Font Forum

TrueType Fonts in TEX

Vladimír Koutný

TrueType fonts are widely used these days; unfortunately, they are not supported by many non-Windows programs, like TEX. Generally, using fonts different from those written in METAFONT usually bring some problems with compatibility or platform dependence. The most common problem is the need for manual font generation at a specific resolution, demanded by TEX, or you need a special device driver for this purpose.

An "easy" solution of all these problems is a conversion of a font into METAFONT format. Once the font is converted, you can use it the same way as regular METAFONT fonts. As long as I wanted to use some TTFs in TEX some time ago and I didn't find any converter, I decided to write my own.

Let's look at the TrueType fonts first. Each character is described by its outline, composed of Bezier curves. Some information used for scaling is also included. Currently, the converter reads only the glyph information for a character. The glyph consists of several closed paths. All paths are filled using invert-filter, i.e., the area filled twice will not be filled at all. These paths should not cross themselves, but, as long as the Windows OS doesn't care about that, some fonts are not drawn properly. This causes problems in METAFONT, which treats this as an error in the input file. (Actually, this error can occur only when there is a crossing on a single path so that a "loop" comes up. The paths are processed independently by METAFONT, so crossing of two paths should not cause problems.) Because of this representation of TrueType fonts, the conversion program also generates a set of Bezier curves forming closed paths.

There is also another kind of glyphs, composed glyphs. These are specified as sets of several other glyphs, which are transformed and joined together. Many accented letters are stored in this way in TrueType fonts.

Different styles of the same font, like bold or italics, are usually stored as different TTF files. There are several possible encodings for use in TrueType fonts, but the one most widely used is UNICODE. Of course not all characters are included in a TTF file; those unused are mapped to a default "warning" glyph, usually an empty square. As long as TEX uses only 256 characters in a single font file, it might be desirable to create several MF files.

Now to the conversion program. The first version was based on "The FREE TrueType Font Engine" written by David Turner, but, because of some limits of this library (e.g., a 64KB memory limit), a new one was created, according to the specification for TrueType Fonts [1].

The conversion itself starts by loading all glyphs from the TTF file. All characters we want to export are specified in a single text-file. This file

An earlier version of this article appeared in *Zpravodaj* (the Bulletin of the Czech-Slovak TEX Users Group) **99**/3, pp. 159–160, in Slovak, with the title "TrueType-fonty v TEXu"; it has been translated by the author, and appears here with permission.

specifies the UNICODE number of each character, together with a character number that will be used in METAFONT. During the conversion, the program requires some free space in the working directory for temporary files, that are automatically removed at the end. Also, the output file will be written here, with the name of the original TTF file, but with an MF extension. Because of this, you should have write access to the working directory. The METAFONT output routines were designed by my schoolmate, Rišo Královič.

The newly created font file can be used directly in TEX, maybe preceded by manually running METAFONT to create a TFM file. Unfortunately, there are some cases when METAFONT claims errors. The most common is the "Strange path (turning number is zero)" — this is the error mentioned above, caused by "loops" in the outline. Another reason for this error is the small size of the font, together with the resolution used for generation. This happens mostly with decorative fonts with many details. Possible solutions: use such a font in bigger size, or use higher resolution.

There are two more aspects we could look at. The first one is the kerning. In most TrueType fonts, there are stored kerning values. Normally, kerning is not supported by Windows, just some specific programs are capable of doing so. In TEX, the kerning is something natural, so this kerning information is quite useful. Therefore, the conversion generates also ligtables with kerning pairs.

Next, there are usually some ligature characters in a TTF font. In most fonts there are the fi (UNICODE 0xFB01) and fl (UNICODE 0xFB02) glyphs, as well as some others, like Œ, etc. There might also be still others, but the problem is that a regular TrueType font does not contain any information about ligatures. So if you know that this glyph is a ligature xy, then you can use it and write this ligature by hand into the MF file, but automatic generation of this kind of ligtables is not working yet.

There is also an extension to TrueType fonts, called TrueType Open [2]. This extension brings support for ligatures and also some other improvements, mostly useful for vertical, right-to-left and similar fonts. These fonts are usually treated as regular TrueType fonts, ignoring additional information. Unfortunately, only a few latin fonts are in this Open format; I've actually found only one (Tahoma; maybe it is because this font also contains Greek, Hebrew and Thai characters).

Another possible upgrade of this converter includes creating a somewhat more user-friendly interface, that will allow the user to select desired characters interactively by shape, not only by UNICODE number, together with the possibility to enter ligature information.

The latest version of the converter can be found at `http://www.ksp.sk/textools`, or at `http://www.linxee.sk/ttf`. Feel free to report any bugs, comments or suggestions concerning the problems described here to my address `kluka@hotmail.com`.

References

[1] TrueType 1.0 Font Files, Technical Specification, Revision 1.66, November 1995, Microsoft

[2] TrueType Open Font Specification, Version 1.0, July 1995, Microsoft

◇ Vladimír Koutný
Osuského 7
851 03 Bratislava, Slovakia
`kluka@hotmail.com`