TGCW07: Encoding the Oxford Milton — Interim report

Dominic Dunlop

31st May, 1991

1 Background

In order that OUCS British National Corpus staff could gain experience in translating from an unpleasant presentational markup to a complex structural markup, Lou Burnard came up with some material associated with another project, the Milton Textbase[**?**]. The material was a half-inch magnetic tape carrying an unknown number of files encoded using Interset Typesetting codes [**?**, appendix3.ii], together making up the Oxford Authors edition of selected works of John Milton[**?**].

This brief paper describes experience to date on the work of translation — work which, at the time of writing, is far from complete.

2 From Tape to Encoding

Consideration of the problem of getting from a tape of unknown format to a text marked up with complex structural encoding led to the drawing up of a list of process steps:

1. Read tape.
2. Convert file(s) to more tractable format.
3. Translate tractable format to Milton encoding by one of the following routes:
   (a) Direct translation.
   (b) Translate tractable format to Corpus Document Interchange Format (CDIF)[**?**], and thence to Milton encoding.
   (c) Translate tractable format to TEX, and thence to Milton encoding.
   (d) Translate tractable format to TEX, and thence via CDIF to Milton encoding.

The need for the second step became apparent as soon as the tape had been read: the files contained no carriage control (that is, they did not contain lines of text delimited by line feed-carriage return, or some similar sequence), and incorporated large tracts of apparently semantically meaningless null characters, together with a smattering of control codes. Such a format is inclined to give many text-processing tools indigestion: such tools expect to process lines of some distinctly finite length, and can react badly to unexpected control characters. The "tractable format" consisted of the original text with all control codes stripped out, with typesetter commands isolated on separate lines, and with text word-wrapped to a reasonable line length.

Several alternative methods of carrying out the third step were considered. The possibility of initial translation into TEX, which can be used as a solely presentational markup, was briefly considered as a means of checking for correct identification of presentational features (white space, page and line numbers, etc.) before translating to a structural markup. Used in this way, TEXmight be useful as a *lingua franca* to which all presentational markups could be translated prior to their conversion to a common structural markup. The idea was quickly rejected, however: it did not seem to deliver sufficient benefit to be worth the effort.

CDIF was also considered as a half-way house — particularly since, after all, the job of the OUCS BNC staff is to generate CDIF, not Milton encoding. The problem with this approach is that CDIF does not encode some of the information captured by the more complex Milton encoding; the information is lost. For example, Milton encoding requires the identification of verse lines which are indented; CDIF discards information about leading whitespace. Consequently, it was decided that direct translation from the tractable format to Milton encoding was the way to go.

3 Problems Encountered

3.1 Reading the tape

The tape turned out to contain 103 files (or possibly 102 files preceded by a label) with a strange blocking factor, and under- or over-sized blocks as the last block in each file. Consequently, it took several attempts to read it correctly, interspersed by the examination of dumps of the file contents in order to determine whether the sum of the files corresponded to the whole of the book. This took about a day all told.

3.2 Conversion to tractable format

Two problems were encountered here. Firstly, the choice of tool was probably inappropriate: the UNIX *lex* lexical analyzer generator is intended to be used for

the creation of production tools — compilers and the like — rather than for quick hacks. It also dislikes codes outside the ASCII character set. Consequently, the process of program creation and generation is slow, but results in applications which run satisfyingly fast. This balance is probably not correct for prototyping applications associated with text capture for the Corpus.

Secondly, the list of Interset typesetter commands given in [**?**] was incorrect and incomplete — a situation which is likely to recur whenever a new file format is encountered. An unanticipated benefit of the use of *lex* was the speed with which the 3.5 megabytes of source could be reparsed in order to analyze the coverage of tweaked lists of typesetter commands. Miscellaneous UNIX tools were also called into play to advantage, so avoiding the need to write programs to perform specific tasks. (Except, of course, that a shell command line consisting of a long pipeline is arguably a program. . .)

In the end, it turned out that, of 183 possible commands, the text used only 78. Of these, some occurred over 10,000 times, while others were used less than ten times. Again, this is likely to be typical of texts received.

3.3 Files

It has already been mentioned that the text was delivered as 103 separate files. These bear little or no relation to the structure of the document: they do not correspond to chapters, works, or whatever. Three possibilities exist for further processing of the text:

- Retain existing division into files;
- Repartition into files reflecting document structure; or
- Process as one large file.

No decision has been reached as to which of these paths should be followed.

3.4 Availability of printed copy

Without a printed copy of the *Oxford Milton*, our work would be much more difficult, if not close to impossible. Having the printed text allows us to interpret the effect on layout of the sequences of Interset commands that we see in the files.

3.5 Style of original markup

With just this single example of a text prepared for publication using Interset typesetter commands, we cannot know which sequences of typesetter instructions are an 'idiom' used by a particular compositor to achieve a particular effect, and which correspond to the accepted, or indeed the only, way to make

something happen. Consequently, if we develop a grammar which can successfully map sequences of Interset commands from the *Oxford Milton* into start and end tags for structural features, we cannot be sure that the grammar will be valid for other texts which also use Interset codes. (Indeed, we can be fairly certain that it will not.)

4 Work Outstanding

Work has only just begun on the task, likened by Gavin Burnage to that of solving a crossword, of examining the sequences of typesetter commands used to introduce particular presentational features, and so developing a grammar which can pick out significant commands, while ignoring those which have nothing to do with the structure of the document. This, the core of the task of translation, will be an iterative and time-consuming process.

5 Lessons Learned

- Getting data off magnetic exchange media can be annoyingly time-consuming.
- We need to try out more text processing tools in order to find the best balance between speed of prototyping and speed of processing.
- In processing a new text format, it is almost essential to have a copy of the corresponding printed work.
- CDIF discards presentational information which might be useful in the subsequent creation of more specialized corpora.
- Translating from presentational to structural markup is difficult.

References

[1] *The Interset System Typesetting Manual,* Interset Computer Systems (UK) Ltd.

[2] *Markup Manual for the Milton Textbase,* Lou Burnard, 30th December, 1990

[3] *The Oxford Authors: John Milton,* ed. Stephen Orgel, Jonathon Goldberg; Oxford University Press, 1990

[4] TGCW01: *Markup scheme for the British National Corpus,* Lou Burnard, 25th April, 1991