

WEB 2.0: An Implementation and Acquisition Primer

By Rafael Collado, Phacil Co-Founder



Mr. Rafael Collado is a successful engineer and IT systems executive, specializing in the telecommunications and IT industries. He has developed network management and diagnostic systems, and designed communications processors for the US Government and private companies. For more information on Mr. Collado, please visit the [Management](#) page.

Michael Islek, Subject Matter Expert

Mr. Islek is an engineer and IT systems executive who has been instrumental in pioneering cutting edge web-based website publication tools and in the design and development of telecommunications equipment. He is a senior system architect, supporting all MDA Knowledge Online (MKO) portal software development, enhancements and support efforts.

Introduction

Beginning last year the term WEB 2.0 began to be associated with all sorts of very successful companies who were very effective in using the Internet to deliver their services. The wave has, in recent months, reached the shores of Government IT and the rush is on to anoint all sorts of Government information technology projects with the fashionable moniker of WEB 2.0. Advances in commercial use of web technology will be difficult to migrate to Government use because of the relatively long acquisition cycles. It is important for those who manage the implementation of Government IT services, nevertheless, to pay attention and understand the fundamental operations and implications of these new technology implementations.

Users of Government IT systems will come to work expecting to be able to interact with their institutional IT resources in a way similar to the way they use Internet resources at home. The urgency for Government IT leadership's understanding of this WEB 2.0 technology environment is self-evident. The ability to efficiently and effectively undertake their work tasks will be directly proportional to how easily workers can leverage the skills they acquire while using the Internet at home. Vendors, whether they are Microsoft, SAP, Oracle, etc. will develop and market these technologies to meet their worldwide commercial customer needs. Government customers must understand what WEB 2.0 is and what implications (benefits as well as liabilities) it has for their agencies or services.

Like glory, all technology is fleeting in the sense that the rapid deployment cycle of IT based technology has a half-life that is increasingly being measured in months rather than years. This means that those of us that manage IT deployment and resources no longer have the luxury of time when it comes to evaluating and implementing new methods. Today's technology might be a "fad" but it is more likely that today's "fad" may become a dominant technology in six months and then replaced eight months after that. The best reason to understand WEB 2.0, and what may follow it, is Botwinick's Theorem. This theorem states that, as it relates to information technology, three things are always true:

1. Implementation is always more expensive than originally projected
2. More bandwidth, cycles, memory, etc. is needed than you originally specified
3. 1 and 2 are the optimistic perspective

This paper is written to provide the senior level manager with sufficient background and facts so that they can gain the perspective necessary to:

1. Evaluate what components of the WEB 2.0 story their institutions will embrace
2. Determine how WEB 2.0 can position them for change
3. Engage in informed conversation with those members of their staff, contractors, consultants and colleagues who are stakeholders in their agency or service's information technology universe

Phacil engineers have been involved in Internet/web development since the early 1980s and this paper summarizes our view on WEB 2.0, which we believe to be the latest evolution of the combination of information processing and communications. This paper is written to provide middle and senior level leadership with a brief recitation of the philosophy, implementation and major issues related to WEB 2.0 enabled IT projects. This paper limits itself to the salient issues of WEB 2.0 since it is written not for the engineers and analysts whose principal task is web application development, but for the senior manager or stakeholder who wishes to gain some understanding of the issues. Included herein will be a working definition of WEB 2.0, major technology supporting WEB 2.0, important considerations in acquiring WEB 2.0 technologies and security/ implementation recommendations with respect to WEB 2.0 projects.

WEB 2.0 Defined

While many people use the term WEB 2.0 as though it were a specific technology or technological execution, the term actually arose to describe the maturation and successful integration in the market of a group of related technologies. Most importantly, WEB 2.0 was used to describe the state of events where static informational web pages were increasingly replaced by more dynamic, interactive web experience.

A simple analogy would be comparison of desktop operating system and its evolution. In early version of Windows 95, the user experience and system capabilities were much constrained due to lack of memory and computational resources. However, Windows XP and or VISTA has considerably enhanced user interface that includes dynamic and animated interfaces. WEB 2.0 is a similar technological evolution in the presentation of information as it relates to Web Based Applications. One of the other important things that WEB 2.0 attempts to achieve is extend user interface and processing capabilities of web based applications to match those offered by a desktop application. In fact, WEB 2.0 has become so close to achieving its goal, the difference between a desktop and a web based application becoming more and more blurred every day.

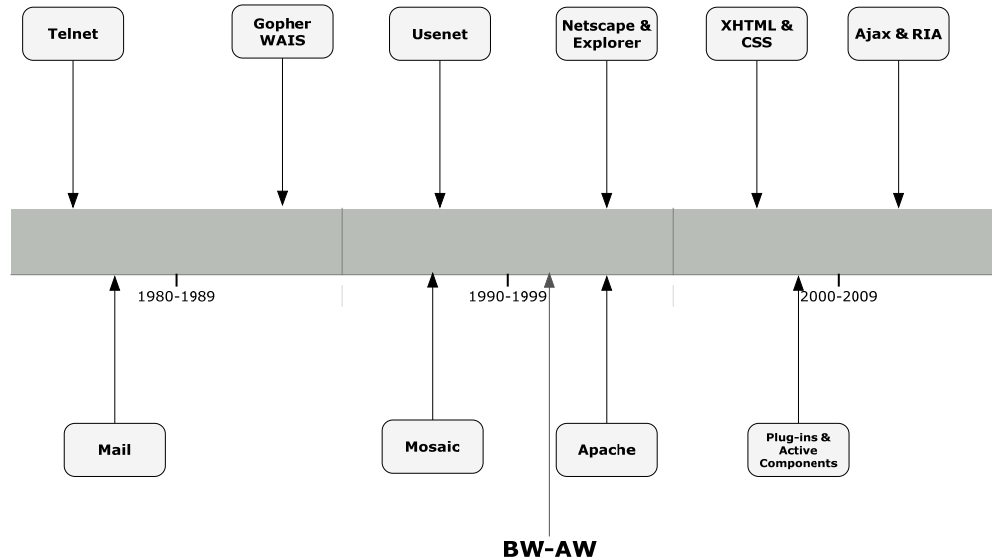


Figure One: The progression of Internet communications from BW (Before Web) to AW (After Web) marked by significant events

As managers of agencies that have to acquire these fantastic processing capabilities and user interface, what are the things we should look for/require and what should we wary of/avoid? The best way to answer this question is to deconstruct the WEB 2.0 experience and drill down to get some fundamental understanding of how this experience is achieved. Figure Two provides an implementation architecture model that we can use to further explore or drill down into what will make WEB 2.0 work.



Figure Two: WEB 2.0 Integrated Technologies

Browser Technology

The browser is the key user window to the web. Many people think that the browser and the web are the same thing, anyone who has been in information technology for any length of time clearly understands that the web is the interlinking of data in a hypertext fashion. The browser is the premier way to navigate these interlinked data points. The totally clueless believe the Internet is the web. Today pretty much the entire world experiences the web, or WEB 2.0, through a browser. If the “eyes are the window to the soul” then the browser is window to the web’s “soul.” As such the browser is at the very core of the WEB 2.0 user experience. At a very minimum, the generic browser

experience in a WEB 2.0 environment should provide support for a “rich Internet application” (RIA) environment. Browsers capable of providing RIA experience are what make the WEB 2.0 experience easier to accomplish and more fulfilling for the user. When using these types of browsers, response is faster (because many things that used to be handled by the server are addressed locally) and it gives the user more of the feeling of a desktop application rather than something that they are doing remotely. The disadvantages are that security may be more difficult to implement and maintain because so much is occurring on users’ computers. We will detail methodologies to address these types of concerns later in this paper.

The underlying technological implementations of these browser technologies are:

- Javascript (or its Microsoft equivalent)
- HTML Request Object
- XML/XHTML

The premier implementations of these browser capabilities are:

- AJAX
- Silverlight
- FLEX
- JavaFx

We have invested the time in laying out some of the facts relating to WEB 2.0 so that we can be precise with respect to what we’re dealing with. Precision in definitions is critical because many, potentially successful, projects have been derailed and achieved much less than their organizations needed because of an initial failure to understand what the underlying technology can deliver and what it cannot. A classic case is when people erroneously refer to a social network project as being WEB 2.0. This leads to many Government agencies believing that including terms like WEB 2.0 in their RFPs or specifications will bring them the benefits of social networking, unified messaging or other things that are closely identified with WEB 2.0 technologies.

In the most simplistic level, WEB 2.0 is a collection of next generation communication protocols, web tools and frame works that allow delivery of dynamic, rich, animated, real-time content and Graphical User Interface (GUI) functionality over the Internet.

Things To Consider When Acquiring WEB 2.0 Capabilities

When you are buying something, be it a car, golf clubs or a house, it is important to understand the parameters and variables that are important in making an informed decision. Suitability of any purchase or acquisition is as much a function of who buys it... as it is on the actual object itself. From the definition above it should be clear that while the term WEB 2.0 is often used to describe a fully functional platform that delivers some specific service, it is more akin to a toolset. From our earlier discussion it is clear that WEB 2.0 is not...

- Social networking
- Better search
- Unified messaging (instant tweets, etc.)
- User generated websites
- Blogs

...though WEB 2.0 techniques can be, and have been, used to accomplish all of the above.

WEB 2.0 is an underlying technological philosophy and capability that can help organizations accomplish all of the above and much more. WEB 2.0, as a technological philosophy, allows organizations to task technology implementers with a canon of methods, technologies, architectural design patterns and best practices that can transform the organization (particularly Government organizations) that uses it in a manner that can lower costs by reducing the friction of intra and inter agency information (intellectual capital) transfer.

User Considerations

It is important to determine the objective that we want to achieve with this technology that we call WEB 2.0. We must also consider the user population. There are five types of WEB 2.0 users:

1. Unaware – those rare individuals who have never heard of Facebook, Twitter, etc.
2. Interested – people who have heard about it, but never used it
3. First-time users – those folks who have used these types of technologies at least once
4. Regular users – those who have found some value in the technology and use it frequently
5. Passionate - typically geeks

In Joshua Porter's very fine book "Designing for The Social Web," he defines a usage lifecycle that we must keep very much in mind when determining and evaluating WEB 2.0 projects. The steps in the lifecycle are illustrated in the graphic below.



Figure Three: Usage Lifecycle

Failure to understand this life-cycle, reflect it in RFPs or evaluate a vendor’s strategy for delivering WEB 2.0 capabilities will result in a project that will have significantly more risk, cost and will probably not achieve the penetration or user acceptance the original business case projected.

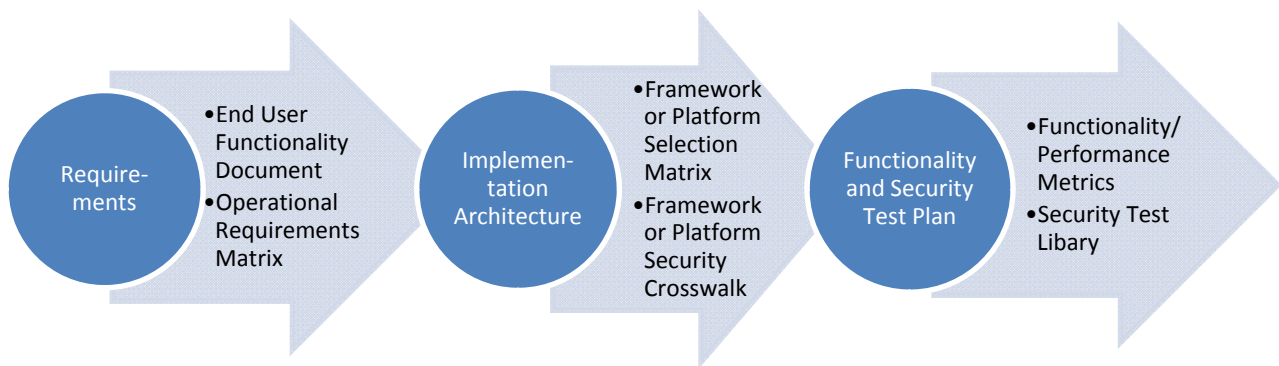


Figure Four: A recommended process for acquiring WEB 2.0 technologies

The user acceptance lifecycle must be a key consideration that should find its way into the Requirements documents.

WEB 2.0: Answering Social Needs

WEB 2.0 technologies should not be considered just for the technological pyrotechnics or to be fashionable. As the concepts of transformation and Netcentric warfare become more critical to operations, any tool that can serve as a collaboration multiplier needs to be harnessed. Earlier we saw that the human considerations of user adoption have to be considered at the beginning. Once considered we need to understand where WEB 2.0 can help.

WEB 2.0 is one of the most critical, facilitating technologies to enable a “N way conversation.” While it may appear that in bureaucracies (military or civilian) everything goes one way... from the leadership to the workers, modern technologies are “flattening” social intercourse so that “N way conversations” are becoming more of the norm. These conversations are not chaos. Whether it is Facebook, MySpace, Twitter, etc., each of these sites facilitates the accomplishment of the user’s requirement to collaborate. In the case of a military service or an agency, we need to be crystal clear in what user objectives we wish to prioritize. Those user objectives need to find themselves into solicitation statements of work and requirements documents. It is ill advised and possibly dangerous to permit solicitations or statements of work to be developed in a context which lets vendors think that the agency is looking for a “secure” or “classified” MySpace. Such a construction is idiotic and foolhardy.

The Phacil website has a series of spreadsheets and programs that can be used to assist in performing these steps (and their sub-steps), or analysis described above, in a disciplined and integrated fashion. The output of these spreadsheets and programs is a report that can be used as an initial requirements/ functionality planning document.

One of the most important aspects of the WEB 2.0 architectural model is the asynchronous processing capabilities that are built into the WEB 2.0 frame work. The traditional web application is based on request and response exchange based between client and servers. At each request, the display was completely re-rendered by the response from server. Not only is complete re-rendering a waste of resources, it is quite clunky and rigid in its abilities. WEB 2.0 allows asynchronous communications between client and server, which in turn allows dynamic and continual updating of a part of a screen. In addition to dynamic and more fluid information rendering, processes can be executed in the background and results rendered as they become available without a round trip data exchange between server and client. In fact different parts of a screen can be served and updated by different web services that adhere to WEB 2.0 system specifications.

WEB 2.0: Operations and Implementations

As mentioned earlier, WEB 2.0 is architected using the stack graphically depicted in Figure Two. From both an operations and implementations perspective, the bulk of the issues or concerns rise from the browser. This means that every successful WEB 2.0 implementation or execution requires that the leadership and organization developing it have a clear statement of what is required of the browser (key interface to the user) and have a disciplined methodology to assure compliance with the requirements. There are two major issues that differentiate WEB 2.0 enabling browser technologies from their predecessors. The first is that these new browser technologies have a larger resource footprint than earlier browsers and the second, much more important, distinction is that this larger software footprint results in a larger security vulnerability exposure if not implemented in a disciplined manner. From the first introduction of the browser, and their endemic potential vulnerabilities, there has existed a growing set of techniques used to exploit these vulnerabilities.

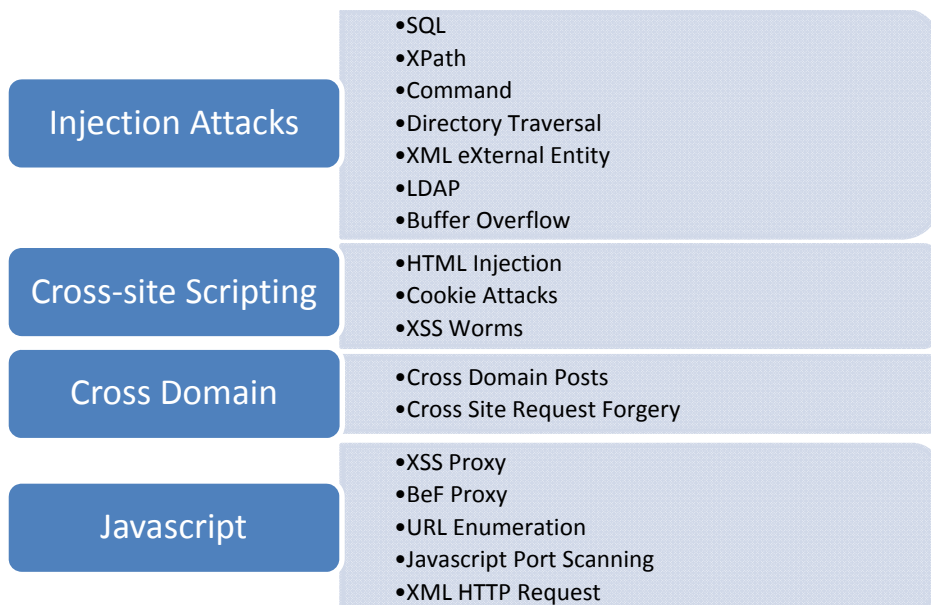


Figure Five: General Types of Attack Vectors

The larger footprint issue is typically addressed by better hardware. The security issues are addressed by a comprehensive security validation. Phacil's security review process for WEB 2.0 applications will be detailed in the next section.

Phacil WEB 2.0: Analysis and Verification Tools

Most popular WEB 2.0 based applications displayed in Figure Six are implemented using commonly available frameworks. These frameworks implement the rich internet application (RIA) and asynchronous javascript /XML (Ajax) features which users associate with WEB 2.0. As mentioned earlier, all of these frameworks increase the security footprint in that they provide the enhanced browser functionality by increasing the amount of operating code in the browser. Every line of code added provides a new opportunity for someone to exploit a coding error or force a malfunction that could compromise security.



Figure Six: Frameworks that Increase the Security Footprint

These frameworks increase security exposure most critically with respect to the various security models used in the browser and in the browser/server application interface. Most browser security models are based on limiting what an attacker can do. These frameworks provide rich tool sets that can be combined in ways that are may not be fully understood by site developers exposing new weaknesses. It is critical that when WEB 2.0 (or social networking projects based on WEB 2.0) is considered, the development frameworks are used in a manner that builds security organically during the development process. Examples of these frameworks are displayed below in Figure Seven. The diagram below uses the intersecting axis to depict sophistication from left to right and browser/server focus from top to bottom. Prototype indicates that it is principally a simple browser based framework and Sharepoint is a more complex server based framework.

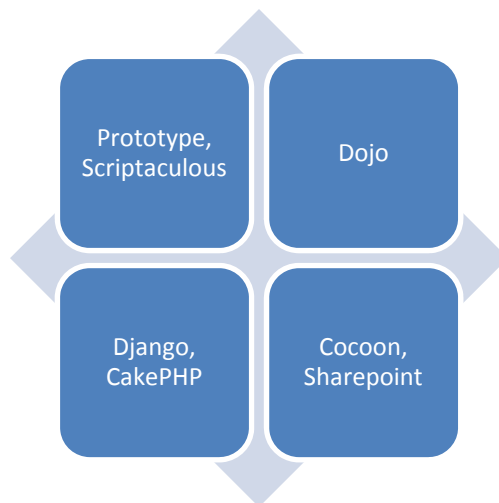


Figure Seven: Development Frameworks

No matter which quadrant they fall into, if applications are to be safe, at Phacil, we demand that each of the general types of attack vectors, outlined in Figure Five, are routinely checked during design, deployment and upgrading of any WEB 2.0 functionality. Phacil has developed a series of spreadsheets that can be used to inventory the signatures of the vectors detailed in Figure Five and they can be downloaded from the Phacil site.

Conclusion

Applications based on WEB 2.0 principles are not a fad and will increasingly become the norm both in and outside of the Federal Government environment. Users will continue to expect a similar experience from their web applications they are accustomed to receiving from their desktop application. IT professionals' success will be measured based on consumer satisfaction on the similarity of the two experiences. From the training and efficiency perspective, user familiarity with the interface of applications will increase productivity and lower costs in providing Government services – whether these services are supervising ballistic missile tests or sharing Homeland Security intelligence information.

Social networking, blogs, RSS feeds and the entire WEB 2.0 universe can be made available safely... if we understand what it is, what it can do and the nature of the risks. WEB 2.0 permits us to provide our military branches and federal agencies enhanced collaboration capabilities by bringing increased knowledge management tools to their desktops over the Internet. As senior leadership carefully considers WEB 2.0, the information contained in this paper should provide the context from which a healthy dialog with vendors, users and other stakeholders can begin.