

5

Section Five

Implementing web2.0 in the Enterprise

Transforming enterprises through introduction of web2.0 technologies into "Enterprise2.0 corporations" offers great potential to achieve a much needed boost in corporate productivity. Successful implementations leverage existing SOA efforts, tackle challenges in technology, culture and governance and avoid pitfalls by embracing open standards. The article demystifies web2.0 terminology, providing insights into concepts, processes and components of enterprise grade web2.0 adoption including key points for successful implementation planning.

“**Stephan H Wissel**
Lotus Technology and
Productivity Advisor, Asia Pacific
IBM Singapore”

Since Tim O'Reilly coined the term web2.0 it has been much talked about, decried as hype and spawned endless discussions about its sense or lack of. The 2.0 meme now can be spotted on almost any initiative (short of cloud 2.0). This article will shed some light on implementation of web2.0 in the enterprise.

1 WHAT IS WEB2.0?

Before one can evaluate the challenge that a web2.0 adoption brings with it, clarity is needed on what exactly is web2.0. The fact that it is a very much overused term doesn't simplify this task, since web2.0 means different things to different people:

- web2.0 is the transition from a publish and consume static internet to an everybody-is-a-creator interactive collaborative application. Key components here are: blogs and wikis.
- web2.0 is social computing, where participants link to each other, form communities and establish common interactions and trust systems. Key components here are: MySpace, Facebook, LinkedIn and Twitter.
- web2.0 is the ability to instantly link and rate content and notify participants about this. Key components are: StumbleUpon, IdeaJam, Delicious or Digg.
- web2.0 is the ability to tap into global expertise to get opinions, advice and insights. Key components are: Answers.com, Yahoo Answers, IQJam or StackOverflow.

- web2.0 is Mashups, RSS, Widgets, situational applications. Key components are: Google Maps, Yahoo Pipes, Google App Engine, OpenAjax or OpenSocial.
- web2.0 is a set of development paradigms like REST, AJAX, RDF or JSON.

Looking at all of the above interpretations, the following common properties can be extracted:

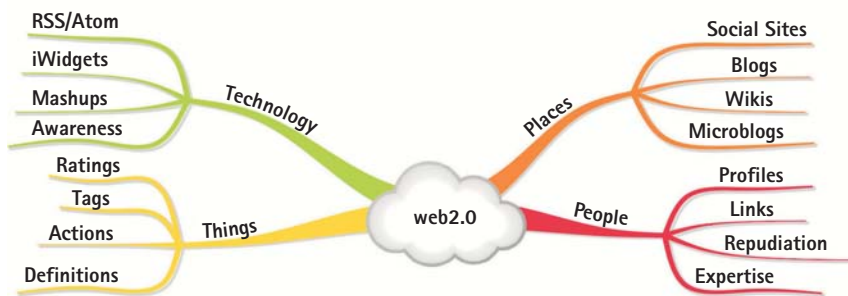


Figure 1: web2.0

2 FROM WEB2.0 TO ENTERPRISE2.0

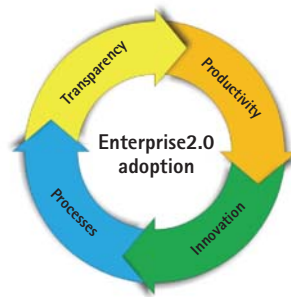


Figure 2: Enterprise2.0 Adoption

Using web2.0 in an enterprise setting complete with new management and go-to-market strategies is commonly called Enterprise2.0. While this sounds just like another buzz word it has real use cases revolving around a number of core needs - improve transparency, accelerate innovation, enhance productivity and streamline business processes:

- **Transparency:** With blog entries, chat logs, discussion forums, rating systems, online activities and micro blogging, information exchange becomes more open and accessible, so more people can stay informed and contribute. Social profiles allow to demonstrate and find expertise.

- **Innovation:** Working in a web2.0 style is already an innovation by itself. Making content more accessible and connectible works almost like brain cells blasting new neural pathways to make room for new thoughts. Without the capability of cross-pollination.
- **Productivity:** Relevant information can be presented in the context where it is needed instead of being buried in the eMail archives or stowed away in disconnected systems. Widgets and situational applications bring together what IT backend systems had separated.
- **Processes:** All businesses are executed using business processes. Advanced corporations have made their processes explicit, documented, standardised and supported by tools like ERP or CRM. However these IT supported processes are neither perfect nor complete. Millions of internal emails sent daily are a living proof for that (If a supporting system would be complete or perfect, there wouldn't be a need for additional communication).

Process clarifications happen outside the supporting system: a chat happens, an email is sent or a conversation in person happens. The outside interaction is not documented in the context of the process or transaction, so decisions and reason become opaque for later reviewers.

The question: "why did we do it this way" stays unanswered or at least requires a search in the email or chat logs if they are still available. When integrating web2.0 in Enterprise process management systems, by using widgets, profiles and mashups, then clarification communication, exception handling and process improvements (Kaizen anyone) are not only available when and where a user needs them (no switching back and forth to the email system) but also are stored in the context of the process or transaction that triggered the collaboration. This makes processes both more efficient and transparent, so actions stay conclusive, satisfying auditors and serving as learning results for staff trying to pick-up new roles by looking at their predecessors work artifacts.

3 KEY PROPERTIES OF WEB2.0 IN THE ENTERPRISE

Dion Hinchcliffe coined the acronym "flatness" (based on earlier work of Andrew McAfee, Associate Professor, Harvard Business School) to sum up 10 key properties of Enterprise2.0 (explanations below by the author):

- **Freeform:** No or no high barriers to authorship. Authoring content must be easy to learn and must not be restricted.
- **Links:** Similar users, content, processes can be grouped together. Grouping can be deliberate (as in forming a community), semi-automatic (whereby users tag entries with the same keywords) or automatic (based on system functions).
- **Authorship:** Every user can author information, every piece of information has an author.

- **Tagging:** Allow every user to tag any content. A user or content can be tagged by anybody with multiple tags. This is different from the rigid structure of a corporate taxonomy that mandates single attribution.
- **Network-Oriented:** All content can be uniquely addressed on the network. This means an end to "local only content" and requires sound and working synchronisation strategies.
- **Extensions:** Based on profiles or past actions, content and users are recommended to each other.
- **Search:** Allowing users to search across the entire information set for users, information or connections. Harvesting information in closed or confidential systems becomes a mandate and a challenge.
- **Social:** Systems need to provide easy access (transparency), allow for diversity in content and community members as well as being open in their structure.
- **Emergence:** Provide systems that can detect, notify and leverage trends in the living network, commonly referred to as "the collective wisdom of the community".
- **Signals:** Allowing people to subscribe to users or content with their preferred notification mechanism like RSS/Atom feeds, Micro blogging alerts, chat notification or POE (plain old email).

4 HOW DOES WEB2.0 FIT INTO A SOA STRATEGY?

SOA started as an initiative to improve module reuse and agility in the enterprise. Its core concept of loosely coupled components by using open standards to communicate is coincidentally a key requirement of a successful implementation of fully integrated web2.0 applications.

The typical sequence in a web2.0 implementation in an enterprise is the introduction of social computing like profiles, bookmarks and blogs only to realise that stand-alone they do improve collaboration but not core business processes. When an organisation reaches this insight one of two things happen:

- Without a working SOA strategy the organisation gets stuck, applies a patchwork of integration points or gives up in their efforts to improve their productivity beyond the basic gains.
- With a working SOA strategy, core applications can be integrated seamlessly with social computing components to take application productivity to a new level. The openness and loose coupling that SOA advocates becomes key to enabling the process and social computing spheres in the organisation.

In a nutshell: SOA + Social Computing components = web2.0 in the enterprise.

5 THE IMPORTANCE OF STANDARDS

Combining data and information wouldn't be possible without clearly defined standards for the exchange of information. The smallest common denominator in today's IT world is XML. Unfortunately there is little agreement on the various formats that should be used when it comes to the social computing parts of web2.0.

One of the reasons for this Babel of format definitions is the heritage of social computing that originates from the private use of social computing in the internet where interoperability typically had been seen as dangerous and undesirable. While the big social computing platforms by now all offer a respective API to their platform, interoperability is still a challenge. Luckily one already can see a number of standard formats evolving:

- **XML ATOM:** ATOM was designed as format to address the shortcomings of the RSS format used in news or feed readers. ATOM has three key properties that put it at the core of data exchange; firstly it can be presented using well established technologies (feed readers); secondly it can be extended to carry additional information; and last but certainly not least ATOM can be used to post back information, so one format is suitable for read and write operations.
- **RDF:** RDF is a long established standard promoted by the w3 as a core format for the semantic web. RDF allows meaningful description of resources and their attributes, to enable automated discovery and processing. The most commonly used part of RDF is the Dublin Core that describes attributes such as author, location, editors etc.
- **JSON:** A rapidly evolving content delivery standard competing with XML. JSON delivers similar capabilities to basic XML but with less overhead (no closing tags) and better integration into browser based computing (JavaScript).
- **iWidgets:** iWidgets are the rapidly evolving standard for UI (also called glass) level integration of content and information. iWidgets are interactive. They not only retrieve and display information from a web2.0 component, they also can send user actions back to it. A hallmark of iWidgets is that these information are not only available in a user interface but in a standardised API as properties and actions, that can be wired up. The output of one set of iWidgets serves as input to another (set of) iWidget(s). iWidgets essentially serve a similar purpose as portlets (JSR 168/268) but breaking free from a specific requirement regarding the platform they run on. One could compare the transition from portlets to iWidgets to the transition from Java's RMI to the platform neutral web services.

When implementing web2.0 one can see two approaches that are used together to deliver systems: back-end integration based on XML (that is where SOA comes to play) and front-end implementation using JavaScript in a browser or similar environment. One main reason is speed. All browsers process data that is delivered in JavaScript's JSON format much faster since there is no conversion step from XML to JavaScript in the process. There is no hard separation between these approaches, there are back-end systems processing JSON as well as front-end systems processing XML.

6 IMPLEMENTATION CHALLENGES

Implementing Enterprise2.0 comes with a set of challenges that require close attention. Even more than the arrival of email in the enterprise, web2.0 is a culture changer. Addressing a web2.0 implementation from a technology perspective alone is a guarantee for failure.

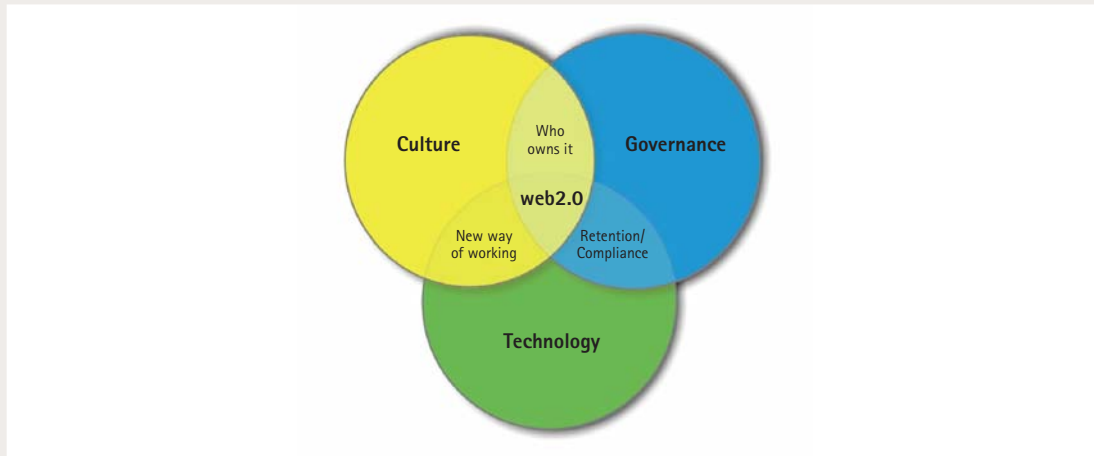


Figure 3: Implementation Challenges

Three aspects of a web2.0 implementation must be carefully balanced. In the authors opinion the technical challenges are the easiest to overcome. The cultural changes might be the hardest to master. The most elusive challenges are governance issues.

- **Cultural challenges:** web2.0 use wasn't driven by organisations or corporate innovators. Senior management might feel threatened by this bottom-up movement and resort to denial for the necessity to adopt. A possible remedy is to make senior management the sponsor of a corporate web2.0 initiative. Participation in a web2.0 environment has been likened to the self-realisation stage of Maslof's pyramid of human needs. For readers who are familiar with this pyramid it is obvious that working at this level requires the lower levels (physical security, sense of belonging) to be given. A corporate environment might be too competitive (lack of belonging and security: everybody is my competitor/enemy, keeping knowledge is a matter of survival) or too insecure (why should I contribute if my job is on the line) or a cultural mis-fit to successfully roll-out web2.0 applications. Providing remedies for this challenge exceeds the scope of this article. For these organisations web2.0 might prove similarly fatal as the meteor was to the dinosaur population.
- **Governance challenges:** web2.0 marks the departure from key concepts of information management: tight control, single point of ownership, closed boundaries and "internal use only". As corporate information systems are gradually opened up to allow partners, suppliers, stake holders and customers to interact with each other the corporate communications loses its monopoly of corporate information management. Their

role changes from gatekeeper to moderator and mediator of trans-corporate collaboration. Clear rules must be defined, communicated, policed and enforced as to how this collaboration is acceptable and in line with corporate goals and values.

A fully deployed web2.0 system stores information in many places: the blogs, the tagging database, the repudiation system. So questions need to be addressed: what happens to the tags of an entry that has reached its lifespan according to enterprise information retention rules?

What is part of a document? Are the tags and comments part of it? The links that point to this document? How should information updates be handled? Do the 140 characters of a microblog entry constitute a document that is subject to enterprise retention rules or are the entries of one day/one month/one year such a document?

Who owns the comments an employee has entered to any system. Can she delete them without impunity? What happens to blog entries, tags, comments etc. when employees change roles or leave the enterprise? How does information ownership transit in such situation? A blog entry about the impact of the new leave regulation surely stays personal, but the ownership of the HR employee manual must transit to the new HR person.

- **Technical challenges:** Implementing web2.0 on a blank slate is an easy exercise: Pick your platform, roll it out. However to maximise the effectiveness and efficiency of a web2.0 platform in the enterprise, integration with existing systems is key. The more legacy applications an organisation is required to maintain, the more difficult it will be to implement web2.0. A working SOA strategy makes implementation much easier.

A common misconception in technology is that web2.0 means everything must be based on access via a browser. While it is true that the browser is the key platform one can see the emergence of a large array of very specific applications (e.g. a chat client, a feed reader or a plug-in for the address book) for specific tasks. A commonality of these specialised applications is that they are fully network aware. More and more these applications are built on specialised network aware, offline capable frameworks, which allow the extension of these applications to mobile devices. Key examples are Adobe's Flex, Sun's JavaFX, Microsoft's Silverlight and IBM's Lotus Expeditor.

- **Mobility challenge:** Workforce expectations shift from "working in the office" to "work anywhere, anytime with any device". Since this shift happens with different velocity it poses both a technical and cultural challenge. Senior management might not anticipate or value the mobile working style the Internet generation is expecting, posing an additional risk for corporate efficiency and talent retention. On the technology side enterprise application architects need to define strategies to accommodate the different working styles tackling physical problems like small screens on mobile devices, limited connectivity (mostly online, but not always) and security. Web2.0 concept of "widedisation" offers a possible solution path: On a mobile device (which today has only 5% of the screen real estate that a desktop computer provides) one widget at a time is shown, while on a desktop all that snippets of information provided by the widgets would be arranged into one concise display.

- **Software development challenges:** Corporate software development is subject to all three challenge areas. Composite application that blend traditional application and web2.0 elements need a different governance and development lifecycle. Many web2.0 components use their own storage mechanism and data schemas (like tagging or enterprise action management). Now both the developers of the application and the developers of the components need to orchestrate their release cycles and provide interfaces that are both forward and backward compatible. Again practitioners of SOA will be in familiar territory. Similar to an SOA approach a corporate application developers need to raise their awareness what components are available or desirable in order not to reinvent the wheel too often.

7 EXAMPLES OF ENTERPRISE2.0 APPLICATIONS

Web2.0 applications in the enterprise range from the use of plain vanilla web2.0 applications (like blogs or wikis) for corporate purposes to web2.0 enhanced classical applications. The following are examples of how such applications are used in the real world (the names of the organisations have been removed to protect the innocent).

- **Corporate Blogs:** All C-Level executives use a blog to share their thoughts and insights about the development of the organisation. The blogs have effectively replaced the internal corporate newsletters. HR announcements are published in a blog rather than delivered through an eMail. The blog comment function established an effective back-channel to form a closer relationship between management and employees.
- **Project Teams:** Standard blogs were used to document the progress of various corporate projects. Not only did it allow shorter team meetings in which everybody was up to speed, but it also allowed the sharing of key insights between project teams that were facing similar challenges in other diverse projects.
- **Tag enhanced CRM:** Tagging capability was added to a CRM system. Sales reps now find proposals with similar tags in the system, thus shortening the time required to propose a solution to a customer.
- **Chat enabled radiological application:** X-ray images are captured in a specialised clinical application. Physicians in different locations display these images using a client application (built on Lotus Expeditor). While examining the images the physicians are connected to peers in different locations. They look at the same image and can use screen annotation, voice, video and text chat to establish a joint diagnosis faster reducing cost and saving lives. Attention can be directed to specific areas of an X-Ray, so obtaining a second opinion about a medical finding is instantaneous. All this collaboration is documented, so it serves as prove for due process and as learning resource for others.
- Disparate data sources about weather, airstrip capabilities, aircraft specifications, warehouse content, local transportation availability are combined into a situational application of a disaster relieve agency thus enabling them to react more quickly in an emergency.

8 HOW TO GET STARTED

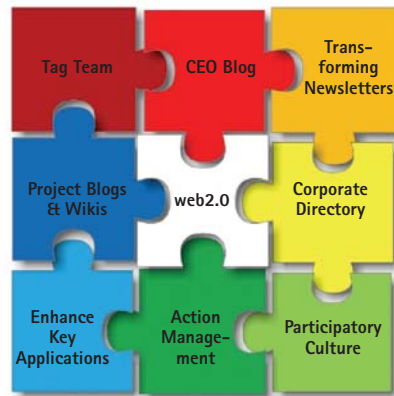


Figure 4: An example of an implementation approach

The following guidelines are based on the author's personal experiences. Based on the situation of any given enterprise an implementation approach will differ either slightly or radically. One must keep in mind that a best practice is "What worked for others in their context of use".

- 1) Assemble a corporate tag team consisting of a senior management sponsor, corporate communications, the IT team and employees that have an existing affinity to web2.0. Look for bloggers and staff with a Facebook or LinkedIn profile.
- 2) Get the CEO to start a blog. It is generally considered to be acceptable to get a ghost writer for the CEO if they are too busy.
- 3) Move all departments that send out regular news letters from newsletters to blogs. Allow a transition period where the email newsletter contains a link to the blog until it ceases.
- 4) Introduce blogs & wikis to those teams in the organisation who are involved in project work.
- 5) Create or enhance your corporate directory to allow personal information and tagging. Populate and synchronise it with your HR systems.
- 6) Make tagging, rating, comments and presence awareness available to key corporate applications.

7) Provide web2.0 software that allows collaboration about actions and activities to be completed, so web2.0 becomes the corporate synonym of "Getting things done". This is a first step towards the emerging challenges of "Enterprise Attention & Enterprise Action Management (EAAM)" EAAM will be a topic for future discussion.

8) Allow a participatory culture. Invite employees to join, don't force them.

Following these steps will give your web2.0 initiative enough momentum to become a self runner spreading fast in the organisation and beyond. Of course, your mileage might vary.

9 IMPLEMENTATION WITHOUT STANDARDS

None of the getting started tips or challenges mentioned above seems, on first view, to mandate the use of any web2.0 related standards.

However that is far from reality. The early consumer driven web2.0 systems like MySpace, LinkedIn or Facebook teach a valuable lesson: It is still almost impossible to move or integrate data from these platforms with each other. Doing this requires substantial efforts accommodating different data models, APIs and integration philosophies. The lack of standards here is based on the fact, that this is a relatively young branch of online offerings, in the still young internet industry, where building a platform and gaining market share made standard consideration an afterthought or even an perceived menace in the quest of locking customers into the "walled garden" of a social networking offering.

All these developments are fully contrary to enterprise IT needs. The ability to re-use, connect and link data from different systems is a core requirement for corporate IT success. Whenever an enterprise gets locked into the propriety standards of a single vendor, the enterprise becomes hostage to the vendor's product policy and upgrade cycles. Adherence to Open Standards work as a vaccine against the "being-locked-in" disease.

Enterprises that want to use web2.0 systems not only internally, but to reach out to customers, suppliers, stakeholders, partners, government and even competitors can lower their vendor dependency, expenditure, effort and risks when using established Open Standards and protocols. The industry at large is learning the lesson - at different pace - that offering standards based products is a competitive advantage in a trustful customer relationship. After all it is easier to sell to a customer who is ones customer because they want to, rather than because they have no choice.

Standards are furthermore a key component in the ongoing effort of improving employee productivity. E.g. A single standard for blog contributions will allow an employee to contribute to internal and external blogs using a single interface, thus reducing training cost and individual effort. One standard for short text communication (like tweets and SMS) makes this communication channels more useful and credible.

10 EMERGING STANDARDS – ACTIONS NEEDED

Web2.0 is rapidly evolving, reaching a critical mass in both Internet and corporate adoption. The nature of standardisation processes doesn't accommodate rapidly evolving technologies, so web2.0 is still lacking in accepted standards. This work in progress opens the opportunity for active participation.

Standardisation is required in two core areas: protocols and data formats. Email only became globally useful once the industry settled on SMTP as transport standard. The world wide web became useful with the HTTP standard.

For web2.0 to achieve its full potential additional protocols need to be standardised and adopted: real time signalling (a.k.a. chat), synchronisation and rich media communication (e.g. Video or composite documents). These areas are currently dominated by vendor propriety protocols that lack interoperability. Today a user of a 3G phone can't participate in a video chat with users of popular Internet or corporate based chat systems. The standards don't fit. Strong participation of interested parties is needed to resolve the lack of standards in this area.

Standards for data look more promising. Especially the work around RDF and micro formats by the w3 point into the right direction. Participating in these working groups will help to gain the knowledge and insights for a faster corporate web2.0 adoption.

There are more standard initiative worth mentioning: Since web2.0 relies on consumption of information widgets a common API for these widget is desirable. While some platforms like IBM's upcoming Websphere Portal 6.5 can consume widgets from different vendors, having only one API to worry about will ease the plumbing web2.0 adoption. The OpenAjax Alliance (<http://www.openajax.org/>) aims to provide this interoperability. With industry heavyweights like Adobe, Cisco, IBM, Microsoft, Novell, Oracle, Redhat and SAP as well as framework providers like the Eclipse foundation, the Dojo foundation and the jQuery team in its member ranks the alliance looks very promising. The current member list is heavily biased towards technology providers. More participation by customer organisations and institutions, from both the private and public sector is needed.

One step further, addressing not the "how" but the "what" of web2.0 is the goal of the OpenSocial foundation (<http://www.opensocial.org/>). The initiative was originally spearheaded by Google and critics cite Google Orkut's lack of success compared to MySpace and Facebook as prime motivator for the launch of the initiative. By now all large public facing social sites, with the exception of Facebook, are supporting this initiative. While the initiative clearly targets the public social networks any enterprise on a web2.0 trajectory will face the need of integration into these platforms sooner later. Participating early in the foundation's activities and getting acquaintance to their reference implementation (code name Shindig) will prevent development delays once the corporate web2.0 is ready to fuse with Internet2.0.

BIOGRAPHY OF AUTHOR



Stephan H Wissel

Lotus Technology and Productivity Advisor, Asia Pacific
IBM Singapore

Stephan Wissel has more than 20 years of experience in the leading edge of collaboration. He works for IBM Singapore as Lotus Technology and Productivity Advisor for Asia Pacific. He is an active blogger (<http://www.wissel.net>) and can be reached via Twitter ([notessensei](#)) or Facebook (and still email: notessensei@sg.ibm.com).