

# Bridging the gap between Web 2.0 Technologies and Social Computing Principles

Giorgos Kormaris<sup>1</sup>, Marco Spruit<sup>1</sup>

<sup>1</sup> Department of Information & Computer Sciences, Utrecht University  
{gkormari, m.r.spruit}@cs.uu.nl

**Abstract.** This research presents a brief review of the different definitions of Web 2.0 and presents the most important Web 2.0 Technologies that underlie the evolution of the Web. We map these Web 2.0 technologies to the Social Computing Principles and describe the different relations and patterns that occur. We argue that creating insight into the relations between Web 2.0 Technologies and Principles will help enable the creation of more successful services and accommodate a better understanding of Web 2.0 and its social aspects.

**Keywords:** *Web 2.0 technologies, Web 2.0 applications, Social Computing Principles*

## 1. Introduction: the theory-technology gap in Web 2.0

Web 2.0 is a term that is nowadays broadly used; people perceive it in many different ways. There are various definitions and opinions about the true meaning of Web 2.0, what it includes and if it has been given a proper title. Despite all this discussion and research, there is still a gap between the theoretical view of the new type of services that have risen along with Web 2.0 and with the actual technologies or applications which are used to build, implement and offer these services.

This research attempts to bridge the gap that exists between theory and practice in the field of Web 2.0, also referred to as Social Computing in [1] and the internet services which are provided it.

Firstly, in *Related Literature* we briefly present the existing literature regarding Web 2.0, the different definitions that have been conceived by scholars or technology experts and present in more detail the technologies that recently emerged and enabled this change to happen will be presented and discussed, describing the ways that each technology contributes to the new social view of the web. In the same section, we present the Web 2.0 Principles, as defined in [1] and map these principles with corresponding technologies.

By doing so, in Section 3 we are going to be able to propose certain guidelines and take a more structured approach towards creating interactive Web 2.0 services. Providing such a structure, will make it easier to define the architecture of their service enable, for one that wishes to provide services through the internet.

Finally, in Section 4, we are going to draw certain conclusions about the facts that have been presented. We will also provide opportunities for further developments and how the present research could evolve in the future.

## **2. Related Literature on Web 2.0**

In this section we present the literature that has served as the basis of our research and that lead us towards the effort of combining theories and technologies of Web 2.0, in order to create a new relationship between the technologies and the principles of Web 2.0, also called Social Computing.

### **2.1 Web 2.0: How can it be defined?**

Ever since the term '*Web 2.0*' was officially coined in 2004 by Dale Dougherty, a vice-president of O'Reilly Media Inc., during a team discussion on a potential future conference about the Web, everyone was in the Web 2.0 hype [2].

Specifically, O'Reilly firstly describes the new trend for the web, by using the phrase "*The Web as a platform*" and by this he means that there are no hard boundaries, but rather a set of principles which outline the area of Web 2.0. If a certain company claims to be "Web 2.0 oriented", then it has to be along these principles, which are translated into core competencies for Web2.0 companies.

Since O'Reilly was the first to talk about a new kind of World Wide Web, a large debate broke out; some were trying to define Web 2.0 and others to deny there is actually such a major difference, after the big web crash during 1999-2000, [3].

Another analyst that approached the subject is Andrew McAfee, who did agree with the overall definition of having a general boundary around what is called Web 2.0 and characteristically says:

*"...most current platforms, such as knowledge management systems, information portals, intranets and workflow applications, are highly structured from the start, and users have little opportunity to influence this structure."*, [4].

As quoted above, it is preferable to create a set of ground rules that serve as a basis for new applications and tools to be created, which will promote the exchange and emergence of new knowledge.

Yet another trend analyst described the same general concept as McAfee and Sir Tim Berners Lee [5]; it was Hinchcliffe, who proposed that this platform should be open and that it should not only be about the Internet. It should include all the connected devices, such as mobile telephones and smartphones, as well as "*rich and interactive user interfaces*" [6].

### **2.2 Web 2.0 Technologies**

We saw that Web 2.0 has many different definitions, by many types of experts but where it all goes down to is creating a unified platform that can be used to connect everyone and all the related devices, fast and safely.

In addition to speed and safety, it is equally important to have a friendly, easy-to-use interface, combining usability and functionality in one. We have seen many

different services emerge during the 00s, such as *Flickr, Facebook, Twitter, Gmail* and many others that have tried to combine all these different elements in their service.

All these services became a reality by having new technologies and standards “*under the hood*”, which enable web developers to create innovative applications that can be used to distribute, share and create information in new ways.

In this section, we are going to discuss the following technologies in detail and mention the reasons why they played such an integral role in the evolution of the World Wide Web. We categorize them according to the OSI 7-layer model: *AJAX, SOAP and REST, Adobe Flash, Flex and AIR, Open APIs and Mashups, RSS Feeds, Microformats and Semantics.*

**AJAX.** The development model of AJAX is probably the one that revolutionized the way web applications and services are delivered, a term which is accredited to Jesse James Garrett and stands for Asynchronous JavaScript + XML, [7].

Ajax is not a new web-based programming language, but it’s a group of technologies combined together, having as a base the Ajax engine, creating a new experience for users and their interaction with web applications. The technologies that it incorporates are already mature, stable, and popular web-based programming languages and script languages [7], [8] and include *(X)HTML, CSS, XML, XSLT, JSON, DOM, XMLHttpRequest, Javascript, VBScript, Adobe’s Flash, Flex and AIR and Microsoft’s Silverlight.*

- **HTML - XHTML & CSS:** *Hyper Text Markup Language*, its extended version XHTML and Cascading Style Sheets are used to change the format and display of the data of web pages, both being thoroughly tested, popular and standardized according to World Wide Web Consortium (W3C).
- **XML & XSLT – JSON:** *EXtensible Markup Language* is a self-descriptive language that was designed to carry data and gives developers the ability to define their own tags. XSLT a way to transform XML documents that essentially change the way data are displayed.
- **JavaScript Object Notation** is an alternate of XML and “*it is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language*”, Ecma International, [9]
- **DOM:** *Document Object Model* is cross-platform and language-independent way of dynamically controlling objects and to describe the data of an HTML, XHTML and XML documents.
- **XMLHttpRequest:** First implemented by Microsoft as an ActiveX object but now also available as a native object within both Mozilla and Apple's Safari browser, enables JavaScript to make HTTP requests to a remote server without the need to reload the page. In essence, HTTP requests can be made and responses received, completely in the background and without the user experiencing any visual interruptions, [10].
- **JavaScript – VBScript:** *JavaScript* was created by Brendan Eich of Netscape and it is an open, cross-platform object scripting language for the creation and customization of applications on enterprise networks and the Internet, [11]. It is mainly used to bring all the aforementioned technologies together.

*VBScript* a scripting language quite similar to JavaScript and it is an Active Scripting language, developed by Microsoft, which uses the Component Object Model to access elements of the environment within which it is running, [12].

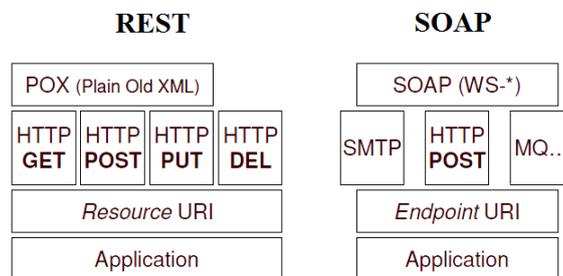
Where Ajax is completely different from the classic web application model, is its asynchronous way of client-server communication, putting a stop to the “start-stop-start-stop nature of interaction”, [7]. The Ajax engine is responsible for the communication and data exchange with the server, which hidden from the user, but also renders the interface that users interact with.

This kind of architecture is widespread and used in a variety of applications which can be very simple, like a small website or even complex services, such as Google Maps. Finally, it has also become quite popular to be used in combination with the *Ruby on Rails* web development framework, in an effort to use Agile Development methodology in web-based projects.

**SOAP & REST Architectures.** Since we discussed the technologies that are used to implement the applications and services that have become known as Web 2.0, we also have to mention the architectures that are used, in order to plan the development of these services.

The choice is always up to the hands of the developer, but as quoted by Anderson in [8], Sean McGrath describes the Web as an enormous information space, littered with nouns (that can be located with URIs) and a small number of verbs (GET, POST etc). Where SOAP is more of a Verb Noun system he argues that SOAP allows the creation of too many (irregular) verbs.

We will not go into much detail, since this topic is rather deep and the subject of a different research, but what is worth mentioning is the dispute between these two schools.



**Fig. 1.** The building blocks of the two different approaches on Web service architectures [13].

*SOAP* is the traditional, standards-based approach; it was developed by Microsoft in 1998 and since then has become the most popular standard in exchanging XML-based messages between applications, [8]. It has become so popular, because it was the first architecture that enabled the usage of new technologies like AJAX and being introduced by a major corporation as Microsoft is, it was bound to become a success. From a technical point of view, SOAP simply is “A protocol ‘framework’, to deliver the necessary interoperability between message-based middleware tools across the entire industry”, [13].

*REST* was developed by Roy Fielding and is the conceptually simpler “*trendy new kid on the block*” and provides a simple communications interface using XML and HTTP, using simple commands, such as *POST*, *GET* and *PUT*, [14]. It relies on simplicity, ubiquity and scalability, since it can support small, simple services and complex services, offered by large service providers, such as Amazon and Google [13].

**Adobe Flash - Adobe Flex - AIR & Microsoft Silverlight.** *Adobe Flash* is a multimedia platform, which is used to create Rich Internet Applications (RIA), giving developers the ability to include animation, interactive graphics and other options to webpages, without considerably slowing down the loading of webpages.

*Adobe Flex* is package for developing such applications and offers a separate IDE for developers to create their RIAs based on the Flash platform. *AIR* is another development of Adobe, which is a cross-operating system runtime that enables you to use your existing HTML/Ajax, Flex, or Flash web development skills and tools to build and deploy rich Internet applications to the desktop.

*Silverlight* was developed by Microsoft and was officially released in 2007, as an alternative way to create multimedia applications for the web. It is “a cross-browser, cross-platform and cross-device browser plug-in that helps companies design, develop and deliver applications and experiences on the Web”, [15].

**Open APIs & Mashups.** According to [6], an Application Programming Interface (API) provides a mechanism for programmers to make use of the functionality of a set of modules without having access to the source code. The addition of *Open* is used when these APIs are free and open for all programmers to use and to take advantage of certain features in their own projects.

We have seen big social networks and major web service providers giving out APIs for developers, most notably *Facebook* and the *Facebook Apps API*, *Flickr* and *Google* with various APIs, with the likes of the *Google Maps API* being rather popular.

This is a growing trend and has become quite popular during the Web 2.0 era, something that was augmented by the growing number of mashups that we have encountered during the past years. Simply put a mashup “...is a customizable application that takes seemingly disparate data sets - both static and real-time - and integrates them to create a new data set”, [16].

To do this easily and successfully, the use of Open APIs has become a *must* amongst developers and this has led to the creation of the alliances such is the “*Open MashUp*” alliance ([www.openmashup.org](http://www.openmashup.org)), having support of major companies such as HP, Intel, Cap Gemini and the *–under development–* project lead by Google, called “Open Social” ([www.opensocial.org](http://www.opensocial.org)), in cooperation with social service providers such as MySpace, Hi5, XING and others.

**RSS feed, Microformats & Semantics.** Since the social aspect of Web 2.0 is rather crucial, we have seen many technologies that give users the opportunity to have personalized information right on their desktop or mobile phone (RSS), personalize their accounts at websites (Microformats) and even to be able to get information specifically for them, based on their interests or previous browsing (Semantics).

According to specifications of *RSS 2.0*, “RSS is a Web content syndication format, its name is an acronym for Really Simple Syndication RSS is a dialect of XML and all RSS files must conform to the XML 1.0 specification, as published on the World

Wide Web Consortium (W3C) website”. As mentioned what RSS does, is give users the opportunity to receive information they have selected as interesting and to be instantly notified about new updates on their favorite subjects, from their preferred blogs or websites.

*Microformats* could be placed on step above RSS, since it’s a technology that provides developers with a way of adding simple markup to human-readable data items such as events, contact details or locations, on web pages, so that the information in them can be extracted by software and indexed, searched for, saved, cross-referenced or combined.

*Semantics* have been and still are a topic of broad discussions, since many analysts and scientists mention another web, called the *Semantic Web*, where semantics play the key role.

Despite this view, since Web 2.0 has become the main point of reference, another trend has emerged; it is broadly discussed in scientific research, [17], [18], that what seems more logical than is having a unified web, where the social aspect of the web, commonly named as Web 2.0 is one and the same with the Semantic Web, which is used to augment the personalized aspect of the web and provide valuable data to developers and/or businesses.

Although some have second thoughts that such an attempt would be successful saying that “...I’ve adopted a cautious outlook toward the prospects of a marriage between *Semantic Web technology and Web 2.0*” [17], others seem to be more optimistic and state that “...there is growing realization that the two ideas complement each other and that in fact both communities need elements from the other’s technologies to overcome their own limitations”, [18].

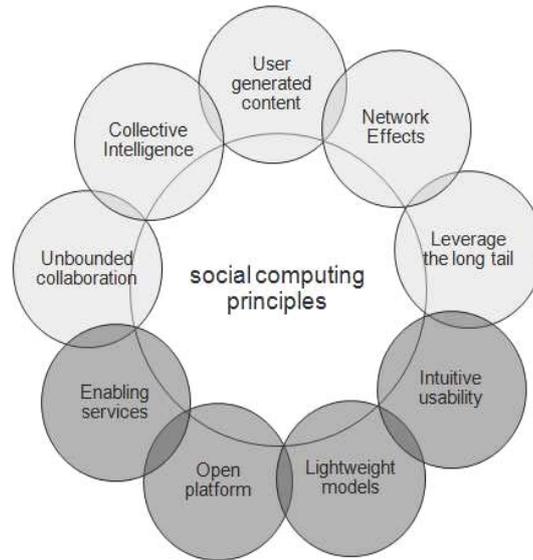
### **2.3 Social Computing and its Principles**

In the previous section we presented the different technologies that have served as the base of the evolution of the World Wide Web, into what is called Web 2.0. Some tend to disagree with the name that has been given to these new services, which have changed the landscape of the web. This new effort to re-define the state of the web has moved towards the term Social Computing, which has been the new center of discussion amongst internet experts, scientists and trend analysts.

According to a thorough investigation that was conducted in [1], all the different definitions that have emerged during the past years were examined and created a new definition which combines all these different opinions by experts like O’Reilly and Hinchcliffe: “...*Social Computing refers to a development where technologies enable empowerment of individuals, or groups of individuals, to express themselves in a more natural way, leading to easier creation, enriching, and finding of content...*” [1].

## **3. Social Computing Principles**

In addition to the definition of Social Computing in [1], a set of nine principles that can be identified as the main characteristics of Social Computing have been defined, by comparing the most popular definitions of Social Computing.



**Fig. 1.** The nine principles of Social Computing as described by Knol in [1].

This review and comparison resulted in defining the following nine principles of Social Computing: *Open Platform*, *Lightweight Models*, *Enabling Services*, *Intuitive Usability*, *Long Tail*, which are more **Technology oriented** and *Unbounded Collaboration*, *Collective Intelligence*, *Network Effect* and *User Generated Content*, which are more **Socially oriented**. In the following subsections we will briefly present these principles according to their orientation.

### 3.1 Technology oriented

- **Open Platform.** This principle dictates that Web 2.0 services provide users the possibility to access information they desire with their browser and puts an emphasis on synergy between different devices and applications that are connected to the Internet. At the same time it does not imply the replacement of desktop computers and classic operating systems but promotes compatibility and collaboration, towards a more social web.
- **Lightweight Models.** When talking about lightweight models, we think of flexible and agile ways of developing a product, thus being able to update, change and re-use much faster than the classic development methods. This is common for a web based service, "...it requires an agile business model, which can handle such a fast update rate..." and it also helps to reduce the costs in organizational change, a process that is expensive in terms of energy usage as well as investments.

- **Enabling Services.** What enabling services define is the differentiation of online services that follow the model of *SaaS - Software as a Service*, a model used extensively, by services such as the applications by Google e.g. Gmail and Google Docs. What is considered crucial for these services is to be characterized by flexibility, openness, scalability and re-usability. These characteristics enable the creation of mashups, faster updating, online management of data and lower transaction costs.
- **Intuitive Usability.** The meaning of this principle is quite clear, but this does not mean that it is simple to implement; Usability is key when it comes to interfaces, which is a main element of all web services. Consequently, interfaces must be easy and simple for all types of users, according to the *walk up 'n' use* mentality, but at the same time to offer expert users more options to personalize their side of the service according to their liking.

For example, *Macromedia* introduced the term of *Rich Internet Applications* as described in the previous section, but things became even more usable when *AJAX* was introduced, exploiting the graphic environment offered by *JavaScript*.

### 3.2 Socially oriented

- **Long Tail.** The basis of this principle is the *80 - 20 Pareto distribution*, meaning that services should aim at both sides of the market; the 20% of customers that generate large profit, as well as the 80% of customers that generate small profit, but are by far larger in number. To achieve this, services must attract users of all orientations, having Amazon as an example which manages to gain a lot of profit from this group of customers, by offering niche products, exclusively from their online store.
- **Unbounded Collaboration.** Social computing is all about collaboration and communication, between users and between services. This can be achieved, by giving developers the ability to create mashups, through the Open Platform principle, for example by using Open APIs (as described in 2.2 Web 2.0 Technologies). Additionally, users meet, interact and communicate in online communities. User contributions within these communities, add value to these services or to the organizations that have created the service.
- **Collective Intelligence.** When talking about *Social Computing* and *Unbounded Collaboration* of users, then we also have to take into consideration the knowledge, which is created by these communities, whether they work in groups or individually.

The main issue for this principle is trust amongst users, since this decentralization of the web suffers from lack of control. Web services that want to harness the vast amounts of information generated by all these users, it is important attract as many as possible to contribute, since the 1% rule applies in these communities.

- **Network Effects.** This principle describes the interaction between users and services, but this relation works both ways; users contribute to make services richer in content, by sharing their knowledge and services attract users, by offering more options and by adjusting their content to every user’s preferences and history. Due to the limited number of users that actually spend time and effort to contribute to an application, attracting users is an ongoing race, where the rule of *first-mover-wins* does not apply, since we constantly encounter new services that make competition harder.
- **User Generated Content.** Social Computing contains all of the previously mentioned principles and technologies, but in its core, data is the main driver of social computing and the owner of the data is key. Managing all this data is a difficult task, since there are many issues regarding this matter; authorship, privacy and security. How content is used and protected by service providers is a major issue and often influences how popular a service is.

#### 4. Mapping Web 2.0 Technologies and Principles

In the previous sections, we have presented the technologies that enabled the evolution of the Web into Web 2.0, which in turn focuses in the Social aspects of computing and the interaction between users and services. We also presented the principles which define the main borders of Social Computing and how they are separated into two categories, *Technology oriented* and *Socially oriented*.

In this section we are going to present a map of the principles, their corresponding technologies and explain why we consider these relations to be valid. In the following table, where we have mapped the relationship of *Principles* and *Technologies*, we have placed columns with the Principles divided in the two orientations that can be identified.

We have placed the Technologies in the rows of our table, in a sequence from the lowest level technologies, according to the OSI 7-layer model of the internet layers, to the highest. The shade of gray is correspondent with the closeness of the relation between Technologies and Principles. Darker shades equal higher relevance and lighter shades equal lower relevance.

The positions of the principles are correspondent to the initial description from [1]. Principles more *Technology oriented* are placed on the left and more *Socially oriented* principles are placed on the right.

**Table 1:** *The relation between Web 2.0 Technologies and Principles of Social Computing.*

PRINCIPLES	
<i>Technology Oriented</i>	<i>Socially Oriented</i>

TECHNOLOGIES	Open Platform	Lightweight Models	Enabling Services	Intuitive Usability	Long Tail	Unbounded Collaboration	Collective Intelligence	Network Effects	User Generated Content
AJAX									
SOAP & REST									
Open APIs & Mashups									
Flash, AIR, Silverlight									
RSS, Microformats									

At first glance, we can see the clear formation of two clusters, one on the top left of the Table 1 and one on the bottom right. Let's take a deeper look into the relations between Technologies and Principles and try to explain why they occur.

#### **Technologies and Technology oriented Principles**

As logic dictates, one would assume that the Technology oriented principles would be related with technologies that belong to a lower OSI layer, which are placed on the first rows of the table; *AJAX* and *SOAP & REST* have most of the relations with the Technology oriented Principles .

These technologies have started a true revolution in web services, since they are open to all developers, especially *AJAX* and *REST*, enabling the creation of lightweight applications that hide unnecessary information from the user. At the same time, they present users *the look and feel* of a normal desktop application, thus also contributing to the Intuitive Usability of these applications.

Multimedia technologies like *Flash* and *Silverlight* do not relate with many Principles, but their importance is not reduced; these technologies give developers the ability to create impressive and interactive applications, thus adding new elements to the *Intuitive Usability* of services. This type of technologies is widespread and is constantly becoming easier to use, so that everyone can create their own personal content, thus attracting a larger percentage of the Long Tail.

Another type of technology that attracts a large number of potential clients, thus augmenting the *Long Tail Principle*, is that of technologies that add a more personalized view on services, such as *RSS feeds* and *Microformats*. Additionally, these technologies that augment personalized *User Interfaces* also have a close

relation with *Intuitive Usability*, since it helps users to access their favorite topics much faster.

**Technologies and Socially oriented Principles.** On the other hand, higher level technologies, like *Flash*, and *RSS* have most of their relations with the *Socially oriented Principles*. They are the ones that promote and augment the interaction between users and enable them to create new content that adds value to existing services.

Using multimedia technologies like *Flash* or *Silverlight*, is accomplished either by complete solutions such as Rich Internet Applications, or by embedding music, movies and other multimedia in web services. Another way of using this technology is quite common nowadays, when users are asked to create their own content (*User Generated Content*), to share it with others and in return they receive various rewards.

Such rewards might include their videos being included in their favorite music group video clip (*'Placebo – Running Up That Hill' music video clip, Prodigy Video Clip Contest*, [19]), or even using their videos to add more depth to their university applications, something that Tufts University recently introduced [20], [21]. Examples like these, point out the significance of user generated content, which is enhanced by multimedia technologies.

In the previous subsection we explained how *RSS* and other personalization technologies are related with Technology oriented Principles, but there is more to these technologies; Personalization means that users have spent time on these applications and have tweaked the interface according to their preferences.

What we can derive from this behavior is that these users have become dependent from these services, thus they are valuable to them and this value keeps increasing. This is also one of the main ideas of the Network Effects principle, making this type of technologies and this principle, closely related.

Finally, these personalization options give users the ability to create their own personalized content and share it with their peers, who can comment, add or respond with their own content, thus explaining the relation with the User Generated Content principle.

**The special case of Openness.** The only technology that has many relations, with both Technology and Socially oriented Principles is that of *Open APIs & Mashups* and the reason for this is simple; Open APIs promote the principle of the Open Platform, giving developers the opportunity to use many different parts of applications to create Enabling Services.

This openness also creates an attraction of large numbers of users, who are potential customers, thus giving organizations the ability to exploit the Long Tail. Such a large concentration of users also results in the creation of communities regarding these new APIs or initiatives for new mashups. The benefits of these communities have been already described in 2.3 *Social Computing and its Principles*, where all of the *Socially oriented Principles* are mentioned.

## 5. Conclusions

In our research we presented a series of technologies that have enabled the evolution of the World Wide Web, the creation of innovative services and new

applications that have been tagged with term *Web 2.0*; since this term is still new and there is confusion surrounding it, we conducted a literature research of all the different definitions of Web 2.0.

Furthermore, we created a list of the most important technologies behind Web 2.0 and presented all these technologies, describing their key elements and how they contributed in changing the landscape of web service and applications. Having done this, the Social identity of this new trend came to light and in order to define it we used the recent work, [1] as our main base. Thus we described the term Social Computing, which is another name for Web 2.0 and the wave of social services that came with it. In order to have a clear view of what Social Computing really means, we presented the principles that govern it and that create the general borders in which is functions. These principles were also divided, according to their orientation, in two categories; Technology oriented and Socially oriented.

The main idea behind this paper was triggered by the complicated and abstract character of Web 2.0 and our main goal was to provide with a more solid structure for Social Computing. To do so, we created a map of the relations that can be identified, between its two main elements; the Principles that govern it and the Technologies that enabled the World Wide Web to evolve into Web 2.0 or Social Computing.

Concluding our research, we explained the relations that were identified and showed the different relations between the two different types of Principles, with Technologies of different OSI layers. Through the creation of this solid structure, if we were to create a new Social Computing service or application, it would be best if we actually looked carefully at the map of Technologies – Principles; we should decide what we wish to achieve with our new service and use the map as our guide in creating a detailed plan about the development of our new product.

## **6. Discussion**

During our research, we encountered several analyses about Web 2.0, Social Computing and the technologies behind it. We also came across new developments in the area of Web 2.0 Technologies, such as the work of Mozilla Labs and the new services that were recently introduced by Google, *Google Wave* and the controversial *Google Buzz*. With new services like the ones mentioned, the relation of principles and technologies becomes even more obvious, since they promote social interaction, sharing of knowledge and user generated content, all based on the technologies mentioned in the present paper [22], [23].

Although the main subject of our research is Web 2.0 Technologies, innovation does not only happen in services, but also in devices, closely linked to the Internet, which create new challenges and opportunities. For example, following the joint effort of Microsoft - HP *Tablet* presentation, Apple introduced *iPad* into the market, products which were launched as a '*middle product*', between smart phones and laptops. These new devices create opportunities for the development of innovative services, which in turn add new value to web based services, such as the new bookstore application supported solely by Apple's *iPad*.

Finally, if we were to reflect on the future of Web 2.0, one thing is certain; there will always be new services, applications and technologies introduced, therefore we must always keep in touch with latest innovations and an open mind.

## References

1. Knol, M., Spruit, M., & Scheper, W.: The Emerging Value of Social Computing in Business Model Innovation. M. O. Lytras, *Electronic Globalized Business And Sustainable Development Through IT Management: Strategies And Perspectives*. IGI Global. (2009)
2. O'Reilly, T., <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
3. O'Reilly, T., [http://radar.oreilly.com/archives/2006/12/web\\_20\\_compact.html](http://radar.oreilly.com/archives/2006/12/web_20_compact.html)
4. McAfee, A. (2006). Enterprise 2.0: The Dawn of Emergent Collaboration. *MIT Sloan Management Review* , 47 (3), pp. 21-28.
5. Berners-Lee, S. T. (1999). *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web* . New York: HarperCollins.
6. Hinchcliffe, D., [http://web2.socialcomputingjournal.com/the\\_state\\_of\\_web\\_20.htm](http://web2.socialcomputingjournal.com/the_state_of_web_20.htm)
7. Garrett, J, <http://www.adaptivepath.com/publications/essays/archives/000385.php>
8. Anderson, P.: *What is Web 2.0? Ideas, technologies and implications for education*. JISC Technology and Standards Watch. (2007)
9. Ecma International.: *Standard ECMA-262 5th Edition: ECMAScript Language Specification*. Geneva: Ecma International. (2007)
10. McLellan, D.: *O'Reilly: xml.com.*: <http://www.xml.com/pub/a/2005/02/09/xml-http-request.html>
11. Netscape Communications Corporation. *Javascript Press Release*, <http://web.archive.org/web/20070916144913/http://wp.netscape.com/newsref/pr/newsrelease67.html>
12. Microsoft Corporation., [http://msdn.microsoft.com/en-us/library/1kw29xwf\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/1kw29xwf(VS.85).aspx)
13. Pautasso, C.: SOAP vs REST: Bringing the Web back into Web Services, Business Integration Technologies, IBM Zurich Research Lab, (2007)
14. Spies, B., <http://www.ajaxonomy.com/2008/xml/web-services-part-1-soap-vs-rest>
15. Microsoft Corporation, <http://www.microsoft.com/silverlight/overview/default.aspx>
16. Benson, J., & Company, H. a.: *EMML Changes Everything: Profitability, Predicatbility & Performance through Enterprise Mashups*. Alexandria, Virginia: Open Mashup Alliance. (2009)
17. Greaves, M.: Semantic Web 2.0. *IEEE INTELLIGENT SYSTEMS* , p. IEEE Computer Society. (2007)
18. Ankolekar, A., Krötzsch, M., Tran, T., & Vrandečić, D.: The two cultures: Mashing up Web 2.0 and the Semantic Web. *Journal of Web Semantics* , pp. 70-75. (2008)
19. Prodigy Video Clip Contest, <http://www.dailymotion.com/group/runwiththewolves>
20. My College Guide, <http://mycollegeguide.org/blog/02/2010/college-applications-video/>
21. Tufts University Admissions, <http://admissions.tufts.edu/?pid=184>
22. Hasija, K., Singh, D., & Mehta, A.: Re-Orientatation of Web 2.0. *International Conference on Information and Multimedia Technology* (pp. 117-120). Jeju Island, South Korea : IEEE Computer Society. (2009)
23. Vossen, G.: *Unleashing Web 2.0, From Concepts to Creativity*. Burlington. Morgan Kaufman, (2007)