column

Don’t give authors the class files!

Kaveh Bazargan
Focal Image Ltd
kaveh@focal.demon.co.uk

Publishers are now used to taking TEX or LATEX files from authors and using them in publishing books and journals. It has become a standard procedure now for LATEX class files to be distributed to interested authors. Publishers will subcontract TEXies to produce these to their particular styles. It is, in some cases at least, a bit like web sites. They want them because everyone else has them – as much PR as anything else.

Based on being involved in the typesetting of tens of thousands of LATEX pages, I want to argue against giving authors these class files. Here’s why:

- Not all class files can be used by the average author. This is not necessarily because they are bad class files, but they need careful use. They need some manual tweaking in the final stages, and this is not always trivial. The trick is to do this with minimal effect on the structure of the source code, so the code can be reused. The author may attempt to do this, but will invariably come up with an inelegant solution. He may even edit the class file in an effort to fix things.
- Authors will get the impression that they are producing camera–ready copy. In a minority of cases they will be able to produce CRC successfully, but again, in an effort to get the pages looking right they will produce ugly code to to generate reasonable output.
- Most class files are written by LATEX specialists for their own use in–house, for example for typesetting material sent in by publishers. (I wish we could banish that term – typesetting!) They sometimes have dirty fixes which they have never had time to clean up, but which nevertheless produce the correct results.
- Commercial fonts cannot legally be copied and sent to authors. So the publisher would have to have multiple copies of the fonts, for multiple platforms, in stock. Alternatively the style would have to be limited to public domain fonts.
- Presently, most publishers do not request an archive of the LATEX source code. They are mostly interested in POSTSCRIPT, PDF, and sometimes SGML headers. They are not worried about the structure of the source code. But as output becomes increasingly electronic, the structure of the code will be important, so that SGML or XML files can be produced with minimal work.

So what is the solution? Well, I think publishers should be more ‘pro–active’ in dealing with authors. In other words, they should change tactics and actively try to come up with a system that works. Authors will respond to well set out instructions, especially if they think it will improve the efficiency of processing their files.

I think that publishers should send out an ‘author kit’ to prospective authors, and active encourage the use of LATEX, rather than just say they will accept the files. Here is what I think should be in the author kit:

- **A ‘generic’ class file.**
  This class file would probably be based on the standard LATEX Article or Book class files, but with modifications. It would use standard Computer Modern fonts, and would be designed with generous line and page breaking glue, so as to avoid overful boxes wherever possible. In short the class file would be designed to make it easier for the author to enter his text. The generic class file would contain all the environments necessary, as envisaged by the publisher, but the visual output would look nothing like the final typeset pages.
- **A user–friendly instruction manual.**
  Written in the spirit of Lamport’s standard manual, a small user guide can gently encourage the author to produce clean, structured text. Any special instructions should be written into a `readme` file. I think that a well written set of instructions will raise the profile of the publishers considerably.
- **A fully working sample file.**
  This is probably the easiest way to commnunicate the workings of the class file. A simple document with examples of each environment available will help further to guide authors in the right direction.
- **Lamport’s manual.**
  Why not? I would say that for a prospective book author, this is not a major expenditure, and will pay off in goodwill.

Here are the advantages of taking this generic file approach:

- Publishers are free to implement more ambitious styles, getting away from the usual 'TEX' look. For example the designers can have a freer hand in choosing fonts, using shading, colour, PostScript tricks, MetaPost etc. All this without having to update the author class files at all.
- Authors can concentrate on what they should, namely the content of their paper – in the original spirit of TEX.
- Support for authors becomes a simple matter, being confined to one file.

I think the procedure outlined means less work for the author and the publisher, and more accurate typesetting. Typesetters would rather work with clean, structured LATEX files than files that look almost like the final product, but which have bad code. Here's another advantage: if the procedures work well, then the typesetter can return to the author the LATEX files used in the final typesetting of the book, by removing any visual codes used in the final stages of pagination. For book authors, the files can be used in preparing the next edition. This very rarely occurs at present. When conventional typesetting systems are used, the native file can only be manipulated by the same (usually expensive) typesetting program.