Comments on "Filenames for Fonts" (*TUGboat* 11, no. 4)[*]

Frank Mittelbach

Abstract

In *TUGboat* 11, no. 4, a nomenclature for font files was proposed by Karl Berry. I disagree with Berry's proposal in some important points and would like to put these in writing in the hope that we will find some suitable agreement before anything is adopted as a standard.[1] I was aware of an ongoing email discussion about this topic but unfortunately didn't pay enough attention to realise that this would lead at this stage to a definite proposal.

Although this article points out several possibilities, it is not meant as a counter proposal. It is written in the hope that re-opening the discussion will lead to the best possible solution. In its current state, Berry's proposal cannot be used for LaTeX 3.0 (cf. sections 2 and 4) and this means that the majority of TeX users will be forced to use something different.[2] Thus, however consistent and rational it may be, his scheme can never become a universal standard.

1 Identifying font characteristics

Berry said that his proposal follows and simplifies the scheme we adopted for the new font selection scheme of LaTeX [11]. But in my opinion it makes the same mistake as we did in our first proposal for a new font selection scheme for TeX fonts [10]. The main idea behind identifying certain properties of fonts individually is the desire to change them independently. If, for example, a designer defines the layout of a heading to appear in 'bold extended' typeface, then a part of this heading that is to be emphasized should appear in a corresponding font, preferably in 'bold extended italic' or at least in 'bold italic'. This is possible if one identifies 'bold'

as the *weight*, 'extended' as the *width* and 'italic' as the *shape* or *variant* of the current font.

However, Berry identifies both 'sans serif' and 'typewriter' as *variants*, whereas we think that these are invariants of a font family and consequently should appear in the font family name. The reason for this decision is the fact that there is practically no font family which consists of both a 'serif' and a 'sans serif' variant or which contains an additional 'typewriter' variant. We do not view the Computer Modern Family as a counter example: it is a meta-family consisting of several independent font families which are only loosely connected by design principles. Otherwise one has to accept Concrete Roman as part of this family[3] and this seems a bit far-fetched.

2 The Computer Modern families

It is true that more and more fonts are becoming accessible through TeX and that it is therefore time to introduce a naming convention which allows them to be handled in a consistent manner. However, the main fonts in use are still public domain fonts generated by METAFONT. This is because, firstly, they cost nothing (or almost nothing) and, secondly, they ensure compatibility since most of them are included in the standard TeX distributions. For these reasons a scheme that does not cover these fonts is only of limited use. Berry never mentions how Computer Modern by Knuth [7] or Pandora by Billawala [2] could be integrated into his scheme.[4]

3 Font names of eight characters

As Berry correctly states, eight characters are definitely not enough to cover all possible font families with all their variations, at least not if verbose naming is used. However, if we encode the font names into arbitrary sequences of letters and

---

[*] The assistance of Chris Rowley is acknowledged with pleasure.

[1] The re-implementation of LaTeX will allow the user to access a broader range of fonts and it would be a big disadvantage if the method used implements a standard different from the one used in other macro packages.

[2] At the moment, only comparatively few users are in a position to actually use the new typefaces and most therefore have to rely on Computer Modern or a virtual variant thereof (implementing a standard TeX encoding).

[3] The Concrete Roman family is constructed by using the Computer Modern METAFONT sources and applying new parameter sets. In our notation the family 'concrete roman' consists of the *variants* 'normal', 'italic' and 'small caps', all in medium *width* and *weight*. Additionally a 'slanted' or 'sloped' *variant* in 'condensed' *width* exists. Examples of this typeface can be found in [11].

[4] While it might be possible to come up with some two-letter combinations for the typeface names and perhaps 't' (i.e. TeX distribution) for the foundry, there is no possibility in Berry's scheme of including virtual fonts that extend METAFONT fonts to 256-character codepages, cf. section 4.

numbers (beginning with a letter) then we can address $26 \times 36^7 = ?$ different fonts.[5] I suppose that nobody would like to remember Times-Roman as z5zcvp49, so perhaps a more readable encoding has to be found, but we should not choose a system that already has difficulties in covering the current range of available fonts.

However, depending on the addressing method used within TEX, the actual names of the files can be of minor importance since they need concern only format developers. For example, in the new font selection scheme for LATEX [11], the user will specify fonts by characteristics which consist, in principle, of arbitrarily long strings.[6] At the most primitive level, this user-interface consists of command sequences such as

```
\family{tim}\series{bc}%
\shape{sc}\selectfont
```

this loads a small caps variant of Times-Roman in weight/width bold condensed (in the current size). This might indeed correspond to some external file named z5zcvp49, without forcing the user to learn this fact. To use such a scheme successfully we have to ensure that there no longer exist situations where the user is forced to return to LATEX's \newfont command or TEX's \font primitive. In particular, this requires proper documentation[7] of the available fonts in the form of tables for \family, \series, etc., together with the ability to access fonts in arbitrary sizes since many fonts can be scaled nowadays by the output devices. This important feature will be added to the new font selection scheme in the near future; the implementation is currently under way.

The following sections deal with individual parts of Berry's proposal. They are nothing more than observations and do not add up to a new proposal for using the available number of characters.

----

[5] Similar considerations apply to the case of user names on length-restricted systems. While pzf5hz is perhaps difficult to connect with "Mittelbach", this approach allows over 60 000 EDS employees access to the company's net without conflicts.

[6] It should be pointed out that the new font selection scheme is independent of LATEX and could therefore serve as a new standard, just like the old one proposed by Knuth [9, pp. 414–415] at a time when only a few fonts were available and subsequently implemented in most macro packages.

[7] A task that has still to be undertaken for LATEX 3.0: any volunteers?

Denoting the foundry While it seems nice to have names that are easy to remember[8], I have some reservations about the foundry table given in [1, p. 517]. It might be possible to cover the major foundries of the western world, but the market is young and will certainly expand enormously in the near future.[9] The present list is nowhere near complete: where, for example, are Monotype and Linotype?

Denoting the typeface family The table of typeface families reads like a PostScript brochure. While at present this is certainly an important source of non-METAFONT fonts used with TEX, one should look closely at all the other families provided by the major printer companies to see whether or not they fit into any proposed scheme.[10]

Denoting the weight and expansion Given the constraints on available characters for use in the font names, I would suggest squeezing this information into one character. One can probably use 'memorable' characters for the most usual combinations and assign the remaining characters to all other combinations. To reduce the number of possible combinations one should drop the distinction between human and automatic scaling in expansion. While this is an interesting fact, I doubt whether any foundry supplies the same typeface in both ways.

Denoting the variant My only concerns here are those I expressed earlier (see section 1) about what constitutes a variant.

Denoting the size To avoid the problem of unspecifiable font sizes, I suggest the use of a two-digit hexadecimal (or even base-36) number. For standard sizes, i.e. those in the range 5pt to 20pt, this is as readable as a decimal number; and for the usual display sizes (e.g. 72pt) one would surely get used to it. This would also allow the packing of additional information into this part of the font name, as explained in the next section.

----

[8] Whatever this means when only a single character is to be used.

[9] Since it is already difficult to assign appropriate letters, one might think of dropping this approach completely to avoid giving certain foundries preference over others.

[10] There exist by now tools to generate .tfm files from down-loadable fonts for many font formats in different printer languages.

4 Font Encoding Schemes

Berry mentions the problem of virtual fonts. It is in principle possible to generate arbitrary fonts by combining characters from different typefaces into one virtual font. While this method allows the creation of an unlimited number of fonts and could certainly blow up any scheme, it seems questionable whether this will actually happen. A natural usage of this concept would be to add certain missing characters or symbols to a font so that it can be used with a standard macro package. In such a case, however, the resulting virtual font would still be clearly identifiable by its major raw font. On the other hand, virtual fonts could completely dispose of the problem of font encoding, provided that the TEX community can agree on a few standard layouts for 'latin' (cf. [3]), 'math', etc.

The use of 'r' for raw .tfm files, as pointed out by Berry, works only for fonts which have no design size and this again rules out any font produced with METAFONT, since virtual fonts for such families (following the coding scheme as proposed in [3])[11] cannot be specified within Berry's naming conventions. I therefore suggest coding this information within the design size, by adding a suitable number to the actual design size to indicate that a raw .tfm file is to be used.[12] If the design size were coded in hexadecimal notation, this would allow design sizes up to 127pt ("00–"7F) for the (virtual) fonts which are actually used (and which have a standard TEX encoding scheme), the accompanying range ("80–"FF) being left for raw .tfm files.

---

[11] To my knowledge, this work (for the Computer Modern families) has already been undertaken in Germany and is at the moment in β-testing.

[12] The use of unusual font sizes for the raw .tfm seems appropriate since these font metric files are of interest only to those who have to set up virtual fonts.

References

[1] Karl Berry. Filenames for fonts. *TUGboat*, 11(4):517–520, 1990.

[2] Nazneen N. Billawala. Metamarks: Preliminary studies for a Pandora's Box of shapes. Technical Report STAN-CS-89-1256, Stanford University, Department of Computer Science, Stanford, California 94305, 1989.

[3] Michael J. Ferguson. Report on multilingual activities. *TUGboat*, 11(4):514–516, 1990.

[4] Donald E. Knuth. Computers & Typesetting. Addison-Wesley, Reading, Massachusetts, 1986. Consists of [9, 8, 6, 5, 7].

[5] Donald E. Knuth. *METAFONT: The Program.* Volume D of *Computers & Typesetting* [4], 1986.

[6] Donald E. Knuth. *The METAFONTbook.* Volume C of *Computers & Typesetting* [4], 1986.

[7] Donald E. Knuth. *Computer Modern Typefaces.* Volume E of *Computers & Typesetting* [4], July 1987. Reprint with corrections.

[8] Donald E. Knuth. *TEX: The Program.* Volume B of *Computers & Typesetting* [4], May 1988. Reprint with corrections.

[9] Donald E. Knuth. *The TEXbook.* Volume A of *Computers & Typesetting* [4], May 1989. Eighth printing.

[10] Frank Mittelbach and Rainer Schöpf. A new font selection scheme for TEX macro packages— the basic macros. *TUGboat*, 10(2):222–238, July 1989.

[11] Frank Mittelbach and Rainer Schöpf. Reprint (with corrections): The new font family selection — user interface to standard LATEX. *TUGboat*, 11(2):297–305, June 1990.

◇ Frank Mittelbach
Electronic Data Systems
(Deutschland) GmbH
Eisenstraße 56 (N 15), D-6090
Rüsselsheim, Federal Republic
of Germany
Tel. +49 6142 803267
mittelbach@mzdmza.zdv.uni-mainz.de