they are “governing” the Web rather than a nation, but the only difference that makes is that their decisions will affect everyone in the world, not just in a single country.

## CONCLUSIONS

It is safe to say that the term “Web 2.0” does not correspond to any specific set of technical features or methods. Moreover, it cannot seriously be argued that it represents a technical advance beyond the Web as envisaged by Berners-Lee in 1990 and ever since. Indeed, it still lags well behind the Web community's most ambitious projects, such as the Semantic Web. While Web 2.0 could

certainly use Semantic Web functionality, the Web 2.0 community — if there is such a thing — is unlikely to produce it.

In a way, Web 2.0 signals a return to the original concept of the Web as a democratic, borderless domain in which individuals and groups could communicate freely — both in the sense of “free speech” and “free beer.” The dot-com years saw an alarming lurch toward corporatism, and many people still see the Web as nothing more than a gigantic cash cow created solely for their benefit. If Web 2.0 succeeds in putting the individual first once more, that alone will be a valuable achievement.

The bottom line is that “Web 2.0” was introduced as a marketing term, a brand name for O’Reilly Media’s series of conferences. Many people have struggled to find a single unifying thread running through all its myriad themes. From a technical point of view, such a thread is hard to find. But a practical explanation is that Web 2.0 embraces a nice wide, varied set of interesting topics to be discussed at a popular, fashionable, lucrative conference. Dave Winer, one of the inventors of Web services, remarks dismissively that “Web 2.0 is a way for certain marketing people to claim they invented stuff that they didn’t invent, without actually claiming they invented it. It’s the kind of double-talk marketing guys love.”<sup>32</sup>

So it makes little sense to assign any wider and more portentous meanings to Web 2.0 — especially since CMP Media has shown its willingness to enforce its trademark rights against another conference organizer. To bracket the technology of the Web with a moneymaking scheme is utterly contrary to the principles laid down by its creator, Berners-Lee; and it seems both crass and futile for anyone to make the attempt. So let us be clear that “Web 2.0” should denote the conference staged by CMP Media and O’Reilly Media and nothing more. Of course, it will continue to be applied, loosely, to any topic suitable for discussion at the conference. But there is only one Web — and it is rich enough in potential to see us all through the next 10 years or more.

#### ABOUT THE AUTHOR

Tom Welsh is a Senior Consultant with Cutter Consortium’s Enterprise Architecture advisory service, author of its survey-based *Executive Updates*, and former Editor of Cutter Consortium’s *Web Services Strategies*. He is an independent consultant and analyst specializing in middleware, object technology, and software engineering. At Digital Equipment Corporation, Mr. Welsh was a hardware technician, software support specialist, corporate software developer, and senior technology consultant, before taking on the task of marketing Digital’s OO software products in the UK.

In his role as a principal analyst at ComputerWire, Mr. Welsh has been a leading contributor to *Client/Server Tools Bulletin*, *Internet Tools Bulletin*, and *Object-Oriented Tools Bulletin*. He has written many reports and papers, as well as chairing and speaking at conferences and seminars. Since 1992, Mr. Welsh has closely followed the work of the OMG and its specifications, including CORBA, UML, XMI, and CWM. He can be reached at [twelsh@cutter.com](mailto:twelsh@cutter.com).

#### ENDNOTES

<sup>1</sup>Wikipedia. “Web 2.0” ([http://en.wikipedia.org/wiki/Web\\_2.0](http://en.wikipedia.org/wiki/Web_2.0)).

<sup>2</sup>Hagel, John III. “What Is Web 2.0?” Edge Perspectives with John Hagel (blog), 25 September 2005 ([http://edgeperspectives.typepad.com/edge\\_perspectives/2005/09/what\\_is\\_web\\_20.html](http://edgeperspectives.typepad.com/edge_perspectives/2005/09/what_is_web_20.html)).

<sup>3</sup>Angermeier, Markus. “The huge cloud lens bubble map web2.0.” Kosmar (blog) (<http://kosmar.de/archives/2005/11/11/the-huge-cloud-lens-bubble-map-web20/>).

<sup>4</sup>*The Economist*. “Aftershock,” 31 May 2005 ([www.economist.com/displaystory.cfm?story\\_id=3819476](http://www.economist.com/displaystory.cfm?story_id=3819476)).

<sup>5</sup>Carr, Nicholas. “The amorality of Web 2.0.” Rough Type (blog), 3 October 2005 ([www.roughtype.com/archives/2005/10/the\\_amorality\\_o.php](http://www.roughtype.com/archives/2005/10/the_amorality_o.php)).

<sup>6</sup>Dvorak, John C. “Web 2.0 Baloney.” *PC Magazine*, 1 March 2006 ([www.pcmag.com/article2/0,1895,1931858,00.asp?](http://www.pcmag.com/article2/0,1895,1931858,00.asp?)).

- <sup>7</sup>Orlowski, Andrew. "What is Web 2.0? You redefine the paradigm." *The Register*, 11 November 2005 ([www.theregister.co.uk/2005/11/11/web\\_two\\_point\\_naught\\_answers](http://www.theregister.co.uk/2005/11/11/web_two_point_naught_answers)).
- <sup>8</sup>Hinchcliffe, Dion. "The Best Web 2.0 Software of 2005" ([http://web2.wsj2.com/the\\_best\\_web\\_20\\_software\\_of\\_2005.htm](http://web2.wsj2.com/the_best_web_20_software_of_2005.htm)).
- <sup>9</sup>Sherriff, Lucy. "MySpace sued for \$30m after assault on 14-year-old." *The Register*, 20 June 2006 ([www.theregister.co.uk/2006/06/20/myspace\\_sued](http://www.theregister.co.uk/2006/06/20/myspace_sued)).
- <sup>10</sup>Out-Law.com. "YouTube wanders into copyright mire." *The Register*, 19 July 2006 ([www.theregister.co.uk/2006/07/19/youtube\\_copyright](http://www.theregister.co.uk/2006/07/19/youtube_copyright)).
- <sup>11</sup>Hof, Rob. "Flock, the New Browser on the Block." *BusinessWeek*, 5 October 2005 ([www.businessweek.com/technology/content/oct2005/tc2005105\\_2789\\_tc024.htm](http://www.businessweek.com/technology/content/oct2005/tc2005105_2789_tc024.htm)).
- <sup>12</sup>Decrem, Bart. "Introducing Flock Beta 1." Flock ([www.flock.com/blog/introducing-flock-beta-1](http://www.flock.com/blog/introducing-flock-beta-1)).
- <sup>13</sup>Levy, Steven, and Brad Stone. "The New Wisdom of the Web." *Newsweek*, 3 April 2006 ([www.msnbc.msn.com/id/12015774](http://www.msnbc.msn.com/id/12015774)).
- <sup>14</sup>O'Reilly, Tim. "Not 2.0?" O'Reilly Radar (blog), 5 August 2005 ([http://radar.oreilly.com/archives/2005/08/not\\_20.html](http://radar.oreilly.com/archives/2005/08/not_20.html)).
- <sup>15</sup>O'Reilly, Tim. "What Is Web 2.0," 30 September 2005 ([www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html](http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html)).
- <sup>16</sup>O'Reilly, Tim. "Hardware, Software, and Infoware." In *Open Sources: Voices from the Open Source Revolution*. O'Reilly Media, 1999 ([www.oreilly.com/catalog/opensources/book/tim.html](http://www.oreilly.com/catalog/opensources/book/tim.html)).
- <sup>17</sup>Anderson, Chris. *The Long Tail: Why the Future of Business Is Selling Less of More*. Hyperion, 2006.
- <sup>18</sup>Wikipedia. "The Long Tail" ([http://en.wikipedia.org/wiki/Long\\_tail](http://en.wikipedia.org/wiki/Long_tail)).
- <sup>19</sup>Berners-Lee, Tim, James Hendler, and Ora Lassila. "The Semantic Web." *Scientific American*, May 2001 ([www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21&pageNumber=1&catID=2](http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21&pageNumber=1&catID=2)).
- <sup>20</sup>Fried, Ina. "Gates: We're entering 'live era' of software." CNET News.com, 1 November 2005 ([http://news.com.com/Gates+Were+entering+live+era+of+software/2100-1016\\_3-5926237.html](http://news.com.com/Gates+Were+entering+live+era+of+software/2100-1016_3-5926237.html)).
- <sup>21</sup>Postel's Prescription (<http://catb.org/jargon/html/P/postcardware.html>).
- <sup>22</sup>Raymond, Eric. "Chapter 1: Philosophy." In *The Art of UNIX Programming*. Addison-Wesley, 2003 ([www.awprofessional.com/content/images/0131429019/samplechapter/raymondch01.pdf](http://www.awprofessional.com/content/images/0131429019/samplechapter/raymondch01.pdf)).
- <sup>23</sup>Fielding, Roy Thomas. "Architectural Styles and the Design of Network-based Software Architectures," 2000 ([www.ics.uci.edu/~fielding/pubs/dissertation/top.htm](http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm)).
- <sup>24</sup>Kaye, Doug. "Jeff Barr — Amazon Web Services." IT Conversations ([www.itconversations.com/transcripts/31/transcript31-1.html](http://www.itconversations.com/transcripts/31/transcript31-1.html)).
- <sup>25</sup>Rip, Peter. "Enterprise 2.0." EarlyStageVC (blog), 29 October 2005 ([http://earlystagevc.typepad.com/earlystagevc/2005/10/enterprise\\_web\\_.html](http://earlystagevc.typepad.com/earlystagevc/2005/10/enterprise_web_.html)).
- <sup>26</sup>Davis, Ian. "Talis, Web 2.0 and All That." Internet Alchemy (blog), 4 July 2005 (<http://internetalchemy.org/2005/07/talis-web-20-and-all-that>).
- <sup>27</sup>Bray, Tim. "Web 2.0." ongoing (blog), 26 May 2006 ([www.tbray.org/ongoing/When/200x/2006/05/26/Web-20-trademark](http://www.tbray.org/ongoing/When/200x/2006/05/26/Web-20-trademark)).
- <sup>28</sup>Licklider, J.C.R. "Man-Computer Symbiosis." *IRE Transactions on Human Factors in Electronics*, Vol. HFE-1, pp. 4-11, March 1960 (<http://groups.csail.mit.edu/medg/people/psz/Licklider.html>).
- <sup>29</sup>Orlowski, Andrew. "Web 2.0 meets its Alamo?" *The Register*, 12 July 2006 ([www.theregister.co.uk/2006/07/12/netscape\\_diggs\\_portal](http://www.theregister.co.uk/2006/07/12/netscape_diggs_portal)).
- <sup>30</sup>Lombardi, Candace. "Google exec challenges Berners-Lee." CNET News.com, 18 July 2006 ([http://news.com.com/2100-1025\\_3-6095705.html?part=rss&tag=6095705&subj=news](http://news.com.com/2100-1025_3-6095705.html?part=rss&tag=6095705&subj=news)).
- <sup>31</sup>Farrell, Nick. "Google says web users are too stupid for semantics." *The Inquirer*, 21 July 2006 ([www.theinquirer.net/default.aspx?article=33143](http://www.theinquirer.net/default.aspx?article=33143)).
- <sup>32</sup>Winer, Dave. Scripting News (blog), 19 December 2005 ([www.scripting.com/2005/12/19.html#busted](http://www.scripting.com/2005/12/19.html#busted)).

# Enterprise Architecture Practice

Today the demands on corporate IT have never been greater. Cutting costs and accelerating time to market for individual line-of-business projects are still priorities, but even that's not nearly enough anymore. Companies are now looking for strategies to better leverage their entire IT infrastructure. They want IT to deliver sophisticated enterprise applications that can provide value across many lines of business and provide marked differentiation from their competitors. The Enterprise Architecture Practice provides the information, analysis, and strategic advice to help organizations commit to and develop an overarching plan that ensures their whole system fits together and performs seamlessly.

The Enterprise Architecture Practice offer continuous research into the latest developments in this area, including Web services, enterprise application integration, XML, security, emerging and established methodologies, Model Driven Architecture, how to build an enterprise architecture, plus unbiased reports on the vendors and products in this market. Consulting and training offerings, which are customized, can range from mapping an infrastructure architecture to transitioning to a distributed computing environment.

## Products and Services Available from the Enterprise Architecture Practice

- The Enterprise Architecture Advisory Service
- Consulting
- Inhouse Workshops
- Mentoring
- Research Reports

## Other Cutter Consortium Practices

Cutter Consortium aligns its products and services into the nine practice areas below. Each of these practices includes a subscription-based periodical service, plus consulting and training services.

- Agile Project Management
- Business Intelligence
- Business-IT Strategies
- Business Technology Trends & Impacts
- Enterprise Architecture
- IT Management
- Measurement & Benchmarking Strategies
- Enterprise Risk Management & Governance
- Sourcing & Vendor Relationships

# Senior Consultant Team

Our team of internationally recognized specialists offers expertise in security issues, e-business implementation, XML, e-business methodologies, agents, Web services, J2EE, .NET, high-level architecture and systems integration planning, managing distributed systems, performing architecture assessments, providing mentoring and training, overseeing or executing pilot projects, and more. The team includes:

- Michael Rosen, Practice Director
- Scott W. Ambler
- Douglas Barry
- Mark Choate
- Don Estes
- Pierfranco Ferranato
- Clive Finkelstein
- Kurt Guenther
- Michael Guttman
- Paul Harmon
- David Hay
- Tushar Hazra
- J. Bradford Kain
- Bartosz Kiepuszewski
- Sebastian Konkol
- André LeClerc
- Jean Pierre LeJacq
- Arun K. Majumdar
- Thomas R. Marzolf
- Jason Matthews
- James Odell
- Ken Orr
- Wojciech Ozimek
- Oliver Sims
- Borys Stokalski
- John Tibbetts
- Sandy Tyndale-Bisco
- William Ulrich
- Jeroen van Tyn
- Jim Watson
- Tom Welsh
- Bryan Wood