Are virtual fonts obsolete?

Boris Veytsman

> [Integrating third-party fonts] is unfortunately a messy topic. Forget about it unless you want to delve into many details of the TEX installation.
>
> TEX Live Manual [1]

Abstract

Virtual fonts (VF) were created to address a shortcoming of TEX fonts: each slot address occupied exactly one byte, so there were no more than 256 different characters per font. Later, when PostScript fonts got popular, VF became the way of choice for integration of these fonts with TEX. Today new font formats can be directly read by the modern TEX engines, and, for example, XETEX can directly work with system fonts. There is a temptation to declare VF obsolete.

In this paper we show that there is much more in VF than just making PostScript fonts available for TEX. There are various tricks developed over the years that use VF technology to achieve new striking effects.

The aim of this paper is to convince the users to learn how to employ VF, and to convince the programmers of the new engines to provide the interface for font manipulation comparable to VF.

1 Introduction

Several years ago I attended a presentation of XETEX by Jonathan Kew. He typed in the editor

```
\font\x="Adobe Garamond Pro" at 11pt
\x This is Adobe Garamond Pro
```

and then clicked "Compile". The preview window showed the phrase — in the beautiful Adobe Garamond Pro! I remember coming to Jonathan after the presentation and telling him, "You just closed one of the sources of my income!" I knew that for decades the use of third party fonts in TEX was one of the most difficult endeavors. It had a certain positive value for a consultant like me — and now this was going to change.

I was wrong. Actually in the years after that meeting I had a number of virtual font projects sponsored by various customers. Also, I learned a lot about fonts in TEX and came to the understanding that there is much more there than just typesetting the text in this or that third-party font.

In this paper I try to discuss some tricks possible with the traditional VF and compare them to the mechanism provided by XETEX.

2 A bit of history

In its early days TEX was practically synonymous with Computer Modern fonts. If you met a TEX document, you could bet it was typeset in Computer Modern. On the other hand, if you saw Computer Modern, you knew this text was typeset by TEX. Many people wanted to use the large number of free and commercial fonts available. However, there were two main obstacles to the use of these fonts with TEX: first, they were not in the METAFONT format, and second, they were not rich enough. TEX expected to find a lot of characters used for mathematical typesetting etc., and many free or commercial fonts lacked them — or did not have them in the proper places.

There were some hacks floating around — until Knuth proposed a unified interface of virtual fonts in TEX [7]. This interface used the fact that TEX itself is quite agnostic about the way the fonts are internally presented. What is needed is *metric information*: the dimensions of the letters, ligatures and kerning. The job of putting the letters themselves on paper or screen is done by *drivers* like `dvips`. The idea of virtual fonts is that the developer should create a metric file and a virtual font description with the instructions for the driver. These instructions could be quite complex, like "Take the letter A from this position in the file `abca`, the letter B from that position in the file `xyzz`, ...". They also could contain transformations of the letters: expansion, contraction, slanting, kerning and ligature changes.

In this way one can overcome both obstacles for using "foreign" fonts with TEX. The font files could be in any format as long as the drivers recognized them. Also, if a font lacked certain characters, they could be taken from another font, maybe created with the explicit purpose to extend the original font for TEX.

The creation of virtual font files involved some repetitious steps. Thus there were several attempts to automate them. The most successful of them is probably the widely used program fontinst [5]. There is a great guide for this program by Philipp Lehman [8] and a very detailed discussion of many related topics including mathematical typesetting in the book by Alan Hoenig [4]. Still, even with these resources the setup of virtual fonts remains one of the most complex tasks for an apprentice TEXnician. Knuth gave his paper [7] a very apt title: virtual fonts indeed provide more fun to grand wizards.

In the next section we explore some tricks with virtual fonts.

3 Some virtual font tricks

As discussed in the previous section, the most direct application of virtual fonts is adding mathematical symbols to the free and commercial fonts that lack them. There are several good reviews of the many fonts created in this way. They include the survey of free fonts by Hartke [3], the "rogues gallery" in Hoenig's book [4], the tables in the LaTeX Companion [9], etc.

A good example of virtual fonts is the `mathptmx` package from PSNFSS [13]. It uses Times and Symbol PostScript fonts for mathematical typesetting (Figure 1).

Journal d'Analyse Mathematique uses Times, but with a slight modification: the letters are expanded in the horizontal direction. Math is provided by Belleek fonts [6]. Both the expansion and addition of math symbols are done with virtual fonts (Figure 2).

This example shows that VF can be used not only for mathematical typesetting, but also for text effects. Let us expand on this topic.

Eric Gill thought that italics should be used with lower case only, and the capitals should be taken from a matching Roman font [2]. To check the appearance of such combinations it is easy to "gillize" common fonts using the VF technique. The package `gillcm` [16] was created to demonstrate the setup of VF using this task. In Figure 3 we combine Roman uppercase letters with unslanted lower case italics.

Many examples of VF tricks can be found in the book [4]. In Figure 4 the Mantinia font is shown. Its unusual ligatures and swashes give the font a high decorative value. Of course the original font had all these characters, but it lacked ligature rules: evidently the designer had in mind manual typesetting only. Alan Hoenig added the ligatures through VF.

While this version of the font is nice to look at, its readability may suffer. Therefore Alan created another, more "subdued" version of this font usable for titling (Figure 5).

We conclude this section with the example of the use of Unicode fonts. The situation with these fonts is in a sense opposite to the situation with non-TeX fonts described in Section 2. They have *many* glyphs in one font — while TeX expects them to be in the different text and mathematical ones. However, VF can help in this case too. We can extract ranges of characters from a Unicode font and assign them to different TeX virtual fonts.

An example of this use of VF is `mathgifg` package [14] for use of Georgia font in text and math with TeX. Georgia is a nice Unicode font from Microsoft distributed with more or less recent Windows installations. It has Greek characters and mathematical symbols. The US Army Corps of Engineers sponsored the creation of TeX support for typesetting text and mathematics with it. The package [14] first separates the font into TeX virtual fonts, and then combines them for typesetting. Note that the Georgia font has "old style" (lowercase) numerals. While some designers used lowercase numerals in math in the past, it is probably too disturbing for a modern eye, so the package uses Franklin Gothic numerals (from another font distributed by Microsoft) in math and Georgia numerals in text. The results are shown in Figure 6.

4 Font setup with XeTeX

XeTeX at present has the full support of virtual fonts, as with other TeX engines. However, in this section we discuss the features specific to this engine which provide another way of dealing with fonts. Namely, XeTeX uses system libraries for font handling, and thus can "see" and directly use all fonts available in the current computer.

One of the striking things about XeTeX is the great ease with which third party text fonts are included in TeX documents. In this section we will discuss the LaTeX variant of fonts support based on fontspec [12] package. However, the corresponding variants for other flavors of TeX are relatively straightforward to set up. Actually the example in Section 1 was written with plain TeX commands.

With LaTeX and fontspec one can define *ad hoc* font selection schemes "on the fly". Consider the following invocation:

```
\fontspec[BoldFont={Helvetica Neue}]%
   {Helvetica Neue Ultralight}
```

As the result of this command *Helvetica Neue Ultralight* becomes the new Roman family. Moreover, *Helvetica Neue* becomes its bold variant, so macros like `\bfseries` and `\textbf` switch to this font.

The setup of mathematics fonts depends on the features of these fonts. If the OpenType font supports mathematics typesetting, then the experimental package unicode-math [11] can be used with the following easy interface:

```
\setmathfont[math-style=TeX]{Cambria Math}
```

However, if no mathematics support is provided by the font designer, then the situation becomes more complicated. The mathspec package [10] is intended to set up such fonts for typesetting mathematics in TeX. It has commands for font selection like `\setmathsfont`, `\setmathrm`, `\setmathsf`, etc.

Theorem 1 (Residue Theorem). Let f be analytic in the region G except for the isolated singularities $a_1, a_2, \ldots, a_m$. If γ is a closed rectifiable curve in G which does not pass through any of the points a_k and if $\gamma \approx 0$ in G then

$$\frac{1}{2\pi i}\int_\gamma f = \sum_{k=1}^{m} n(\gamma; a_k)\operatorname{Res}(f; a_k).$$

Theorem 2 (Maximum Modulus). *Let G be a bounded open set in* $\mathbb{C}$ *and suppose that* f *is a continuous function on* G^- *which is analytic in G. Then*

$$\max\{|f(z)| : z \in G^-\} = \max\{|f(z)| : z \in \partial G\}.$$

ΑΛΔ∇BCDΣEFΓGHIJKLMNOΘΩ℧PΦΠΞQRSTUVWXYϒΨZ 1234567890
aαbβc∂dδeεεfζξgγhħℏıιiȷjkκϰlℓλmnηθϑoσςϕφ℘pρϱqrstτπuμνvυwωϖxχyψz ∞ ∝ ∅⌀dð ϶

Figure 1: Times text with Symbol math (`mathptmx` package [13]), from [3]

Theorem 1 (Residue Theorem). Let f be analytic in the region G except for the isolated singularities $a_1, a_2, \ldots, a_m$. If γ is a closed rectifiable curve in G which does not pass through any of the points a_k and if $\gamma \approx 0$ in G then

$$\frac{1}{2\pi i}\int_\gamma f = \sum_{k=1}^{m} n(\gamma; a_k)\operatorname{Res}(f; a_k).$$

Theorem 2 (Maximum Modulus). *Let G be a bounded open set in* $\mathbb{C}$ *and suppose that* f *is a continuous function on* G^- *which is analytic in G. Then*

$$\max\{|f(z)| : z \in G^-\} = \max\{|f(z)| : z \in \partial G\}.$$

ΑΛΔ∇BCDΣEFΓGHIJKLMNOΘΩ℧PΦΠΞQRSTUVWXYϒΨZ 1234567890
aαbβc∂dδeϵεfζξgγhħℏıiȷjkκϰlℓλmnηθϑoσςϕφ℘pρϱqrstτπuμνυυwωϖxχyψz ∞ ∝ ∅⌀dð ϶

Figure 2: Times expanded text with Belleek math (`jamtimes` package [15])

Properly speaking, there is no such thing as an alphabet of italic capitals, and where upright or nearly upright italics are used ordinary upright Roman capitals go perfectly well with them. ¶Eric Gill, "An Essay on Typography", 1931.

Properly speaking, there is no such thing as an alphabet of italic capitals, and where upright or nearly upright italics are used ordinary upright Roman capitals go perfectly well with them. ¶Eric Gill, "An Essay on Typography", 1931.

Figure 3: Comparison of Computer Modern Italics with "Gillized" Computer Modern Italics: unslanted italic lowercase and Roman uppercase (package `gillcm` [16])

QUONIAM AD HUNC LOCUM
PERVENTUM EST, NON ALIENUM ESSE
VIDETUR DE GALLIÆ GERMANIÆQUE
MORIBUS & QUO DIFFERANT HÆ
NATIONES INTER SESE PROPONERE IN
GALLIA NON SOLUM IN OMNIBUS
CIVITATIBUS ATQUE IN OMNIBUS
PAGIS PARTIBUSQUE, SED PÆNE
ETIAM IN SINGULIS DOMIBUS
FACTIONES SUNT, EARUMQUE
FACTIONUM PRINCIPES SUNT QUI
SUMMAM AUCTORITATEM EORUM
IUDICIO HABERE EXISTIMANTUR,
QUORUM AD ARBITRIUM
IUDICIUMQUE SUMMA OMNIUM
RERUM CONSILIORUMQUE REDEAT.

Figure 4: Mantinia font with swashes and ligatures (from [4])

IV

THE CRUELTY, FOLLIES, AND MURDER OF COMMODUS · ELECTION OF PERTINAX · HIS ATTEMPTS TO REFORM THE STATE · HIS ASSASSINATION BY PRÆTORIAN GUARDS · INDIGNATION

The mildness of Marcus, which the rigid discipline of the Stoics was unable to eradicate, formed, at the same time, the most amiable and the only defective, part of his character. His excellent understanding was often deceived by the unsuspecting goodness of his heart. Artful men, who study the passions of princes and conceal their own, approached his person in the disguise of philosophic sanctity, and acquired riches and honours by affecting to despise them.[1] His excessive indulgence to his brother, his wife, and his son, exceeded the bounds of private virtue, and became a public injury, by the example and consequences of their vices.

Faustina, the daughter of Pius and the wife of Marcus, has been as much celebrated for her gallantries as for her beauty. The grave simplicity of the philosopher was ill calculated to engage her wanton levity, or to fix that unbounded passion for variety which often discovered personal merit in the meanest of mankind. The Cupid of the ancients was, in general, a very sensual deity; and the amours of an empress, as they exact on her side the plainest of advances, are seldom susceptible of much sentimental delicacy. Marcus was the only man in the empire who seemed ignorant or insensible of the irregularities of Faustina; which, according to the prejudices of every age, reflected some disgrace on the injured husband. He promoted several of her lovers to posts of honour and profit, and, during a connexion of thirty years, invariably gave her proofs of the most tender confidence, and of a respect which ended not with her life. In his Meditations he thanks the gods, who had bestowed on him a wife so faithful, so gentle, and of such a wonderful simplicity of manners.[2] The obsequious senate, at this earnest request, declared her a goddess.

[1] *See the complaints of Avidius Cassius. These are, it is true, the complaints of faction; but even faction exaggerates rather than invents.*

[2] *The world has laughed at the credulity of Marcus; but Madame Dacier assures us (and we may credit a lady) that the husband will always be deceived, if the the wife condescends to dissemble.*

Figure 5: Mantinia font (titling) and Galliard font (body text). From [4]

Theorem 1 (Residue Theorem). Let f be analytic in the region G except for the isolated singularities $a_1, a_2, \ldots, a_m$. If γ is a closed rectifiable curve in G which does not pass through any of the points a_k and if $\gamma \approx 0$ in G then

$$\frac{1}{2\pi i}\int_\gamma f = \sum_{k=1}^{m} n(\gamma; a_k)\mathrm{Res}(f; a_k).$$

Theorem 2 (Maximum Modulus). *Let G be a bounded open set in $\mathbb{C}$ and suppose that f is a continuous function on G^- which is analytic in G. Then*

$$\max\{|f(z)| : z \in G^-\} = \max\{|f(z)| : z \in \partial G\}.$$

ΑΛΔ∇BCDΣEFΓGHIJKLMNOΘΩ℧PΦΠΞQRSTUVWXYϒΨZ 1234567890
aαbβc∂dδeϵεfζξgγhħℏıȷjkκϰlℓλmnηθϑoσςφϕ℘pρϱqrstτπυμνυvwωϖxχyψz ∞ ∝ ∅⌀dð ϶
Text numerals: 123567890
Math numerals: 123567890

Figure 6: Georgia and Franklin Gothic fonts (`mathgifg` package [14])

One can use these commands to tell TeX to use a certain font for one of TeX's mathematical alphabets, for example

```
\setmathsfont(Digits)%
   [Numbers={Lining,Proportional}]%
   {Minion Pro}
\setmathsfont(Latin)%
   [Numbers={Lining,Proportional}]%
   {Minion Pro}
\setmathsfont(Greek)%
   [Numbers={Lining,Proportional}]%
   {Minion Pro}
```

The fontspec package allows for selecting font variants and alternates if these are provided by the font designer, for example

```
\fontspec[Variant=1]{Zapfino} Zapfino 1
\fontspec[Variant=2]{Zapfino} Zapfino 2
...
```

The same is true for variants in ligatures and kerning, for example

```
\fontspec[Ligatures=Rare]%
  {Adobe Garamond Pro} ...
\fontspec[Ligatures=NoCommon]%
  {Adobe Garamond Pro} ...
```

This very short discussion of the possibilities of XeTeX shows that it can help you to get from the font everything the font designer put there. Since OpenType format is very rich in features, this may be quite a lot indeed.

However, this approach has a flip side: it allows you to get from the font *only* what the font designer put there. Thus it assumes an ideal world of wise and TeX-aware font designers. If a font designer does not envision TeX-like automatic typesetting and, for example, assumes manual selection of characters for ligatures and manual kerning adjustments, you are out of luck.

In the past most font designers were of the second category (see the description of font adjustments in [4]). They often did not care about anything but either the crude typesetting without ligatures or the manual work with the so called *expert fonts*. It is true that OpenType provides many features for automatic typesetting; whether the font designers are going to use them is quite a different question.

5 Conclusions

The technique of virtual fonts, initially developed to allow the inclusion of third party fonts in TeX, became quite versatile in many different tasks. They can be used to achieve various effects in text and math. Unfortunately, they are known for their steep learning curve.

XeTeX provides an alternative way to incorporate third party fonts in TeX. This way allows using the full potential of such fonts — which can be impressive for feature-rich OpenType fonts. However, the "old" technique of virtual fonts allows low level font manipulation, including mixing different fonts. This manipulation is useful for adding to the font new features, not foreseen by the font creators.

References

[1] Karl Berry, ed. *The TeX Live Guide*. TUG, July 2010. `http://tug.org/texlive`.

[2] Eric Gill. *An Essay on Typography. With a new introduction by Cristopher Skelton*. David R. Godine, Boston, 2007.

[3] Stephen G. Hartke. *A Survey of Free Math Fonts for TEX and LATEX*, 2006. http://mirror.ctan.org/info/Free_Math_Font_Survey.

[4] Alan Hoenig. *TEX Unbound: LATEX and TEX Strategies for Fonts, Graphics, and More*. Oxford University Press, USA, 1998.

[5] Alan Jeffrey, Rowland McDonnell, and Lars Hellström. *fontinst: Font installation software for TEX*, December 2004. http://mirror.ctan.org/fonts/utilities/fontinst.

[6] Richard Kinch. *Belleek: Free replacement for basic MathTime fonts*, August 1998. http://mirror.ctan.org/fonts/belleek/.

[7] Donald Knuth. Virtual fonts: More fun for grand wizards. *TUGboat*, 11(1):13–23, 1990. http://www.tug.org/TUGboat/Articles/tb11-1/tb27knut.pdf.

[8] Philipp Lehman. *The Font Installation Guide*, December 2004. http://mirror.ctan.org/info/Type1fonts/fontinstallationguide.

[9] Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley. *The LATEX Companion*. Addison-Wesley Series on Tools and Techniques for Computer Typesetting. Addison-Wesley Professional, Boston, 2nd edition, 2004.

[10] Andrew Gilbert Moschou. *The mathspec package*, September 2009. http://mirror.ctan.org/macros/xetex/latex/mathspec/.

[11] Will Robertson. *Experimental Unicode mathematical typesetting: The unicode-math package*, June 2010. http://mirror.ctan.org/macros/latex/contrib/unicode-math/.

[12] Will Robertson and Khaled Hosny. *The fontspec package*, June 2010. http://mirror.ctan.org/macros/xetex/latex/fontspec/.

[13] Walter Schmidt. *Using Common PostScript Fonts with LATEX. PSNFSS Version 9.2*, September 2004. http://mirror.ctan.org/macros/latex/required/psnfss.

[14] Boris Veytsman. *LATEX Support for Microsoft Georgia and ITC Franklin Gothic in Text and Math*, July 2009. http://mirror.ctan.org/fonts/mathgifg/.

[15] Boris Veytsman. *Expanded Times Roman Fonts As Used in Journal d'Analyse Mathematique*, July 2010. http://mirror.ctan.org/fonts/jamtimes/.

[16] Boris Veytsman. *Unslanted Italic Computer Modern Fonts Based on Eric Gill's Ideas*, July 2010. http://mirror.ctan.org/fonts/gillcm/.

◇ Boris Veytsman
Computational Materials Science Center, MS 6A2, George Mason University, Fairfax, VA 22030
borisv (at) lk dot net
http://borisv.lk.net