WORKING NOTE AC115

Alan Creak
25 September 1997

BACKGROUND FOR A DOCUMENT GENERATOR

Reviewing my experience of building the 1998 department handbooks by hand in 1997, I consider possibilities for automating the process in future.

THE JOB.

Producing a new handbook is a matter of reviewing the previous year's edition, identifying any specific changes, and then collecting information - usually amendments, but sometimes new material - from many people in the department. The resulting changes must then be brought together and appropriately formatted in at least two ways : for the printed handbook (in fact there are two, but the mechanisms are precisely the same in both cases), and for the world-wide web (www) version.

It was clear from the start that some of this process can be automated. Identical requests for revisions, with the addition of the text to be changed, are sent to many people; these several mail messages can be completely automated given the required information. On the other hand, it seems that other components must remain manual for some time at least, though in some cases there is potential for automatic assistance. This year, I made little attempt to implement any automatic procedures; it seemed more appropriate to work through the whole operation with some care to gain more experience of the issues involved before making any judgments which might turn out to be unfortunate.

There is also considerable scope for expansion. The handbook information overlaps significantly with other sorts of information required for other purposes. Lists of course information are required for the Calendar, information on examinations and tests for faculty and registry, and information on set books for the bookshop are regular examples. Lists of who-does-what are useful in many ways.

THE HANDBOOK TASK.

The whole process is executed in several stages. The procedure followed in 1997 can be described roughly (and with some liberties taken to make the description more useful) as composed of these operations :

1 : The head of department determines the lecturing duties for the next year, and perhaps also determines some other responsibilities of various members of the department.

2 : The head of department and information coordinator review the current handbooks, identifying specific details that should be changed. These are mainly matters of the structure of the handbooks as a whole; details of recurring items, whether descriptive material, information about specific courses, or personal information of members of the department, will all be checked by those responsible as a matter of course.

3 : The information coordinator prepares a skeleton for the new handbooks. Most of this is likely to be much the same as the previous year's version, but any changes must be incorporated.

4 : The information coordinator sends copies of each item in the handbook, by electronic mail, to whoever is responsible for its content, requesting a reply by electronic mail.

5 : The information coordinator collects the replies. Each must be checked, and any questionable points cleared up by further enquiries from the person responsible.

6 : The information coordinator must then assemble the required components for each required output, and convert each into the appropriate form - formatted in (for example) Word for the printed text, and HTML for the www material.

7 : Finally, the complete document in some legible form is checked by anyone available - in this case, the text for the handbooks were checked by the head of department and information coordinator - for general consistency, and corrections made as required.

INFORMATION FLOW.

That description of the procedure is deceptively simple. In fact, there are a number of dependencies between the items.

- Items 1 and 2 are independent, and are the only independent items so far as the handbook is concerned. (That isn't quite as bad as it seems; there is at least one other significant collection of information which is independent, as will appear shortly, but that wasn't clear at the beginning of the exercise because the different collections weren't separated.) They therefore determine the timing of the whole exercise. In 1997, item 1 didn't happen until the end of the inter-semester break, which was too late.
- Item 3 depends on Item 2.
- Item 4 depends on Items 2 and 3.
- Item 5 depends on Item 4, and also on Item 5.
- Item 6 depends on Item 5.
- Item 7 depends on Item 6.

ANALYSIS.

Analysis of this pattern of behaviour, confirmed by experience (or perhaps experience, supported by analysis of the pattern of behaviour) brings to light some interesting features. The diagram on the next page is an attempt to present these and the relationships between the various parts of the process, excepting Item 6. The diagram is, inevitably, idealised, and some of it hasn't quite happened yet, but the major features of the process are clearly shown. The top row lists the six collections of data which seem to be the most useful bases for building the handbooks. These were not all readily visible at the start of the exercise (indeed, the only obvious one was the timetable - all the rest of the information was distributed, so that (for example) course information appeared in four different places, people information in several, functionaries hardly at all, and so on.) Defined in a little more detail :

The **Functionaries** file is a list of all the responsibilities (lecture courses and administrative tasks) of - in principle - everyone in the department. At the moment, it includes only academics, because that's the major requirement for the handbooks,. but that will have to change.

The **Skeleton** file is simply the ordered list of parts for the handbooks. It includes the identity of each component, and where to find it.

The **Text** file is the collected text for the handbooks, excepting that found in other repositories.

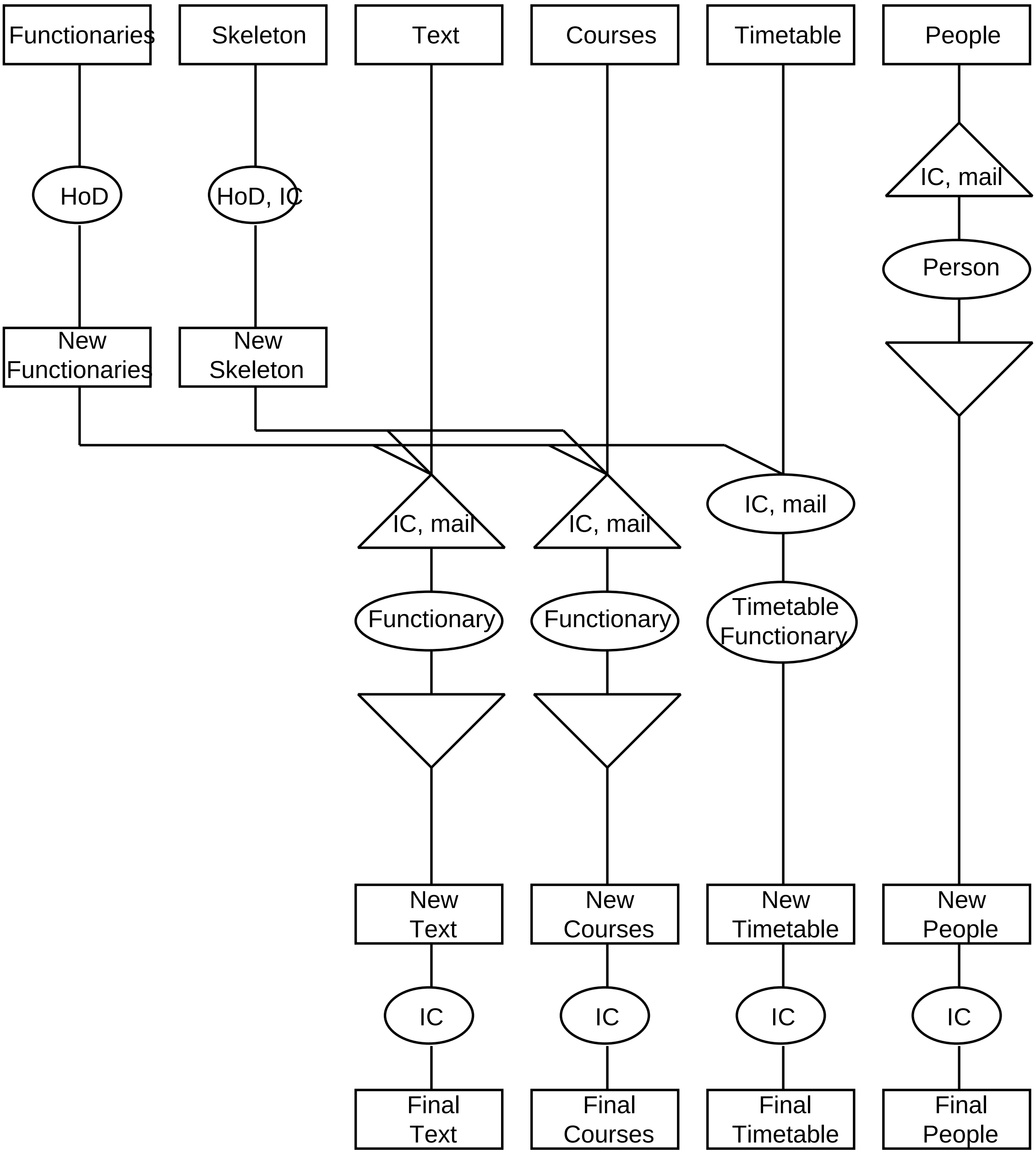
The **Courses** file is the complete list of lecture courses offered in the department, including information such as prerequisites, textbooks, course description, and so on. It need not include the lecturers (available from the Functionaries file) or the lecture times (from the timetable), etc., though it might be useful to include these items simply for convenient reference to the information.

The **Timetable** file is what it says - the timetable for lectures and for tests.

The **People** file is the collection of information about members and staff of the department, such as personal interests, publications, and the like.

With this organisation for the data, the processing becomes comparatively simple. It is clear that there are three activities (not two, as I originally thought) that can be started without waiting for any others :

- Reviewing the People file. This is possible because there is no doubt of who are the appropriate functionaries - the people who know about the People file are the people.



- Reviewing the Functionaries file. This is possible because it is a matter of department policy, the sole responsibility of the head of department, and independent of any other information.

- Reviewing the handbooks Skeleton file. This is again a policy matter, and the responsibility of the head of department. I've included the information coordinator because that's what we did for the 1998 handbook, and it was helpful to work through the decisions, but strictly it's up to the head of department to make the decisions.

The other activities depend on a selection of these primary activities in obvious ways; revisions of text and details of lecture courses are carried out by the responsible functionaries, and these are not known until the New Functionaries file is decided, while the new Timetable file depends on knowing the timetable functionary, but also on who is involved in which lecture course.

The last stage shown has the information coordinator producing final versions of the files from the results of the information gathering exercise, the new files. This is a stage of critical review, simply summarised as making sure there are no silly mistakes in the files. It's surprising how very clever people can be incapable (or unwilling) to conform to very simple requests, and in consequence produce nonsense. It is very difficult to see how this step, or some equivalent, could ever be automated.

ON DRIVING THE FUNCTIONARIES.

Many parts of the handbook (strictly, all of it, but not all of it happened this year) are written and maintained by the people I have identified as functionaries - in some cases people with specific responsibility for part of the material, and, in the case of the People file, by the people concerned.

In each case, I sent a copy of the material, including certain formatting symbols, to the functionary, requesting that the material be amended as seemed appropriate and returned. I particularly requested that the formatting symbols be left unchanged, expecting that the significance would immediately be clear to my computer-literate colleagues. Returns fell into several categories.

- The majority understood, and went slightly out of their way to return exactly what I wanted. I could simply copy and paste from the returned mail to my file.

- A significant minority came close, but used the ordinary "Reply" function of the Eudora mail package, which gave almost what I wanted but with portions of the original decorated with a > symbol at the beginning of each line. That was almost as easy to deal with as the ideal reply.

- Others just returned the bits they wanted to change. That gave me a significant editing task to do.

- Yet others ignored my formatting, and wrote me either a short essay on what they wanted me to do, or gave me a copy of how they'd like it this year without any of my formatting symbols. These were quite hard work.

- Several chose not to reply without a reminder, or at all.

It is clear that I can't - ever - rely on my highly intelligent colleagues to reply helpfully, and that it will always be necessary at least to monitor the returns to check the formatting, or lack thereof.

Nevertheless, the majority reaction seemed to be favourable, so it is probably sensible to continue with this method of soliciting changes.

STEP 6 : ASSEMBLING THE FINAL DOCUMENTS.

The information files are *not* intended to be components of the final documents; they contain the material that will be used for the final documents, but it is very much raw material. There are two reasons for proceeding in this way. First, because the raw material will be put together in at least two, and probably more, ways to produce different final documents, so it is not sensible to impose any format upon the material which might not be appropriate for all circumstances. Second, because the information files are edited and generally messed about by other people - the various functionaries who revise them - and they cannot be relied upon to preserve such formatting material as might be included, and, more particularly, to add required formatting to new material. It is much more sensible to keep the data as raw as possible, and to impose any required formatting later where it is safe from interference.

When I received the files, they were roughly formatted for the handbooks, with a strong accent on "roughly". It was very clear that several people had edited them at some time or other, and that few, if any, had any real idea of how to use a word processor effectively. The files were littered with extraneous tabulations, the usual style was Normal with arbitrary local style changes and added tabulations or (worse) spaces to format things manually, dotted leaders for tabulations were inserted manually, so any minor change in format destroyed the layout completely, and so on. There were also traces of an attempt to combine Word and HTML in the same file using hidden text; much of this had then been edited by someone with no idea at all of what was happening. All this made it very clear that the policy of separating data collection and storage from formatting was greatly to be desired.

The separation was not quite complete in this year's exercise, but two classes (lecture courses and personal information) were identified and moved into their own files. This proved very satisfactory; it immediately became very straightforward to select specific information for (for example) all courses by simple operations using Word. The trick is managed by explicitly labelling every field of each file record with a distinctive label; the pattern chosen was !!<sometext>||.

I am tempted to call this notation YAML (Yet Another Markup Language). It certainly *is* a markup language; why do I need a new one ? I have two reasons, neither irresistibly compelling, but together perhaps sufficient justification. First, I thought it better to avoid some notation with which any of my colleagues might be familiar in case they were tempted to manipulate it; the object of the exercise is to achieve my sort of formatting, and to use an existing notation could make that difficult. Second, I didn't want to commit myself to any other language in case it turned out to be inadequate.

Given this labelling, then, suppose for example that it is desired to extract from the Courses file a table of course numbers (label !!number||) against assessment details (label !!assessment||). The record for a course looks something like this :

```
!!number||
415.340

!!title||
Operating systems

......

!!assessment||
70% examination, 30% coursework

......
```

This sequence of operations will generate the required list :

```
Change all !!number||^p to ^t
Change all !!assessment||^p to ^t^t
Text to table
Delete column 1
Table to text
Change all ^t^p to ^p
Repeat
     Change all ^p^p to ^p
until no further change
Change all ^p^t to ^t
```

(^p denotes a paragraph mark, and ^t a tabulation.)

There is an element of the ad hoc about that procedure, but it works, and proved very useful when there was a requirement for just that list. Nevertheless, I do not recommend it as a method for regular use; it appears here only to illustrate the power of the Word operations. A similar sequence can be used for quite complicated formatting operations; for example, this sequence :

```
Change all !!assessment||^p to !!assessment||^t
Change all !!assessment|| to (nothing) style "details"
Change all !!assessment|| to Assessment : format italic
```

converts the original text into a set of lines something like :

Assessment : 70% examination, 30% coursework

depending on the definition of the style "details". An alternative sequence can be written which gives the rows of an HTML table. By this means, given sufficient appropriate labels and a suitable template with styles defined as required, a whole document can be formatted uniformly and simply using a predefined Word macro. Notice particularly that only the labels and text are required in the raw files; any other formatting imposed by the functionaries can be overridden with ease.

There remains only the question of how the raw document is brought together ready for formatting. The answer is in principle simple : given the files identified above, with items appropriately labelled, the instructions implied by the Skeleton file guide the selection of records from the other files quite precisely. Some (mainly those without repeating components) are identified explicitly; others, particularly those from the People file and Courses file, are assembled by iterative operations on each record throughout a complete file. Some are defined indirectly; for example, the list of lecturers for any lecture course is obtained by reference to the Functionaries file for the people with lecturing responsibilities for the course.

None of this is difficult, though some is tedious. All of it can be programmed without much difficulty, though with some expenditure of time. It seems close to certain that the expenditure of time will be well justified by very considerable savings over the succeeding years. Even if the details of the handbook change greatly, the low-level structure is unlikely to be very different, and given a careful design of the automatic system it should be readily adaptable to any new pattern.

STEP 7 : FINISHING TOUCHES.

In the final step of the preparation, the document is prepared in a comparatively easily legible version close to its final form, and is read as a whole by any available experts. Inevitably, these turned out to be the head of department and the information coordinator. They carry out a final check for anything at all, but this is an opportunity to inspect the overall structure and sequence, and to look for duplications, or contradictions.

A significant number of mistakes turned up, some at the local level despite the earlier cycles of checking. A few others were also identified; some enrolment information was duplicated, anomalous entries in various lists were noticed, and other details sorted out. These were corrected in the version finally accepted.

One further necessary task is to reflect these changes back into the collection of files. This operation was carried out manually; in principle it can be automated, but the necessary overhead might not be worth the trouble. This year, several changes were necessary, but many of those were the consequence of failures to collect the correct information in the first place, and should be much reduced with an automatic system.

REQUIREMENTS.

This year's experience shows that several things are necessary for the smooth running of any scheme of this nature, whether or not it is automated. Some things I needed to know, but weren't always easy to find :

- Details of new members of the department, and who has left. Perhaps there is a current list somewhere, but I don't know where. This is all part of a people information system which should operate in real time as people come and go - and it includes the notice boards which I mentioned in my first exploratory note[1] on information coordination.

- Who is responsible for every part of the handbook. Some of it's obvious; all lecture Courses file have supervisors, there are functionaries for diplomas, examinations, etc. After dealing with those, though, there is a great deal of unowned material. This year, a lot of it went to the head of department by default, but that isn't obviously a very good idea unless there's no alternative. There are also project Courses file without official supervisors; someone should take responsibility for each of these.

- What new developments should be in the handbooks. This again is left to the head of department. Ultimately, that's correct, but in this case it would be better if there were a mechanism for designating events as they happen as potential handbook material.

- Reliable specifications for prerequisites, etc. (Here, "etc." includes the course titles themselves !) My academic colleagues are clearly too busy to check obvious things like that, but it is fairly important that they be correct and consistent between all our documents.

On the other hand, there was at least one thing which I found I didn't need to know, because it came out of the Functionaries file :

- New Courses file. The details come from the functionaries in charge (the course supervisors), but the list of courses should always come from the Functionaries file (or perhaps from the Timetable file - at least the two should be checked for the same list of courses). That's because it is necessary to omit any course not given in the year covered by the handbook. Just working from the previous year's list is likely to lead to confusion.

Given these, and other similar details, and the development scheme laid out earlier, the essential tasks of producing the handbooks should be quite well defined and controllable. But things are not always quite as simple as might appear.

COMPLICATIONS :

The process is simple in principle, but has very many components, and a significant proportion of these are prone to sub-optimal performance in various ways. Some examples :

- Objections can arise during the process. It can turn out that the Skeleton file determined at the beginning of the exercise is flawed. Perhaps someone, while contemplating some part of the collection of data for which he is responsible, notices some anomaly. Obviously, any such event must be dealt with, but the effect on the process is unpredictable.

- Electronic mail is not completely reliable, and addresses can be wrong. All being well, it might be hoped that this problem should solve itself, as electronic mail addresses known to be reliable can be kept in the People file. Still, it caused some trouble this year.

- People don't do what they are asked to do. For whatever reason, they don't conform to simple editing conventions (like "don't touch the formatting signs in the text"), and instead send what they think is adequate. Some people don't even read mail, and therefore not only don't receive the requests for revision, but - having missed other messages earlier - don't understand why they are expected to do things. (Two academics, not having read an earlier message from the head of department defining people's responsibilities in 1998, did not understand why they received requests to provide information on certain Courses file.)

- The department's data management is such that there is little coordination between different activities, so that some things happen twice for different reasons. An example which occurred in this exercise was the request for details of course assessment independently of the request for the same information for the handbook. All being well, this should stop as the organisation improves.

- Interdependencies within the handbook data themselves have not been sorted out. There is still duplication, and unnecessary requests for information which can be worked out from existing files. For example, the list of lecturers for each course can be worked out from the Functionaries file.

- Various lists are not kept up to date. To get information for the People file, I used input from the Functionaries file, the department's www pages, the telephone directory .... As another example, there was no clear record of the department's prizes which had been awarded in the previous year. (Part of that wasn't our fault; there was not, and rarely is, any record of information from the faculty examiners' meeting at which some prizes are awarded.)

- Finally, because of these possible pitfalls, it is a mistake to update the old files as information comes in. It is preferable to keep everything until the decisions are made, then to construct the whole new file as a unit. A simple example is the values of section numbers in the handbook; I had used these to identify electronic mail messages, but because of reorganisation some numbers changed. Because of that, I had to remember a number of arbitrary transformations. As it happened, I did so without mishap, but it would have been very easy to confuse things. Eventually, section numbers should be allocated automatically; that implies that any cross-references should be identified with symbolic names which can be set automatically to the correct values.

OTHER MATTERS.

This account of events is primarily directed at the central process of compiling the handbook and the preliminary steps in developing a more automatic system, as experienced in 1997. It is worth recording a few additional notes which extend beyond those temporal and topical limits. Here they are.

An obvious eventual goal is to automate the rest of the process. This year's experience suggests that some of the operations, though almost certainly not all, can usefully be made automatic. The most significant component is the initial cycle of requests for revision; the People file can be run at any time, while the Courses file depends only on the Functionaries file. So, in principle, does the other material, provided that appropriate functionaries can be identified for each item. An operation which almost certainly cannot be automated is dealing with the material returned in response to the requests. Because people don't necessarily follow instructions, manual comparison with the original item is necessary on return, though perhaps it can be simplified with software help.

Having more or less sorted out how to handle information retrieval from people within the department, the next question is how to cope with people not in the department. We use quite a lot of such information; details of instructions for enrolment, various student services, the SMIS director's message, and so on are examples. This year I have relied on www information, history, telephone calls, and other arbitrary sources, and have probably amassed a reasonable selection of almost correct information. This is not entirely satisfactory, and a search for better sources would be worth while. Ideally, everyone else will happily fall in with my system; in practice, that's a bit improbable, so I should find some way to collect whatever is most convenient for them, and massage it into the form we want as painlessly as possible. It would not be unreasonable to isolate these components as further files in the data collection - a SMIS file, an Enrolment file, etc.

There are also links with other data : the science faculty handbook contains some information about our courses, which is not the same as ours, but follows a similar pattern; the commerce faculty handbook also contains a list of certain courses, with less extensive descriptions than we use; the Calendar includes a list of all courses, again with different descriptions; and so on. All these must be revised each year, and it probably makes sense to include them all in the Courses file so that the appropriate functionaries can review all of them at once.

CONCLUSION.

I've put in a lot of work on this in 1997, but it has probably been worth while. It seems that the prospects for automation of much of the process are good; in so doing, we shall build up a set of files which can also be used for information which extends beyond the range of the handbooks themselves. I have some bright ideas on how I can implement this system, which have several intriguing side effects (I learn Java, and it advances my research); all I need now is about two clear free years before the 1998 cycle starts.

REFERENCES.

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