New technology and public perceptions

Bill Johncocks

The challenge that emerging publishing technologies present to indexing is complicated by two common misunderstandings: first, that indexing depends on pagination, and second, that it simply locates term occurrences. The profession's approach should surely be based on a clear separation of the intellectual process of indexing, whose mechanization we need to resist, from the development of device-independent location systems, which we should embrace and attempt to influence. A brief overview of current technologies is attempted.

New technologies began to impact indexing more than 30 years ago, well before the introduction of Macrex in 1981, while embedded indexing was well established by the early 1990s and even XML was codified in 1998. It seems to be the rapid rise to prominence of the ebook that has presented indexers with a challenge which suddenly appears urgent, but there are already so many approaches to electronic publishing and techniques for facilitating retrieval that some indexers have confessed themselves to be (or unwittingly revealed themselves as) confused. To make sense of the impacts of the various approaches, we need to define a few terms. That need is greater because the relevant technologies have not always been helpfully named in the first place. It might also be useful to put the various approaches into a historical context.

Breaking free from the page

One of the factors that make these new technologies seem strange is our long-established reliance on a far-from-perfect locator formalism, the page number. Many non-indexers even assume that indexing without page numbers is impossible, so they see new technologies which do not rely on any pre-existing stable pagination as requiring alternative retrieval mechanisms. And yet the page number is a truly terrible locator. Suppose you wanted to find references to minor planets in a 300-page book on astronomy. If the index provides just five separate single page numbers, it still leaves you to scan one-sixtieth of the entire text for a concept which, in some cases, may be expressed obliquely or synonymously (as, say, 'asteroids', 'Kuiper Belt objects', 'Ceres' or 'Vesta'). In the same book, a machine-generated concordance or an inept human indexer might offer 50 unqualified page references for the planet Jupiter, which would lead the hapless seeker after Jovian material to scan one page in every six, a substantial fraction of the whole book.

Some of us already thought this situation unreasonable, long before readers met the modern alternative of hyperlinked indexes which, in contrast, present the start of each relevant treatment unambiguously at the very top of their screens. Horrors like that Jupiter entry result from a misguided reliance on mechanical word-spotting allied to an inappropriate (that is, page-based) location technique. Pages are only second best anyway: have you ever heard anyone say 'It's on page 1468 of the Bible'? They don't of course, not just because the established book–chapter–verse system is finer-grained and faster, but because it is an unchanging characteristic of a stable text, rather than a mere accident of presentation. This second advantage also applies to date-based texts like Pepys' *Diary* or Boswell's journals, or to any text with visibly numbered sections or paragraphs.

So clearly the fundamental requirement for a working index is not a uniformly paginated edition, but a stable text containing enough unambiguous positional indicators that we can direct readers to. It just happened that Gutenberg presented us with standard paginations a few years before the current division of the Bible into verses was universally adopted: pages got there first and have held almost undisputed sway ever since. Were they not overdue for a serious challenge in any case?

Parenthetically, the true parent of the index of course is not unambiguous pagination, which simply provides a convenient location mechanism, but the standardization within language groups of alphabetical order and spelling. For the former, we might have to thank the Phoenicians, but the latter was largely a 17th-century achievement, in England at least. As soon as one strays from alphabetical order and tries to sort numerical or symbolic listings, it becomes very evident that the key usability feature of an index is a universally recognized and intuitively accessible order for the entries. However, for the orthographically challenged, search engines that supply and retrieve the correct versions of common errors have undeniable advantages.

An overview of page-independent indexing

It can help, when surveying unfamiliar technologies, to fit them into some context or classification. There are though numerous ways to group individual systems. The simplest perhaps is to understand them all as reactions to perceived deficiencies of traditional page-based indexing, and then to look at their diverse approaches to page-independent locations. Let's probe the mechanics of four key modern techniques using an unrealistically pared-down example of a fragment of text and the single index heading we might wish to assign. Here is the text:

> The cat sat on the mat.

and the required index entry:

> domestic pets

The traditional approach of course is to await not just stabilization of the text but final pagination and then append the page number as in:

> domestic pets, 94

This works well for a single printed edition but has a number of well-known practical disadvantages. First, it takes an unconscionable time. Once a book has been paginated, its author and publisher want to put it on sale, not to wait for some far-off freelancer to perform the arcane and time-consuming ritual that furnishes it with an index. The second disadvantage is more fundamental: being wedded to a particular instance of pagination, the index will fail if applied to any slightly changed version of the text. From moving an illustration to adding a few updating paragraphs, excising a whole chapter or printing a new edition on a different page size, any change that will affect pagination is instantly fatal to an existing page-based index. A related but newer threat comes from display technology; if we present the text online, or render it on a hand-held device, page size is not predefined and the index locators become meaningless or even misleading. The key to overcoming these problems is to abandon our fixation on the page as locator.

Embedded indexing (EI)

The first technique to appear that circumvented these disadvantages was still designed for text that would eventually be paginated; it simply allowed the indexing operation to be moved to before the page layout stage. Embedded indexing involves inserting actual index terms at appropriate locations in the electronic version of the indexed document, in a form whose visibility can be switched on and off (Lamb, 2005).

After pagination, EI software builds the index automatically from this markup by associating the final page locations with each embedded index entry and then, like any of the familiar indexing software systems, performs the sorting, suppression of duplication and formatting necessary to produce a usable index with no further involvement by the indexer. Here is an example of our text with the index term embedded:

> The cat sat on the mat.

That was a little disingenuous, but it demonstrates two essential features of the embedded index which are not always appreciated. First, the indexing exists as *hidden* text, and in our example, its visibility has been turned off. Second, there is no separate index file: modifications to the original document wholly contain and retain the index. EI systems are included in most of the common word processing and document preparation systems, like Microsoft Word and LaTex. Here is what our index might look like using Microsoft Word and with 'show hidden text' turned on. These XE codes are the ones the Word software recognizes when assembling the eventual index:

> The·cat·{XE·"domestic·pets"·}sat·on·the·mat.

Revisiting the drawbacks of the traditional page-based, back-of-the-book index, we can see that they have been neatly overcome with EI. First, depending on the client's software choices, indexing the author's manuscript can be performed well before pagination, and once pagination has occurred, simply running the software generates a finished index. There need be no opportunity cost associated with human indexing. Second, the indexed text is in principle tolerant of subsequent deletions, format changes and content rearrangements, provided they are carried out before the final index-generating software run, because the index terms simply disappear (in the case of deletion) or move to new locations (in the case of rearrangements) along with the portion of text they describe. In an extreme case, lose half the chapters and you will be left with only about half the index terms; just those describing the surviving text. There should be no need to trouble the indexer again.

The resulting index may be unbalanced and might look rather odd to a trained evaluator but it will still work. It is quite ironic that, if exact positions are converted to the page numbers readers are more used to seeing, the precision with which text treatments are identified will have been thrown away. Also the unpredictable intrusion of page boundaries makes locator ranges problematic. In principle, although word-processor-based EI software defaults to allocating page numbers, the embedding would be valid for display on a website or an ebook page: it is just a question of adding further software to render the resulting index appropriately. Thus EI indexes facilitate repurposing: index once – publish many times!

Tagging with code markers

Since EI appeared, publishers have developed several alternative options for indexing unpaginated documents. Three major companies, Cambridge University Press, Oxford University Press and Elsevier, use a variant of what – in the United Kingdom at least – is loosely called 'tagging', an unfortunate term considering the specific meaning of the word 'tag' in markup languages. Unlike EI, tagging does involve supplying a separate index file but it still avoids page numbers. Instead it relies on inserting some kind of unique code (rather than the index terms themselves) into the text at the places where indexed topics are discussed. These codes are employed as placeholders, acting as temporary locators in the accompanying index and being replaced (in the case of hard-copy publications) by page numbers once the latter have been assigned.

Positions in the text can be marked in two different ways, with either a meaningful sequence number or an arbitrary code. The only requirement of any location code is that it must be unique within a given document. The first is probably faster, because a computer can assign section, paragraph, line or sentence numbers in advance, but it depends on a stable, definitive text. The latter requires the indexer to assign the codes (as do some EI extent designations that will result in page ranges) but, having no positional significance, should survive subsequent rearrangement of text elements, for example swapping two sections around, which would undermine any sequence numbers. Here is our example using a sequential code number:

```
The cat sat on the mat.[2478]
```

domestic pets, 2478

and an indexer-assigned code might look very similar. So far as I can discover, when UK publishers use either technique to mark up unpaginated text, in all cases a printed book is eventually produced. When the codes are used to generate page number locators, the final index will look indistinguishable from our first, traditional example, and as with EI, the index's precision will be degraded by the conversion process.

Hyperlinking (with a nod to ebooks)

The third distinct way of marking a text position as a locator for an index entry is hyperlinking (Wright, 1997). EI requires an electronic version of the text but hyperlinking takes us yet further from the printed page in that it is inapplicable to hard copy indexes. Instead of inserting the index terms (as in EI) or a placeholder code (as in tagging), we can use the same markup language that renders the document to link the index to the text treatment. In the case of a web page, this will normally use two instances of the HTML 'anchor' tag, linked by matching attribute values at each end. The text markup might be:

```
<a name=“moggy”>The cat sat on the mat</a>.
```

while the index entry is:

```
<a href=“#moggy”>domestic pets</a>
```

which could result in the index entry:

domestic pets

provided that the entry is unique and so can be set up to take the index user straight to a single text location. Otherwise, where more than one treatment needs to be accessible from the same entry, perhaps separate matching attributes (in this case three pairs) would lead from location links to discussions of pets:

domestic pets, 94, 112–3, 123

or (since what are presumably residual page numbers are now irrelevant):

domestic pets, 1, 2, 3

Clicking on the entry in the first case, or on one of the three occurrences in the second and third, will of course cause 'The cat sat on the mat' and its following text to appear instantly at the top of the reader's screen. The choice of 'moggy' is again arbitrary; once more the sole requirement is uniqueness. The rendering software will need to match the 'href' attribute at the clickable end of the link (the index term) to the 'name' attribute of the link target (the labelled text location).

Website indexing is a specialized area of activity at present, but hugely significant because ebooks are structurally analogous to very large web pages – a continuous stream of text with no predetermined structure or breaks through which the reader slowly scrolls – and current systems employ coding based on HTML or XHTML to render the book in reader-specified screen sizes. Because ebooks currently have no fixed pagination, any page-based indexes prepared for their hard-copy versions will self-evidently be inappropriate. Nevertheless, these, complete with meaningless locators, are often apologetically supplied and perhaps shamefacedly made less than obviously accessible. The fall-back seems (at least in the case of the Amazon Kindle, currently dominant in the UK market) to be keyword searching linked to automatically generated 'locations' of perhaps a sentence or two, which is almost as precise as hyperlinking (or indeed as Biblical verses).

If an ebook's text has no obvious structure below, say, the level of chapter numbers, indexing would have to depend on either links to sequential tags or markup-language anchors. Of these more thoughtful approaches, the most attractive from the readers' viewpoint seems likely to be a human index hyperlinked to embedded anchors as just described. Unfortunately, the scramble for market share which sustained ereader development concentrated on the kind of entertainment fiction where indexes would not have been provided even for their hard-copy versions.

Whether and how human indexes will be incorporated once the ebook corpus extends fully into works of reference remains to be seen. For the moment, the problem seems to have been laid aside, its solution not seen as a high priority, and we read the depressingly defeatist statement from Amazon that 'indexes are not recommended at this time' (Amazon.com, 2007). It seems unrealistic to expect pressure from non-fiction readers to persuade manufacturers and publishers to provide more sophisticated subject access, but the very fluidity of the technology coupled with the influence of market forces or standardization may present us with opportunities to campaign for genuine, index-based retrievability in future generations of ereaders, rather than the fig-leaf of keyword searches.

XML (Extensible Markup Language)

The three page-independent techniques so far described – EI, tagging and hyperlinking – represent different ways of encoding text positions. Sadly, XML cuts across that neat classification but fortunately, there is a comprehensive and readable account of XML embedding, the purest usage, by Michele Combs on page 47 of this issue. XML is a powerful and flexible data description language, allied to the familiar HTML that drives websites. It allows almost limitless repurposing of the marked-up document, for different display systems like websites and handheld devices as well as hard-copy printing, but the detailed implementations are client-specific. Two other significant features from the indexer's viewpoint are that it is massively verbose and utterly fault-intolerant.

Direct embedding of index terms in XML codes as described by Combs works rather like EI using Word, though the rendering software is hugely more powerful and more flexible, allowing the generation of hard copy, hyperlinked text and potentially ebooks, but few publishers adopting XML-first production methods have so far required their indexers to master its technicalities. This too is an area where formulating requirements now might ensure that systems do not simply ignore the index. XML embedding on our sample text might something look like this:

```
The <indexentry><primary>domestic pets</
primary></indexentry>cat sat on the mat.
```

There is one final act of disambiguation required here. Indexing with the well-known CUP-XML system is a tagging technique and does not involve direct XML embedding as described by Combs, nor does it expose the indexer to any XML whatever. Few systems have been more unhelpfully named.

Drawing a line in the sand

At present, public attitudes to indexing are being distorted by a misconception far more damaging than the assumed reliance on page numbers. This one strikes at the heart of what an index is and what it is for. It has spread from IT people to most authors and several publishers, and is even tacitly accepted by many readers and some misguided indexers. It is the conviction that an index essentially lists just term occurrences and their positions; a sterile enumeration with no enrichment of the author's vocabulary, no subentries, no cross-references; in fact no discernible contribution from human intelligence at all. What is being set up as a replacement for indexes is of course simply a concordance, the kind of thing that a computer can produce and moreover that it could produce faster and more reliably than any indexer.

This threat to analytical indexing, to attentive reading, and perhaps even to full education of our citizens, comes originally from IT. Programmers are influential with publishers, and markedly incurious about our specialized contribution. There are two reasons why they seek to replace the irreducible intellectual component of indexing with mechanically selecting words. First, it is easy for computers to do, and second, IT people cleave to a false model of reading. Put crudely, computers cannot read, but they are good at recognizing words, so the IT industry, certainly for the past 40 years, has determinedly played to their strengths without bothering about what an index is and does.

Back in 1994, Nancy Mulvany remarked that 'It would also be helpful if designers would speak with professional indexers and find out how we work. Before developers can provide efficient tools they must understand the process of creating authored indexes' (Mulvany, 1994), while in 1995, Hans Wellisch dismissed 'people whose idea of an index is an alphabetical list of words extracted from a text plus their locators' (Wellisch, 1995: 170). *Plus ça change!* Those two authorities might not have foreseen how the demands of electronic devices would force the retreat from page-based locators, but they certainly recognized that the intellectual part of indexing – the analysis of meaning, significance and uniqueness, then modelling the likely behaviour of human readers and providing for their predicted access paths – cannot be automated.

When academics and respected journalists unthinkingly reveal that they see an index as a list of term occurrences, we can pin the blame on the keyword-based retrieval methods made familiar by search engines. Even limiting ourselves to word searches, the interrogation techniques are actually less sophisticated today than those available in the 1970s. Search engines are great for finding cheap flights, attributing well-known quotations or assembling material for a homework assignment; but for accessing books, they are woefully inadequate. They pull off the difficult feat of combining low recall (missing all treatment where a concept is only expressed synonymously, elliptically or at a different level in a hierarchy) and low precision (overwhelming the readers with strings of undifferentiated occurrence positions, many of them negative, repetitive, figurative or insignificant).

Keyword searching fails with reference-type documents because it is based on an entirely false paradigm. Human writers do not communicate with human readers by the unvarying repetition of identical noun phrases (or reading would be intolerably dull), and that fact, a problem for computers, is something a human indexer is uniquely well placed to recognize and allow for (Johncocks, 2005).

Nevertheless, software providers have been quick to accommodate this naïve view of information retrieval, and crucially, even some indexers contribute to the prevailing confusion. Bowing to the convenience of author-indexers, Microsoft Word's built-in indexing module provides a 'Mark All' option, allowing every occurrences of a term or phrase to automatically generate an identical index entry. So in our 'Jupiter' example, the 'index' will faithfully but pointlessly direct the reader to all 50 locations including 'except Jupiter'; 'as we saw in the Jupiter chapter' and 'in mythology was the daughter of Jupiter'. Few professional indexers, I hope, will use 'Mark All', but I was shocked when both SI and ASI invited presentations on TExtract, essentially a word-spotting tool unashamedly marketed in the past as an alternative to human indexing, at their recent annual conferences. Had they not noticed that word-spotting indexes are almost entirely responsible for the widespread (and often correct) conviction that a search facility is a superior alternative? Hardly surprising then that Amazon's Kindle reader offers a word-search facility but makes so few concessions to index use.

The superficial simplicity of word spotting might also have persuaded some publishers that indexing is an easy and mechanical process, leading them to resist offering any premium for learning newer and more complex methods of working. Though it might pay in the short term by allowing the faster compilation of a plausible caricature of an index, embracing word spotting is essentially slow professional suicide. Indexers cannot hope to compete by merely doing slowly and fallibly what computers can do better and faster. We have to add value, and of course we do have value to add.

The limits of flexibility

Though we must remain determined to condemn keyword 'indexes' for serious documents, each of the pagination-independent indexing techniques described here can be reconciled with genuine analytical indexing. Essentially, they all affect locators, not headings. We need energetically to insist on retaining control over headings, but we can afford to welcome pagination-independent and device-agnostic delivery techniques. Some grumble about the transfer of effort, but this happened too when we moved from cards and slips to PCs, and from proofs to emailed PDFs; new business processes are a fact of life, and in the wider world of publishing we are just the pipers, not the payers.

Nevertheless, it would be optimistic to expect the transition to be painless. All the technologies described require the indexer to master a different set of supplementary skills. For freelancers contemplating what appears to be a one-off commission, the learning curve associated with an unusual system may be too steep to make economic sense, but if a regular customer converts, most indexers will see the advantage in adapting their technique to accommodate the new way of working. Publishers need to meet indexers half way on this though; some interfaces are needlessly cumbersome and, for example, make it difficult for the indexer to view the developing index, so consistency becomes elusive. An additional risk with embedded and hyperlinked techniques is that, by requiring a remote indexer to access and modify the source text, they introduce potential conflicts, where the indexer may inadvertently corrupt the text or a later editorial intervention may harm the index. Techniques are available to avoid this danger, like a concurrent versions system (CVS), without which publishers may be tempted to train and retain a team of dedicated indexers. Finally, if it is necessary to buy any software to work this way, the costs can be substantial.

Indexers are all free to reject new technology, although there is a price to pay (as there is with refusing to drive a car or carry a mobile phone). Individually we can opt out, but it would be another route to professional suicide were our institutions to omit to train new entrants to the profession in techniques to which they are certain to be exposed at a time when they are temperamentally most receptive. Of course we always need to stress indexing fundamentals, but it would surely be folly to equip new indexers only to meet the needs of the last century.

A basis for engagement

As a first step, we all need to be clear, in our dealings with the publishing industry and its IT advisers, about precisely what human indexing can offer. The IT industry has never really wanted to talk to indexers (we represent a reality they have not so far been prepared to acknowledge), but standards bodies might, and so might the designers of the next generation of ebook readers, interested in giving their product an advantage in an expanded market. With publishers, the challenge is to gain access to the decision makers, rather than the commissioning editors who simply implement policy made by remote management, and whose awareness of their chosen technology can often be shallow.

There are a few areas where we could improve our approach to technology.

- We need to agree and insist on the essential elements of human indexing. Not all indexers have renounced the blind alley of word spotting.
- We need to ensure that emerging best practice and standards can accommodate all the important features of analytical indexes.
- Indexing societies are organized nationally, and though we face a global technological challenge, there has been little international cooperation and exchange until now.
- In a fast-changing world, we need to supplement both training for new indexers and continuing professional development for established ones, by devising skills-updating modules addressing specific technologies.
- Confused about the various technologies, some indexers are making misleading claims, which can undermine the technological credibility of the whole profession. They should be persuaded of their folly.
- Some developments, like assembling one-off 'books' from fragments of others, may require even more fundamental changes in our approach, and have been deliberately left out of this survey.

It appears that our first test will come over ebooks, where with the proof of concept spectacularly delivered and an income stream assured by fiction titles, we can hope that ereader manufacturers will become receptive to a different retrieval paradigm as they expand fully into non-fiction. Machine searches might yet triumph over human indexing, which would, I think, be bad news for literacy and indeed for culture, but it is too soon to abandon the struggle. We need to approach clients with a positive, welcoming attitude but every indexer should be equipped to explain why there is a line beyond which we will not retreat.

References

Amazon.com (2007) 'Kindle direct publishing', available at: https://kdp.amazon.com/self-publishing/help?topicId=A2RY017TIRUIVI#back Quoted in Lamb, J. (2011) 'Kindle and the index', available at: http://ccgi.jalamb.com/2011/05/kindle-and-the-index/

Johncocks, B. (2005) 'The myth of the reusable index', *The Indexer* **24**(4), 213–17.

Lamb, J. (2005) 'Embedded indexing', *The Indexer* **24**(4), 206–9.

Mulvany, N. C. (1994) 'Embedded indexing software: users speak out', talk for Changing Landscapes of Indexing: the Proceedings of the 26th Annual Meeting of the ASI. (Also at www.bayside-indexing.com/embed.htm)

Wellisch, H. (1995) *Indexing from A–Z*, 2nd edn. New York: H.W. Wilson.

Wright, J. C. (1997) 'How to index online.' *The Indexer* **20**(3), 115–20.

Bill Johncocks is a freelance indexer specializing in scientific texts, an SI tutor, member of the SI Publishing Technology Group and the current editor of the SI newsletter, SIdelights. *He lives on the Isle of Skye. Email:* bill.johncocks@btconnect.com