

fonts: tutorial

NFSS: using font families in L^AT_EX 2_ε

Siep Kroonenberg
Kluwer Academic Publishers
Dordrecht
siepo@cybercomm.nl

abstract

This paper gives a brief overview of the L^AT_EX 2_ε NFSS font machinery and font definition files. It also gives examples of ad-hoc font changes with low-level NFSS commands.

keywords

NFSS, font families, font attributes

Plain T_EX doesn't support font attributes that you can vary independently: if you specify `\bf\it` then you get italics; if you specify `\it\bf` then you get bold. L^AT_EX 2.09 had the same shortcoming. The *New Font Selection Scheme*, NFSS for short, changed this: it provides mechanisms for organizing fonts into families, in which the different members are identified by different font attributes which can be modified independently.

NFSS is now integrated into L^AT_EX 2_ε.

This paper introduces you to the low-level L^AT_EX font machinery.

Font attributes

A font is specified by the following attributes:

- `\fontencoding{encoding}`: which characters are available at what slots. Normally, you don't explicitly concern yourself with encodings; the format file will have picked a default, which may sometimes be overruled by your class file.
- `\fontfamily{family}`: this may be e.g. be Computer Modern or Helvetica; [5] in this journal describes the rather desperate Berry system of naming used by most T_EX distributions; see also [6]. You could typeset your document in Times instead of Computer Modern by adding a command `\renewcommand{rmdefault}{ptm}` to the preamble.
- `\fontseries{series}`: these are really two attributes rolled into one: width ((medium-)condensed–medium) and weight (light–bold)
- `\fontshape{shape}`: options include upright, *italic*, *slanted*, SMALL CAPS

- `\fontsize{point size}{baselineskip}`: usually specified through high-level commands `\normalsize` or `\footnotesize`

Font families

For normal text font attributes may be {OT1} (old encoding), {ptm} (Times), {m} (medium weight) and {n} (normal shape). A command `\bfseries` would change the weight attribute from medium to bold and leave the other attributes unchanged. L^AT_EX needs mappings from sets of attributes to actual font files. Such mappings are defined in *font definition* (.fd) files. One .fd file lists all fonts available for one encoding and one family. For instance, a file `ot1ptm.fd` describes all fonts belonging to the ptm (Times Roman) font family with the OT1 (Old T_EX) encoding. It looks something like this:

```
\ProvidesFile{ot1ptm.fd}
[1997/09/30 Fontinst v1.6 font
definitions for OT1/ptm.]

\DeclareFontFamily{OT1}{ptm}{}

\DeclareFontShape{OT1}{ptm}{b}{n}{
<-> ptmb7t}{}
\DeclareFontShape{OT1}{ptm}{b}{sc}{
<-> ptmbc7t}{}
\DeclareFontShape{OT1}{ptm}{b}{sl}{
<-> ptmbo7t}{}
\DeclareFontShape{OT1}{ptm}{b}{it}{
<-> ptmbi7t}{}
\DeclareFontShape{OT1}{ptm}{m}{n}{
<-> ptmr7t}{}
\DeclareFontShape{OT1}{ptm}{m}{sc}{
<-> ptmrc7t}{}
\DeclareFontShape{OT1}{ptm}{m}{sl}{
<-> ptmro7t}{}
\DeclareFontShape{OT1}{ptm}{m}{it}{
<-> ptmri7t}{}

\DeclareFontShape{OT1}{ptm}{bx}{n}{
<-> ssub * ptm/b/n}{}
\DeclareFontShape{OT1}{ptm}{bx}{sc}{
<-> ssub * ptm/b/sc}{}
\DeclareFontShape{OT1}{ptm}{bx}{sl}{
<-> ssub * ptm/b/sl}{}

```

```
\DeclareFontShape{OT1}{ptm}{bx}{it}{
  <-> ssub * ptm/b/it}{
\DeclareFontShape{OT1}{ptm}{l}{n}{
  <-> ssub * ptm/m/n}{
\DeclareFontShape{OT1}{ptm}{l}{sc}{
  <-> ssub * ptm/m/sc}{
\DeclareFontShape{OT1}{ptm}{l}{sl}{
  <-> ssub * ptm/m/sl}{
\DeclareFontShape{OT1}{ptm}{l}{it}{
  <-> ssub * ptm/m/it}{
```

```
\DeclareFontShape{OT1}{ptm}{m}{ui}{
  <-> ssub * ptm/m/it}{
\DeclareFontShape{OT1}{ptm}{b}{ui}{
  <-> ssub * ptm/b/it}{
```

\endinput

The line

```
\DeclareFontFamily{OT1}{ptm}{}
```

specifies the font family to be defined by the file. This should match the filename, except for the case of the encoding.

The following line

```
\DeclareFontShape{OT1}{ptm}{b}{it}{
  <-> ptmbi7t}{}
```

associates a combination of encoding, family, weight/width and shape with a font on disk, i.e. with a .tfm file¹. Because this is a scalable font, no mention needs to be made of specific sizes.

For Computer Modern, this would look a good deal messier; a fontshape definition from *e.g.* ot1cmtd.fd looks like

```
\DeclareFontShape{OT1}{cmtt}{m}{n}
  {%
  <5><6><7><8>cmtt8<9>cmtt9%
  <10><10.95>cmtt10%
  <12><14.4><17.28><20.74><24.88>cmtt12%
  }{}
```

The fifth parameter assigns fonts to (sets or ranges of) font sizes.

Both `\DeclareFontFamily` and `\DeclareFontShape` have a final parameter in which ‘loading settings’ can be given but which is usually left empty.

You may have noticed a third type of declaration above:

```
\DeclareFontShape{OT1}{ptm}{l}{n}{
  <->ssub * ptm/m/n}{}
```

says that weight ‘light’ (l) is unavailable and that weight ‘medium’ (m) should be used instead. A similar declaration substitutes bold (b) for bold extended (bx).

Pound for pound

A weird one is

```
\DeclareFontShape{OT1}{ptm}{m}{ui}{
  <-> ssub * ptm/m/it}{}
```

which says that fontshape italic (it) must be substituted for ‘upright italic’(ui). Upright italic is similar to italic, but without the slant:

agi

Why bother with upright italic when nobody uses it anyway? Well, one might use it unintentionally: in OT1 encoding, dollars and pounds occupy the same slot, dollars coming from a ‘Roman’ font (upright or slanted), and pounds from an italic font, again, upright or slanted. This declaration should ensure that with Times in OT1 encoding upright pound symbols are rendered as italic pound symbols rather than upright dollars.

Mix and match

You may think that T_EX ought to be able to figure out by itself which fonts combine into which families by looking at font names – what good would the Berry naming scheme (see *e.g.* [5]) be otherwise.

Well, maybe it could, but it doesn’t. For the user, this means additional complexity, but also an opportunity to mix and match his own families. Here is the MAPS font family definition:

```
\DeclareFontFamily{OT1}{ptfs}{}
\DeclareFontShape{OT1}{ptfs}{b}{n}{
  <-> s * [0.95] pfrb7t}{
\DeclareFontShape{OT1}{ptfs}{b}{sc}{
  <-> pfrbc7t}{
\DeclareFontShape{OT1}{ptfs}{b}{it}{
  <-> s * [0.95] pfrbi7t}{
\DeclareFontShape{OT1}{ptfs}{m}{n}{
  <-> ptmr7d}{
\DeclareFontShape{OT1}{ptfs}{m}{sc}{
  <-> ptmrc9t}{
\DeclareFontShape{OT1}{ptfs}{m}{it}{
  <-> ptmri7d}{
\DeclareFontShape{OT1}{ptfs}{m}{sl}{
```

1. Actually, this is a virtual font, although that is of no concern to T_EX itself. When the dvi-driver has to render the font, it will first look for a corresponding virtual font, *i.e.* a .vfi file. This file will contain instructions to typeset the original characters, using characters from possibly different fonts at possibly different positions in the encoding vector.

```

<-> ssub * ptfs/m/it}{ }
\DeclareFontShape{OT1}{ptfs}{bx}{n}{
<-> ssub * ptfs/b/n}{ }
\DeclareFontShape{OT1}{ptfs}{bx}{it}{
<-> ssub * ptfs/b/it}{ }
\DeclareFontShape{OT1}{ptfs}{bx}{sl}{
<-> ssub * ptfs/bx/it}{ }
\DeclareFontShape{OT1}{ptfs}{b}{sl}{
<-> ssub * ptfs/b/it}{ }
\DeclareFontShape{OT1}{ptfs}{bx}{ui}{
<-> ssub * ptfs/bx/it}{ }

\DeclareFontShape{OT1}{ptfs}{b}{ui}{
<->sub * ptfs/b/it}{ }
\DeclareFontShape{OT1}{ptfs}{m}{ui}{
<->sub * ptfs/m/it}{ }

```

This family combines medium weights from Times with bold weights from Adobe Frutiger. Moreover, it scales the Frutiger fonts to 95% of their original size, to compensate for their larger x-height.

X-height refers to the height of lower-case letters without ascenders or descenders, such as ‘x’. Font sizes are measured including ascenders and descenders. A font with a large x-height looks larger than a font with a small x-height at the same point size. Without such a correction, Times and Frutiger would not harmonize very well.

Ad-hoc font changes

L^AT_EX has been designed for structured documents. Most of the time the class file will take care of font changes, and most of the time that is a good thing.

If you use L^AT_EX for everything else, then you may also want to use it for one-off projects such as invitations or flyers, even though L^AT_EX is not the tool of choice for such documents. Here one often wants to create a headline at a size or in a font that the L^AT_EX 2_ε class files don’t provide for.

One could use plain T_EX font commands such as

```

\font\headlinefont=bchb7t at 22pt
:
\headlinefont Headline

```

which produces

Headline

There are problems with this approach: because it bypasses the NFSS engine, implicit font changes are not handled properly, as the following example demonstrates:

```

{\headlinefont
Head \textit{italic} $\alpha$ tail}

```

produces

Head *italic* α tail

The new font is picked relative to the original font. Presumably, we wanted something like

Head *italic* α tail

We could have achieved this by defining `\headlinefont` as follows:

```

\newcommand{\headlinefont}{%
\fontfamily{bch}\bfseries
\fontsize{22pt}{22pt}\selectfont}

```

The current MAPS classfile still uses the old OT1 encoding. Using plain T_EX font commands in combination with T1 encoding can produce even nastier effects:

hei“g

instead of

heiig

This could happen because some characters, such as the dotless *i*, come from another font.

If you bemoan the increased bulk and complexity of L^AT_EX 2_ε, keep in mind that the innovations in it solved some real problems.

References

1. Michel Goossens, Frank Mittelbach and Alexander Samarin, *The L^AT_EX Companion*, Addison-Wesley, 1994.
2. Michel Goossens, Sebastian Rahtz and Frank Mittelbach, *The L^AT_EX Graphics Companion*, Addison-Wesley, 1997.
3. The L^AT_EX3 Project Team, *L^AT_EX 2_ε font selection*, CTAN/macros/latex/base/fntguide.tex. This is the definitive reference on L^AT_EX 2_ε font selection.
4. Leslie Lamport, *L^AT_EX: A Document Preparation System*, Second Edition, Addison-Wesley, 1994.
5. Maarten Gelderman, *How to install a Type 1 font using FONTINST*, MAPS 22 (this issue)
6. Tomas Rokicky, *DVIPS manual*, distributed with DVIPS.