Web 2.0 and the Semantic Web

Group Homework of Internet Services & Protocols
12.06.2006

Chao Xiaojuan
Shen Li
Wu Weiwei
Wu Binbin
History of Web: From Web1.0 to Web2.0

**Web1.0**
- Static HTML Pages
- Updated rarely

**Web1.5**
- Dynamic HTML web pages
- Ever-changing content

**The Semantic Web**
- Web pages understandable by computers

**Web2.0**
- the Web as platform
- Data as the driving force
- Share and Interaction
Part I: The Semantic Web

Outlines

1. Motivations
2. Main Principles
3. Layers/Components/Specifications
4. Application Examples
Motivations

Traditional web pages are designed to read by people, not machines.

The Semantic Web aims to make web pages understandable by computers, so that they can search websites and perform actions in a standardized way.

The benefits are that computers can harness the enormous network of information and services on the Web.
The goal of the Semantic Web is to develop enabling standards and technologies designed to help machines understand more information on the Web so that they can support richer discovery, data integration, navigation, and automation of tasks.

With Semantic Web we not only receive more exact results when searching for information, but also know when we can integrate information from different sources, know what information to compare, and can provide all kinds of automated services in different domains from future home and digital libraries to electronic business and health services [Berners-Lee 2001].
The definition of the Semantic Web

The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries. It is a collaborative effort led by W3C with participation from a large number of researchers and industrial partners. It is based on the Resource Description Framework (RDF), which integrates a variety of applications using XML for syntax and URIs for naming. (W3C)
Semantic Web Main Principles

**Principle 1: Everything can be identified by URI's**

People, places, and things in the physical world can be referred to in the Semantic Web by using a variety of identifiers.

The vocabularies we choose to use for describing resources are also defined by URIs.
Semantic Web Main Principles

Principle 2: Resources and links can have types
Semantic Web Main Principles

**Principle 3: Partial information is tolerated**
Some of the linked resources may cease to exist or the addresses may be reused. The Semantic Web tools need to tolerate this data decay and be able to function in spite of that.

**Principle 4: There is no need for absolute truth**
The applications decide what they trust by using the context of the statements; e.g. who said what and when and what credentials they had to say it.

**Principle 5: Evolution is supported**
The Semantic Web uses descriptive conventions that can expand as human understanding expands.

**Principle 6: Minimalist design**
The aim of the W3C activity is to standardize no more than is necessary.
Semantic Web Layers and Components

The Semantic Web principles are implemented in the layers of Web technologies and standards.
Semantic Web Layers and Components

The Semantic Web comprises the standards and tools:

**URI**  A global naming scheme.

**RDF** is a simple data model for referring to objects ("resources") and how they are related. An RDF-based model can be represented in XML syntax.

**RDF Schema** is a vocabulary for describing properties and classes of RDF resources, with a semantics for generalization-hierarchies of such properties and classes.

**OWL** adds more vocabulary for describing properties and classes: among others, relations between classes (e.g. disjointness), cardinality (e.g. "exactly one"), equality, richer typing of properties, characteristics of properties (e.g. symmetry), and enumerated classes.
Resource Description Framework (RDF)

RDF was designed as a metadata model using XML; now used as a general method of modeling knowledge, through a variety of syntax formats (XML and non-XML).

The RDF metadata model is based on the idea of making statements about resources in the form of a subject-predicate-object expression, called a triple in RDF terminology. (“The sky has the color blue.”)
Part II: Web2.0

Outlines

1. Web2.0 principles and techniques
2. Several Web2.0 examples
3. Markets and Future
Like many important concepts, Web 2.0 doesn't have a hard boundary, but rather, a gravitational core. You can visualize Web 2.0 as a set of principles and practices that tie together a veritable solar system of sites that demonstrate some or all of those principles, at a varying distance from that core.
Principle 1: The Web as platform

**Traditional concepts:**
1. "the web as platform" in terms of the old software paradigm: the web browser is a product.
2. the web was about publishing, not participation; that advertisers, not consumers

**Web 2.0 concepts:**
1. Customers pay for the services used that delivered by web
2. "the long tail", the collective power of the small sites that make up the bulk of the web's content
3. *the service automatically gets better the more people use it.*
Principle 2: Harnessing Collective Intelligence

It is a truism that the greatest internet success stories don't advertise their products. Their adoption is driven by "viral marketing"—that is, recommendations propagating directly from one user to another.

Network effects from user contributions are the key to market dominance in the Web 2.0 era.

The technique based on this rule:
- **Yahoo!**
- **Google's PageRank**
- **eBay**
- **Amazon**
Principle 3: Data is the Next Intel Inside

Who owns the data?

In the internet era, one can already see a number of cases where control over the database has led to market control and outsized financial returns.
Principle 4: End of the Software Release Cycle

One of the defining characteristics of internet era software is that it is delivered as a service, not as a product. This fact leads to following mainly fundamental changes in the business model:

- Operations must become a core competency.
- Users must be treated as co-developers.
Based on above principles, what changes in Web 2.0

<table>
<thead>
<tr>
<th>Web 1.0</th>
<th>Web 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal websites --&gt;</td>
<td>Blogging</td>
</tr>
<tr>
<td>DoubleClick --&gt;</td>
<td>Google AdSense</td>
</tr>
<tr>
<td>Akamai --&gt;</td>
<td>BitTorrent</td>
</tr>
<tr>
<td>mp3.com --&gt;</td>
<td>Napster</td>
</tr>
<tr>
<td>Personal websites --&gt;</td>
<td>Blogging</td>
</tr>
<tr>
<td>publishing --&gt;</td>
<td>Participation</td>
</tr>
<tr>
<td>Content management systems --&gt;</td>
<td>Wikis</td>
</tr>
<tr>
<td>.... --&gt;</td>
<td>....</td>
</tr>
</tbody>
</table>
Examples of Web2.0 <1>-BLOG

A weblog (usually shortened to blog, and occasionally spelled web log) is a web-based publication consisting primarily of periodic articles.

A blog entry typically consists of:
- **Title**
- **Body**
- **Permalink**
- **Post Date**

A blog entry optionally includes:
- **Comments**
- **Categories (or TAGS)**
- **Trackback**
Examples of Web2.0 - Wikipedia

Wikipedia, an online encyclopedia based on the unlikely notion that an entry can be added by any web user, and edited by any other users.
Examples of Web2.0 <3>- Google Adsense

Google's AdSense program lets you sell advertising space for other people's ads on your website--and not just any ads that Google chooses, but ads that are relevant to your site's content pages.

The service is free, and you earn money every time someone clicks on an ad.

It’s success came from an understanding of the collective power of the small sites that make up the bulk of the web's content.
Examples of Web2.0 <4>- Google Map and Ajax

**Google Maps** is a free, web map server application and technology provided by Google. It offers draggable street maps, a route planner and business locator (city centers only) for many nations, and satellite images for the whole world. It is implemented by Ajax.
Examples of Web2.0 <4>- Google Map and Ajax- continued

Ajax (Asynchronous JavaScript and XML) isn’t a technology. It’s really several technologies, each flourishing in its own right, coming together in powerful new ways. Ajax incorporates:

- Standards-based presentation using XHTML and CSS
- Dynamic display and interaction using the Document Object Model
- Data interchange and manipulation using XML and XSLT
- Asynchronous data retrieval using XMLHttpRequest
- JavaScript binding everything together.
Market and Future

We expect to see many new web applications over the next few years, both truly novel applications, and rich web reimplementations of PC applications.

1. Gmail has already provided some interesting innovations in email, combining the strengths of the web (accessible from anywhere, deep database competencies, searchability) with user interfaces that approach PC interfaces in usability.

2. Web 2.0 will also remake the address book.

3. A Web 2.0 word processor would support wiki-style collaborative editing, not just standalone documents.

4. Web 2.0 revolution will not be limited to PC applications, the web can also be used to deliver software as a service, in enterprise scale applications such as CRM.
Conclusions

Is Web 2.0 killing the Semantic Web?

Web 2.0 is people-oriented and doesn’t care about technology or standards. It let people create, collaborate, share and interact.

The Semantic Web is the polar opposite: standardize all your data in RDF; encode it in XML.

So whilst Web 2.0 is about high-level (user experience) and immediate benefits, the Semantic Web is a low-level (data), long-term solution.

We can take advantage of the flexibility of Web 2.0. As it is technology agnostic, we can use Semantic Web technologies in our Web 2.0 applications and get the best of both worlds.
References

1. Wikipedia (Web2.0/The Semantic Web/RDF)
2. Tim O'Reilly: What Is Web 2.0 --Design Patterns and Business Models for the Next Generation of Software
5. Dan Zambonini: Is Web 2.0 killing the Semantic Web