

Notes on setup of PostScript fonts for L^AT_EX 2_ε

Sebastian Rahtz

1994/02/03

Contents

1	Introduction	1
2	Portability	2
3	Standard installation	2
4	Installation of non-standard fonts and packages	3
5	Testing	3
6	User interface	3
6.1	Important — variant OT1 font encoding	4
7	New fonts, and package-writers interface	4

1 Introduction

This set of files offers a complete working setup for L^AT_EX 2_ε's NFSS and POSTSCRIPT fonts; it is based on the system I created between 1989 and 1991 for Southampton Computer Science, checked and updated for NFSS2, and subsequently L^AT_EX 2_ε. Package files are provided for the common PostScript fonts. You should be familiar with the standard L^AT_EX 2_ε files and their use to follow this document. Thanks are due to many people for correcting errors and helping, including Michel Goossens, Berthold Horn, Alan Jeffrey, Howard Marvel, Frank Poppe, Andrew Trevorrow. . .

Packages are offered to set documents in the common PostScript fonts, plus common free fonts (Charter, Utopia etc). All Lucida fonts are supported, MathTime fonts, and a selection of other Adobe and Monotype fonts. It is assumed that Karl Berry's naming scheme is followed. Textures users, however, can get a basic set of .sty and .fd files using Textures font names by running T_EX on `textures.ins`.

NOTE: I have included *new* font metric and virtual font files for PostScript fonts, for those who wish to use Cork encoding. These have been generated with Alan Jeffrey's *fontinst* program, and need a driver which understands virtual fonts.

2 Portability

My personal setup uses a POSTSCRIPT printer, Karl Berry's names for POSTSCRIPT fonts in T_EX, virtual fonts to map POSTSCRIPT to CMR layout, and I process my .dvi files with Rokicki's dvips. Other systems may fall over.

3 Standard installation

This distribution is provided as a set of .dtx files which need to be unpacked using *docstrip* to create user files. There are two types of file which result:

1. .fd files which describe a particular font encoding/family combination
2. .sty files which change the font defaults to use some new group of fonts (often just one default is changed)

Scripts are provided for *docstrip* in the form of .ins files, which simply need to be run through T_EX; when that has been done, install all the .sty and .fd files that result in a directory where L^AT_EX will find them.

You have an important decision to make at some point — are you going to use fonts encoded in the ‘Cork’ layout, or the old ones which look like the CM fonts described in the T_EX book? This manual will not attempt to explain why you should or should not use Cork fonts...Font description (.fd) files are created for both T1 and OT1 encoding by the installation procedure.

If you follow the Cork-encoding route, you need different T_EX font metric files and virtual font files. Suitable files for the standard 35 fonts, in both T1 and OT1 encoding, are in the `fonts/metric` CTAN directory, where you will also find a Unix-style ‘Makefile’ which shows how to create suitable metrics for extra Type1 fonts you may have — you will need the *fontinst* package to make this work.

Important. If you use the Cork (T1 in L^AT_EX 2_ε scheme) encoding, you will probably also need the ‘dc’ CM fonts to go with them, for maths and so on.

Family name	Full name
pag	Adobe AvantGarde
pbb	Adobe MBembo
pbk	Adobe Bookman
pcr	Adobe Courier
phv	Adobe Helvetica
pnc	Adobe NewCenturySchoolbook
ppl	Adobe Palatino
ptm	Adobe Times Roman
pzc	Adobe ZapfChancery
psy	Adobe Symbol font
pzd	Adobe Zapf Dingbats

Package	Sans font	Roman font	Typewriter font
times.sty	Helvetica	Times	Courier
palatino.sty	Helvetica	Palatino	Courier
helvet.sty	Helvetica		
avant.sty	AvantGarde		
newcent.sty	AvantGarde	NewCenturySchoolbook	Courier
bookman.sty	AvantGarde	Bookman	Courier

Table 1: Effect of package files

4 Installation of non-standard fonts and packages

If you want to create font files and package files for a greater selection of faces (see Appendix A for details of fonts, and table 2 for packages), run \TeX again on the other `.ins` files. These are a fairly random selection of fonts which the author needs, not put together in any rational way. You will not be asked about encoding in this case — T1 is assumed.

To create font and package files for Lucida and Lucida Bright (including Lucida Bright maths), run \TeX on `lucida.ins`.

To create an $\LaTeX 2_\epsilon$ package file for typesetting Object Z, run \TeX on `oz.ins`.

5 Testing

All installers should run `test0.tex` through \LaTeX and print the result, after installing their chosen setup, to ensure that things are more or less working. `test1.tex` will exercise your supply of PostScript fonts.

Do not worry if nothing but `test0.tex` works! `test0.ps` is a prebuilt version of `test0.tex` for you to compare.

6 User interface

The daily user will simply use one of the packages `times`, `newcent`, `helvet`, `palatino` etc to change the default text fonts for one or more of the roman, sans-serif and typewriter faces. Table 1 lists the effects of the package files created in the installation procedure.

The special package `pifont` gives access to the Dingbat and Symbol fonts. This is described in *The \LaTeX Companion*. Note that maths fonts will stay the same unless you have suitable fonts to load. If the Adobe Lucida Maths fonts have been purchased, and appropriate metrics obtained, loading `lucmath` will remove all reference to CMR fonts in the document. Alternatively, purchase the Lucida Bright font set and use the `lucbr` package.

6.1 Important — variant OT1 font encoding

The package files assume that you have already made the choice of which text font encoding scheme you prefer (T1 or OT1), and that it is the default when the \LaTeX job starts. However, you may need to override the defaults if you use the OT1 `tfm` files distributed with `dvips`, as some characters are not in the expected places. Similarly, Textures users will find things not quite right. For this purpose the package file `ot1var` is provided; this changes some macro definitions to make the necessary fixes, with options `[dvips]` and `[textures]` (default its `[dvips]`).

7 New fonts, and package-writers interface

Setting up new fonts, so as to create `.fd` files, is exactly as documented in the main $\LaTeX 2_\epsilon$ documentation. Suitable models for a large multifont installation can be found in `psfonts.fdd` (covering both old \TeX encoding and Cork encoding), but a `.fd` file for a single font is easy to write by hand, once you know which font encoding it uses. Thus the declaration file `OT1ppl.fd` for old-style \TeX -encoding Palatino (OT1) could look like this:

```
\DeclareFontFamily{OT1}{ppl}{}
\DeclareFontShape{OT1}{ppl}{m}{n}{<->pplr}{}
\DeclareFontShape{OT1}{ppl}{m}{it}{<->pplri}{}
\DeclareFontShape{OT1}{ppl}{b}{n}{<->pplb}{}
\DeclareFontShape{OT1}{ppl}{b}{it}{<->pplbi}{}
\DeclareFontShape{OT1}{ppl}{m}{sc}{<->pplrc}{}
\DeclareFontShape{OT1}{ppl}{m}{sl}{<->pplro}{}
%
\DeclareFontShape{OT1}{ppl}{b}{sc}{<->sub * ppl/m/sc}{}
\DeclareFontShape{OT1}{ppl}{b}{sl}{<->sub * ppl/b/it}{}
\DeclareFontShape{OT1}{ppl}{bx}{n}{<->sub * ppl/b/n}{}
\DeclareFontShape{OT1}{ppl}{bx}{it}{<->sub * ppl/b/it}{}
\DeclareFontShape{OT1}{ppl}{bx}{sc}{<->sub * ppl/m/sc}{}
\DeclareFontShape{OT1}{ppl}{bx}{sl}{<->sub * ppl/m/sl}{}
\endinput
```

Once the font family and encoding is declared (I use the three-letter unique combinations devised by Karl Berry for all PostScript fonts), each combination of series and style is mapped to the name of a `tfm` file; we don't need to worry about which sizes are available, because the PostScript fonts can be scaled to any size. The second part of the file sets up some substitutions for situations where we may have no font available (bold small caps, bold slanted, or bold-extended series).

If you need to create a new style file which can be used in the same way as the ones provided, simply use standard $\LaTeX 2_\epsilon$ constructs. This is `times.sty` (with banner information removed):

```
\ProvidesPackage{times}
\renewcommand{\sfdefault}{phv}
```

```

\renewcommand{\rmdefault}{ptm}
\renewcommand{\ttdefault}{pcr}
\def\bfdefault{b}
\endinput

```

Note the switch to make ‘b’ default for bold rather than ‘bx’.

If appropriate you could `\input` the style files described above, and use the utility macros to tailor your environment.

Appendix: non-standard fonts

Family name	Full name
bch	Bitstream Charter
hlc	B&H Lucida Bright
hlcs	B&H Lucida Sans
hlct	B&H Lucida Bright Typewriter
mgn	Monotype Garamond
mim	Monotype Imprint
mnp	Monotype NewsPlantin
mnt	Monotype Times New Roman
mpi	Monotype Plantin
mpp	Monotype Perpetua
pgm	Adobe ITC Garamond
pgs	Adobe MGillSans
pgs	Adobe MGillSans
plc	Adobe Lucida
plcs	Adobe Lucida Sans
pnb	NewBaskerville
pop	Adobe Optima
pun	Adobe Univers
put	Adobe Utopia-Regular
unmr	URW NimbusRoman-Regular
unmrs	URW NimbusSans-Regular

Package	Sans font	Roman font	Typewriter font
basker.sty		Monotype Baskerville	
bembo.sty		Bembo	
charter		Bitstream Charter	
garamond.sty		Adobe Garamond	
gill	Gill Sans		
imprint	Monotype Imprint		
mathtime	Helvetica	Times	Courier
mtimes		Monotype Times	
nimbus	URW NimbusSans-Regular	URW NimbusRoman-Regular	
optima		Adobe Optima	
perpetua		Monotype Perpetua	
plantin		Monotype NewsPlantin	
univers	Adobe Univers		
utopia		Utopia	
lucid.sty	LucidaSans	Lucida	Courier
lucbr.sty	LucidaSans	LucidaBright	LucidaTypewriter

Table 2: Effect of extra package files

Notes: a) `lucbr.sty` uses the font names for Lucida Bright which conform to Karl Berry's scheme. Use package option 'yy' to use the font names supplied by Y&Y. b) the `mathtime` package sets up the MathTime fonts for math. If you want to use just standard PostScript fonts for math, Alan Jeffrey's `mathptm` package does as good a job as possible (though it still needs access to some CMR math fonts).