

INTERVIEW:

XML Past, Present and Future:

An Interview with Tim Bray

Andrew Updegrove

There is essentially no computer in the world, desktop, hand-held, or back-room, that doesn't process XML sometimes...XML won't be the last neutral information-wrapping system; but as the first, it's done very well. - *Tim Bray, W3C.org press release, "XML is 10!" - 2-12-08*

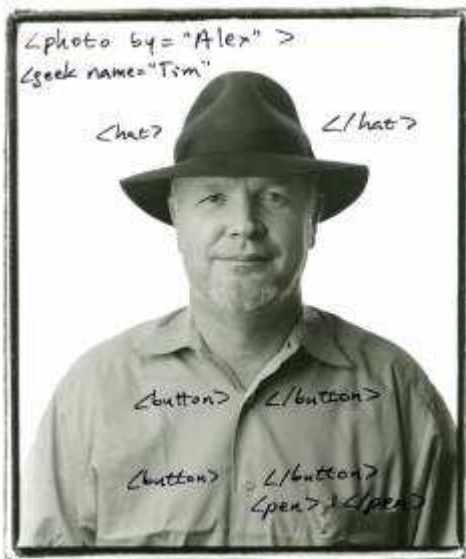


Photo by Alex Waterhouse-Hayward
alexwaterhousehayward.com

It may seem as if standards materialize out of nowhere, but of course that's never been the case. They are the product of a collaborative process that typically includes many experts, drawn from a variety of backgrounds. Every successful standards development effort also requires a few individuals willing to play a more central role, as working group chairs, to keep things moving efficiently, fairly and on the right course, and also as editors to control and write the text of the standard itself.

Serving as a standards editor is a highly technical task that can only be learned in the breach (there are no courses that teach it). At the same time, it requires not only satisfying the demands of those entitled to vote on whether to accept or reject a final draft, but also laying out what must be done in such a

way that those with no prior contact can easily produce uniformly compliant implementations. Needless to say, the likelihood of a standard's becoming widely and successfully implemented in the marketplace can be greatly influenced (for better or for worse) by the skills of its editor.

In the old days of standards development, editing a standard was a leisurely process. That all changed as the pace of innovation ramped up exponentially in the information technology sector. In the late 1990s, even greater pressure was brought to bear to generate the standards needed to keep the accelerating

locomotive of the Internet and the Web on track. In the fall of 1996, what must have been a new record was set when the first draft of an important new standard was produced in only twenty weeks.

Two co-editors made that possible (later there were three), one of whom was a Canadian raised in Lebanon who had already helped create one of the first successful Internet search engines. His name was Tim Bray, and the standard he helped create became one of the most influential standards of the Digital Age to date: the Extensible Markup Language, more commonly known simply as XML.

Tim's recruitment for that role was a combination of availability and capability. At the time what became the XML Working Group was chartered, Bray was an Invited Expert with the World Wide Web Consortium (W3C) and a friend of Jon Bosak, the project leader. He was also working as an independent consultant, making him the master of his own schedule. On the capability front, he had previously been the manager of a major text digitization project: the conversion of the Oxford English Dictionary. The rest, as they say, is history (you can read Tim's own highly personal account of the people, the times and the process [here](#)).

While Tim's primary role is as a technologist (his ongoing research is described at his [Concur.Next](#) Web site) he has continued to drive important Web-relevant standards efforts, including as a member of the W3C Technical Architecture Group (2001-2004), as co-editor of the Namespaces in XML W3C standard (1996-1999), and as the Co-Chair of the Atompub Working Group of the Internet Engineering Task Force (IETF) (2004-2007). When he's not editing standards, he serves as Distinguished Engineer and Director of Web Technologies at Sun Microsystems, Inc.

In this interview, Tim shares his thoughts on where XML has been, where it is now, and where it's going next.

I. The Past

AU: *For starters, what would the Internet and the Web look like today if XML had never been created?*

TB: I really have nothing beyond wild guesses; alternate histories are hard to make believable. On the downside, there is a huge amount of application integration with real business value-adds that would have been more difficult or impossible. In particular, the rise of REST [*ed*: Representational State Transfer, a distributed software architecture useful for the Web] might have been hampered if there hadn't been such a useful general-purpose format to ship around in the bodies of resource representations. On the upside, the huge waste of energy and investment that went into the failed WS-* project, which was originally presented as "XML Web Services", might have been prevented.

AU: *While XML was an outgrowth of SGML, it was a new start, rather than a new version of SGML. What were the problems you were trying to solve when you helped create XML that required a fresh start?*

TB: When you ship a new version of something, it's usually grown, compared to the previous version. XML, on the other hand, was radically smaller than SGML, so it could hardly be presented as a new version. Also, several of us were impressed with Tim Berners-Lee's then-new Web consortium and thought it might be a more fruitful place to get work done than the ISO SGML committee.

AU: *All standards need sponsors, usually from the business world. Who kicked the XML effort off, and why?*

TB: XML had some sponsorship from Sun, in that John Bosak's manager authorized him spending half his time on it. But the primary business-world backer was Microsoft, which saw the opportunity to do more business computing on the Web; it was pretty obvious that neither HTML or SGML was the right vehicle for this, but the space between them was crying out to be filled. Aside from Sun and Microsoft, there was support from some small SGML-community players, but none of the big corporates; until it became obvious that XML was catching on.

AU: *XML was created in the middle of a wild ride (the Internet Bubble) with enormous financial, technical and social dimensions that researchers will be studying for decades. What was it like working on XML – under tremendous time pressure – while all of this was going on?*

TB: It was about as much fun as you can get paid for; which reminds me that I should point out that a lot of the labor was volunteer: myself and James Clark at least. While Michael Sperberg-McQueen was employed, his employer, I suspect, didn't realize they were supporting the creation of what became XML.

[Editing XML] was about as much fun as you can get paid for

The XML Working Group was a like-minded bunch and we had a mostly-shared vision, based on experience, of what needed to be built. We got along well and were blessedly free of problem personalities. Jon was a capable and efficient leader.

AU: *XML achieved wide use very quickly. Was there more to this than simply the rapid growth of the Web? In other words, why did XML take off so rapidly, while many other worthwhile standards don't?*

TB: Let me turn that question around: Why on earth did it take until the late Nineties before someone cooked up a neutral data interchange format? There had been some attempts, most notably ASN.1. The time was long-overdue and the need was huge. XML, seen in the rearview mirror, is far from perfect, but it could be made to work for interchanging more or less anything between more or less any two computing systems, and the world really needed one of those.

Another important reason is that in parallel with designing XML, we (James Clark, myself, people at Microsoft) were building open-source software to process it. So by the time people got around to looking at it, there was already reasonably-good free software that you could put to use right then.

I guess I shouldn't underestimate the importance of the fact that XML got internationalization right via its tight coupling to Unicode in a way that turned out to pretty well just work.

Finally, the fact that XML was quite useful for encoding documents, not just relational records or persisted objects, was a major value-add.

When we (chiefly Jon Bosak and I) went out on the road to sell XML, it was like hurling your weight against a door that wasn't even latched; everyone said "Oh yeah, we can use that."

AU: *The flip side of success for a standard is often contentiousness in its development and maintenance. You had a taste of that early on when you went to work for Netscape, resulting in a third co-editor (from Microsoft) being appointed. Ten years later, we had the ODF-OOXML saga. Do you think XML bears any lasting scars from the level of energy that major vendors put into its development and maintenance?*

TB: Netscape, despite the fact that they hired me, never put any energy in. My reports on its progress were more or less completely ignored. Netscape was already well into the progress of its case of terminal arrogance.

But the answer to your question is "yes." XML 1.0 itself was designed and shipped by a small group of experts who really had no motives aside from making it work. Once it became successful, the space around all the standards-building tables became crowded with company representatives, who had neither the same level of technical expertise, nor the same focus on doing the right thing. Examples of negative results were the low quality of specifications like XSD and WSDL; and, as you point out, the OOXML debacle.

AU: *Conversely, were there any benefits from this level of attention to help offset the frustrations?*

TB: Not that I'm aware of.

AU: *How did you expect XML to be used, and by whom? Is that what actually happened, or did it take on a different life of its own?*

TB: Our primary objective was that Web servers deliver payloads suitable for processing by computer programs, as well as display to humans. We also knew that most of what was being done with SGML could be done much more easily and cheaply.

Obviously, it took on not one but a hundred different lives of its own, many of them still frankly astonishing to me. I've noticed that people who help build general-purpose technologies are usually bad at predicting how they'll be used.

AU: *While XML is known as a remarkably flexible standard, every standard inevitably includes constraints. If you had it to do over again, are there any things you would do differently based on how you've seen technology and usage develop?*

TB: Oh, yes. The big thing we'd do is leave out DTDs. In the real world of data interchange and processing, schemas are second-class citizens. Also, DTDs brought along with them a bunch of features which turn out to be less than useful or arguably even actively harmful.

Also, XML Namespaces, which were done at more or less the same time as XML 1.0, get a lot of hate. There are a couple of pieces of that design that could be improved, and there's also a case to be made that they actually could have been dispensed with.

There are a bunch of other pieces of fine-tuning that we can see in the rear-view mirror, but those are the big ones.

AU: *How might the Internet and the Web look and function differently today if you'd gone down that road instead?*

TB: Not much, to be honest. We're fortunate in that it was possible, in practical terms, to either ignore or work around the irritants in XML.

AU: *Fundamental standards often influence many other standards decisions, and also how architectures evolve in a broader sense. Do you see such wider effects on how the architecture and/or infrastructure of the Internet and Web have evolved that you can trace back to the creation and success of XML?*

TB: First, XML is an existence proof of the possibility of data-interchange formats that are language-neutral, OS-neutral, database-neutral, and so on. We've had a couple more since XML that have gotten some traction: YAML [*ed.:* a much more human readable data format] and especially JSON [*ed.:* Javascript Object Notation, a data exchange alternative to XML, often used in Ajax programming]. I particularly like JSON for the things it's good at, which pleasantly enough mostly happen to be things where using XML is awkward.

Why on earth did it take until the late 'Nineties before someone cooked up a neutral data interchange format?

The notions of Web Services and especially REST depend crucially on the assumption that you can ship things around the infrastructure that any flavor of computing infrastructure can produce and any other flavor can consume. These days, any time there's an argument as to whether some information resource should be open or not, it is a pure policy argument; because of XML, there are typically only minor *technical* barriers to opening up information. That seems like the big deal to me.

II The Present

AU: *XML has been adapted to handle everything from sports information, to advertising handling, to human resource data, to financial reporting information. Where do you think its impact has been greatest?*

TB: The most successful application, in terms of volume of information and number of users, has been syndication: Atom and RSS. Also, offerings like Amazon Web Services depend crucially on XML. But look behind the firewall at any large enterprise, private or public sector, and you're apt to find a whole bunch of XML sloshing back and forth being used to stitch different applications and components together; in many cases even when they weren't designed for such integration.

AU: *Are there any areas where you're surprised that XML isn't yet being used to its full potential? Which, and why do you suppose that is?*

TB: Some of us hoped that XML would replace a lot of the usage of HTML on the Web, simply because dealing with real-world HTML is such a major pain in the butt. That hasn't happened, simply because the cost of HTML parsing is already a sunk investment, and so there was no real upstream pressure to produce XML.

XHTML has been a success and quite a few of the better Web designers use it just because that eliminates certain classes of problems you can run into. But now we see that the HTML5 project is moving in quite a different direction; its leadership is actively disdainful of XML.

The real answer to your question, though, is that XML is being used far beyond what any of us could have dreamed its full potential to be.

AU: *We both remember the ODF - OOXML competition well. If that process revealed any flaws in the standard setting infrastructure and process, what were they?*

TB: This was my first exposure to the ISO/IEC JTC1 process and culture, and I was horrified at the pervasive corruption and incompetence. I would prefer never to work in that context again. I would be eager to participate in a reform effort, if there were the political will to launch such a thing.

AU: *Do you think the existing IT standards development structure (e.g., the ISO/IEC process plus innumerable consortia) is sufficient for today's demands, or do you think we need new types of organizations, such as ones that would rate the "openness" of standards developers?*

TB: I have experience in the W3C, IETF, ISO, and OASIS contexts. Among those organizations, I find I generally prefer the IETF culture and process. Having said that, standards are created by people, and the individuals who end up as committee members, editors, and chairs end up having a huge influence.

I'm unconvinced that the world needs any new standards organizations.

AU: *The Obama administration in the US has pledged to spend tens of billions of dollars on several major technology based initiatives that involve masses of data - a major electronic health record initiative, as well as a total redesign of the electronic power grid that is intended to turn it into an interactive ecommerce platform. Similar efforts of varying size are in progress in other countries and regions. What role do you see XML playing in these enormously expensive undertakings? Will they require further development of XML?*

TB: Almost by definition, a high proportion of this information, especially in the health sector, takes the form of documents. If you want to represent documents in a form that's open, highly interchangeable, re-usable for unforeseen purposes, and free from vendor clutches, you really can't beat XML. So I'm assuming that it will be the default choice for a lot of this stuff.

On the other hand, some of the work, for example in "Smart Grid", seems to me like it involves interchanging numbers and database records rather than documents; something like JSON may be a much better fit.

Of course this doesn't mean that the costs, complexity, and openness in these projects won't be driven in the wrong direction by technology vendors and especially blue-suit consultant solution providers, whose business interests are not aligned with lightweight, open, flexible, technology deployments.

AU: *What's being done with XML 1.0 and 1.1 development today that you think people should be aware of?*

TB: Not much. XML 1.0 pretty well just works. The XML group at W3C continues to tinker with internationalization, mostly because Unicode is a moving target. I don't agree with some of the stuff they've

The big thing we'd do is leave out DTDs. In the real world of data interchange and processing, schemas are second-class citizens

done, but on the other hand it doesn't seem to be actively harmful. XML 1.1 was a mis-step, which fortunately has been largely ignored by the marketplace.

III The Future

AU: *First we had XML 1.0 (in 1998), which is now in its 5th edition, then XML 1.1 (in 2004), now in its 2nd edition. Naturally, people talk about whether there should be an XML 2.0. In your view, should there be, or has XML, like SGML, reached a point where any thing significantly different should mark the launch of a new standard?*

TB: I don't think XML needs any more features. I suspect that opinion is widely shared. I have proposed something I called "XML-SW" where SW stands for Skunkworks, which is just a cleanup. See <http://markmail.org/message/hzxocbofmmmgxeah> and <http://www.textuality.com/xml/xmlSW.html>

It integrates three or four of the low-level XML 1.0 standards that everyone implements: namespaces, the information set, xml:base. Also it decouples DTDs. Finally, it reorganizes the XML specification to make it more readable and usable. I think the resulting document is quite a bit cleaner and more useful to implementors. But realistically, the world seems to getting along reasonably well without it.

AU: *If it's time to go to some sort of "next generation" XML, what do you think its mission should be?*

TB: I don't think it's time. The world is reasonably well-served by the XML and JSON tandem for information and document interchange. Let's invest at the higher level, in applications and data resources that impact users, not the boring stuff in the engine room.

AU: *To what extent, and how, do you see the Semantic Web as an extension of XML?*

TB: Not in the slightest. The fact that RDF has an XML syntax is an unfortunate historical accident, because XML was definitely flavor-of-the-month at the time RDF was being built. Also unfortunate because that syntax is horrid; hard to read, hard to write, hard to work with. I am well-known to be generally a Semantic-Web skeptic anyhow; there has been considerable energy and hype going into the project for a decade or so, and remarkably little useful software coming out. By the way, it seems that the Semantic Web has now been rebranded as "linked data."

AU: *Looking way out into the future - say 10 or 15 years - where do you see data creation and sharing headed? What should we be able to do in the future that we can't do now, and what role will standards have to play in order to make that possible?*

TB: That, quite properly, is a matter of policy not technology. The barriers for sharing information are not technological in any crucial way. Where there is the political will or business case for sharing information, you can start now; no need to wait for technology.

I think the most interesting thing going on in the world of information sharing is the advent of low-cost mobile-phone technology in the underdeveloped portions of the world, bringing the benefits of the Internet, albeit in a less-polished forms, to a couple of billion people who stand to realize benefits which will impact their lives more than the Internet has impacted ours.

AU: *I wrote a piece recently called [Digitization and the \(Vanishing\) Arts of the Book](#). As the world moves more and more from fixed to electronic media, do you think that we need to make more room for aesthetics in standards development? If so, how would we go about that?*

TB: In response to your piece, I have to point out that electronic display media have been playing catch-up these last few decades. Paper display technology offers immensely higher resolution and a vastly larger palette of colors compared to any electronic medium, and has been more convenient to carry around and use. With things such as the Kindle, we're making progress on convenience, but I think we're still years and years from catching up on resolution and color.

Now, if you're reading my blog posts, a popular novel, or a Humanities textbook, who cares? I live in a part of the world where forestry is an environmentally fraught issue, and I have no patience with the cutting down old-growth timber to print Stephen King (and I like Stephen King). I'm a book lover, my house is stuffed

with them; but in the future, the preserve of books will (properly in my opinion) be the antiquarian domain and those places where high-quality display is essential: Art, coffee-table books, graphically-intense textbooks; perhaps poetry.

Now, you asked about aesthetics in the standards domain. Engineering aesthetics are a different kind of beast; we worship at the temples of simplicity, flexibility, and minimalism. Which are only occasionally appropriate in the world of human aesthetics.

AU: *This has really been great, so just one last question: How long have you been wearing the <hat>?*

TB: A couple of decades. I've always like wearing a hat. On top of which I'm a pale white bald guy, and developed some lesions on my head that had to be blasted off with liquid nitrogen; ouch! So thus my fashion sense and medical advice are pointing in the same direction.

Copyright 2009 Andrew Updegrove

Sign up for a [free subscription](#) to **Standards Today** at

<http://www.consortiuminfo.org/subscribe/2.php?addentry=1>

Standards Today is a free bi-monthly electronic Journal sponsored by the Boston law firm of Gesmer Updegrove LLP. The current issue of **Standards Today** and a subscription form may be found at www.consortiuminfo.org/bulletins. Questions or comments about the articles in this issue or about ConsortiumInfo.org may be directed to Andrew Updegrove at updegrove@consortiuminfo.org, or by telephone at 617/350-7800. To learn more about Gesmer Updegrove's standards and open source practice, visit http://www.gesmer.com/practice_areas/consortium.php, or contact Andrew Updegrove.

© 2009 Andrew Updegrove. All rights reserved.