A courseware authoring dashboard using Schematron

As with busses, sometimes you can wait ages for a Schematron and suddenly a whole pack of them come along together*.

For those of you who don't know:

> In Markup Languages, Schematron is a rule-based validation language for making assertions about the presence or absence of patterns in XML trees. It is a simple and powerful structural schema language. It typically uses XPath to describe patterns.
>
> (Wikipedia contributors 2008)

Instead of the all or nothing syntactic approach that you get with other kinds of schemas Schematron lets you pick and choose things to worry about. So instead of saying “all course books must begin with a Learning Outcomes section” you can write a rule that simply reports on whether there's a Learning Outcomes section or not without letting there be any variation. Why? In some courses it might be important to add something before that section while I have heard arguments that in some situations specifying learning outcomes upfront scares off potential students.

We've discussed using Schematron to provide reports on ICE content but have never got around to using it. This week it has resurfaced in couple of contexts.

Relevant to **ICE as a course-authoring system**, the Learning and Teaching Support Group at USQ have a checklist, *The USQ course writing guide* which authors can use to see if their courses meet our standards for fleximode courseware. At the moment it's a manual process to tick the boxes. We met with Michael Sankey from LTSU this week, and it's pretty clear that Schematron could play a part in automating lots of the checklist.

As part of our ongoing exploration of how we might create an automatic or semi-automatic **system for inferring structure in documents** Ian Barnes has pointed out that Schematron might play a role there too.

Ian's insight was prompted by a recent post of Rick Jelliffe's about a project to add annotations to a corpus of (presumably) word documents in the the OOXML zip package format:

> The brief was for an organization with a large number of documents from multiple sources, but with each source supposed to use stylesheets. The idea was to make a rules base that would distinguish all the different ways that a few structures (titles, table of contents, potentially citations, etc) were represented. This would allow classification of documents according to the structures found, the discovery of outliers and exceptions (e.g. incorrectly marked up documents, or where additional rules were needed), and automated annotation back to the original documents.
>
> http://news.oreilly.com/2008/08/a-standardsbased-expert-system.html

I'll come back to Ian's structure guesser (or as I like to call it the structure sniffer) in another post and talk here about the possibilities for adding validation or **dashboard** services for courseware written using ICE, via Schematron.

Rick's idea of Schematron rules that can reach inside Zip files would be perfect for the USQ courseware context as our content is in Open Document Format files (actually some of it is Word docs but we convert it to ODF as part of the process). We could translate a lot of the checkboxes in the *USQ course writing guide* into Schematron rules to do things like check that there is a an acknowledgements section in the course introduction. Not only could the system report issues, it could open up the documents in question for you and take you to the trouble spots and insert comments in the documents.

* Although come to think of it I don't think I've ever seen two busses in a row in Toowoomba.

Not everything needs to be seen as a validation issue though, just some reporting would be useful to create a kind of dashboard for courseware. “Module 4 contains no activities” might a worthwhile thing to report along with word counts for various modules and how many citations there are, etc.

Another place we could use Schematron to report on course structure would be in the course organizer, which is part of the IMS package manifest file in every ICE course. An organizer is a kind of table of contents for the course, and it is used to generate the navigation. Schematron could easily be used to validate things such as “There must be a *Study schedule*”, and check things like whether the links to study modules have names that are not just like “Module 1” but convey a bit more about what's in the module.

A few years ago Ron Ward and I were involved in a project that used Schematron. There we used it to validate metadata for documents as they were uploaded into a content management system – Schematron would look for patterns in the metadata and complain when it was wrong. The complaint took the form of an HTML form that the user could fill-out to fix the metadata to the Schematron system's satisfaction. The Schematron rules worked well to create a true declaratively specified interface, but our implementation was a bit inflexible, like my attitude at the time, so usability suffered. Lesson learnt, I hope.

I think that **presenting this as a dashboard** that lets you know what your course is like will be better than presenting it as *validation* which has connotations of centralized control, something that doesn't always go down well in a university, even when we do have agreed standards to maintain.

It will be a little while before we get to implementing this I just wanted to record our current thinking.