# • Snowbooks' submission for the innovation of the year award 2007: XSL Project

Whilst the judging requirements for this award state that it is not about 'technological wizardry', our submission for the Innovation of the Year concerns computers and computing. What doesn't, in today's modern publishing company? An aspect of Snowbooks on which we pride ourselves is that we do not have a separate IT strategy, distinct from business as usual. Instead, computers and programs are the tools we use to make our business as efficient as possible; to free up our time to focus on the more creative side of the business that gives us our spark.

The innovation that we will describe here has saved us, conservatively, a month a year. It has got us extra sales from home and overseas, and has provided us with an invaluable selling tool for retailers, wholesalers, libraries, non-book sellers, other merchants and individuals. It has garnered us column inches and speaking engagements. It has avoided the untold cost of lost sales through inaccuracy or missing data.

We have developed the capability to produce all our paper and online book marketing materials - webpages, AIs and catalogues, in .html and .pdf form - in a four step process that takes two minutes per book title. The process is as follows and the total number of mouse clicks involved at each stage is given in brackets:

First, input all data into the database. We use the Anko Publishing Manager database.

» Export one title's record from the database in XML format (2)

» Open the XML file in Cooktop (a freeware XML editor made available by the OpenSource movement) (2)

» Open the XSL file in the same program (2)

» Click on 'transform' (1)

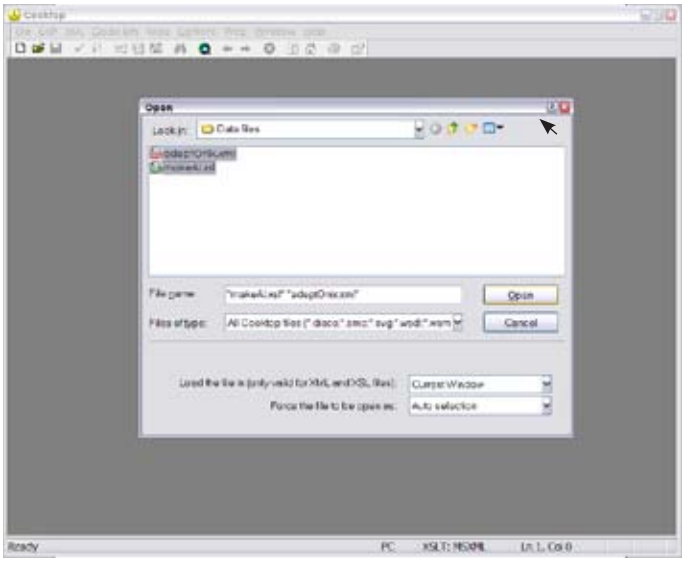
The output file can then be uploaded to snowbooks.com to form part of the catalogue. In total, it requires 7 clicks, which is very swift. (Needless to say, we also use the original XML (ONIX) file to populate our Nielsen records.)

XML is short for eXtensible Mark up Language. It's the language that ONIX is written in, and is a way of tagging up content so that you know what the content is. For instance, here is a way to describe a book:

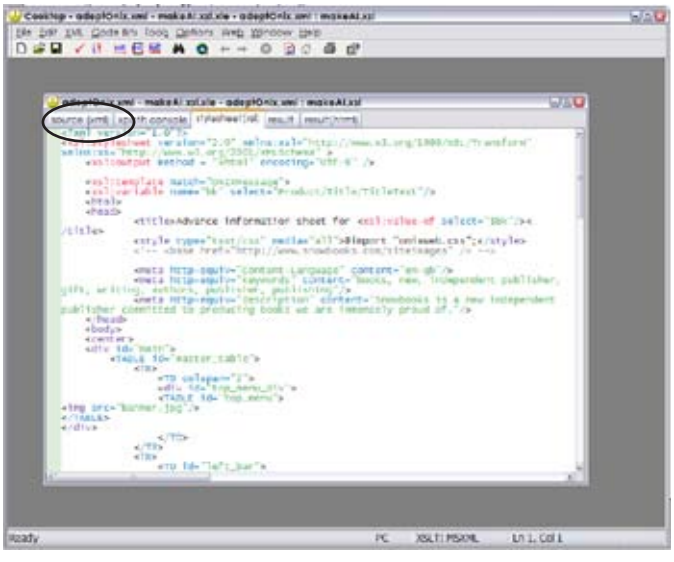
```
<title>Persuasion</title>
<author_first_name>Jane</author_first_name>
<author_last_name>Austen</author_last_name>
<ISBN>9781905005111</ISBN>
```

It's not very complicated, just some content wrapped in tags.

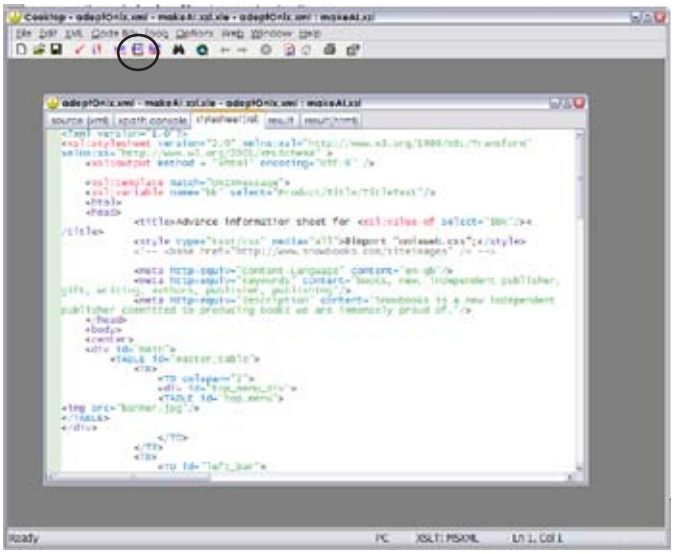
XSL, in turn, understands the XML tags and does what it's told with them. For instance, if we want a block of bibliographic data to feature on our end document, we get the XSL to look for the tags that describe the appropriate content - say, ISBN, Pub date, Format, Size and Number of pages - and to display the data from within the tags.



Open the XML and XSLT files in Cooktop



The open files. This screenshot shows the XSL stylesheet; clicking on the first tab, circled, would show the XML



Next, click the 'Transform' button, circled


The result: an html file (webpage) with data drawn from the original database. Pictures and formatting defined by CSS (cascading style sheets, the gold standard for web design).

What are the main benefits?

» It's very quick.

» It's very accurate. No re-keying of data means no mistakes - only the original database needs to be correct, not ten versions dotted around in different files

» This accuracy means that we aren't losing sales from wrong or absent data. Lost sales are hard to put a value on, but it's an important point.

» It's consistent. Each page has the same type of data.

» It makes our information available for anyone with a web browser to see. I can email a URL of an AI to a customer and they have it immediately. That has been particularly helpful in overseas sales and sales to non-book retailers.

» Once the data is in an XML format, it can be used to populate other files. For instance, we also use XML to fill in Indesign catalogue layouts, and the many different spreadsheets that the different retailers require for titles to be considered for selection. We also use it to populate Word templates - see below.

» It has got us recognised as leading the market - not just among independent publishers, either. I was invited to talk about this at the E4Books seminar in the summer and the feedback was inspiring. Some professional IT managers didn't know this stuff was so straightforward or productive.


XML used to populate fields in a Word document

How did we do it? We had bought the Anko Publishing Manager so had a database containing all our data. It seemed daft to have to keep typing out data for things like AIs, catalogues and so on, so I started to learn about this thing called XSL. I bought a book or two on the subject, read them, and then wrote the code. It's interesting: I am an arts graduate, running a book publishing company, with a love of literature. I am in no way a geek or a technically-minded person, and have no IT training. But with a bit of patience and a burning desire to make my company as efficient as possible, I have learned how to write code from scratch. The motivation was threefold:

» **I hate wasting time.** I'd much rather spend a weekend learning how to use XSL, or another way of automating something, than spend one day a week, every week, retyping or copying and pasting data from one document to another

» **I hate making mistakes.** Rekeying data, no matter how careful you are, simply always results in one or two typos. Before you know it, the data on Nielsen, Amazon, Waterstone's.com and your own website is subtley different. And what happens when the publication date changes, or the page extent? Rather than trying to correct every bit of paper, every partner's website and records, it's much more simple and reliable to have one version of the truth.

» **I hate spending money.** I would much rather read a big, thick, £27.99 book on XSL than pay a consultant £3000 to write an application for me. Firstly, I'd spend £3000 I didn't have to. Secondly, I wouldn't have the knowledge to write the next useful application that I want. By reading the books, and learning this myself, I'm adding to the knowledge assets of Snowbooks and increasing our value as a business.

I thought long and hard about which initiative to submit for this award: we have done lots of interesting things, from enlisting cycling guru Richard Ballantine's services as a consultant editor on our new Cycling list, to launching the SnowCase, a new way for unpublished authors to get their work showcased on our blog. But I think that using XSLT to generate all our marketing materials has saved us more money and time than any other initiative we have done, and is a great example of how you don't need formal IT training, or a budget of any description - just effort and determination - to have world-class data management.