

A lament for the vanishing index

What a whirlwind of technological evolution has swept through the last 40 or so years: desktop computers, mobile telephony, cloud computing and, of course, the web. Not to marvel at The Age of Computing and its manifold spin-offs is, surely, a sign that one is lacking the genes that stamp us as *Homo sapiens*.

But silver linings sometimes harbour clouds. Not everything born of The Age of Computing deserves unqualified admiration. The democratisation of publishing might well be extraordinarily empowering, and continue to change the way we live for the better. But it is not difficult to see some downsides, such as cyber-addiction, the ease with which the meek can be bullied online, and the whittling away of something we will later regret losing, namely privacy. As Shakespeare wrote, no doubt in a melancholy moment, “all that glisters is not gold”.

Of more relevance to the work of technical writers, the democratisation of publishing—and in particular the ease with which everyone’s musings can be disseminated—adds a further layer of complexity to our craft. Language use has never been fixed, but the rate at which it is changing has reached a new high. Anomalous usage, once largely still-born, can now quickly become a fad, a fad a trend and a trend a new convention. (Look, for example, at the birth, and quick maturing, of the *open* hyphen.) Only those deaf to language will fail to recognise that English has changed over the last 20 or so years.

Why is that a problem for technical writers? Well, are we not obliged to communicate in ways that require the least effort on the part of our readers and cause them the least distraction? That lies at the heart of the philosophy behind our oft-repeated mantra that we should write in an “audience-centric” way, that we should be the “readers’ advocate”. Thus we need to write in ways that are maximally *familiar* to our audience, and this implies that we keep abreast of changes in language. (It would distract many a contemporary reader if we stuck to the so-called laws of language etched in our minds by the gerund-grinders of yesteryear.) In stirring up mud—among the occasional gold flecks—in the stream of general communication, language-change imposes an additional, a weightier, burden on writers who need to deliver transparent, immediately digestible meaning. Thus technological change is not necessarily an *unqualified* blessing. Good though it might be, no end of the beam balance is resting on the bench, proud and satisfied.

Like most, but perhaps more so, technical writers are fascinated—and some possibly bewitched—by new technology. It bursts through the dull grey, the stifling monotony, of quotidian routine, and ignites our narcoleptic imaginations. And when new technology comes with a promise of money to be saved, it is hard not to be blinded to the downsides. Some changes following in the wake of new technology are impossible to resist. Language change born of an anarchic blogosphere falls into this category. But other changes can be resisted: if not the technology itself, then at least its application to areas of dubious benefit.

Take, for example, online learning, a relatively new technology that promises to open up knowledge to all. Think of knowledge without borders and the consequent prospect of a New Enlightenment. But then think too of the narrow, pre-defined avenues of exploration it offers, and the vanishing of extemporaneous learning and instantaneous clarification. Only misguided educationalists (and miserly governments) could consider the one-person classroom—aka the

screen—to be a superior learning environment. It is at best an adjunct, not a substitute for teacher-led learning.

Think of structured authoring and the promised simplicity of format-free writing; but then think too of how it shackles instructional creativity and denies the writer a chance to exploit the communicative power of textual appearance.

Think of the customer-created wiki with its democratic invitation to *Everyperson* to share their wisdom: but then think too of the distracting anarchy of styles, the unexamined guesses camouflaged as truth, the un-thought-of gaps, the shaky reliance on the generosity of others, none of which is found in texts meticulously sculpted by technical writers.

Think of content management systems, with their tidy, *waste-not* philosophy that enables every snippet of text, very table and figure, to be re-used and repurposed: and think too of the clashing styles, the inconsistent vocabulary and the clanking transitions that bedevil many mash-ups sewn together from such op-shop left-overs.

Finally, think of the full-text electronic search and the speed with which it delivers information to the reader: and then think too how the ease and speed of its implementation has led to the near-demise of a once essential adjunct to self-paced learning—the humble index—without which the branch we occupy in the gnarled tree of cultural evolution would be more brittle and no doubt stunted. (Can anyone honestly attribute their educational achievements to reading texts that lacked an index? A corollary: is our current default position of omitting indexes condemning future readers to an education far less rich than we enjoyed?)

So, what has happened to the index? Let's look at some claims made against indexes: they are too expensive; they add to the production time and thus can delay a product's release; and a full-text search facility is far better than an index.

Is an index expensive? Let's imagine a 200-page user guide. By Australian standard AS4258, such a manual would take a technical writer about 600 hours to write. At current contract rates, that would cost the contracting company about \$40,000. Now a 200-page manual would take, at most, 30 hours to index. This is one-twentieth as long as it took to write the manual. So indexing would add about \$2,000 to the cost. This is chickenfeed to most companies, and an amount more than offset by the goodwill an index will generate in readers of the manual.

But need such an index take 30 hours to compile? Well it might if the manual was sent to a indexer who knew little beforehand of its contents. But there is an alternative: get the technical writer to index—and *index as they write*. Rather than indexing only when the writing is finished—an oddly common practice—indexing *while writing* would see the cost of indexing largely absorbed in the cost of drafting (for it takes very little time for a writer to insert an index marker as they go). Moreover, the resulting index is likely to be superior, and for two reasons: (a) during the struggle of drafting the writer will no doubt be assessing various synonyms and thus, given their intimacy with the text, will have a better idea than an external reader (such an indexer) of what words are worthy of indexing and (b) the index can then be added to the purview of whoever reviews the manual during its development (thus leading to incremental improvements in the index along the way).

Another objection is that an index adds to the production time and thus can delay a product's release. That might be so if indexing is done post-drafting. But if the technical writer indexes as

they go, little time is added to the schedule. Even so, why can't documentation development begin earlier if that would be needed to accommodate indexing?

But by far the most common objection to including an index in technical documents is that such documents are mostly provided in electronic form, and the most common forms—PDF and web help systems—offer a full-text search facility, a facility supposedly superior to an index. But is it superior? The standard by which this needs to be judged must, surely, be that of *usability*, and one of the three pillars of usability that underpins all informational writing is that information must be *easy to find*. So, is information found via searching found more *easily* than if it were found via a human-created index? The answer is surely no, and for a number of reasons.

Firstly, a search facility is of no use to the many who—in recognition of the fact that reading on screen is slower, is subject to the siren-call of distraction and leads to lower comprehension—print out documents.

Secondly, a search facility could leave a searcher thinking that a particular topic is not covered in the text—even though it is—if the searcher happened to be searching on a *synonym* of the term that the author actually used. For example, searching for *Basedow's disease* in a medical manual written outside Europe is likely to yield no hits, as Basedow's disease is known as *Grave's disease* elsewhere. Same disease; different names. Or searching for *kerosene* will yield no hits if the author's preferred term was *paraffin*. A human-created index will alert the reader to synonyms with a *See* cross-reference: *kerosene See paraffin*. (Likewise with spelling variations: searching a document for *tyres* might yield no results if the author preferred USA spellings.)

Thirdly, a human generated index will alert the reader to *related* topics by way of a *See also* cross-reference: *epilepsy See also seizures*. The reader, unsure about the terminology used in the document, might well find what they are looking for only by going to a cross-referenced topic. Such cross-referencing is not found in full-text searches.

Fourthly, a human-generated index can suggest to the reader where most information about a topic can be found. This could be indicated by the presence of *span* of page numbers (45–51) as opposed to a single page number, or because the page number is set in a different style (such as bold). At best, a search engine might give a hit a ranking, but this is often a blunt indicator (being little more than a measure of how often the search term appears at a particular location).

Fifthly, an index is a better guide than a table of contents to the sweep and depth of knowledge available in the text, revealing far more opportunities for discovery and learning.

Sixthly, I am going to be stuck if I am not sure of the spelling of the term I am looking for. For example, if I enter *antidiluvian* as my search term rather than *antediluvian*, I will, in all likelihood, get no hits. A quick glance at an index would alert me to the correct spelling—and to where the term appears in the index. (Some search engines offer *fuzzy* searches, which might overcome this problem for some terms. For example, the fuzzy search available with the online version of the Macquarie Dictionary displays *antediluvian* if I enter *antidiluvian*, but it gives no suggestions when I enter *Xrays* or *anoreksic*, and it gives the wrong suggestions if I enter *mison* when looking for *meson*.)

Finally, a full-text search can lead to numerous time-wasting partial hits. Having found no hits using, say, *income reports* as my search term, I might then search on *reports*. But before I get a hit on *revenue reports*—the author's preferred term, unbeknown to me—I might have to skip over scores of irrelevant hits (*sales reports, shareholder reports, debtors reports* and so on) before, a hundred or so

pages later, I get a hit that is relevant. A human-generated index would have entries for *reports: revenue* and *income reports*. See *revenue reports*, enabling me to quickly see the author's preferred term and go straight to the relevant topic.

In all these cases, the typical back-of-book index (even one without page numbers, as in an online help system) provides greater usability than a full-text search. Information will often be found more quickly, where it is found at all. Thus a claim that a full-text search facility is far better than an index needs a measure other than usability to justify it. But what might that be?

It is sometimes said that our clients always want to maximize efficiency and that shedding indexes aids us in achieving that goal. Now efficiency might be a laudable goal, but if it were our sole goal—if our readers' needs were not part of the equation—then it could be met quite easily by writing pure, unadorned text: exactly what you get from Notepad. Most clients would balk at that. They know that readers of product manuals would naturally, even if unfairly, generalise their first impressions: shoddy manual, probably shoddy product. At the other end of the spectrum is the incorporation of every bell and whistle ever imagined in tech-writer heaven. Most clients would balk at that too. The cost of producing such documentation would no doubt outweigh the accumulated benefits. So somewhere along the spectrum from maximizing efficiency to delivering the greatest usability to the greatest number is a client's sweet point. This point won't be the same for every client, and the problem is that most clients don't know where their sweet point is.

And in this ignorance lies an opportunity that is often wasted: the opportunity to educate a client about what end-users want (and thus an opportunity to give end-users what they need). Instead, some of us gratefully, sheepishly, do just what a client asks, even if we suspect that the client has thought little, or knows nothing, about usability in documentation. We forego the opportunity to up-sell, to truly be the readers' advocate that we endlessly brag about, perhaps thinking that a suggestion to include, say, an index might be interpreted as nothing more than a wish to fatten our own purse. This is an area where we need to lift our game, to fine-tune our rhetoric, so as to better convince our clients that giving end-users what they need is more than just a win for the technical writer. It is also a win-win for client and end-user. A win-win-win all round, in fact.

Geoffrey Marnell