Another rant about how XML underneath your word processor is not enough

Jon Udell returns to XML word processing document formats, while ignoring the current standards debate.

> I have steered clear of the politics surrounding XML document formats both before and after joining Microsoft. But I was, and will always be, an outspoken advocate for the idea of XML documents. That's a message that doesn't make headlines but bears repeating. We have hardly begun to appreciate or exploit the value of XML. A couple of articles in the current issue of CTQuarterly, a journal about how cyberinfrastructure enables science, illuminate that point.
>
> In Next-Generation Implications of Open Access, Paul Ginsparg writes:
>
> > One of the surprises of the past two decades is how little progress has been made in the underlying document format employed. Equation-intensive physicists, mathematicians, and computer scientists now generally create PDF from TeX. It is a methodology based on a pre-1980s print-on-paper mentality and not optimized for network distribution. The implications of widespread usage of newer document formats such as Microsoft's Open Office XML or the OASIS OpenDocument format and the attendant ability to extract semantic information and modularize documents are scarcely appreciated by the research communities.

I'd like to pick up on one of the key things you'd want to do with any document. **You'd want to put it on the web**. Never mind integrating semantic data, such as I demoed with CML for chemsitry data, the first thing is to be able to publish it properly. Then we'll talk about easy-to-use integration with e-Research tools. Really, we will. But can we learn to make web documents? The web is in its late teens and word processors don't get it yet.

(And when we do talk about integrating data into documents I don't think the Microsoft Approach of dumping arbitrary XML in the midst of documents is going to be the answer for most of us. That way lies expensive development and lots of ongoing maintenance. I think we'll get a lot more traction with a microformat type approach, where we co-opt features that are already there and tie them to other services.)

It's not sufficient to use OOXML or ODF, to make research accessible **you need to be able to make web pages, not just PDF**.

The word processors that use OOXML and ODF don't make HTML very well at all.

Regular readers know what comes next. If you want to (a) use a word processor and (b) make your research available in more than just PDF then you need to find tools that support you. One such tool (and there are not many, I assure you) is ICE, which Jon has linked to before with kind comments.

ICE gives you a default-but-extensible stylesheet, and you can use it to write articles, theses, books and so on, with both PDF and HTML output – this is the point of the ICE-RS project.

Early next week we release a update of ICE which talks ATOM publishing protocol, which means that you can post to the growing number of ATOM-ready web services. When more of them support media libraries we'll add support for images and PDF as well.

Daniel de Byl has lef the silent era of screencasting and is now speaking when he does his demos. See him in action demonstrating atompub from ICE.

But as Tom Worthington notes ICE is still a bit hard to get going with, so beyond next week we're about to release a few things that should really help:

1. A server-based version of ICE. Nothing for authors to install but the Word template or OpenOffice.org toolbar and they're away.
2. A server-free version so that you can convert documents from your word processor and do what you like with them (including using ATOM to push them to a server).

   Initially you'll need to download and install OpenOffice. org and ICE which together perform the conversion. But once installed you can then ignore them and use Word to write if you wish.

   Later we'll probably hook it up to a web service too so you don' t need to run much locally at all, which brings us to;
3. A RESTful web service so you can get all Web 2-dot-0 with your other web apps.
4. A commandline tool (and/or Python library) that you can use as part of another application.

Oh, and remember, people, to use styles.