Exploiting Rich Fonts

Sivan Toledo

1 Introduction

Rich fonts with hundreds or thousands of glyphs are becoming widely available. Such fonts are now bundled with operating systems, printers, and consumer software. Rich fonts enable us to achieve better typographic results than possible with conventional fonts. This paper explores three kinds of rich fonts, their typographic features, and ways to exploit these features in TEX systems.

The three kinds of fonts that the paper explores are:

1. PostScript 3 fonts. Adobe defined a core set of fonts to be included in all PostScript 3 devices [1]. Two families in the core set, Hoefler Text and Apple Chancery, are particularly interesting, since their fonts contain hundreds of glyphs. This article focuses on one family, Hoefler Text, a general-purpose text family. The glyph contents of Apple Chancery is fairly similar to that of Adobe Poetica, for which TEX support has already been developed [7, 5].
2. WGL4 Fonts. WGL4 (Windows Glyph List 4) is a glyph list published by Microsoft and intended to support all European languages, including Greek and Russian. Many fonts bundled with Microsoft's Office products are WGL4 fonts, and the inclusion of Greek letters in them offers new creative possibilities for mathematical typesetting.
3. Palatino Linotype. Palatino Linotype is a font family bundled with Microsoft's Windows 2000 Professional operating system. These fonts are WGL4 fonts, but they also contain many more Latin glyphs than most WGL4 fonts, and thus allow more typographic options.

Both Hoefler-Text and Palatino-Linotype enable one to achieve higher typographic quality than is possible with other font families, including Computer Modern and families for which there are expert fonts. The fonts in these families include several sets of numerals for different uses and more standard ligatures than other text typefaces. They also contain quaint ligatures, alternate glyphs and swashes, but these are not particularly rare.

We can expect additional rich fonts to become available. Adobe is preparing new OpenType "Pro" fonts, which will also include multiple sets of numerals and other features. One "Pro" font family is already available, Tekton Pro, which is bundled with Adobe's InDesign page layout program. Adobe

Hoefler Text is a rich font family. It includes small capitals (*in all the shapes* ***and weights***). It includes both old-style numerals, such as 69978, and lining numerals, such as 69978. It includes many standard ligatures, such as fj for fjord, quaint ligatures such as ſt for adding intereſt. It also includes alternate and swash characters, as in Quick, *wow, Hoefler & Hoefler, fore & aft*. The fonts also include components for fractions, such as 6²³⁄₄₇. The fonts contain both fixed width and variable width numerals, for tables and running text, respectively. Compare the two columns:

Variable Width	Fixed Width
1111.11	1111.11
4567.23	4567.23
1111.11	1111.11
4567.23	4567.23

Figure 1: The typographic features of the Hoefler Text family (most but not all).

Palatino Linotype is a rich font family. It includes small capitals (*in all the shapes* ***and weights***). It includes both old-style numerals, such as 69978, and lining numerals, such as 69978. It includes many standard ligatures, such as fj for fjord, quaint ligatures such as ſp for ſpecial occasions. The fonts also include components for fractions, such as 6²³⁄₄₇. Since the fonts contain Greek letters, we can use them for mathematical typesetting, as in $\max 2x + 5\alpha = \Sigma_{i=1}^{\beta}(8\xi + \Omega)$. There are even two variants of theta (θ and ϑ), phi (ϕ and φ), sigma (σ and ς), and pi (π and ϖ). There is only one version of epsilon, however (ε). The Upsilon (Υ) is different from the Latin Y. The fonts contain both fixed width and variable width numerals, for tables and running text, respectively. There are only variable-width lining numerals; there are no old-style variable-width numerals. Compare the two columns:

Variable Width	Fixed Width
None	1111.11
None	4567.23
1111.11	1111.11
4567.23	4567.23

Figure 2: The typographic features of the Palatino-Linotype family.

Franklin Gothic is a family of WGL4 fonts. Such fonts don't have a rich set of Latin glyphs, but they do have glyphs for Greek and Cyrillic letters. We can use the Greek letters and several symbols for mathematical typesetting, as in $T_1(\alpha, \xi) = \Theta(n^2) = \Sigma_i x_i$. In many of these fonts, however, the shape of some of the letters, especially α, is not suitable for mathematics.

Figure 3: The typographic features of the Franklin-Gothic family.

has announced that it will issue additional "Pro" families, including Adobe Garamond, Minion, and Myriad. The TEX community should, therefore, prepare to exploit these fonts.

Adobe's OpenType fonts are beyond the scope of this paper because their glyphs are stored in a so-called CFF format, a form of compressed Type1 outlines. To the best of my knowledge, there are no tools available today which enable the use of such fonts with TEX systems. The other fonts that this paper discusses have outlines in TrueType format which is relatively easy to use with TEX.

2 TrueType and OpenType Fonts

All the fonts discussed in this article are TrueType or OpenType fonts. The TrueType font format was originally developped by Apple and is now widely used on Machintosh, Windows, and Unix computers. The glyph in a TrueType font are described by outlines (quadratic splines). The hints that ensure high-quality rendering at low resolution are essentially little program that move the control points of the outlines to fit the discrete pixel grid.

TrueType fonts offer two advantages over PostScript Type1 fonts, another widely used outline font format. First, since the hints are programs it is possible to achieve higher quality rendering at low resolution than with Type1 fonts, which have declarative hints. (Hinting TrueType fonts well generally requires a significant effort by an expert font engineer, however.) Second, TrueType fonts combine the description of glyphs, metric information (glyph metrics and kerning pairs), and auxiliary information into a single file, so TrueType fonts are often easier to install and manage than PostScript fonts.

There are now several variants of TrueType fonts. First, Windows and Macintosh computers use slightly different formats. This article deals only with Windows fonts, which are also usable on Unix and Linux. TrueType GX, now known as AAT (Apple Advanced Typography), is an extension of the original TrueType font format that was designed by Apple to offer advanced typographic refinement and control [2]. Apple commissioned and distributed a few GX fonts as part of the Macintosh's operating system, including Hoefler Text, Apple Chancery, and Skia. The only major font manufacturer that offers GX fonts is Linotype. GX fonts offer the typesetter pair kerning, tracking, ligatures, baseline adjustments, vertical substitution, optical alignment of line edges, accent positioning, several sets of numerals, fractions, and glyph variations. Most of these features work automatically; several are under the control of the typesetter (actually the typesetting program), such as selection of glyph variants and type of numerals.

OpenType is another enhancement to TrueType. The OpenType specification was developped by Adobe and Microsoft. OpenType offers two major advantages over conventional TrueType. First, OpenType fonts support advanced typographic features such as glyph substitution (ligatures, small capitals, old-style numerals, contextual shaping of arabic, and so on). Second, The glyphs in OpenType fonts can be described using the normal TrueType machinery or using Type1 machinery. This allows lossless conversion of Type1 fonts to OpenType.

Both GX and OpenType enhance TrueType buy adding so-called *tables*, or data structures, to the font file. The basic TrueType structures are maintained, however, so GX fonts and OpenType fonts with TrueType outlines are also valid TrueType fonts (OpenType fonts with Type1 outlines are not valid TrueType fonts).

3 PostScript 3 Core Fonts

PostScript 3 is new enhancement of the PostScript language (the previous standard is called PostScript Level 2). Several high-end laser printers already support PostScript 3.

Among the enhancements in PostScript 3 over PostScript levels 1 and 2 is a new set of core fonts that must be supported by all PostScript 3 devices. The core sets includes 136 fonts:

- The 35 standard PostScript fonts (Avant Garde, Bookman, Courier, Helvetica, Helvetica Condensed, New Century Schoolbook, Palatino, Times, Symbol, Zapf Chancery, and Zapf Dingbats).
- Some of the core fonts in Microsoft Windows operating systems (Arial and Times New).
- Some of the core fonts in the Apple Macintosh operating system (Chicago, Geneva, Monaco, and New York), as well as two of Apple's GX font families (Apple Chancery and Hoefler Text).

- Several fonts that are bundled with Microsoft Office products and with HP printers.
- Additional Adobe font families, such as Stempel Garamond, Carta, Joana, and Univers.

The 136 core fonts are resident in PostScript 3 printers, such as HP's 2500CM, Lexmark's T series, and Xerox's DocuColor CP40. Furthermore, to enable printer users to utilize all the fonts resident in their printers, printer manufacturers bundle screen fonts with the printers. Thus, users of PostScript 3 printers have access to all 136 fonts in either PostScript Type1 format or in TrueType format. Both HP and Xerox distribute the complete set of screen fonts for both Macintosh and Windows (the Windows-format fonts can be used on Unix and Linux systems as well). Thus, users of Microsoft Windows, Unix and Linux systems who purchase such printers now have access to fonts that were once available only on Macintosh systems. These fonts can be easily used by TEX and Omega[1].

4 The Hoefler-Text Font Family

Hoefler Text, designed by Jonathan Hoefler, was originally commissioned by Apple. The fonts were originally available in Apple's TrueType GX format only. It is now available from the Hoefler Type Foundry (www.typography.com) in PostScript Type1 as well. It is also available in Windows TrueType format as a screen font bundled with several printers, as explained above.

The Windows TrueType versions of the Apple fonts in the PostScript 3 core set retain all the glyphs that are present in the GX fonts, but not all the metric information. The glyph sizes are, of course, present, and so are the kerning pairs (the number of pairs ranges from 1784 in the regular font to 6907 in the black italic). The other metric and substitution information cannot be represented in non-GX TrueType fonts and is lost. (Most of the extra GX font information can be represented in OpenType fonts, a newer enhancement of the TrueType format, but my fonts were not OpenType fonts.)

PostScript 3's core font set includes five fonts from the Hoefler Text family: regular, italic, black, black italic, and ornaments (bold-weight fonts are available from the Hoefler Type Foundry, but are not part of the PostScript 3 core fonts). The regular font actually hides another font consisting of engraved capitals. The text fonts include between 711–881 glyphs each. These glyphs can be classified into the following categories:

- Latin letters, accents, and precomposed characters.
- Swash capitals in the italic fonts; most letters have one swash form, A and M have two, some have none. There is also a long-tail Q in the upright fonts.
- Initial and final swash variants for a few lowercase letters in the italic fonts.
- Small capitals in all the fonts. The italic fonts also contain a few variant small capitals.
- Standard, quaint, and archaic ligatures. The so-called standard ligatures include not only fi, fl, ff, ffi, and ffl, but also fb, fj, ffb, and many more (16 altogether in the italic fonts, less in the uprights). The quaint ligatures include ligatures for ct and st. Again, the italic fonts have many more than the uprights. The archaic ligatures are ligatures involving long-s, which is also present in all the fonts. The standard ligatures are supposed to always be used, the quaint ones under manual control of the typesetter. There are also swash forms for some of the ligatures.
- Four sets of numerals: variable-width old-style numerals (the default), variable-width lining numerals, fixed-width old-style and lining numerals. The fixed-width numerals are designed for tables, the variable-width for all other applications.
- Three sets of punctuation marks and monetary symbols. The default set is designed for text composed with capitals and lowercase, another set for all capitals, and a third set for text composed with small capitals only. The monetary symbols of the second set are also suitable for using with lining numerals in normal text, since their hight is aligned with that of the lining numerals.
- Fraction components and a few precomposed fractions. The components include a fraction slash and numerals for the numerator and denominator. When composed, the height of the fractions is aligned with the height of the capitals and lining figures.
- Inferior and superior lowercase letters and numerals designed for superscripts and subscripts. These are not particularly useful for TEX because the character sets are incomplete, but they should be suitable for marking footnotes.

[1] Omega is a TEX-like program supporting 16-bit characters. It is now a standard part of some TEX distributions, such as teTEX, so most users can use it directly without any need to install new software. Omega preloaded with LATEX is invoked using the command `lambda`.

- Symbols: symbols used in text (e.g., section and paragraph symbols) and a few mathematical symbols.

5 TeX Usage

From the user's point of view, the Hoefler Text family for TeX includes the seven font families shown in Figure 4:

- The default family uses variable-width old-style numerals and contains eight fonts: upright medium and bold, italic medium and bold, and four caps-and-small-caps fonts.
- The engraved family with one font, containing only capitals (lowercase characters are mapped to uppercase glyphs).
- Three families with four fonts each (regular, italic, bold, bold-italic) containing either variable-width lining numerals, fixed-width old-style numerals, or fixed-width lining numerals. These are upper-and-lowercase fonts.
- A family of four (regular, italic, bold, bold-italic) fonts that map lowercase to upper case and use punctuation and monetary signs that align with the capitals (e.i., they are slightly raised compared to the default signs).
- A family of four fonts that map both upper and lower case to small-caps glyphs and use punctuation and monetary signs that align with the small capitals.

The fonts use the T1 encoding. A LaTeX package that I have written defines the families using the New Font Selection Scheme.

These fonts activate the so-called standard ligatures (including fb, fj, etc.) automatically. The fonts also contain quaint ligatures and alternate and swash glyphs, but these are not used automatically. To select an alternate character, user must insert the `^` characters (ASCII circumflex) after the base letter (this assumes that `^` is a letter; more on that later). Two consecutive `^`'s after a base letter generates the second alternate glyph, if it exists. To activate a quaint ligature, the user inserts a `^` between the constituent letters of the ligature. Figure 5 shows how to use this technique.

A very similar mechanism is used to typeset fractions. To typeset a fraction whose numerator and denominator are separated by a fraction slash, the user types `/^` to start the fraction, then the numerator, a slash, and the denominator. This mechanism works in horizontal mode, not math mode. Figure 5 demonstrates this mechanism as well.

TeX must see the code for the `^` letter itself, which normally needs to be specified in LaTeX input file using the macro `\textasciicircum`. I mostly use LyX[2] to prepare input files for LaTeX, so I simply type the `^` and LyX inserts the macro in the LaTeX input file. If you use a text editor to prepare input files, a character that TeX considers to be a letter, perhaps the asterisk, would be preferable.

6 Technicalities

Let us first explain the basics of using TrueType and OpenType fonts, rich or conventional, with TeX. While some TeX implementations, such as TrueTeX, support TrueType fonts directly, most TeX systems require that TrueType fonts be prepared before they can be used. The purpose of the conversion is to produce metric files (.tfm, .ofm for Omega) and sometimes virtual fonts (.vf, .ovf for Omega) for TeX or Omega, and to produce a "glyph container" for the output processor (e.g., `dvips`). The metric and virtual-font files are produced in two steps. First, a utility extracts the metric information from the TrueType fonts and stores it in an AFM file in the format that Type1 metrics is stored in. Several free utilities, such as `ttf2afm` and `ttfps` can perform this conversion. Second, we need to convert the TrueType font file to a format that the output processor can process. We can produce bitmapped fonts at various sizes using the utility `ttf2pk`. The resulting pk font files can be used by most DVI processors. The disadvantage of bitmapped fonts is that PDF files containing them don't render very well on the screen. We can also convert the TrueType font files into scalable font files that can be embeded in PostScript files. One method is to convert the outlines to Type1 outlines. This is a lossy process in that the hints in the original font are lost, but it produces fonts that can be used on any PostScript device. Another method is to convert the TrueType font file to a Type42 PostScript font file. A Type42 font is simply a TrueType font file wrapped in a PostScript program. Ghostscript can process such fonts, as well as most PostScript Level 2 printers, but some older PostScript printers cannot. Adobe distiller can embed such fonts in PDF documents. The free utilities `ttfps` and `gfontview` can convert TrueType fonts to Type42 fonts. PdfTeX can embed TrueType fonts directly in its output, but it does

[2] LyX (`www.lyx.org`) is a visual editor for LaTeX. LyX is not a what-you-see-is-what-you-get system, but its screen display is a reasonable rendition of the LaTeXfile: section titles appear in a large bold font, mathematics are shown using greek symbols and large delimiters, and so on. Mathematics can be entred using LaTeXcommands: type `\alpha` and hit Escape and a greek α appears.

Variable width old-style	(Notice the Punctuation, 2691)
Fixed width old-style	(Notice the Punctuation, 2691)
Variable width lining	(Notice the Punctuation, 2691)
Fixed width lining	(Notice the Punctuation, 2691)
Caps and small caps	(NOTICE THE PUNCTUATION, 2691)
All capitals	(NOTICE THE PUNCTUATION, 2691)
All small capitals	(NOTICE THE PUNCTUATION, 2691)
Engraved capitals	(PUNCTUATION, 2691)

Figure 4: Hoefler font families

Input	**Typeset output**
`Quick Q^uick More M^ore M^^ore`	Quick Quick *More More More*
`w^ow at^ ffk^ & &^ &^^`	*wow at ffk & & et*
`c^t s^t s^p i^s g^g gg T^h`	ct st *sp is gg gg Th*
`/^1/8 /^11/64`	⅛ 11⁄64

Figure 5: How to use alternates, swashes, quaint ligatures, and fractions

need TEX metric files (.tfm). To the best of my knowledge, there are no utilities that can prepare OpenType fonts with Type1 outlines for use with TEX output processors or with pdfTEX.

To use the Hoefler fonts with TEX and Omega, we must generate metric files. To use the fonts with `dvips` and `odvips`, we also have to generate PostScript fonts. I generated `afm` metric files from the TrueType fonts using `ttfps`. I then used the afm files to build the TEX/Omega virtual fonts and the TEX/Omega font metric files. `Ttfps` can also generate PostScript Type42 fonts files. It turned out, however, that `ttfps` produced defective Type42 fonts because the Hoefler fonts are large. I therefore produced another set of Type42 fonts using a program called `gfontview`; these were fine. (The problem with `ttfps` is that it encodes the ttf file in one PostScript array, and these array grow too large on large fonts. The ttf file should be broken into several arrays on table boundaries, which is what `gfontview` does.) The PostScript output files that dvips and odvips produce using these fonts can be processed by ghostscript (e.g., they can be screen previewed), can be printed on most PostScript Level 2 printers, and can be converted to PDF using Adobe Distiller. Of course, if you have a PostScript 3 printer there is no need to embed the Type42 fonts in the output file at all. The TrueType fonts can also be embedded in PDF files directly by pdfTEX—there is no need for Type42 fonts here. A final word on using these fonts in PostScript and PDF documents: you should check the license that come with the printer software to determine what uses of the fonts are legal and which are not.

Let's turn our attention to the virtual fonts that represent the fonts for TEX and Omega. The TEX virtual fonts manage to contain both all the glyphs defined by the T1 encoding as well as all the alternates, swashes, extra ligatures, and fraction components because they are 16-bit Omega fonts. That is, the lower 256 glyphs in the virtual fonts are the glyphs defined by the T1 encoding. Beyond glyph 255, the fonts contain all the remaining glyphs in the Hoefler fonts in a random order. The extra glyphs are accessed using ligatures of characters in the range 0–255. Thus, an 8-bit input file that uses only one font can generate an output file with more than 256 glyphs. For example, the characters f and j are automatically combined into an fj glyph using ligature specification in the file. The index of the fj glyph in the font is larger than 255. Likewise, the characters M and ^ are combined into a glyph of a swash M.

Fractions are supported in a similar way. The `/^` combination is transformed into a little used glyph, a superior comma. A superior comma followed by a numeral is changed into a superior numeral. Any superior numeral followed by another numeral changes the second numeral into a superior numeral as well. A superior numeral followed by a slash changes the slash into a fraction slash, and so on. This requires a lot of ligatures (111 to be precise) but it works remarkably well.

Not surprisingly, activation of quaint ligatures also uses the same mechanism: a lowercase letter followed by a ^ changes the letter into a superior letter, and a superior letter followed by a letter generates the quaint ligature.

I generated the fonts using a program that I wrote called `afm2ofm`. The program is a modification of the C program `afm2tfm` from the `dvips` distribution. The new program, called `afm2ofm`, differs from the original in several ways. First, it generates several 8-bit container fonts that together contain all the glyphs in the TrueType font. For each such font, the program generates an encoding file (.enc), to be used by odvips, and a tfm file, to be used both by Omega and by odvips. Because the ordering of glyphs in the 8-bit fonts is unimportant, they can be generated automatically. Second, it maps all of these 8-bit fonts into the virtual font that it generates, and the virtual font can thus contains more than 256 glyphs. The lower 256 conform to an input encoding (T1 in my case, or T1 modified to use small caps, etc.). The other glyphs in the TrueType font are mapped into free slots beginning at index 128 (this is actually the original behavior of `afm2tfm`; all I had to do here was to let continue beyond slot 255). The rest is standard: the ligatures are specified using the `LIGKERN` command in the input encoding file.

The same task can probably also be accomplished by `fontinst`.

One remarkable feature of this method is that it degrades only slightly if you insist on using 8-bit TeX rather than Omega. If you run `afm2tfm` and `vptovf` rather than `afm2ofm` and `ovp2ovf` the same input encoding files, you get perfectly usable `vf` and `tfm` files. They will include all the glyphs in the T1 encoding, including the ligatures ff, ffl, and ffi, but you will not have access to the alternates, swashes, extra ligatures, and fraction.

7 WGL4 Fonts

WGL4 is a list of glyphs defined by Microsoft, which contains all the letters that are used in European languages, including Greek and Russian. So-called WGL4 fonts are TrueType or OpenType fonts that contain all the glyphs in the list. Beyond the fact that one can use WGL4 fonts to set text in many languages, the Greek letters can sometimes be used for mathematical typesetting.

Many WGL4 fonts are available. Quite a few are freely available from Microsoft at `www.microsoft.com/typography` (Andale Mono, Arial, Arial Black, Georgia, Verdana, Comic Sans, Impact, Times New Roman, and Courier New). Others are bundled with Microsoft's various Office Products, such as Word and Publisher and/or with Windows operating systems: Franklin Gothic, Century Gothic, Monotype Garamond, Century Schoolbook, Bookman Old Style, Corsiva, Mistral Tahoma, Lucida Console, and Lucida Unicode. (Microsoft Publisher 2000 comes with a few additional latin-only rich fonts: Californian FB, Berlin Sans FB, Agency FB, and High Tower Text, Niagara, Magneto.) Figure 6 shows samples of many of these fonts.

The WGL4 list contains 652 glyphs, but many fonts contain a few less or more glyphs. Comic Sans, for example, contains only 574 glyphs, even though it is classified as a WGL4 font by Microsoft. Several fonts contain a lot more glyphs. Arial, Times New Roman, Courier New and Tahoma contain over 1200 glyphs, supporting also Arabic, Hebrew, and Thai.

Not all the Greek glyphs in these fonts are suitable for mathematical typesetting. In particular, the small alpha is often indistinguishable from a Latin small 'a', and the capital Upsilon is sometimes indistinguishable from a Latin capital 'Y'. The small beta is always different from a small Latin 'b', but it's shape is sometimes quite different from the betas that we are used to see in mathematical contexts.

In one of the WGL4 families, Georgia, the numerals are old-style, and hence not particularly suitable for modern mathematical typesetting. Version 1.0 of Verdana, however, had lining (actually semi-lining) numerals, so if you still have it, you might want to use its numerals in math mode together with glyphs from recent versions of Georgia.

Finally, the selection of Greek letters in WGL4 fonts is smaller than the selection in TeX's fonts. In particular, these fonts contain only one glyph for epsilon, theta, sigma, and phi, whereas TeX has two for each.

These deficiencies does not mean that you can't use the Greek glyphs in WGL4 fonts for mathematics, but that you might have to use a few glyphs from other fonts when you build macros to use these fonts.

Two fonts which seem to me particularly useful for mathematics are Georgia (good alpha and beta, Upsilon is Y) and Franklin Gothic, a highly readable sans serif (even better alpha and beta, Upsilon is Y). The Franklin Gothic family also includes condensed medium and demi-bold fonts, which are suitable for titles.

From a technical point of view these fonts can be treated just like the Hoefler Text fonts. Since the set of useful glyphs is smaller, however, 8-bit fonts are probably fine. To simplify the use of these fonts in math, I also produced virtual fonts with OMS and OML encoding and declared the fonts in LaTeX as a new math version. This is not a complete solution, since it does not provide substitutes for glyphs missing in WGL4, but it enabled me to taste the possibilities. Figure 3 shows a bit of math set in Franklin Gothic Book.

Ideally, one should design glyphs to complement these fonts, in much the same way that Haralambous did for Baskerville [6]. Some glyphs can be borrowed from other fonts. For example, one might use the Hebrew aleph from Arial, a sans serif font, when typesetting math with Franklin Gothic say.

Many of these fonts contain extensive pair kerning data. The fonts in the Franklin Gothic family have 1000 pairs, and fonts in the Arial, Times New Roman, Garamond, families also have hundreds of pairs. The fonts in Georgia, Verdana, Century Gothic, and Monotype Corsiva, on the other hand, have no kerning pairs at all.

8 Palatino Linotype

Palatino Linotype is a rich font family bundled with Windows 2000 Professional. As such, we can expect it to become widely available in the future. The fonts are WGL4 fonts, but they also have advanced typographic features not found in most WGL4 fonts. (They also contain many precomposed glyphs for Vietnamese and Greek.) The typographic features in the fonts include three sets of numerals (fixed- and variable-width lining and fixed-width old style), numerous standard and quaint ligatures, precomposed fractions and fraction components, small capitals (and some small punctuation marks to go with them), superior letters and numerals, currency symbols, and a few mathematical symbols. Figure 2 demonstrates many of these features.

The glyphs of the Greek letters are suitable for math typesetting[3]. The alpha and Upsilon are well differentiated from the Latin a and Y, and both them and the beta are suitable as math symbols. There are also variant forms of theta, phi, sigma, and pi. There is only one variant of epsilon, but this is less important, because they two variants in Computer Modern are too similar to be used together in the same document.

According to design information contained in the font files, Herman Zapf (the designer of Palatino) "has drawn numerous additional characters to include an extensive range of ligatures, numerals, fractions, and support for both Cyrillic and both monotonic and polytonic Greek." I do not know whether Zapf designed the Greek letters with math in mind, but this is possible considering his experience in designing math fonts gained when he designed AMS Euler, based on specifications and critique from Knuth [8, 9].

The fonts have pair kerning data that is extracted by `ttfps`, but the number of pairs is not extensive, between 71 and 148 pairs per font. The kerning pairs that are present are for unaccented Latin glyphs and for punctuation. There is no kerning data for accented glyphs, non-Latin glyphs, or small caps. It is possible that there is more kerning data in the "GPOS" table of the fonts (named tables hold data structures in a TrueType font file), not in the "kern" table, which `ttfps` uses to build the `afm` file, but I have not checked that.

Technically speaking, the fonts are converted to use with TeX just like the Hoefler Text fonts.

9 Conclusion

Macro packages, especially LaTeX, should support the features of rich fonts. In particular, I believe that macro packages should automatically selects the correct numerals for most situations. The enhancement to LaTeX that I envision would automatically select fixed-width numerals in tables, and would automatically select lining numerals when there numerals are isolated (as in page numbers and the table of contents) or when the numeral is likely to be followed by a capital (as in section numbers). There are also cases where the LaTeX style can also automatically select the best punctuation, especially parentheses, brackets, and braces. Essentially, when lining figures are enclosed alone in parentheses, the parentheses should also be lining, as in reference number in many bibTeX styles and in equation numbers.

These enhancements would be easier to implement if the font selection scheme supported these font features, which are orthogonal to series and shape, but an easier hack might be to change the encoding (say from T1 to T1Lining or T1Fixed).

Either way, the wider availability of rich fonts, such as Hoefler Text, WGL4 Fonts, Palatino Linotype, and more importantly, the immanent introduction of Adobe's rich "Pro" fonts, imply that the community should discuss this issue if TeX-based systems are to remain in the top tier of typesetting.

The Greek letters in WGL4 fonts, in particular, enable a significant improvement in the typography technical documents. For the first time it is now possible to use several font families in a document with a matching mathematical style for each family. For example, we can use Palatino for the text of the document, with a matching math style with most of

[3] Diego Puga designed an unrelated set of greek letters for mathematical typesetting with Palatino. His package, `mathpazo`, contains Type1 fonts with his greek letters and a few other symbols, as well as virtual fonts that combine his greek letters, latin letters from the Palatino fonts, and mathematical symbols from Computer Modern. The package is available on CTAN.

Arial	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	אבג
	The quick brown fox	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	אבג
Courier	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	אבג
New	**The quick brown fox**	**169**	**ΔΛΨΩαβγζξ**	**ЉЖЗИЮбвдж**	**אבג**
	The quick brown fox	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	אבג
Times New	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	אבג
Roman	**The quick brown fox**	**169**	**ΔΛΨΩαβγζξ**	**ЉЖЗИЮбвдж**	**אבג**
	The quick brown fox	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	אבג
Comic Sans	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
Andale Mono	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
Georgia	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
	The quick brown fox	**169**	**ΔΛΨΩαβγζξ**	**ЉЖЗИЮбвдж**	
	The quick brown fox	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	
Verdana	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
	The quick brown fox	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	
Bookman	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
Old Style	*The quick brown fox*	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	
Century	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
Gothic	*The quick brown fox*	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	
Garamond	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
	The quick brown fox	**169**	**ΔΛΨΩαβγζξ**	**ЉЖЗИЮбвдж**	
	The quick brown fox	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	
Century	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
Schoolbook	*The quick brown fox*	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	
Corsiva	*The quick brown fox*	*169*	*ΔΛΨΩαβγζξ*	*ЉЖЗИЮбвдж*	
Mistral	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	
Tahoma	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	אבג
Tekton Pro	The quick brown fox	169	Small caps	Old-style numerals 169	
	The quick brown fox	**169**	**Small caps**	**Old-style numerals 169**	
	The quick brown fox	*169*	*Small caps*	*Old-style numerals 169*	
Californian FB	The quick brown fox	169	Small caps	Quaint ligatures ct	
	The quick brown fox	*169*	*Small caps*	*Quaint ligatures ct*	
High Tower	The quick brown fox	169	Small caps	Old-Style numerals 169	
Text	*The quick brown fox*	*169*			
Berlin Sans FB	The quick brown fox	169	Alts WWSS&&ggOO	Old-style numerals 169	
	The quick brown fox	**169**	**Alts WWSS&&ggOO**	**Old-style figs 169**	
Agency FB	The quick brown fox	169	Alternates AAKKMMRRVVWWXXYY		
Niagara	The quick brown fox	169	Small caps	Small numerals 169	
Magneto	The quick brown fox	169	Alternates A A H H f f i i w w x x		
Apple	*The quick brown fox*	169	Small caps	*Old-style numerals 169*	
Chancery			*Alts AAAAAAA*	*Quaint ligatures ctThctsh*	
Lucida Sans	The quick brown fox	169	ΔΛΨΩαβγζξ	ЉЖЗИЮбвдж	אבג

Figure 6: Samples of some of the rich fonts that are mentioned in the article. The samples show numerals and latin text in all the fonts, and rich text features such as greek, cyrillic, and hebrew letters, small capitals and old-style numerals, quaint ligatures and alternate glyphs. Not all fonts in each family are shown.

the symbols taken from Palatino fonts, and Franklin Gothic for captions and titles. There are bold version of the mathematical symbols, so the math can match the text in weight, not only style. This would represent a significant improvement over today's typography, with regular-weight serifed math in bold titles and sans serif captions, for example.

Rich fonts such as Hoefler Text also raise the issue of visual editing. Glyph variations are meant to be selected visually. TEX and LATEX packages that support rich fonts, such as Alan Hoenig's package for the Adobe Poetica family, rely on a combination of plain-text markup to select glyphs and/or on automatic variant substitution, say a swash at the end of every word. Although Hoenig has achieved magnificent results with this method, it is not the most direct way of using such fonts.

A better way would be for LATEX editors, particularly LYX, to allow the user to select glyphs visually and to automatically produce the plain-text markup in the LATEX input file. LYX would use the font, say Hoefler Text, as a screen font and allow the user to select glyphs for which there are variants or to activate individual quaint ligatures. This, however, requires significant changes in LYX.

References

[1] Adobe Systems Inc., *PostScript 3 Core Font Set Overview*, Technical Note 5609, 1997. Available online at http://www.adobe.com. A PDF document listing samples of the fonts in the Core Set is also available online.

[2] Apple Computer Inc., *Inside Macintosh: QuickDraw GX Typography*, Addison-Wesley, 1994. Available online at http://developer.apple.com/techpubs/mac/GXTypography/GXTypography-2.html.

[3] Apple Computer Inc., *TrueType Reference Manual*, 1997. Available online at http://fonts.apple.com.

[4] The Hoefler Type Foundry, *Catalog Number 2*. No date. Available online at http://www.typography.com.

[5] Yannis Haralambous and John Plaice, "First applications of Omega: Adobe Poetica, Arabic, Greek, Khmer," *TUGboat* 15(3), September 1994, pages 344–352.

[6] Yannis Haralambous, *Une police pour la Societe mathematique de France: le SMF Baskerville*, Les Cahiers GUTenberg 32, 1999, pages 5–20.

[7] Alan Hoenig, "The Poetica family: fancy fonts with TEX and LATEX," *TUGboat* 16(3), September 1995, pages 244–252.

[8] Donald E. Knuth and Hermann Zapf, "AMS Euler—A New Typeface for Mathematics," *Scholarly Publishing* 20(3), April 1989, pages 131–157.

[9] David R Siegel, "The Euler Project at Stanford," Technical Report, Stanford University, 1985.

◇ Sivan Toledo
School of Computer Science,
Tel-Aviv University
Tel-Aviv 69987, Israel
sivan@math.tau.ac.il