

Typography

Typographers' Inn

Peter Flynn

'C' stands for Euro

January 1st came and went, and we survived the introduction of the Euro, the planet's ugliest-named currency. I can now write cheques in €, do electronic transactions, and even lodge credits, should anyone be generous enough to send me money. As a L^AT_EX user, I have discovered the `\texteuro` command in the `textcomp` package (which I should have mentioned last time, but *mea culpa*, I am a recent convert to L^AT_EX and am still finding stuff squirrelled away that I didn't know about). And, I'm pleased to say, L^AT_EX's € is a much more suitable design to go with those serif fonts which have none of their own than the strange C-like designs put out by Microsoft in their TrueType replacement fonts (designed by and licensed from Monotype, of all people!). But `\texteuro` uses PostScript fonts (the T1 encoding) and while that's fine by me, it's not for everyone.

Full marks, therefore to Henrik Theiling for his `eurosym` package, which implements the original EU (sans-serif) design in roman, bold, italic, and outline using METAFONT, so it's usable in T_EX-based systems anywhere. No seriffed version yet, but the following table shows some of the glyphs available.

	n=sc	sl=it	ol
m	€	€	€
bx	€	€	€

Adobe also has the Euro in PostScript fonts available for download, including sans and serif versions (with serifs top and bottom, too!), and there's already a Euro in the `china2e` package (a METAFONT font). Bitstream sells a standalone pi font with the Euro symbol, and will customize your existing fonts for you (for a charge). Linotype sells nearly 200 Euro symbols for DM 100 but makes the same mistake as Monotype in pretending the E is a C in the serif versions.

The EU has laid down that the official design is to be used regardless of the surrounding font (in both style and colour, see <http://europa.eu.int/euro/html/entry.html>). Fortunately I don't know anyone daft enough to want to follow that diktat.

As I write, each of my Euros appears to be worth \$1.30 Canadian, so I've started to save for the TUG meeting in Vancouver.

L^AT_EX and glue

I've pretty much settled down to using L^AT_EX now. I don't make so many mistakes and I've stopped typing plain T_EX on the rare occasions when I actually write a document in raw code. Most of my text is authored by other means and converted to L^AT_EX for formatting, so my misgivings about the default L^AT_EX appearance mean that most of what I do is writing or modifying style files to publishers' specs. I've started to turn some of the ideas which have spun off from this into class and package files in preparation for a project I mentioned online recently and which I will be presenting in Vancouver.

I had several responses to my suggestion that we ditch the weird concept that the default style for reports should have chapters, most of them supporting a change. It's probably inadvisable to change the source of `article.cls`, as too many people have too much private code rigged to cope with its peculiarities, and they rightly rely on the stability of T_EX systems to maintain their text. What I'm aiming at is a package that repairs this and other leaks and seals them up so that authors have to spend less time fiddling and thus have more time for writing¹. Articles should work more like authors and publishers expect them to, books more like books, and reports more like reports. Maybe this will even help stem the flow of FAQs about these problems on `comp.text.tex`.

And what was that about glue? When you repair a puncture in a tire, you glue a patch of rubber to the inner tube. In the early days of cycling and motoring, tire rubber needed heat treatment after repairs to ensure the patch was properly bonded, and this treatment was called 'vulcanising'. To avoid the bond degrading as the rubber flexes in use, the glue is made of our good friend latex (plus assorted chemicals). Those of you with long memories may recall childhood cycle repairs with 'self-vulcanising' glue, which replaced the need for heat-bonding. Hence by a tortuous path the name `vulcan` for a package which seals the leaks in L^AT_EX — come to Vancouver and see (tell your boss you're off to learn about latex bondage or something: Dan Quayle will explain).

Backquotes

Maybe it's just the way that once you notice something once, you repeatedly see it all around you, but I've spotted the reversed quote (‘) several times,

¹ But where should we be without the inveterate fiddling author inventing new styles?



Figure 1: Type fac-simile of the SGML97 logo

including some extremely public displays which included the logo which appeared on all the posters, leaflets, proceedings, and assorted publicity for the XML'97 conference in Washington, DC. They got it right on the title page, and anywhere that it was reproduced from typed characters, but the logo itself, approximated in Figure 1 with CM fonts, used the reversed quote. I'm curious to know why so I've sent a message to the designer and I'll let you know.

Get writing

The new journal I mentioned last time, *Markup Languages: Theory & Practice* [1], is up and running. It's quarterly, peer-reviewed, and the first one of its kind devoted to text markup. I'm therefore repeating my call for articles: as the markup we use for \TeX and \LaTeX was one of the major advances in the move towards logic-based or structural markup, I feel that there is plenty of scope. Dip your quill in the ink and start writing now.

Postscript

Probably like many of you I've been using PostScript fonts for years. They're portable, convenient, reasonably accurate, and although the hinting isn't a full substitute for design-sizing, in most practical situations they work just fine. I don't do many jobs requiring extremely large sizes, so I haven't run into the problems that I am told exist in advertising work, for example.

But my guess is that most \TeX systems, particularly in research or academic sites, don't have PostScript fonts (a font survey would be interesting). There is a cost involved once you go beyond Charter and the other free PostScript fonts, and although it is small per font, it can be outside the budget of many individuals, especially students, and even some projects. The bigger stumbling-block is the installation: I had my own problems when I did the first few fonts, but I was lucky to have generous and helpful people on call who patiently explained what I needed to do; and this was long before the new \TeX Directory Structure.

I have therefore finally gotten around to writing a new PostScript font installer. The old `mkvf` program which I wrote to take the Virtual Font route was a shell script, and fairly crude; the `mkcd` batch

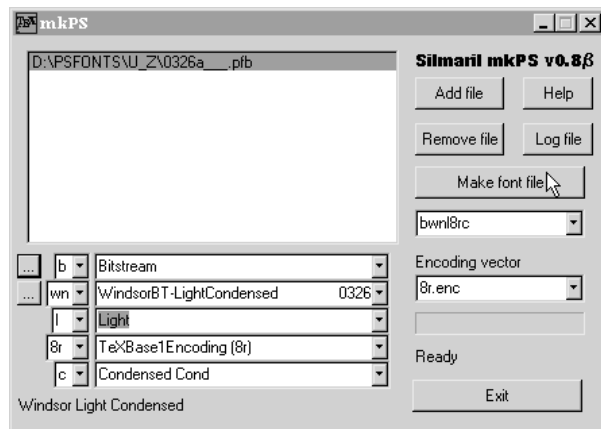


Figure 2: The MKPS PostScript font installer

file for DOS/Windows systems which followed was never satisfactory as the restricted operating environment available precluded it doing all that `mkvf` did. This time I have taken the plunge into Windows'95 and rewritten it as a windowing utility: it's the most prevalent platform I support. The tool I used, *Visual DisplayScript*, is a very simple and effective way of tying together a simple interface to make a little utility (see Figure 2).

It assumes the TDS, although you can change that if you store your fonts elsewhere, and it makes reasonably intelligent although by no means fool-proof attempts to deduce the Karl Berry fontname abbreviations from the extended descriptive name in the AFM file (part of this grew out of having to rescue a client's broken installation where all the AFMs were corrupt and I had to try and dig into several hundred PFBs for the same data). I hope to have a distributable beta release by the summer: if anyone can recommend a similarly simple environment for producing X Window mini-apps, I'd be happy to hear of it.

The core that does the work is about four lines: `afm2tfm`, `vptovf`, some file-copying, and the appending of the relevant line to `psfonts.map` (it does assume `dvips`: it's all I know about). What takes the time, as usual, is handling the configuration, the deducing, the file-loading, and working out the name (and checking in the fontname map files). The user should be able to drag and drop a PFB file onto it, check that it has correctly resolved Bonemontano, Inc's *Gracatia Sancta* Skinny Weird DemiBold into `zgsdw8ro` and then just go ahead and do everything, including creating a skeleton FD file. If you're tired of hearing people complain that \TeX systems have only got one font, and it's soooo hard to get it to

work with anything else, mail me to go on the beta list. And no, I'm not offering €1.00 per bug.

References

- [1] *Markup Languages: Theory & Practice*. MIT Press, Cambridge, MA. ISSN:1099-6621.

◇ Peter Flynn
Computer Centre
University College Cork
Ireland
pflynn@imbolc.ucc.ie
<http://imbolc.ucc.ie/~pflynn/>