WORKING NOTE AC127

Alan Creak
1999 February 24

LESSONS FROM THE 1998 HANDBOOK EXERCISE

This is a collection of notes made since the 1998 spurt of activity. It isn't particularly carefully organised, but I've tried to categorise things, and drawn a few tentative conclusions.

Cross-references.

However we construct the handbooks, we're always likely to want cross-references of some sort. (Perhaps we should be trendy and call them hyperlinks, but as they've been called cross-references for a very long time I'll stick to it.) To give a cross-reference, you must be able to identify a target, which is some more or less remote part of the material. The target can be specified in several, significantly different, ways :

- **By content :** Some item in the material (I'll assume it's text, though that isn't essential to the argument) is specified in sufficient detail to be identified unambiguously. This could in principle be a part of the text as given, but it would be difficult both to ensure that a reference was unique and to search through a large document to find it – and it would be impossible to guarantee that the target would remain unique in a document which was still under development. It is therefore more practicable to insert an artificial target, either guaranteed by the system to remain unique or sufficiently odd that one has reasonable confidence in its uniqueness. The most common example is perhaps the use of links to literature references. A serious flaw of methods of this type is that they are not robust to editing operations on the target text; the target might be changed or removed, and there is no simple way to check that this does not happen.

- **By accident :** Some property of something irrelevant to the material becomes firmly associated with it by some accident of production. Page numbers are in this category. They work very well, because they are – by definition – unique, and because once a document is printed they are rigidly associated with parts of the text. But they are quite useless in a document which is never printed – and worse than useless in a printed document if they're wrong. (An example from 1998 : after carefully working out the page references for her M.Sc. thesis, Natalie Spooner printed it, only to find after the thesis had been bound that some vagary of Word had changed the pagination. She had to reprint parts of the thesis and bind it again.)

- **By component :** Most documents of any size are hierarchically structured, being assemblies of sections, chapters, paragraphs, etc. It is usual for the smallest components to be quite small, and therefore to be useful items to quote in cross-reference – just as page numbers are useful. Such a reference has the significant advantage that its semantics is determined solely by the document; accidents such as pagination cannot affect it. It is also possible to decide on the document structure at a "high" level and be reasonably confident that it won't change drastically; the components of the document can therefore be specified beforehand and then used throughout the document for cross-reference with some confidence.

For present purposes, we seek a means of implementing cross-references which will work for the handbooks. There are two features of the handbooks which bear strongly on this decision : the handbooks are written by many authors working independently; and the same text is used to produce both printed and HTML versions of the document.

Methods based on content are unlikely to be satisfactory, as the author of the section containing the reference might not be the author of the target section, and the machinery needed to ensure that targets were present, and stayed present, would not be easy to put into effect.

Why can't we quote the page numbers ? – because there are no pages in the HTML version. They won't do even if we consider only the printed handbooks, because the page numbers are determined by the printers, who do the final formatting. Why don't we do the final formatting ? – I don't know. But even if we do it, the pagination will be determined by something like Word or Pagemaker, and won't be available for building into the system, short of possible further reliance on Word facilities. (Unless we do it all with – say – Imperial[1] ? Note that the demands of formatting the handbook are non-trivial, and might push LaTeX a bit hard – though doubtless one could meekly follow everyone else and give away decent layout for the convenience of LaTeX. But one wouldn't, any more than one would wait for Imperial to be completed.)

We are left with sections. The requirement is that, at any arbitrary point in the handbook, we should be able to insert something which means "look at <some specified section>". This link must satisfy two important criteria : the author must be able to insert it without much difficulty; and it must work simply in the final product.

To insert the link, the author must be able to identify the intended target. I can think of only one satisfactory way to do so; to give each section of the document a trivial name, and to make the names available. Section numbers are likely to change as topics are added to and deleted from the handbooks, or as editorial decisions move parts around. Section titles are also likely to change as perhaps minor developments in the department's activities alter the focus of the section – or even as pedantic editors change the punctuation.

In 1998, I used fixed names, which don't appear in the handbook material and therefore need not have any particular meaning; they seem to work well. The table of names can be kept as an additional field in the skeleton file[2], and turns out to be a very convenient source of file names for the individual files in the various data collections.

Not everything is perfect. To make up the names I took the initial letters of the words in the section titles, so they do have a sort of meaning. This turned out to be useful, as I didn't have to look up the name of a section if I knew its title; with strictly arbitrary names that wouldn't be possible. But if I don't regard the names as strictly arbitrary, they're no use for cross-references. There is a possible conflict of interest here between truly arbitrary names, fully independent of the data, and mnemonic names, which are necessarily dependent on the data to some extent. I don't think that the conflict is resolvable, but I'll think about it. (Curiously enough, this topic turned up as part of the operating systems examination[3]; this is real research-based instruction.)

The second criterion is that the references must work easily. For the HTML handbook, that's trivial, as a reference to (say) pqr can be converted into a simple HTML link to the appropriate HTML file – sensibly called pqr.html. For the printed version, that won't work, because the link has to be implemented by the person reading the document, and a reference through a table of arbitrary identifiers is unlikely to appeal. I have already explained why page numbers cannot easily be used; there remains the expedient of section numbers. While these cannot be built into the handbook text, they can be worked out automatically once the handbook composition is determined. It is therefore necessary for the formatting programme to begin by allocating section numbers using the layout determined in the skeleton file, after which they can be used in the table of contents and in any cross-references required.

Identifying the sections.

Having based our cross-references on sections, we had better work out how to identify them. At present, it's simple : section headings are explicitly identified by the mark-up symbol |!|Section heading!|. Further mark-up symbols are used to define chapters, subheadings, and (occasionally) subsubheadings. Clearly, something of the sort must eventually happen so that the format is properly produced in the visible document, but just what constitutes a section should be an editorial decision, not under the control of individual authors.

A good way *not* to do it is the Word device of associating specific styles with headings of different sorts. I tried that in 1998, mainly to manage the automatic numbering in the text version; it turned out to be less than completely satisfactory, because I wanted to do the numbering twice, once for the text and once for the contents. Doubtless it's possible, but I couldn't find out how quickly enough. (As usual, we had no readily accessible manual, apart from the unusable computer version.)

It is also desirable that subsidiary levels of text be available which are not tied to the level-numbering hierarchy; in principle, the formatting details and the document structure are independent, though they are often used in conjunction.

These requirements can be satisfied if authors can use mark-up symbols to identify a "title" for each file, and "headings" within the file. The hierarchic level of the "title", and whether or not the "headings" should be numbered as subsidiary to the title, are editorial decisions which should be recorded in the skeleton file. Perhaps the simplest way to manage this is to establish the convention that only titles will be included in the contents and section numbering, with hierarchy as imposed by the skeleton file. By requiring that the files be such that at most one chapter or section heading appears in each, the titles and the specified levels are sufficient to determine which titles are chapter or section names, and the numbering can then be managed automatically. The restriction to at most one section in a file was imposed in 1998, and was not a constraint; while some people received several files to edit, all files were sensible components of the text.

People information.

The people collection is growing; now it includes (I think) everything specifically about people which appears in the handbooks, with room numbers, telephone numbers, www pages, etc. as well as history and publications. This makes it more cumbersome for its original purpose in constructing the handbooks, as there's a lot more to delete, but overall it's much better. People just get one record to correct instead of things in bits and pieces, and I have a check on the correctness of the files telephone numbers, etc.

The collection does not include items better indexed by people and something else – so it doesn't include lecturing or administrative responsibilities, and other such details. These still come from the functionaries collection, and I think that's right.

Whence cometh the list of people ? I originally made up the people collection from the functionaries collection, but this is less than completely satisfactory. There are several sorts of people – all academics, I think – without jobs, who can reasonably be included in the handbook as potential supervisors, but who have no formal duties. These include one honorary lecturer who's been there for years, two honorary research fellows new for 1999 (as it happens, both have jobs in 1999, but there's no guarantee that that will continue), and research fellows; there might be others, but I haven't been told about them. And the question is : who is supposed to tell me about them ?

The answer is probably that it doesn't much matter, because it's unlikely to happen anyway. I "solved" the problem this year by sending E-mail to the academics alias to check that everyone had received a request for information; several hadn't. This isn't entirely satisfactory, as there might be potential project supervisors who arrive at the beginning of the year, but of whom I never hear until the next year is upon us. The "correct" action is for me to be notified at the same time as the entry goes into the academics alias.

This is part of the bigger problem of how I should be notified of all changes to the material. People get promoted, they come and go, they publish, and so on, and this reaches my collections by accident if at all. This gets us right back to the original problem of information coordination[4], which I don't want to revisit in this document. It is worth pointing out, though, that none of the problems I mentioned in my earlier note have gone away yet.

There is a question in principle of whether my people collection should be regarded as a long-term repository of information on the people in the department, or simply as a resource for the handbooks, or at most for the current year's activities. I favour the short-term alternative, but there would be some advantages in the long-term proposal, particularly in compiling lists of the department's publications, past research students, and so on. No, we don't want to go back for ever in the handbook, but it seems to be not a bad idea to have a complete record somewhere. Doesn't it ? (Most of the material is in people's www pages, but that disappears when they leave.)

A final related item is the "List of theses". We are in the curious situation that, while our PhD coordinator keeps a list of the PhD students and tries to keep up with their progress, there is no list of master's thesis students. The only reference is the library catalogue, which is good once the theses get there, but that doesn't always happen very promptly. Here again, there is information which I should be able to acquire earlier.

Publications.

People persist in giving me their publications lists in all sorts of curious formats. Well, I can understand that – I have preferences of my own, and like to stick to them. It makes producing neat output hard, though. I'd like to try to keep the material in standard form, perhaps using something like BibTeX ? I don't particularly want to use BibTeX for the same reasons that I don't want to be stuck with TeX, but perhaps it's the simplest way.

In 1998, I had no time to do anything about it, so put up with the mess. In 1999, I'll try to do a bit better. A possibility is to use a pattern matcher of some sort to try to do the translation automatically. We'll see.

Courses information.

Dealing with the Courses collection was comparatively straightforward, perhaps because the topic was clearly defined. The tricky bits were in coordinating information for different versions of ostensibly the same course presented in different semesters or on different campi, and usually by different people; perhaps it would make sense to send all the versions to all the course supervisors involved and ask them to sort it out between them.

In 1997 I got round this by having only one file for each course number, ignoring the semester and campus; if there were several courses with the same number, they shared the same file. This was messy in that special treatment for parts which differed between the versions (notably the lecturers) was required, and effectively destroyed by the abandonment of the university's original requirement that courses with the same number should have essentially the same content. One course (210) had required separate descriptions in 1998, and I'd handled that as a special case; as version proliferate, it seems better to allow for the worst case, but expect that many prescriptions will consist largely of references to details in the first version presented.

Another complication from having several versions of courses comes from my device for automatically inserting www links in the HTML version wherever a computer science course is mentioned. I've done this by stripping the semester and campus suffixes, which leaves me with an unambiguous number for each course. This make sense in many cases, where as the courses do move between campi and semesters the stripped version provides a convenient standard, and as we are accustomed to referring to courses by number only the numbers are natural identifiers. It follows, though, that the HTML files must themselves be similarly combined, so that the single HTML file for 210 (say) includes details of both the 210 courses. For 1999, I did it all manually; that is less than completely satisfactory, but the result is quite good, as the direct comparison between versions is easy, and the references to the "master" version are immediately resolvable.

File management.

It's important to keep the working files separate from the real long-term mark-up files. The main reason is the obvious information protection requirement; the real files are our source of information, and mustn't ever be lost.

But where do you draw the line ? Many of the raw files come from my E-mail requests for information, and they are rather arbitrarily formatted to the extent that they are not immediately suitable for the real mark-up files. Should I keep all the raw ones ? At present I do, but it's a nuisance having real raw data which you can't easily treat systematically, and which you certainly don't want to send out again for further editing next year.

I think that some of these difficulties might disappear when I get away from using Word as my major editing tool; I think I'll be able to ensure that nothing important is lost in going from raw file to standard-form mark-up file.

Data collections.

Data collections are the things I called files in early documents[2]. In fact, they turned out not to be very suitably organised as single files[5], so I split them into individual files. There are two sorts of data collection :

- **Narrative** collections, in which there is little formal structure, and no natural records into which the file can be divided. These are commonly collections of text intended for people to read; the handbook text is largely composed of narrative files. Structure can be imposed by external considerations. In the handbooks, the effective structure which determines the separation into distinct files follows from the requirement that different parts be edited by different people, though section headings are also useful boundaries in case it is desired to shuffle the order of treatment. Mark-up symbols are used primarily to identify textual components – headings, chapters, parts requiring special formatting – and can appear pretty well anywhere.

- **Database** collections, with many sections (records) of essentially similar structure, each recording information about one item of a particular type. The Courses collection and People collection are of this type. Mark-up symbols are used primarily to identify fields within the record.

In both cases, additional mark-up symbols can be used to control local formatting (type style, etc.).

It does seem to be convenient to separate large collections into comparatively small physical files. Whether or not this is a peculiarity of the handbook files isn't obvious to me. Certainly, one of the big advantages of modular files for the handbook is the comparative ease with which they can be sent to different people for editing, and that requirement might not be important in other contexts. With both sorts of collection, though, there is a clear sense in which the collection is the "real" unit, and corresponding to this each collection has some sort of list in which all items of the collection are recorded, perhaps in some significant order. The skeleton file serves this purpose for the handbook; in the other cases, the list file has not so far been a feature of the records, but has existed implicitly in the functionaries list and the timetable. It is becoming clear that new and explicit lists will be necessary to drive the automatic system properly. It is noteworthy that the lists themselves are (likely to be) database collections, each presented as a single file, as they are composed of sequences of items of similar nature.

The distinction between narrative and database collections is significant, because the different types of collection are used in different ways. A narrative collection is simply read and formatted, because there's not much more to do with it; processing is essentially serial. With a database collection, access is more likely to be directed to a specific entry, and the fields of the entry need not be used in the order in which they appear in the file. This is the pattern used in my discussion of the document factory[6], in which I designed a machine to deal with the Courses collection; this pattern would not be appropriate for a Narrative collection.

Collecting stuff.

I can handle E-mail decorations (>, etc.) quite effectively now using Word macros. It should be even better with the document factory[6], though it does seem unlikely that I'll be able to manage unfamiliar formats automatically. But I can't handle my colleagues. They continue to reply in inappropriate ways :

Not at all. This is not common, but has caused a couple of problems with the way I manage the files. This year, with the intention of ensuring that historical information disappeared when it went out of date, I didn't try to replace the old files. Instead, I sent those out for modifications and changes, but used the replies to construct a new set of files. This made sure that entries from people who had died or left or were otherwise uninvolved didn't get carried forward; equally, though, it made sure that entries from lazy people didn't get into the handbook. Perhaps that was deserved, but it was also unintentional.

The answer is presumably to go back to the replacement strategy, relying on editorial checks, assisted by an automatic transaction log, to make sure that material no longer useful is replaced. That's a pity, but as my colleagues are quite capable of sending me outdated material anyway (see below), the editing is a necessity. (I noticed the phenomenon in this cycle because I maintained a log manually; without the log, I wouldn't have noticed except by some sort of final check for consistency and completeness.)

Send the wrong material. People get details wrong – particularly prerequisites and restrictions, though at least one textbook wasn't properly described. That's why expert editing is necessary. At least I can then get the correct material into the files so that it will have a chance of getting in next year.

Send editing instructions. People don't read the messages I send, so tell me how to edit the material (which they have deleted from the message) rather than just editing it themselves. This is particularly annoying when the reply amounts to "no change". This is the main obstacle to serious automation. (I was able to turn this silly behaviour to some advantage by using it as the basis for an assignment[7], but that doesn't count as an ameliorating factor.)

Making links automatically.

I insert some links into the HTML version as part of the automatic processing. These are primarily references to courses, which I recognise as 415.???, but I also try to pick up the department, SMIS, and the university.

As I know where all the course files are, that's easy; as I'm sometimes wrong, it's less than perfect. I'm wrong when a reference is to a course that ran last year but not this year, which really requires a reference to a previous year's entry. I can do that by brute force (only change the numbers I know are this year's, then assume that the others are last year's, for example), but it's worth looking for a better solution in the factory. Note that this substitution is particularly valuable in the HTML handbook, where you can't just turn over the pages, and almost useless in the paper version. That's why it's sensible to do it automatically without explicit mark-up if I can. (THINKS : is there anything else which can be similarly automated as a special case ? – lecturers to profiles in the graduate handbook ? – and to home pages everywhere else, including the profiles ?)

The other automatic links (department, SMIS, and university) are not so useful, and perhaps a bit gimmicky. They also go wrong sometimes – while "the university" almost always means Auckland, there are several cases (particularly in the graduate handbook profiles section) where it turns up meaning other universities, and the automatic links to Auckland are not appropriate. It is probably better to eliminate these; explicit links can still be inserted using the ordinary mark-up facilities where required.

Skeleton file.

There is a skeleton file for each handbook; it lists the files which go to compose the handbook, and identifies who's responsible for each. As time goes by, it acquires new bits and pieces, but the purpose is always to record information needed to construct the handbooks.

The skeleton file began as a list of fragments, in order, with a note of who's responsible for each. That's generally the name of a functionary, and in the course of time it is intended that the translation into real people should be automatic, but at the moment I do that. The fragments are not necessarily handbook sections; they might be chapter headings, or subsections, or whatever. At present, the level is determined by a mark-up notation associated with the headings within the text, but as I pointed out earlier a better solution in the long run might be to keep the level in the skeleton file. That makes it easier to change, and still works provided that everything is assembled serially. Note that the mark-up symbols for the different levels are still likely to be used for formatting, but they'll be inserted automatically as the stati of the headings are determined as the file is assembled serially.

Other accretions to the skeleton file have been abbreviated titles (also discussed earlier) and indications of differences in the source of material for the printed and HTML handbooks. Such differences usually correspond to cases where the information is originally found elsewhere; for example, the year's timetable for the university comes from a registry source, and it is sensible to direct a www enquiry to the registry's www page for the topic, as any changes will be reflected there. (A counterexample : in at least one case, my collection of information about people was more accurate than that in the department's www material. That's an argument for unifying the whole system.)

A complication at the moment is that certain sections have more than one author. The theory is that the authors collaborate on the final version, then return it to me; in practice, they tend to send me copies at every intermediate stage. A rather similar thing happens when people apply to other people for help. A sensible policy which should cope with this phenomenon is to accept only the last version, but it might be advisable to keep a few versions for safety. That's also easy to implement, provided that they don't change the E-mail subject : save all the E-mail as at present, and convert everything that arrives, replacing an older version if there is one.

Mark-up notation.

I can perhaps simplify the main features by keeping heading levels out of the way, but complication within the text is increasing. Notation for specifying URLs isn't simple; references to other sections might be non-simple too. I've devised a notation for pictures, but it's incomplete (it only works for the HTML files) and guaranteed not to last for long.

How much mark-up do I want ? Clearly, the minimum possible consistent with providing authors with the facilities they require without increasing the load on the Information Coordinator. (The Information Coordinator doesn't much mind the work, but the result is likely to correspond more closely with the author's wishes if the instructions can be encoded precisely and executed automatically.)

Things get even worse when the actions to be taken in different circumstances differ greatly. Two examples from this year's experience :

- I mentioned earlier that there are cases where a text file must be inserted to produce the printed handbook, but for the HTML version a reference to an authoritative www page is preferable. These are usually concerned with references to faculty or university information, where the printed information has been taken from some source equivalent to the external www page in the first place.

- Versions of courses with the same number but presented in different semesters or on different campi. (See above for further discussion.) These are handled normally, as individual files, for the printed handbooks, but for the HTML version it is reasonable (well, I've done it for the 1999 handbooks) to collect all versions together into a single HTML file; cross references are common, and it makes comparison easier in cases where the versions differ slightly. Even if some different presentation is selected, special treatment for the files with similar names is still likely to be necessary.

It is perhaps significant that my current "solutions" to these problems do not depend on mark-up methods. The first is handled through different entries in the skeleton file, while the second is so far managed manually.

CONCLUSIONS.

There are not many conclusions. Indeed, there are too few conclusions, as many of the items I've mentioned require some sort of decision. These decisions will have to be taken in due course, but that hasn't happened yet. Meanwhile, though, I think there are three generalisations which I can safely offer :

Metainformation : As the system (and particularly the mark-up characters) becomes more complex, I have to provide information for people writing stuff. I would prefer to keep it all simple, but I don't think I can – if there are to be several versions, I have to get the information somehow. I already have a WWW page with notes on mark-up conventions. That should be expanded, and generally better organised. It also needs a summary of the codes for cross-reference (that is, the skeleton file).

Decisions : There are a lot to take, and I think that many could usefully be discussed by more people than me. Examples are : how to present multiple versions of a course (101, etc.); what sorts of mark-up are desired and what are tolerable; whether I should maintain histories of publications, students, etc.. What's the best way of getting sensible opinions ?

Extent : How far should these information collections extend ? Do we really want a unified structure for the department ? If we do, people are going to have to follow specific procedures more diligently, though we should end up with a better result. How much regimentation is tolerable ?

REFERENCES.

1 : G.A. Creak : *Regal and Imperial*, unpublished Working Note AC126 (in preparation, 1999 March).

2 : G.A. Creak : *Background for a document generator*, unpublished Working Note AC115 (1997 September).

3 : G.A. Creak : Question 2, in the 415.340 Examination (G.A. Creak, R. Sheehan (eds), 1998 November).

4 : G.A. Creak : *Information structures*, unpublished Working Note AC111 (1997 May).

5 : G.A. Creak : *Handbook preparation : the future (perhaps)*, unpublished Working Note AC120 (1997 December).

6 : G.A. Creak : *Designing the document factory*, unpublished Working Note AC117 (1997 November).

7 : G.A. Creak : 415.340 Assignment 1 (1998 July).