Indexing in an XML context

Caroline Murray

Based on Cambridge University Press's experiences, this paper provides an overview of the advantages and disadvantages of using XML to index books prior to typesetting.

For the last three years, Cambridge University Press's Academic groups (publishing tertiary-level textbooks, reference titles and monographs) have been using an indexing process which enables the index to be compiled during the copy-editing stage of the book's production, so that the index can be run out with the first text proofs.

Several separate reasons lay behind this development. Firstly, the decision had already been taken to capture the content of all our titles in XML mark-up language, so that in addition to the then (and still now) primary output of PDFs for the printing of the traditional book, all our books would have the potential for repurposing as e-books in any of the variety of output types then in development. Our expectation is that as XML becomes the 'industry standard' for encoding text and illustrations of whatever complexity, we are creating an asset which will not become outdated as technology continues to develop. Moreover, we decided to capture all our titles in this way (and not merely those whose 'e-book potential' seems obvious at the time), because experience over the last 10 years has demonstrated that we cannot predict what the readers of our books will want, now or in the future, to own or borrow via an electronic rather than the print medium.

We further decided to do the 'XML capture' at an early stage in the production process – after the delivery of the final files by the author, but before the copy editing begins. The initial reason for this was the need to capture the 'marketing fragments' – XML coded in a consistent manner – for output to aggregators and bibliographers, as well as for use on our own website, at the earliest practical moment.

The main limitation of e-books generated by retrospective XML coding (i.e. coding up the files after the book has been typeset conventionally) is that the index cannot be used as a precise searching tool in contexts other than the one in which it was created. Linking index entries to text at this stage is a time and thought-consuming process which cannot be automated, and should not have to be done more than once. If the publisher decides that having some sort of linking done retrospectively is worth the extra cost and time involved, links created between the index and the text will take the reader only to the top of the page(s) on the file to which the index refers, rather than to a precise point or span in the text. The index is thus only of use if the electronic repurposing maintains the pagination of the printed book, either as an exact look-alike or by indicating page numbers and breaks somewhere on the screen.

We therefore decided that we would attempt to develop a system which would embed the links between text and index as part of the original process of XML coding and typesetting. This would have several advantages: the links would be created as part of the origination process, rather than by revisiting already typeset files (which would help keep down costs), and the links, being embedded in the text at precise points, would remain constant regardless of the format in which the text was subsequently output. This latter quality also means that it does not matter how much the text is chopped around on page proof – not that we want to encourage our authors to do this, but if for any reason the text has to be reflowed, the page numbers in the index adjust automatically to reflect the new text layout. Moreover, if the index is run out at first page proof, we will know the final extent of the book, and thus be able to do the final costings, fix the price, and print the jackets/covers (the spine width having been confirmed) at a much earlier stage than formerly; and the large number of titles that in the past needed a revised proof stage only because the index had to be proofread at that stage can be sent to press much sooner, improving stock dates and reducing the amount of capital tied up in work in progress.

Our system was developed by Christopher Hamilton-Emery (formerly Production Director of the Cambridge Branch of the Press) together with our typesetting partner, Techbooks Inc., where boundless enthusiasm combined with immense technical expertise were provided by Gurvinder Batra, President and CTO of the Professional Publishing Group, and his team. In Cambridge, Karl Howe, Academic Production Manager, has driven the project forward, and done a great deal of work with the professional indexing community in the UK.

We cannot pretend that our current method is anything other than steam-driven, in spite of the slick results it provides. We have had many teething problems, and a great deal of very useful feedback both from our hardy and courageous pool of professional indexers and from our authors, the majority of whom (about 90 per cent in Humanities and Social Sciences, about 70 per cent in STM) compile their own indexes. Briefly, the process works as follows. You, as the author or indexer, are supplied with an 'XML typescript' – a double-spaced printout of your original text which has been XML coded but looks like output in Word. If you are not indexing your own book, you use this typescript when responding to queries from your copy editor, who works from an identical printout. But if you are indexing, you work on the typescript (either the hard copy or PDF files, if you have Adobe Acrobat 5 and its tools). You write the index using your preferred method – Word, WordPerfect, a stack of index cards in a shoebox, whatever – but instead of referring to page proofs for the index references, you use unique numbers which you also write on the XML typescript (the right-hand margin is set up for this purpose), and you highlight the point (or span) in the text to which the number relates, using a highlighter pen. The end products you send to us are an index file (which looks perfectly normal except that the numbers after the headwords are those not of pages but of specific references in the text) and a marked-up XML typescript or PDF files which show where in the text these index references need to be linked. (For more detail on the process, see our website, `https://authornet.cambridge.org/information/productionguide/`.) The typesetter takes the XML typescript and adds the link coding to the XML files; when the proof is run out, the unique numbers 'know' which pages they are on.

Once you have grasped the principle, the process is easier to do than to describe. Those of our authors who have had difficulty with it have usually tried to do something rather more complicated than we need. On the other hand, some have assumed a vastly more sophisticated system, whereby having marked the links, they do not actually have to write the index – 'the computer' will intuit the headwords and subheads. (Actually, this works fine for an index of names only, since the highlighted names can be extracted and alphabetized by the typesetter – except that the author will have to go through the proof adding first names or initials and untangling alpha-order issues among the usual suspects such as De Kooning and Van Gogh.) One frequent stumbling point is the meaning of the word 'unique' – it does not mean 'use 27 for each reference to Hals, Frans'. Making an analogy with a 'conventionally' compiled index usually sorts out this confusion. Some authors (and most professional indexers) prefer to limit the digits in each unique reference by prefixing the chapter number (thus 20.59 for the 59th item coded in Chapter 20, rather than 2304, for example), and for those books (often law titles) that are numbered throughout by paragraph, the index references are often anchored to the paragraph number rather than to a specific point.

Of course, this system is not readily compatible with the indexing programs that many professionals use, nor is it intended to be: most of our indexes are not written by professionals (see below). However, we have been very pleased at the number of professionals who have been prepared to grapple with it, have suggested improvements, and are currently working with us to develop the next stage. We have also been pleased – and relieved to the point of astonishment – at the number of authors who have got to grips with the system uncomplainingly, in spite of its recognized clunkiness. Of 300 or so authors who have partaken in the process, fewer than a dozen have taken serious exception to it (and with the benefit of hindsight there were two or three among these whom we should not have asked to undertake the process). This is not to say that they enjoyed it – another handful of authors told us from the outset that they would not do it 'our way' because their colleague X had been forced to produce his index this way and had found it appalling (though X did not complain to us at the time). We do not in fact force anyone to it – we explain the advantages for the book production process (including earlier publication, which is often a powerful inducement), and the prospective ghettoization of any e-product, and if the author remains obdurate, s/he can do a conventional index on the page proofs.

Our next stage of development is intended to get away from having to mark up the hard copy. Techbooks has produced a 'point and click' system based on a Word front-end behind which lies the XML coding, and we are currently trialling this with both Mac and PC users. (As with the original development, this system ties in to other initiatives we are taking, especially copy editing on-screen in Word from standardized files.) This tool enables the indexer to highlight words or sections of text with the mouse, and to select 'headword', 'subhead' or 'subsubhead' for each entry. It generates the unique numbers automatically, and alphabetizes the index. This is a giant step forward in our terms, though again we are aware that our tool (which Techbooks proposes to license to users) is not as sophisticated as many of the commercially available indexing systems. I am sure, however, that it is only a matter of time before someone out there comes up with a program that satisfies the needs both of the professional indexer and of the XML-based publisher – if only because it is becoming increasingly clear that many other academic publishers are moving, by various routes, in the same direction as Cambridge University Press.

Finally, another issue keeps coming up in the ongoing debate about how we generate our indexes. If one of our driving forces is the need to provide an index that can be used as a search tool in the electronic product, why do we persist in obtaining amateur-written, short, non-analytical, largely name-based indexes for most of our books? There are two answers, of which the more obvious is money: authors produce indexes for free, or if they choose not to do it themselves, they pay, either upfront or as a deduction from their royalty. There is unrelenting pressure to keep production costs down in the increasingly difficult publishing climate for academic books, and professional indexing is a 'luxury' which many books simply cannot afford. Secondly, it may be a truism to say that nobody understands a book like its author, but it is definitely true that all authors believe the truism. We frequently encounter authors who do not want to do their own index, and are prepared to pay for a professional's work, but who then reject the index on the ground that they would not have done it that way, and that 'lots of things' (always unspecified) have been left out. Unfortunately, the answer (through gritted teeth) 'Well, why didn't you do it your way, then?' is, however tempting, not tactful.

Caroline Murray is Academic Production Director at Cambridge University Press. She began to work in publishing as a freelance copy editor while completing her PhD, and joined the production staff in Cambridge in the early 1990s. Email: `cmurray@cambridge.org`