Scholarly HTML

A few weeks ago, I felt there was something in the air to do with desktop tools for eResearch. We went with that in the form of The Desktop Fascinator and have been enjoying a productive conversation with others in the, um, what's the word for the twitter-o-blog-o-sphere?

This week I feel the time is right to look with fresh eyes at something we have been over and over in the ICE project; the place of HTML in scholarly publishing. There have been a couple of things happen recently that made me look at the issues in a new light.

I'm going to propose here an application profile of HTML for scholarly works, with some conventions for representing an article or chapter as a single web page, with referenceable paragraphs, semantic web via a linked data approach, simple inline citations and protocols for representing quotes, examples etc; “Scholarly HTML”. This is only a working title and it's a bad one because if you search for that you get a lot of hits for filenames; `scholarly.html`.

I am sure I am not the first to think of some of these ideas – any pointers to prior-art would be appreciated.

Background

Lets start with Rick Jelliffe, who talked about Dennis Hamilton's notion of standards floors and ceilings in relation to my relentless ranting about HTML:

> Dennis Hamilton over at the ODF TC made an interesting point recently about *floors* and *ceilings* and interoperability: both are needed. I think one of Peter's main points is this issue of having an adequate floor: that HTML should be this floor but its support by desktop vendors in their applications continues to be ratty.

I like this idea of a floor and that's where I think we need to work if we want to mobilise the micro-crowds that John Wilbanks talks about:

> The difference is that .005% of all web users gets us Wikipedia. .005% of geneticists gets us a table at T.G.I. Friday's. My point was that the math breaks down for crowds and science.

If we further break that down into 50% of the geneticists using LaTeX and 30% using Word 2007 and 20% using something else (I made this up what do geneticists use to write their papers and theses?) then we have made an even smaller pool of users who will be able to tag their work with terms from genetic ontologies. I'm talking about using the Word 2007 add-in that John Wilbanks has been working on with Microsoft. But what if we worked out a way to do semantic-web-scholarly publishing in HTML that would be useful for all of the people at the table at T.G.I. Friday's, whatever that is.

Then we have Mark Pilgrim doing his update to Dive into Python in HTML. Bye bye DocBook. Hello HTML.

> Anyway, I now realize that there were some hidden assumptions behind my design decisions in 2000. Some of those assumptions turned out to be wrong, or at least not-completely-right. Sure, a lot of people downloaded DiP, but it still pales in comparison to the number of visitors I got from search traffic. In 2000, I fretted about my “home page” and my “navigation aids.” Nobody cares

> about any of that anymore, and I have nine years of access logs to prove it.
>
> So, I am writing DiP3 in pure HTML and, modulo some lossless minimizations, publishing exactly what I write. This makes the proofreading feedback cycle faster — instead of "building" the HTML output, I just hit `Ctrl-R`. I expected it to make some things more complicated, but they turn out not to matter very much.

There's no discussion in this post about what the publisher will do to make it into a book, but whatever toolchains these people use should be able to be adapted to HTML, particularly if there was a profile which people could use to say 'this blockquote is an example'. There are other bits which I'm sure could be cleaned up too with a bit of jQuery – like automated numbering of examples or headings.

I am guessing Mark Pilgrim used emacs to write, sorry 'lovingly hand craft' this thing, but with the set of guidelines he gives for how to format stuff you could do it in a wiki, or WordPress or you guessed it, ICE. And as for cruft in the HTML, well provided people use the key things we care about, like headings, and block-quotes and so on then we can strip out the rest.

ICE makes PDF and HTML – but for the purposes of a new scholarly publishing model we can set PDF aside as something we will generate later on, from whatever is most convenient. The HTML is the bit that matters now.

This reminded me of Peter Murray Rust on PDF:

> Why should documents have page numbers? why should be have two columns per page. Because the publishers force it on us.

And what Tim Bray said, ages and ages ago before he went quiet on this whole topic. Here's Jon Udell on Tim Bray:

> Musing on the European Commission's findings, Tim Bray wondered if maybe XHTML had gotten short shrift:
>
> > They considered, and rejected, XHTML as a standard office document format. I think that it can do most things you need in a modern office document and has remarkably few real drawbacks. [ongoing]
>
> I'm not ready to go along with the other conclusion he reaches in that posting -- that custom schemas are a red herring. But I agree that XHTML is more valuable than most people think. For the vast majority of useful documents, it can have as much structure as we need, and for the rest it can be extended internally with namespaced inclusions. But the real power arises from its hypertextual nature. For me, increasingly, **there is no office, and there is no desktop, there is only a network of linked documents**. A successful open document format will have to be supremely well-adapted to that environment, as XHTML is.

Rules of the game

As we're talking about laying a floor here rather than swinging from the ceiling I want stuff that could be authored in a generic HTML editor, or in any other tool that can produce HTML, like a wiki, or Moodle, or WordPress, or using something like Google docs. So that means really simple mechanisms and microformats and nothing that depends on anything that costs money to use.

What I'm thinking is, everyone can do links so lets go crazy with links.

Unfortunately even RDFa is out, as you can't do it via a word processor or a typical WYSIWYG HTML editing tool. I reckon adding the rel attribute it too hard but what would make it easy for everyone is if the CC people set up a has-licence page where the rel is part of the endpoint. So this example from the RDFa primer:

```
All content on this site is licensed under
<a rel="license"
href="http://creativecommons.org/licenses/by/3.0/">
    a Creative Commons License
</a>.
```

Becomes this, where you bundle the predicate and the object together and link to them from the subject.

```
All content on this site is licensed under
<a
href="http://creativecommons.org/licenses/by/3.0/haslicense">
    a Creative Commons License
</a>.
```

The trick is not to link to an identifier for the license but an identifier for license-plus-verb.

What it should be like

Here's a list of stuff the Scholarly HTML should do, off the top of my head. Argue with me:

- **Headings. Documents should definitely have headings.** They don't need nested sections because the headings are enough for a machine to work out the nesting. If a journal/discipline really really must have some fixed structure you could validate that the first heading element contains the text 'Introduction' and so on.

  The page should have title, and the title should be put in the top of the document in an `<h1 class="title">` element. Thereafter the next heading should be a plain-old `h1`. If naughty users jump from `h1` to `h3` we'll just normalize that back to `h2` later on.

  If you really really need more formal structure then how about putting an empty link in the heading to a web page for something like an NLM section, meaning this 'h1' is the same as that NLM element. But do you really need all those elements if the paper *is* the HTML?

- **Protocols for representing things like examples**. Mark Pilgrim says he uses blockquotes, but there should be some class that lets you tell an example of your own from a quotation. We don't need a standards committee yet, we need some examples.

- **Metadata, using a linked data approach** (which I am going to describe in another post, but which I have touched on before).

  > What if my local ePrints repostiory had a page for me-as-author. It might look like http://eprints.usw.edu.au/authors/PeterMacolmSefton and resolve to a page that describes me and the works that are attributed to me with a note there that says "if you want to indicate that

this person is an author of a paper, link their name as it appears on the work to this page". That is way simpler to implement than the stuff we talked about in our paper on embedding semantics in word processing documents.

The journal itself could provide is-author pages for those who don't have them yet as they collect data about the authors in their journal websites anyway. We'll work out a way to relate the is-author pages together later. We have to start somewhere.

- Links from terms mentioned in the text to **ontologies** that describe them. This is going to be kind of silly if all we do is link the same term to the same ontology over and over again like the discussion about 'Potter syndrome' we had here last week as that string unlikely to be ambiguous, but could be useful if you link different strings of text to the same ontology. Like, say you could link "distinctive lightning shaped scarring on forehead" to Potter syndrome.
- **Linkable paragraphs** – via a mechanism like Tim Bray's purple pilcrows. These could be used to cite the paper but could also be key to stand-off annotation/discussion services. This does not need to be done by the author – we can make scripts that do that when the page crosses the curation boundary. ICE has cute tricks it uses to provide Ids for pages that are partly based on their content so we can tell when a paragraph has changed, but keep it around if someone has commented on it.
- **Dead simple reference management** via links to trusted sources. In many disciplines the manuscript should be able to do what Peter-Muray Rust said and just use a DOI – the publisher can be responsible to fetching the metadata and making a bibliography. How about you let the user choose the referencing style and select it on the fly using Zotero's growing CSL library? Or skip that altogether an let readers import the references into their own libraries.

Illustrations? I don't know, that's one we will have to work through. Maths likewise.

Out of scope for now is how ORE would be used to describe a data-supported paper with the text and all its associated images and data files as an aggregation. This is partly because there are very few end-user tools that would let someone package a document in this way but I think they will start to appear. There's one for WordPress for example.

Beyond just putting stuff on the web, I think this approach would make it simpler to write lots of the tools that are going to machine-read the semantic-scholarly web; make them all work on Scholarly HTML rather than trying to deal with several different upstream formats.