

A Web 2.0 Primer

What Enterprises Should Know

>
accenture

High performance. Delivered.

• Consulting • Technology • Outsourcing

The term Web 2.0 was coined in 2004 and promptly trademarked by O'Reilly Media. It's not a very specific term; rather, it's an umbrella that covers many technologies and approaches. It encompasses blogs and wikis, peer-to-peer file sharing and open source, web services, cost per click and multi-platform delivery.

The full collection of Web 2.0 architectures, business models and technologies is sufficiently broad and eclectic that it can't be addressed in a single short paper or fit into a single consistent framework. There *is* one (nearly) general thing we can say about Web 2.0, however. To some extent, its theme is mass participation, where participation takes the form of creating, editing, or ranking content.

For purposes of this paper, we've elected to concentrate on Web 2.0 "phenomena" that, we believe, will have a **significant impact on the enterprise**. Hence BitTorrent, AdSense, Napster, search engine optimization and others have been passed over in favor of **blogs, wikis, folksonomies, crowd-sourcing, mashups and Rich Internet Applications**. These six fit solidly under O'Reilly's conception of Web 2.0 and their effects will be felt at the enterprise level.

Blogs, for example, will provide significant improvements in internal and external communication while Wikis will make it easy to bring many minds together on a single deliverable. Crowd-sourcing will make it possible to harness thousands (or even millions) of inexpensive resources to solve a problem or create content.

Folksonomies seem likely to improve enterprise knowledge management. Mashups will make it possible for less-technical people to build systems and Rich Internet Applications will increase the power (and reduce the TCO) of software.

This paper briefly examines these six phenomena from an enterprise perspective.

Rich Internet Applications

Web 1.0 brought an unprecedented level of information and functionality to individuals and the enterprise. However, its page-oriented user interface represented a step backwards to the time of the green-screen terminal. Gone was the instant, continuous response we had come to expect from PC applications—back came the waits as the whole screen updated each time we pressed a button or hit the Enter key. Even at its best, Web 1.0's page orientation couldn't begin to match the fluidity and engagement of desktop software.

Web 2.0 has a solution to the page-after-page challenge in the form of Rich Internet Applications (RIAs). RIAs are highly interactive clients delivered through a Web browser. When you arrive at a RIA page, code is automatically downloaded to your browser. That code in turn manages most of your interactions with the application. It will sometimes asynchronously retrieve data from the server, but for the most part it operates locally and autonomously—like a conventional PC program.

RIAs bring back the instant, continuous response that client/server applications taught us to expect. In so doing, they increase the amount of information we see (i.e., they improve "user bandwidth") and extend the reach of our software to anywhere a browser can go—most important, however, RIAs can engage customers to a degree unknown in Web 1.0 applications. Two examples (from "IDC, Rich Internet Applications," November 2003):

MINI USA (the car company) wanted to develop a web site that would inform and entertain prospective customers. They created a fun, dynamic RIA on which it was possible to easily configure your own MINI. As you specified parameters, the application instantly updated pictures (interior and exterior) to reflect your choices. During its first year, visitors to the site configured an average of 1.48 MINIs each, and the conversion-to-sale rate for those who did create a configuration was an astonishing 30%. Although correlation is not causality, it seems plausible that the fluidity and responsiveness provided by the RIA was a key component of the site's (and MINI's) success.

The Broadmoor Hotel's online reservation system was once a typical page-after-page, Web 1.0 application. In an effort to improve conversion rates, it turned to a RIA-based reservation system that implemented most of the reservation process (date selection, room selection, credit card entry, totals) on a single page. As you change dates, the available room types change, instantly and without a page refresh. The total cost of your stay is shown dynamically as you change dates and room types. The site is as fluid and engaging as any Windows application. After it was released, the hotel's conversion rate went from 2.7% to 22%; its reservation count increased 20%; it saw a 50% increase in revenue booked and a 66% increase in room nights.

Inside the enterprise, RIAs can have a similarly powerful effect. If internal applications are built on Web 1.0 (or the mainframe), the switch to RIA (like the switch to client/server) can yield significant usability benefits. These translate into reduced training, improved productivity, and reduced user support.

A RIA will sometimes asynchronously retrieve data from a server, but for the most part it operates locally and autonomously, like a conventional PC program.

Rich Internet Applications have three broad implications. First, improvements to the user interface, as discussed above. Second, a (possible) reduction in bandwidth requirements because pieces of data (rather than whole pages) are sent from the server to the client.

The third implication is the retention of Web 1.0-style release management, in which you're free to fix and enhance an application without worrying about installing new client-side code. This is in contrast to "classical" client/server architectures, in which installing, upgrading and patching client-side code is a significant challenge. The impact can be significant: Gartner's "TCO Comparison of PCs with Server-Based Computing," June 2006, found that server-based computing is anywhere from 12% to 48% less expensive than for comparable PCs.

One area where RIAs will clearly excel is in the delivery of Software as a Service (SaaS). We can expect to see the typical SaaS application—often a page-oriented Web 1.0 system today—become much more sophisticated, which in turn should broaden its appeal and drive sales.

For all their power, RIAs are clearly challenged when it comes to disconnected operation: unplug a typical browser application and it ceases to work. Some types of RIA (text editors, for example) may have to overcome this limitation if they are to succeed. Local, cross-session caching of code and data will be required (which, of course, will blur the line between RIAs and conventionally installed applications).

It seems likely that RIA development costs will be roughly comparable to the development costs of the client-server applications they so closely resemble. Deployment costs, by contrast, should be lower—closer to those associated with Web 1.0 applications.

RIAs are clearly poised to have a significant impact on the way enterprises deliver applications and we strongly encourage early experimentation. The tool sets and environments are different from those of Web 1.0 and new skill sets will be required—writing test applications will let

you evaluate tools and assess your developers' readiness to work in a RIA-centric world.

It's clear that RIAs are not just gimmicky: pretty buttons, video, drag-and-drop icons and animation. They are a powerful way to engage users and thus to increase revenues, reduce costs, and keep customers coming back.

For an in-depth look at Rich Internet Applications that covers all of the topics above and more, see the Accenture AIMS whitepaper, "Rich Internet Applications - State of the Art."

Mashups

A mashup is a Web site built by integrating data from two or more other sites using web services APIs. The most common type of mashup takes information from Google Maps and combines it with geo-data from another site in order to plot features of interest—all the golf courses in New Zealand, for example. The next most common mashups are photo-based applications, usually built around Flickr. Search applications (across Amazon, eBay, Google, YouTube, and others) are the third most numerous. A current list of mashups can be found at www.programmableweb.com. At this writing, there were over 2,100.

Within the enterprise there's the potential for something more significant than golf maps. Web services can be used to wrap both new and legacy enterprise software so as to expose functionality and data. These services can then be invoked by an abstract "orchestration layer" to deliver new "composite" applications. Such a framework is called a Service Oriented Architecture (SOA).

SOA's benefits include faster application development, increased flexibility in the face of business process change, and

higher code quality. (Note that mashups, while consistent with and reminiscent of SOA, are not themselves SOA applications.) For more information, see www.accenture.com/soa.

One example of an enterprise mashup comes from Accenture, whose Performance Workspace combines data and functionality from line-of-business applications (like CRM) with knowledge bases, expert directories, and other components to create significant business value. An implementation of the Performance Workspace for BT Retail's call center drove a 25% productivity improvement and a 75% improvement in time-to-competency.

Another (very successful, ad-supported) mashup, Zillow, combines real estate data from a wide variety of sources with Google Maps imagery to produce aerial views of homes annotated with their estimated value and other information. Zillow's presentation of these disparate sources of publicly-available information makes them much more accessible than they would be in isolation. Zillow is also (as are many mashups) an RIA.

Tools are increasingly available that allow mashups to be created by less-technical people, which raises an interesting value opportunity around the IT backlog—which (according to CIO Magazine, August 2006) is in excess of 50 projects for 71% of U.S. companies. To the extent that the needed systems feature search, reporting and data visualization, it may be possible for the business unit to develop its own applications and thereby put a significant dent in the backlog. (This is not as far-fetched as it sounds. The spreadsheet made it possible for end users to manipulate numbers and produce reports entirely without the aid of IT. All we need is the "spreadsheet" of mashups.) IT can (and should) facilitate the process by making data and functions available to the business units via web services. IT should also get ahead of the tool adoption process by picking, disseminating and supporting a mashup tool set.

Crowd-sourcing is the recruiting of large groups of (usually unpaid, often amateur) contributors to create content or solve problems.

Crowd-Sourcing

Crowd-sourcing is the recruiting of large groups of (usually unpaid, often amateur) contributors to create content or make decisions. Examples abound. The Wikipedia is crowd-sourced (it has a total of five paid employees and 17,000 regular contributors). YourNextDrink.com solicits drink recipes from visitors and leverages their taste buds in order to get the recipes rated. The highest-rated recipes bubble to the top of search results, which makes YourNextDrink a source of particularly good recipes, none of which it developed or rated itself. Monetizing what people willingly do for free is perhaps the central tenet of crowd-sourcing.

Amazon does something similar to YourNextDrink with product reviews—visitors can submit their own and rate others', from which unpaid labor Amazon derives significant competitive advantage.

A company called InnoCentive, which was founded by Eli Lilly in 2001, lets companies crowd-source their research. Companies (including 35 of the Fortune 500) post

research challenges that in turn may be tackled by any of InnoCentive's 90,000 registered researchers in 175 countries. Interestingly, the solutions to these problems often come out of "left field"—a physicist will easily solve what is nominally a difficult chemistry problem, for example. Awards (here's a case of compensated crowd-sourcing) run from \$5,000 to \$1 million—which may be very low compared to the value of a solved R&D problem. Many of the researchers are amateurs or retired. InnoCentive claims a 30% success rate, which for R&D is extremely impressive.

Threadless, a T-shirt company, accepts designs from visitors (who serve here as illustrators); has visitors rate those designs (now they're serving as market researchers); then turns the most popular designs into T-shirts which it sells...to visitors (who have now become customers). Threadless's two founders made \$20 million on the crowd-sourced efforts of their unpaid visitors.

Crowd-sourcing has potential within the enterprise as well. One research project being pursued by Accenture, the Innovation Grapevine, is attempting to create "virtual experts" out of crowds of less-experienced people. This "Wisdom of Crowds" approach should enable more effective use of scarce expertise.

Another project, this one for a client, used a wiki to induce dozens of tellers and other employees to contribute and refine what turned out to be very good ideas for transforming a bank.

It is possible to create participation among stakeholders; there seem to be two key facets to the challenge. First, usability: the wiki (or whatever tool is chosen) must be very easy to use. Second, there need to be incentives in place. These may take the form of prizes, explicit recognition, or just the opportunity to perform well in front of peers and superiors. A third facet, which you can exploit only once, is novelty: the first time it's attempted, a crowd-sourcing exercise may be attractive simply because it is new.

Folksonomies

Folksonomies are a new way of doing manual indexing that hold out the hope of significantly alleviating certain knowledge management problems. Conventional manual indexing goes back centuries, and has always worked thus: a few experts create a list of index words, which in turn are used by a larger group of experts to categorize material (usually documents). These words are then used by a very large group of searchers to find what they're looking for. (Think here of the Dewey Decimal System.)

The technique has some well-known problems. In particular, inter-indexer consistency is poor and the list of words tends to become less relevant with time since new material "drifts away" from it. Eventually, indexers find themselves "shoe-horning" documents into categories that no longer fit the material, and consistency (and thus search accuracy) suffers even further.

Under what is sometimes called a "folksonomy" (folk taxonomy), users create the index words and assign them to material. If you've uploaded a picture to Flickr (a photo sharing service) and want

to categorize it, for example, you might first search through Flickr's existing categories for a word that fit. If there wasn't one, you'd create your own and add it to the list of words, then tag your photo with it. This approach nicely addresses the "drift" problem—since new words can be added at will, the list of categories will always remain relevant.

The difficulty (one would expect) comes with inter-indexer consistency. A small group of experts has a hard enough time consistently indexing a corpus of material—a problem that results in significant difficulties for searchers. It seems reasonable that when you have millions of "experts" creating and applying index words, the difficulties with consistency and redundancy (and hence search) will be significantly magnified. Counter-intuitively, however, at least one study has shown that folksonomies can be quite stable and effective ("Usage Patterns of Collaborative Tagging Systems," Golder and Huberman, 2006).

Folksonomies create a compelling opportunity around intranets. According to one study ("IDC Update: The Hidden Costs of Information Work," April, 2006), the cost of not finding needed information is some \$5,000 per employee per year, or \$50,000,000 per year for a 10,000-person enterprise. It's clear that significant value would be unlocked if folksonomies could be leveraged to improve search effectiveness. Enterprises needn't necessarily begin building folksonomy infrastructures—we can expect content management system vendors to make folksonomy functionality available in their products. It is important, however, to recognize folksonomies' power and be ready to leverage them as they become available.

Internal blogs serve to communicate (an immediate coordination benefit) and to provide an accessible history of the organization's activities (a long-term knowledge management benefit).

Blogs

A blog is a chronologically-organized online journal. A blog post may be of any length, though a paragraph is typical. What's interesting about blogs is the degree to which they interlink: often, a post will consist of commentary on (and a link to) a post from another blog. Readers may add comments to the original post, or to other comments on that post, thereby creating a persistent conversation among author and audience.

Blog authors can publish their content, and readers can subscribe to that content, using RSS (Really Simple Syndication) feeds. RSS feeds eliminate the need to check blogs repeatedly to see if any new posts have been added—instead, notifications of new material come to you. This notification mechanism makes it easier to follow blogs' content—which means you tend to follow more blogs.

RSS has another, counterintuitive advantage: it's a qualitatively new channel—people don't mind getting RSS alerts the way they mind getting email. Thus RSS can serve as an alternative

(perhaps superior) way of letting people know about key events—project status, updates to shared documents, schedule changes, etc.

In the enterprise, blogs can be of two types: internally-facing and externally-facing. Internal blogs serve to communicate (an immediate coordination benefit) and to provide an accessible history of the organization's activities (a long-term knowledge management benefit).

Blogs' long-term knowledge management benefits are an important dividend. In an ideal world, its intranet would house all of the documents created by the enterprise. As a practical matter, 1) it's infeasible to get 100% contribution and 2) many of the enterprise's activities aren't documented in the first place. However, to the extent that a firm's actions and decisions are reflected in its blogs (which are essentially conversations), they form a recoverable record.

One easily overlooked advantage of internal blogging is that it can decrease email traffic. Many of the posts in a blog represent emails (particularly group emails) that were never sent. 10 such posts * 10 replies to each (for example) is 100 emails that never entered your inbox.

About 8% of the Fortune 500 are blogging externally—Microsoft, with its 3,000 externally-facing bloggers, is perhaps the most famous example. Why would an enterprise encourage its employees to blog? Possibly in self-defense—after all, Microsoft's customers, developers and other stakeholders are certainly blogging about Microsoft. Joining the conversation can be a good way to shape its direction.

Public blogs can be an excellent way to maintain relationships with stakeholders. The risks revolve around safety and reputation: if employees are relentlessly upbeat (as they might be if their material is vetted), readers will dismiss them as tools of the marketing department.

On the other hand, unmonitored blogging can lead to embarrassment if employees criticize their employer (or its competitors). Microsoft seems to have come down on the side of "free speech:" its bloggers regularly highlight perceived faults in Microsoft's products and strategy. This unconstrained expression of opinion has made many of its bloggers quite credible with stakeholders—an important benefit to Microsoft as it tries to convey its messages.

One potential risk associated with unmonitored, industrial-scale blogging is that it may turn an enterprise into a "transparent organization"—unable to keep certain kinds of sensitive information (such as product plans) out of the public eye. Each blog can be a source not of leaks, exactly, but of clues that can add up to insights into the organization's activities.

External blogging is valuable for the same reasons advertising is valuable—and it can reach stakeholders at a fraction of the cost. If research shows that your stakeholders read blogs (and not all

enterprises will find that this is true), it may make sense to begin an external blogging "campaign."

The most effective initial approach is probably that adopted by General Motors, which runs a blog that is fed by a wide variety of authors, including vice-chairman Bob Lutz. The blog is presided over by a dedicated editorial team, and the posts are well-written and thoughtful without sounding like "marketing-speak." Having multiple contributors to a single blog reduces the burden on any one individual and makes it easier to keep volume high.

Internally, the multi-employee blog can also be a good idea, though another approach is to create a single blog that's read and contributed to by an entire workgroup. Such a blog is used to

communicate events, discuss issues, and generally keep people informed of the group's thoughts and activities.

Internally, blogging is valuable for the same reasons conversation, email, voicemail and instant messaging are valuable—it facilitates information flow, improves creativity, and reduces duplication of effort. Blogs' value is such that they will probably become an integral part of the organization—much as email and IM are today.

It's common for upwards of 40 people to collaborate on a single Wikipedia article.

Wikis

A wiki (Hawaiian for "fast") is a collaborative hypertext authoring environment within which multiple people can contribute to the creation of a document. Contributions may take the form of composing or editing, and it's common for authors to edit (and even compose in close proximity to) one another's work. The concept is deeply counterintuitive: by all rights, fine-grained group-composition of documents should create a non-converging mess and a legacy of latent hostility. Experience demonstrates that this is not the case.

The most famous wiki is the Wikipedia, a 1.7 million-article encyclopedia created almost entirely with unpaid labor. Again counter-intuitively, one study found that its error count is quite close to that of the much more "formal" Encyclopedia Britannica (with 65,000 articles)—a dramatic confirmation of the open source software dictum that "with enough eyeballs, all bugs are shallow." (Note that Britannica denounced the study as "fatally flawed" and no follow-up study has been performed. Note also that Wikipedia has experienced abuses that don't bedevil Britannica: "edit wars" among passionate contributors, for example.)

It's common for upwards of 40 people to collaborate on a single Wikipedia article—meaning that the "brains-per-page" count is startlingly high. Interestingly, Wikipedia operates for the most part on consensus—contributors must resolve disputes through debate, not mediation.

Wikis are being used within the enterprise as well. Internal wikis can cut through bureaucracy and across silos by giving everyone involved in a project an equally powerful outlet for their ideas. At Accenture, the most successful wiki effort to date was run by the Accenture Delivery Architectures group, which with 50 people developed some 1,500 pages of best practices material over the course of nine months.

Microsoft is experimenting with using an external wiki to enhance its documentation for Visual Studio 2005 and .Net Framework 2.0. Developers are encouraged to add comments and code snippets for the benefit of the rest of the community. Note that it is not possible to edit existing material: contributors can append, but not change.

F5 Networks, a networking hardware firm, has a wiki it uses to collaboratively brainstorm, specify, design and implement new applications. F5 says that the wiki is in active use by some 10,000 customers and employees. This arrangement almost certainly inspires a very high level of customer loyalty and engagement—a level that might not be possible using any other approach. F5 points the way toward the use of wikis for creating non-text artifacts (code). Wikis that support the editing of presentations, Visio diagrams, images, audio, 3D objects and so on will undoubtedly be developed. These will improve the effectiveness of design teams—particularly distributed design teams.

As with blogs, bringing wiki software in-house and making it generally available is inexpensive, low-risk, and recommended. Resist the temptation (it's caused by "Wikipedia envy") to create a single "superwiki" to which hundreds or thousands of employees are expected to contribute in their "spare" time. Instead, start modestly: have small groups work on specific deliverables. As you gain experience with the technology, you can evaluate expanding your horizons.

Conclusion

The technologies and approaches outlined in this paper vary in their maturity, complexity, and in the degree to which we understand how to deploy them effectively. All, however, have potentially significant business value.

Blogs and wikis can create large improvements in collaboration and communication, but because they are both fundamentally social technologies, they (like IM and email) tend to defy cost-benefit analysis. IM's analysis, for example, was never performed—it entered the organization through the back door and spread virally. To this day we don't know how much the enterprise is gaining (or losing) by its use. (But just try to take it away.) In enterprises of any size, something similar is probably happening with blogs and wikis. Advice for users: experiment. Advice for CIOs: move quickly to select, deploy and support blog and wiki tools...before your users do it themselves.

Folksonomies' potential business value is easier to characterize, since we do have numbers (quoted above) around losses caused by poor knowledge management practices. Organizations probably needn't

worry about folksonomies entering through the back door—the effort necessary to build a folksonomy infrastructure and integrate it with the intranet would be significant and hence difficult to hide. Where folksonomies will first appear is in commercial content management systems, which increasingly will include folksonomy functionality. Take advantage of them when they arrive.

Crowd-sourcing is the most challenging of the Web 2.0 phenomena presented. On the one hand, it can lead to very inexpensive creation of high-value content. On the other hand, each enterprise will have to decide for itself what content it makes sense to solicit; how to attract contributors (potentially very difficult); how to retain them; and how (if at all) to compensate them. There is no one-size-fits-all formula, yet by addressing the people, process and technology dimensions, Accenture has seen great success in helping clients develop effective crowd-sourcing programs.

Because of their accessibility to less-technical developers, **mashups** hold out the hope of 1) increasing user satisfaction

We cannot emphasize strongly enough the need to begin experimenting with RIAs as soon as possible.

with IT and 2) reducing the application backlog. Mashups do not necessarily let you ignore IT; rather, they represent a new, more productive way to partner with IT.

Finally, **Rich Internet Applications**. Unlike much of Web 2.0, RIAs have clear business benefits (reduced TCO and significantly improved user experience) and the tools and platforms necessary to deploy them are becoming more sophisticated by the day. We cannot emphasize strongly enough the need to begin experimenting with RIAs as soon as possible.

The component technologies of Web 2.0 are new, innovative, largely untried, and perhaps somewhat scary. But Accenture's research indicates that innovation is a key feature of high-performance businesses...and we expect high-performance businesses to embrace Web 2.0.

About Accenture

Accenture is a global management consulting, technology services and outsourcing company. Committed to delivering innovation, Accenture collaborates with its clients to help them become high-performance businesses and governments. With deep industry and business process expertise, broad global resources and a proven track record, Accenture can mobilize the right people, skills and technologies to help clients improve their performance. With more than 158,000 people in 49 countries, the company generated net revenues of US\$16.65 billion for the fiscal year ended Aug. 31, 2006. Its home page is www.accenture.com.

About Accenture Technology Labs

Accenture Technology Labs, the dedicated technology research and development (R&D) organization within Accenture, has been turning technology innovation into business results for 20 years. The Labs create a vision of how technology will shape the future and invent the next wave of cutting-edge business solutions. Working closely with Accenture's global network of specialists, Accenture Technology Labs helps clients innovate to achieve high business performance. The Labs are located in Chicago, Illinois; Silicon Valley, California; Sophia Antipolis, France; and Bangalore, India. For more information, please visit our website at www.accenture.com/accenturetechlabs.

For more information, please contact
Ed Gottsman at
edward.j.gottsman@accenture.com

Copyright © 2007 Accenture
All rights reserved.

Accenture, its logo, and
High Performance Delivered
are trademarks of Accenture.