

The Web according to W3C

How to turn your idea into a standard

Bert Bos, W3C, December 2005

W3C celebrated its 10th anniversary in 2004. In those 10 years, both W3C and the Web have become more complicated. Not for the users, we hope, but certainly for developers of Web-related software.

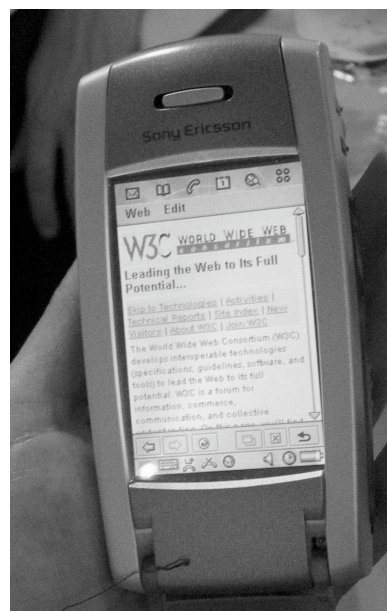
Let's take a (brief) look at the methods W3C tries to use to reach consensus (because *consensus* is the basis of W3C's decision making), at the ways in which people can follow and participate in the work, and at a few of the technologies that are expected.



Figure 1. The W3C 10th anniversary logo

When W3C started, the Web was simple: the IETF had taken on the task of defining URLs; W3C and the IETF worked together on HTML and HTTP; W3C developed CSS; and a group of people donated PNG to W3C. There were plenty of people helping out and although some had trouble understanding W3C's vision of a Web on other devices than PCs, the architecture was simple and progress was quick.

Now the Web is big, slow and complex. There is an ever increasing demand for new technologies, for security, b2b, multimedia, accessibility, privacy, and what not, and although W3C's vision is still the same, it needs more and more discussion in more and more groups to harmonize all the technologies being proposed. But at least everybody now wants the Web on small devices...



*Figure 2. The W3C home page normally has three columns, but on handheld devices it uses a different style sheet.
(Photo: Bert Bos)*

W3C's motto is "Leading the Web to Its Full Potential" but that doubtlessly doesn't tell you much. What *is* the full potential of the Web? According to W3C, the Web will have reached its full potential when it has become the "Semantic Web."

The *Semantic Web* is an ideal. It is a bit like Artificial Intelligence, which comprises everything we think computers should be able to do for us, but cannot do yet. But as soon as they can do something, we no longer regard it

as intelligence... The Semantic Web is the same. We want the Web to give us information, i.e., semantics, not just data, but in fact we “only” become better at manipulating the *syntax* of the data.

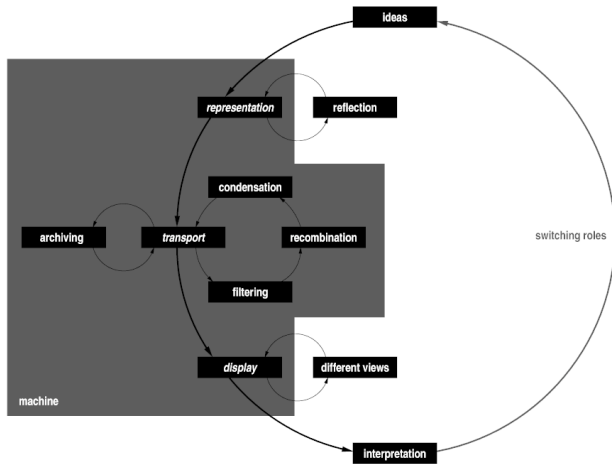


Figure 3. A diagram of communication when the Web is the communication channel.

Another way to define the Semantic Web is as the “re-usable Web.” Information isn’t just presented to a human on a computer screen, but it can also be presented on other devices with smaller screens or no screen at all (in voice or braille, e.g.) and it can be transformed and integrated with other information to increase its usefulness. All this automatically, without the need for a human editor.

Once you know what W3C’s goal is, it is easy to see how its various technical activities try to advance that goal. The activities work on different aspects and at different levels. Some examples:

SVG (Scalable Vector Graphics) provides a way to describe graphics as a set of components with attributes, which helps a bit to make those graphics adaptable to different devices. It doesn’t help to make the information available on non-graphical devices, though. For that, we need to grab the information at a higher, more abstract level.

The Mobile Web Initiative (MWI) is a project in W3C with several working groups that look specifically at technologies and

guidelines for creating *device-independent content*, so that you can access the Web with your mobile phone. Depending on how “smart” your phone is, the Web may be more or less accessible, but at least you should get the best possible experience within the constraints of the device.

The Web Accessibility Initiative (WAI) is another such project. It provides guidelines, technology and training on how to make the Web *accessible* to people with handicaps. (The aging of the population means that most of us will end up handicapped in some way...)

CSS implements the concept of *separation of structure and style*. The structure is in HTML, SVG or some other format and the style in CSS. There can be multiple styles, each for different devices or different kinds of users, and more styles can be added without having to change the structure. Separation of style and structure is one of several rules of thumb for implementing re-usable content. It often helps, but not always.



Figure 4. The CSS and RDF logos.

If even HTML isn’t high enough a level for encoding information in a reusable way, the ultimate format is RDF (Resource Description Framework). RDF itself is a model of knowledge representation in which all information is decomposed into triples of (subject, predicate, object): John is-author-of book1, John is-a human. Each such triple is an arc in a network: subjects and objects can be the start and end of multiple arcs. It is even recursive: the whole triple can be a subject or object in another arc.

On top of this abstract model, W3C and others have developed concrete models for various areas of life, which are called “ontologies”. Basically, they are vocabularies of keywords and rules for how to combine them.

Now you maybe ask why we develop SVG, HTML, CSS and other formats if RDF can express everything and is the most reusable

of all? Well, try to write something in RDF, then you'll know why... Maybe for this audience I can use a comparison with programming languages: why do you use so many different languages, isn't assembly language enough? If we want everybody to be an author (and we do), we'll have to find compromises between abstraction and usability. And hopefully with better tools and by comparing the author's content to other information on the Web, the computer can add enough metadata to make the content reusable slightly beyond what the author himself catered for.

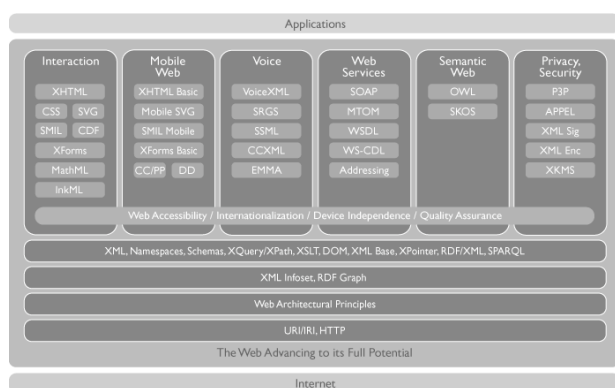


Figure 5. The W3C formats that are based on XML form a stack with HTTP, URI and XML at the base and the most specialized formats at the top.

Now we know *why* W3C decided to develop certain things, but we don't know yet *how* it decides. W3C has over the years developed a certain process, that reflects its structure.

At the heart of W3C is a function called the *Director*, a role that is currently filled by Tim Berners-Lee. The final word on any technology is always the Director's. He decides whether a working group will be created and, when a specification has been developed, whether it may be called a W3C Recommendation. Of course, he can delegate and he never takes any decisions without consulting the team and the members.

The team keeps the computers running, manages the finances, etc., but most of the team consists of technical experts, whose task it is to lead working groups and workshops,

coordinate among working groups, represent the viewpoints of people who aren't members of W3C and in general look after the overall architecture of the Web and of W3C's technologies in particular. As of December 2005, the team consists of 67 people. The three “host sites” are Sophia-Antipolis (France), Tokyo (Japan) and Cambridge, MA (USA), but many team members are living elsewhere.



Figure 6. Most of the communication in W3C's activities passes over the Web, e-mail and the phone, but face-to-face meetings are indispensable.

W3C is a consortium, and thus the third component of W3C is its members. The members are companies and organizations, currently about 400. Part of the money that keeps W3C running comes from membership fees (the rest from donations and sponsored projects). In return for the membership fee, organizations have the right to send somebody to the Advisory Committee and to review specifications and proposals for working groups. They can also send people to the working groups, which is of course the most effective way to both learn about new technologies and influence them. Participating in a working group is effective, but it takes time, and for small companies that is usually where the real cost of membership lies, more so than in the membership fee.

Members or the team propose new working groups when they notice the need (and opportunity) for new work. Typically, the first step is a workshop to which anybody can send papers; a program committee selects the relevant ones. The goal is to determine if W3C should start one or more working groups and, if so, what their scope should be.

If a working group looks desirable, the team writes a proposed charter and asks the members to review it. Based on that review, the Director decides to create the working

group or not. Next, members are invited to join. Working groups can also invite non-members, the so-called Invited Experts. Many Invited Experts are students, who often have very good input, but obviously don't work for a W3C member organization.

W3C SNAKES AND LADDERS WD

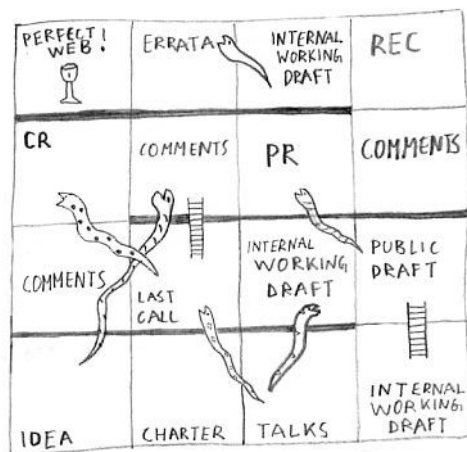


Figure 7. The path from member-only draft to Recommendation has both shortcuts and backwards steps. (Drawing: Ian Hickson.)

Working groups write specifications, which go through various stages on their way to become *W3C Recommendations* (i.e., standards). They start as Working Drafts, then become Candidate Recommendations, and finally Recommendations. Between those stages are periods of review, respectively called Last Call and Proposed Recommendation. During the first of these, anybody, not just members, can send comments, which the working group has to answer. The Director will check if all comments have been answered satisfactorily. During the Proposed Recommendation stage, members review the specification and the Director checks if it has been implemented, because, in general, specifications can only become Recommendations when interoperability has been proven in practice.

All working groups also have a public mailing list, where anybody can join to discuss the specification that is being developed.

W3C has other groups: Interest Groups, Coordination Groups and Incubator Groups. They don't write specifications but they coordinate. IGs provide a platform for members to coordinate, CGs help with scheduling among WGs and Incubator Groups, have a role similar to workshops in that they result in a report on whether there is a need for new technology, except that they can take a year to do so.

Apart from Working Drafts, Candidate Recommendations and Recommendations, W3C also publishes Notes and Submissions. The difference is that Notes are written by WGs, while Submissions come directly from member organizations. They have no status, other than that the Director agreed that it was useful that W3C publish them. Notes often contain background information or historical notes, Submissions are often about proprietary specifications that may or may not be merged into W3C technologies later.

So the four ways to become involved in W3C are: (1) join a public mailing list, (2) get invited to a WG as an Invited Expert, (3) make your employer a W3C member and join a working group and (4) get a job at W3C.



Figure 8. W3C recommends that people check their HTML, CSS, RDF, P3P and XML Schemas with the W3C validator service, but a Web page doesn't need to be validated to be valid.

The process sketched above has over the years become more detailed. There are several required steps that remain largely invisible from the outside, but that have been added to

avoid various errors. One of the biggest reasons for these steps are issues around intellectual property rights, including copyrights, but especially patents.

Copyrights are fairly simple to deal with and the issues around them were already well known when W3C started. (But even so, it has taken us several years to get the wording right: it's much easier now that we have team members who are trained in legal matters.) Basically, everything W3C publishes falls under either the W3C document license (free for any use, but you may not modify the document) or the W3C software license (an Open Source license). Anything you contribute to W3C, such as Submissions or test suites, must be re-distributable under either license.

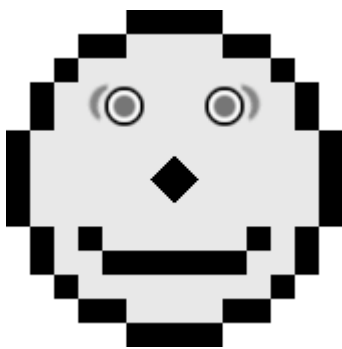


Figure 9. *Acid2 is a test of support for certain aspects of CSS, HTML, HTTP and URLs, made by the Web Standards Project. If the browser is compliant, the test page shows this face.*

The patent problems are more severe and have had a lot of impact on the W3C process. Ten years ago, nobody thought about them and if anybody mentioned them at all (such as Eolas), people just laughed at them. The Web was new, free and open.

But as companies started making money from the Web and claimed patents on Web-related technologies, the situation changed. W3C developed a patent policy to protect its technologies and its members. At various stages in the development of a specification, W3C members now have to declare patents they have, and if they want to join a

working group, they have to license any essential patents royalty-free. (If they can't do that, either the working group will not be created, or its charter will be changed to create a slightly different technology.) Everything that comes out of W3C is thus royalty free, at least as far as patents by W3C members are concerned.

W3C is not a “patent pool” like some other organizations. The technologies are free for use by *everybody*, not just for W3C members, and the license is automatic: you don't have to ask for it.



Figure 10. *Tim Berners-Lee and Robert Cailliau on the stage during the European celebration of 10 years of W3C. Behind them is a photo of a dinner during a W3C meeting in Tokyo in 1997, with Tim, Robert and Ted Nelson. (Photo: Bert Bos.)*

On the other hand, we can only guarantee royalty-free licenses for any patents held by W3C members. There may be people or companies outside W3C who hold patents and may try to claim that they apply to our technology. (If that happens, W3C member companies may retaliate and exclude those people or companies from any royalty-free license that they provided for W3C technologies.)

Of course, it would have been easier if software patents hadn't existed, but that is not something W3C can define. W3C's patent policy provides the best possible environment for software makers in the current situation. As far as I know, no other organization has a policy that gives more freedom to developers.

Patent declarations thus add overhead to

our process, during publication and during the creation of a working group. Working group charters have to be precise. They can't just say in general what the group will work on, as in the early days. They have to be sufficiently precise that companies can determine whether they have any essential patents, even before the first line of a specification is written.

W3C isn't the only organization that makes technical standards. There is ISO, of course, but also ETSI, ECMA, ITU, OMA, IETF and many others. They each have their own domain, but those domains often touch and W3C therefore has liaisons with many standards organizations to make sure that standards are compatible, but also to lobby with them to adopt Web standards in their domains. OMA, e.g., is now using XHTML instead of the old WML and is actively coordinating new standards and guidelines for mobile devices with W3C. Several digital TV standards also use W3C technologies, such as HTML, CSS and the DOM.

In Europe, W3C is a member of the ICTSB, the Information & Communication

Technologies Standards Board, which coordinates IT standards in Europe among W3C, ETSI, CEN, CENELEC, EBU and others. W3C is also a partner in a European FP6 project called COPRAS, whose task it is to be the interface between other European FP6 projects and standards bodies.

Standards don't always originate in standards organizations. PNG is an example of a format that was developed by an ad-hoc group on the Internet and then standardized by W3C. In fact, HTML and CSS also started before W3C existed. More recently, some formats in the Web Services area were developed out of specifications contributed by W3C members.

One task of W3C is therefore to be on the look-out for promising developments that fit the Web architecture. The worst thing that can happen to an organization like W3C is that it starts to suffer from the "not-invented-here syndrome." Luckily, that is not yet the case. Some of the interesting initiatives that we closely follow are microformats and Web Forms 2.0. There is still a lot to be done on the Web...