Lemon8-XML: Interview with MJ Suhonos

Martin Fenner, Gobbledygook

February 27, 2009

Finishing an exciting research project and writing it up in a paper are the first two steps in getting your work published. The next two steps – submitting your paper to a journal and getting it through the review process – have changed dramatically in the last 10-15 years. No longer do we have to print out our manuscript using one of the few available laser printers in the department, paste our gel pictures on cardboard and number the figures with Letraset. And then send it off with the mail. And then repeat the process for every revision of the manuscript.

Now of course we submit our manuscripts online using a manuscript submission system such as Editorial Manager or eJournalPress. Which is not to say that the process is necessarily easy or fun. Many of us can tell stories of spending hours or whole days until the manuscript is finally submitted. We struggle with the conversion of the different parts of the manuscript into a single PDF file, have problems with fonts, have to deal with different graphics formats (e.g. PDF, JPEG, EPS, TIFF), don't use the correct style for our references, etc.

The flip side of this is the time and money spent at the journal to format your accepted manuscript into something that can be turned into a journal article published online or in print.

Some of these problems wouldn't exist if we used a common document format for manuscript writing, manuscript revisions and manuscript printing and online viewing. That common document format does exist and is called NLM Journal Publishing DTD (and no, it's not LaTex). This document format is used in the workflow of many journals, but until now has rarely been used by authors. Last November I talked with Pablo Fermicola about a free tool that allows Microsoft Word to save manuscripts in that format (Microsoft Word Authoring Add-In). eXtyles NLM is another tool for Microsoft Word with similar functionality.

Lemon8-XML takes a different approach in helping academic authors convert their manuscripts into the NLM DTD format. Version 1.0 of the software was released today, so this was a great opportunity to talk with the lead developer MJ Suhonos about Lemon8-XML.

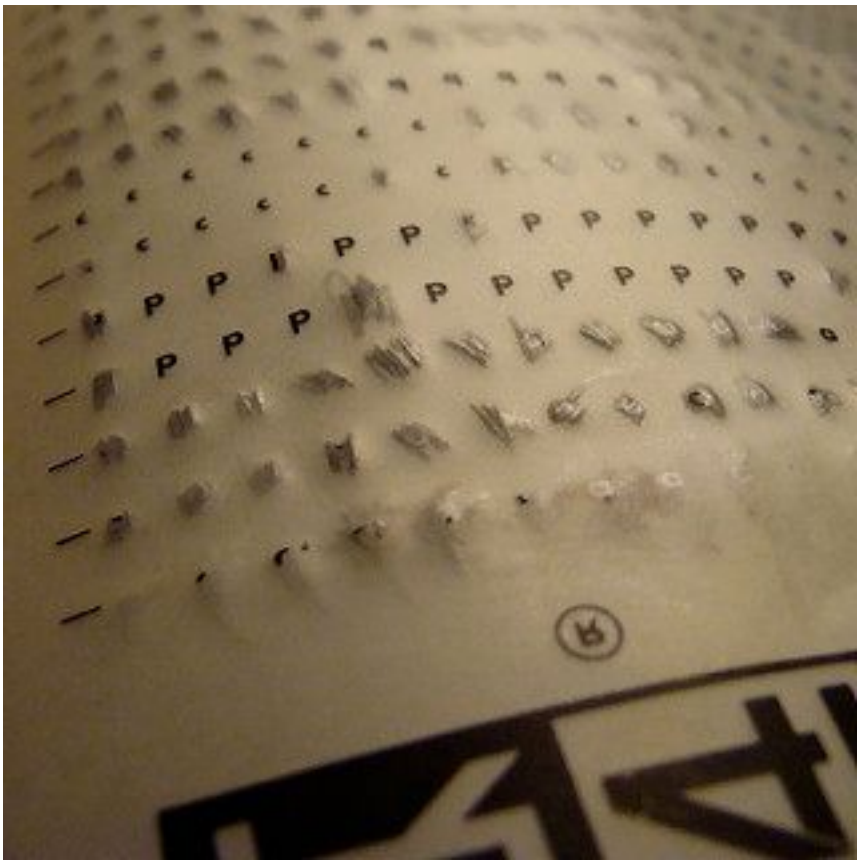

Figure 1: Flickr Photo by synaptic.happenings.

1. Can you describe what Lemon8-XML is and does?

Lemon8-XML is a freely-available, open source web application that aims to help academic authors and editors convert scholarly articles from layout formats like Microsoft Word to structured XML formats without requiring a great deal of knowledge of XML or the schema that they're working with. It's based on the OpenDocument format internally, and the NLM Journal Publishing DTD as the default export format.

Originally, Lemon8-XML came from a bunch of scripts that performed a series of tasks:

- convert an author's article from a myriad of document formats to OpenDocument,
- parse the article and try to extract metadata, determine section structure based on layout information, and parse citations into their disparate elements,
- export this semantic data as XML.

The work to date on Lemon8-XML has been building these functions into an easy-to-use web UI that also provides some editing facilities; for example, to fix problems from incomplete or incorrect parsing. One of the most promising features of Lemon8-XML is its ability to parse citations in a wide range of formats, and automatically try to correct or enhance them by searching for their complete records in, eg. PubMed, CrossRef, and WorldCat.

2. What are the advantages of using Lemon8-XML over traditional word processors such as Microsoft Word?

I should start by clarifying that Lemon8-XML wasn't really intended to be used as a word processor, but rather a conversion tool to be used after the writing was done. This was to fit in with existing practices, where authors upload papers to a journal and the editing is done afterwards. Of course, the advent of online word processors challenges this notion, and has caused us to think differently about the editing aspects of Lemon8-XML. Cross-platform deployment and online collaboration are things that obviously the web is great for. An author could, for example, upload a paper to Lemon8-XML, and work together with an editor on that same copy, regardless of what desktop tools they have. One advantage of this idea is that the burden of work is shared between more people, whether that's an author and a copyeditor, or multiple editors. This is a big deal for small journals who don't have funds for dedicated staff.

Within PKP in general, we also try to keep the requirements for our software as low as possible, both technically and financially. So, for journals like Open Medicine, who are entirely volunteer-run, paying $230 per copy for Microsoft Word on each computer actually represents a significant cost. Using OpenOffice on the desktop and a single copy of Lemon8-XML on a server — both of which are free — lets them spend this out-of-pocket expense on other things to keep

their journal alive and growing. Microsoft Word also has an unfortunate history of being variable and inconsistent across platforms and versions. We wanted to help people escape that, which is one of the reasons for converting everything to OpenDocument as early as possible. In addition, stewarding documents through an online workflow can be a laborious and frustrating process, so the possibility for journals to make the editing process more centralized and interactive by integrating Lemon8-XML is a compelling one.

3. How does Lemon8-XML compare to the Microsoft Word Article Authoring Add-In, that also produces NLM Journal Publishing DTD output?

In many ways, we're working in parallel with Pablo Fermicola's group at Microsoft — they're basically building an editor for placing a structured NLM schema on top of the Microsoft WordML (DOCX) XML format, while Lemon8-XML places the NLM schema on top of the OpenDocument (ODT) XML format. Of course, there's nothing saying that Lemon8-XML couldn't be modified to read DOCX files generated with the Article Authoring Add-In or vice-versa. This difference is really a reflection of the two competing standards, and the tools they use, with the same ultimate goal: to allow users to generate semantically-structured documents from layout-based ones. In addition, the Article Authoring Add-In is built on Microsoft Word 2007 as a platform, which I think has some limitations; for example, most of the PKP team don't use Windows, so the Article Authoring Add-In is basically inaccessible to us. But overall, our goals appear to be very similar.

To my mind, the main difference between the projects is in how they approach the user: the Article Authoring Add-In presents new tools for adding semantic mark-up in a (somewhat) familiar interface; many authors are already familiar with Microsoft Word and are comfortable working in that environment. My concern with this approach is that it still has strong ties to the layout paradigm, so, for example, marking text as "article title" may not be sufficiently different in authors' minds from marking text as "16 point bold". I also think the idea of presenting an entire document in a single free-form editor reinforces the layout paradigm, as opposed to identifying the distinct elements which comprise a scholarly article. Lemon8-XML, on the other hand, builds from these individual elements, and assembles them within the user interface based on their structural relations. You can see this reflected in the tabs in the current interface: "Metadata, Sections, Citations" — these are the front, body, back matter of an article.

This places restrictions on how a user can edit their article, which is sometimes frustrating, but it forces them to think about what they're doing and the meaning of their content: why do I have to place it here, what content is valid in this element, etc. In a future version, this will all be on the same web page, similar to how it would appear in Microsoft Word, but again with the semantic structure visually enforced instead of being totally free-form.

4. What is the difference between Lemon8-XML and online word processors such as Google Docs?

Unlike most word processor software, Lemon8-XML is built around the semantic notion of a document, not its appearance. We wanted to help people begin to think about the meaning and structure of their articles, not just whether they look good on the screen or as a PDF. It turns out this is a tough challenge, though — people have a hard time with WYSIWYM (What You See Is What You Mean) editors, and there's often a lot of complex structure to present visually for editing. One thing people seem to be comfortable with, or at least used to, is entering metadata and information in web forms, so much of the Lemon8-XML user interface is built that way.

Lemon8-XML is also specifically aimed at modeling and editing scholarly articles, which have a long history of practice and some very rigid conventions that aren't applicable more broadly. This means it's not a general-purpose editor like most word processors, but that also means we can focus on the specific things it should do, and refine them rather than trying to please everyone and falling victim to feature-creep.

5. Why did you pick the NLM Journal Publishing DTD as document format?

I did a survey in 2003 of available DTDs to represent scholarly journal articles, looking for something I could use as a common source for generating HTML and PDF. I selected the NLM DTD above the others (eg. BioMed Central, DocBook, Erudit, Text Encoding Initiative) since it seemed to strike a good balance between comprehensiveness and granularity; it can be as simple or as complex as you need it to be. It also had a very thorough citation model and a modular design, so extensibility wasn't a concern — I liked the "journal articles plus" concept that it was built with. And, of course, being stewarded by the National Library of Medicine meant it would likely be well-maintained and remain open. The fact that it became the central XML standard for PubMed Central as an archival format was just icing on the cake.

Another reason is that, among the STM journals at least, a lot of established practice has developed around generating HTML and PDF from NLM XML specifically, and we want to support that. Actually, we want to expand adoption of the NLM DTD by making it accessible to humanities and social science journals, as well. The information contained in the XML is essentially identical regardless of subject, which means we just need to develop discipline-specific rendering; for example, for different citation styles. I think there's an incredible amount of room for growth and improvement in this area in particular.

6. Does Lemon8-XML integrate with manuscript submission systems such as Open Journal Systems?

Not yet, but that's the next major area of development immediately following the 1.0 release. OJS has been so successful, one of our biggest challenges has been keeping it flexible enough to support a wide range of workflows, and we want to continue with that by taking as much of the technology from Lemon8-XML as possible and folding it into OJS. I don't know if we'll see a side-by-side kind of integration immediately, but rather strategic enhancements of certain aspects of OJS: automatic document conversion, annotation directly within articles, extraction and stripping of metadata to improve blindedness, automatically parsing and linking citations, etc. By going this way, we give users the ability to choose specific features that are useful for them, and at the same time build upon the huge community that already exists for OJS in terms of development, testing, and feedback.

There's definitely still value in a stand-alone Lemon8-XML for cases where people don't want journal workflow, or where they're integrating with a different kind of system. So, we will be working on a Lemon8-XML 2.0 that's built on the same modular, reliable framework as the rest of the PKP suite, and back-porting code from OJS.

7. What is the Public Knowledge Project?

PKP is a small research group distributed between Simon Fraser University, University of British Columbia, Stanford University and Arizona State University that has been quietly developing open source software for online publishing and knowledge sharing for the past 10 years, under the direction of Dr. John Willinsky at Stanford. Our major aim is to provide tools for improving access to academic research, as well as helping improve the quality and efficiency of its production. Our philosophy is also to create partnerships between researchers, librarians, publishers, to help them build sustainable and globally accessible scholarly infrastructure. In my mind, it's also about giving people options and choices aside from what's being offered commercially, especially to those who can't afford them, like a lot of developing-world journals.

We have four applications in addition to Lemon8-XML: Open Journal Systems, our most popular, is being used by over 2500 journals in over 50 countries. We also produce a conference management system (Open Conference Systems), an OAI metadata indexing system (Open Archives Harvester), and a monograph publishing system (Open Monograph Press), currently under development. All of our software is freely available under the GPL open source license, and we have an active community of over 1500 users. We've worked with small, volunteer-run humanities journals, to major international society conferences, to high-profile, ISI-ranked medical journals.

8. What are your responsibilities within the Lemon8-XML project?

Because our group's so small, we all share responsibilities — there are only a handful of people to do almost all of the development, and that's in addition to handling the support, correspondence, and software maintenance of the rest of the PKP suite, not to mention providing various international workshops. So, we often divvy roles based on expertise or interest. Since bringing Lemon8-XML to PKP, I do the vast majority of Lemon8-XML development, but I also manage collaborations with our development and research partners, and write white papers on the software design we use to share our ideas, on top of contributing to the daily stuff above.

Probably the biggest struggle in developing Lemon8-XML has been that it's one person, about half-time, doing all of the coding, testing, etc. We already have a number of very exciting partnerships with groups who will be testing and providing feedback, and have contributed a talented developer who's recently started half-time as well. Of course, we can always use more help with the coding side of things. Our team is quite spread-out geographically, so we have a lot of experience with working across time zones and with distributed development practices. I'm hoping that after the 1.0 release, more people will become interested and will want to help get involved. Whether contributing plugins for citation processing, or helping us improve the UI — really, anything that helps the software grow will benefit more authors and editors, which is the main goal.

9. What did you do before starting to work on Lemon8-XML?

I worked for a two-person medical informatics journal that was publishing around 50 articles a year using Microsoft FrontPage and managed entirely via email. My job was to transition it to an online manuscript management system (naturally, I chose OJS), convert around 300 back articles from FrontPage HTML into valid NLM XML, develop a custom rendering system that would create HTML and PDF galleys from that XML, get the journal accepted into PubMed Central, and establish an impact factor from ISI (which is now 3.0) — oh, and continue publishing 50 articles a year. The journal also ran an international medical conference during that time (using OCS). Three years later, I left to do my Masters degree in Library/Information Studies, during which time I joined PKP and started working on Lemon8-XML, based on a lot of the things I'd learned from my time at the journal.

10. Do you want to talk about future plans for Lemon8-XML?

The most common question I'm asked is, "when will it be ready?", which for many people really means, "when will it be a one-click magic bullet for getting my journal into PubMed Central?". I'm proud of the Lemon8-XML 1.0 release, and I feel that it represents a significant milestone given how far we've come — but there's still a long way to go before we get to that point. I hope people will

view the 1.0 version as a stable tool that's already being used in production by at least one journal, and a strong indicator of PKP's commitment to developing research and software in this area.

I mentioned the Open Journal Systems integration, which is both a strategic and pragmatic move; this will be our main focus, as well as integration with OMP from its earliest inception. We also want to improve the editing aspect of Lemon8-XML — there is some great work being done in the area of using style templates for providing structural mark-up in traditional word processors; we will be working more closely with Peter Sefton's group on the ICE project, who have a lot of experience. I'd also like to re-visit the idea of using web-based WYSIWYM editors like WYMeditor or an enhanced TinyMCE as part of a user interface overhaul, and possibly integrating more closely with Google Docs if that's an option. Certainly, there's room for adding more citation lookup plugins: OAIster, Amazon.com, CiteSeer, etc. as well as improving the existing ones. Finally, we want to try applying the approach we've used with citation parsing and lookup with other scholarly article elements; for example, checking quotations for correctness and plagiarism, extracting and linking author/contributor identifiers, and so on. We also have some ideas around OpenURL that may or may not come into Lemon8-XML development.

We're also starting a major side-project based entirely around the NLM DTD in two parts: 1) building mappings from various other XML schemas to NLM; and 2) building a standard set of rendering tools for generating HTML and PDF from NLM XML, in a way that can be easily customized for an individual journal. There are already a lot of groups out there using NLM but currently the practice is quite fragmented — we'd like to see these journals and publishers become more connected and share their work as a community. As I say, we want to help increase adoption as a way to raise the bar on journal quality and improve options for publishing and archiving.

One of the most important things we've learned during the development of Lemon8-XML has been the value of user feedback early and often, and remaining aware of related work that's going on, so we can remain efficient by building partnerships instead of working in parallel. We'll continue pursuing this approach, and I'd encourage anyone who is interested in any of these areas or has ideas of their own to contribute to get in touch with us.