Fonts

Filenames for Fonts

Karl Berry

As more typeface families become available for use with TeX, the need for a consistent, rational naming scheme for the font filenames concomitantly grows. Some (electronic) discussion has gone into the following proposal; I felt it was appropriate now to bring it before a wider community. In some respects, it follows and simplifies Mittelbach's and Schöpf's article in *TUGboat*, volume 11, number 2 (June 1990).

Here are some facts about fonts that went into the hopper when creating this proposal:

- TeX runs on virtually all computers, under almost as many operating systems, all with their own idea of how files should be named. Any proposal regarding filenames, therefore, must cater to the lowest common denominator. That seems to be eight characters in length, not counting any extension, and with case being insignificant. Characters other than letters and numerals are probably unusable.
- Most typefaces are offered by several vendors. The version offered by vendor A is not compatible with that of vendor B.
- Typefaces typically come in different weights (hairline to extra heavy), different expansions (ultra condensed to wide), and an open-ended range of variants (italic, sans serif, typewriter, shadow, ...). No accepted standards exist for any of these qualities, nor are any standards ever likely to gain acceptance.[1]
- The Computer Modern typeface family preserves traditional typesetting practice in at least one important respect: different sizes of the same font are not scaled linearly. This is in contrast to most commercial fonts available.

---

[1] Editor's note: A draft international standard, ISO/IEC DIS 9541, Font Information Interchange, attempts to define these qualities, among others, in a manner acceptable to font suppliers and usable by a wide variety of typesetting software, including (indirectly) TeX. The Editor has been a participant in this working group for several years, and is trying to make sure that the needs of TeX users are heard.

Here is how I propose to divide up the eight characters:

FTTWVEDD

where

- F represents the foundry that produced the font, and is omitted if there isn't one.
- TT represents the typeface name.
- W represents the weight.
- V represents the variant, and is omitted if both it and the expansion are "normal".
- E represents the expansion, and is omitted if it is "normal".
- DD represents the design size, and is omitted if the font is linearly scaled from a single tfm file.

See the section on virtual fonts (towards the end) for an exception to the above.

The weight, variant, and expansion are probably all best taken from the original source of the typeface, instead of trying to relate them to some external standard.

Before giving the lists of abbreviations, let me point out two problems, to neither of which I have a good solution. 1) Assuming that only the English letters are used, two letters is enough for only 676 typeface families (even assuming we want to use all possible combinations, which is doubtful). There are many more than 676 typeface families in the world. 2) Fonts with design sizes over 100 pt are not common, but neither are they unheard of.

On to the specifics of the lists. If you adopt this proposal at your own installation, and find that you have fonts with some property I missed, please write to me (see the end of the article for various addresses), so I can update the lists. You can get the most up-to-date version of these lists electronically, by anonymous ftp from the host ftp.cs.umb.edu. I will also send them to you by electronic mail, if necessary.

I give the letters in lowercase, which is preferred on systems where case is significant. The lists are in alphabetical order by the abbreviations.

Foundry

a	Autologic
b	Bitstream
c	Compugraphic
g	Free Software Foundation (g for GNU)
h	Bigelow & Holmes (with apologies to Chuck)
i	International Typeface Corporation
p	Adobe (p for PostScript)

r   reserved for use with virtual fonts; see below

s   Sun

Typeface families

ad	Adobe Garamond
ag	Avant Garde
ao	Antique Olive
at	American Typewriter
bb	Bembo
bd	Bodoni
bg	Benguiat
bk	Bookman
bl	Balloon
bv	Baskerville
bw	Broadway
cb	Cooper Black
cl	Cloister
cr	Courier
cn	Century
cs	Century Schoolbook
hv	Helvetica
gm	Garamond
go	Goudy Oldstyle
gs	Gill Sans
jo	Joanna
lc	Lucida
lt	Lutetia
nc	New Century Schoolbook
op	Optima
pl	Palatino
pp	Perpetua
rw	Rockwell
st	Stone
sy	Symbol
tm	Times
un	Univers
uy	University
zc	Zapf Chancery
zd	Zapf Dingbats

Weight

a	hairline
b	bold
c	black
d	demi
h	heavy
i	extra light
k	book
l	light
m	medium
r	regular
s	semi
t	thin
u	ultra
x	extra bold

In order of lightest to heaviest (more or less):

hairline	demibold
thin	semi
extra light	bold
light	extra bold
book	heavy
regular	black
medium	ultra

Variant

b	bright
c	small caps
e	engraved
g	grooved (as in the IBM logo)
h	shadow
i	(text) italic
l	outline
n	informal
o	oblique (i.e., slanted)
r	normal (roman or sans)
s	sans serif
t	typewriter
u	unslanted italic

If the variant is r, and the expansion is also normal, both the variant and the expansion are omitted. When the normal version of the typeface is sans serif (e.g., Helvetica), r should be used, not s. Use s only when the typeface family has both serif and sans serif variants.

Expansion

c	condensed (by hand)
e	expanded (automatic)
n	narrow (automatic)
o	extra condensed
	regular, normal, medium (always omitted)
w	wide
x	extended (by hand)

In order of narrowest to widest (more or less):

extra condensed	extended
condensed	expanded
narrow	wide
regular	

Expansion of fonts is sometimes done automatically (as in PostScript **scale**), and sometimes done by humans. I chose 'narrow' and 'expanded' to imply the former, and 'condensed' and 'extended' to imply the latter, as I believe this reflects common usage.

Virtual fonts

In concert with releasing TeX 3.0 and META-FONT 2.7, Don Knuth wrote two new utility programs: VFtoVP and VPtoVF, which convert to and from "virtual" fonts. Virtual fonts provide a general interface between the writers of TeX macros and font suppliers. In general, therefore, it is impossible to come up with a general scheme for naming virtual fonts, since each virtual font is an individual creation, possibly bringing together many unrelated fonts.

Nevertheless, one common case is to use virtual fonts to map TeX's default accent and other character code conventions onto a vendor-supplied font. For example, dvips (by Tom Rokicki) does this for fonts given in the PostScript "standard encoding". In this case, each font consists of a "virtual" tfm file, which is what TeX uses, a "raw" tfm file, which corresponds to the actual device font, and a vf file, which describes the relationship between the two.

This adds another dimension to the space of font names, namely, "virtualness" (or rather, "rawness", since it is the virtual tfm files that the users want to see). But we have already used up all eight characters in the font names.

The best solution I have been able to think of is this: prepend r to the raw tfm files; the virtual tfm files should be named with the usual foundry prefix. For example, the virtual Times Roman tfm file is named ptmr, as usual; the raw Times Roman tfm file is named rptmr. To prevent intolerable confusion, I promise never to give a foundry the letter r.

This scheme will work only as long as the virtualized fonts do not have design sizes; if they do, another foundry letter will have to be allocated, it seems to me.

A pox upon the houses of those who decided on fixed-length filenames!

Examples

In closing, I will give two examples. First, the fonts in the Univers typeface family were assigned numbers by its designer, Adrien Frutiger. (You can see the scheme on, for example, page 29 of *The Art of Typo.icon.ography*, by Martin Solomon.) Naturally, we want to give them names.

45 (light): unl
46 (light italic): unli
47 (light condensed): unlrc
48 (light condensed italic): unlic
49 (light extra condensed): unlro
53 (medium extended): unmrx
55 (medium): unm
56 (medium italic): unmi
57 (medium condensed): unmrc
58 (medium condensed italic): unmic
59 (medium extra condensed): unmro
63 (demibold extended): undrx
65 (demibold): und
66 (demibold italic): undi
67 (demibold condensed): undrc
68 (demibold condensed italic): undic
73 (bold extended): unbrx
75 (bold): unb
76 (bold italic): unbi
83 (extra bold extended): unxrx

Second, here are names for the 35 standard PostScript fonts:

AvantGarde-Book: pagk
AvantGarde-BookOblique: pagko
AvantGarde-Demi: pagd
AvantGarde-DemiOblique: pagdo
Bookman-Demi: pbkd
Bookman-DemiItalic: pbkdi
Bookman-Light: pbkl
Bookman-LightItalic: pbkli
Courier-Bold: pcrb
Courier-BoldOblique: pcrbo
Courier: pcrr
Courier-Oblique: pcrro
Helvetica-Bold: phvb
Helvetica-BoldOblique: phvbo
Helvetica-NarrowBold: phvbrn
Helvetica-NarrowBoldOblique: phvbon
Helvetica: phvr
Helvetica-Oblique: phvro
Helvetica-Narrow: phvrrn
Helvetica-NarrowOblique: phvron
NewCenturySchlbk-Bold: pncb
NewCenturySchlbk-BoldItalic: pncbi

NewCenturySchlbk-Italic: pncri
NewCenturySchlbk-Roman: pncr
Palatino-Bold: pplb
Palatino-BoldItalic: pplbi
Palatino-Italic: pplri
Palatino-Roman: pplr
Symbol: psyr
Times-Bold: ptmb
Times-BoldItalic: ptmbi
Times-Italic: ptmri
Times-Roman: ptmr
ZapfChancery-MediumItalic: pzcmi
ZapfDingbats: pzdr

Please contact me if you have any comments or additions.

◇ Karl Berry
135 Center Hill Rd.
Plymouth, MA 02360
karl@cs.umb.edu

Arabic, Persian and Ottoman TEX for Mac and PC

Yannis Haralambous

¡El peor con TEX,
es la gente
que lo gusta!
— Anonymous

The whole thing started when my friend Siavash Mirshams Shahshahani from the Sharif University of Technology, Tehran, asked me to find some arabic-alphabet TEX for the PCs of his department. I looked around and found out that there is none, at least not in the public domain. Unfortunately, not even TEX-XET, the right–to–left version of TEX is available for Mac or PC. So I decided to fill the gap. For this I had to create the necessary font yarb — which is suitable for arabic, persian and ottoman as well — and a Pascal preprocessor yarbtex. This work has been done on a Mac SE/30 with OzTEX, MacMETAFONT, and Think Pascal v3.0.

The arabic alphabet

The common arabic alphabet has the following 28 letters:

ا ب ت ث ج ح خ د ذ

ر ز س ش ص ض ط ظ ع غ

ف ق ك ل م ن ه و ى

To cover also persian and pre-1932 turkish, one has to consider the following 5

پ چ ژ گ ڭ

Finally ڤ is used for the arabic transliteration of the sound "v", as in "Vienna": ڤینا.

Let's see now how letters are combined to form words; with the letters ت, ك and ع we can form the word تكع (right to left!). As you can see, letters are written in a different way according to their position inside the word. Consequently, arabic letters appear in four forms: *initial* (like ت for ت), *middle* (like ك for ك), *final* (like ع for ع) and *isolated.* The letters ا, د, ذ, ر, ز, ژ, و have only final and isolated forms. So when they occur inside a word, the next letter is initial (as in ولد).

As in german, there are short and long vowels. The long ones are ا, و and ى. The short ones are not represented at all, except in classical or educative texts, where they are represented by accents: ˘ (fatha), ˍ (kasra), ˊ (damma), respectively for the sounds of a, i and u. In this case, the lack of vowel is represented by ° (tašdid). Also there are other accents (˜, ˵, ˏ, ̗) for special purposes.

The preprocessor yarbtex

While typing the input, it would be too tedious to think in which form each letter should be written, and since this problem is much too complicated to be solved only by ligatures inherent to the font, it was necessary to make a preprocessor which converts an xxx.arb input file, into a xxx.tex TEX-file. The usual procedure

$$.\texttt{tex} \xrightarrow{\text{TEX}} .\texttt{dvi} \xrightarrow{\text{dvitops}} .\texttt{ps}$$

is now one step longer

$$.\texttt{arb} \xrightarrow{\text{yarbtex}} .\texttt{tex} \xrightarrow{\text{TEX}} .\texttt{dvi} \xrightarrow{\text{dvitops}} .\texttt{ps}.$$

Let's take a look now at an xxx.arb file: to typeset in arabic, persian and/or ottoman (in short APO) you have to enter into "arabic mode" by typing a vertical bar |. The | will change to arabic mode regardless of whether you are in text or math mode, so you should use $\vert$ to obtain | in non-arabic mode.

Once you are in arabic mode, you can choose between the following submodes:

a. *text and arabic numerals* (the numerals used in arabic typography),