Linus Romer

The

FETAMONT

Typeface

Design and Constructions
April 15, 2017

Contents

1 Introduction

The logo font, known from logos like METAFONT or METAPOST, has been very limited in its collection of glyphs. The new typeface *Fetamont* extends the logo typeface in two ways:

- Fetamont consists of 256+ glyphs, such that the T1 (a.k.a. EC, a.k.a. Cork) encoding table is complete now.
- Fetamont has additional faces like "light ultracondensed" or "script".

The `fetamont` package provides LaTeX support for the Fetamont typeface. Both the package and the typeface are distributed on CTAN under the terms of the *LaTeX Project Public License* (LPPL).

This document describes the design and the constructions of the typeface itself. The LaTeX support for the Fetamont typeface is described in [Romer17]. For Greek, the use of LuaTeX or XeTeX is recommended.

2 Comparison With Existing Logos

The following picture shows the METAPOST and the METAFONT logos written in Fetamont (gray) and Taco Hoekwater's Type 1 version of the logo font (outlined).

There are hardly any differences; only the "S" is significantly different, because its shape was changed by D. E. Knuth in 1997. The other faces of Hoekwater's *Logo* are also very similar to their corresponding Fetamont faces. Widths and kernings may rarely differ by one unit (except for the "A" in *Logo 9*, which has a strange width).

A comparison with the METATYPE1 logo from [Jackowski01] shows virtually no differences as well.[1]

The following picture compares *Fetamont Bold Condensed 40* with a traced version of the *Title Font* from `manfnt.mf`.

[1] I have never seen the original sources of the "Y" and the "1" but I think that my imitated "Y" and "1" are extremely close to the original.

3 Compiling The Sources

Since version 2017/03/13, Fetamont contains more than 256 glyphs, but METAFONT is only capable of storing 256 glyphs. Therefor, METAFONT has been replaced by METAPOST for compilation. Additionally, a special base file called `mf2outline.mp` has to be used. There is a Python script called MF2OUTLINE (`github.com/linusromer/mf2outline`) that can produce the necessary outline font formats. Store `mf2outline.mp` and `mf2outline.py` in the same place you can make outline fonts in your terminal with something like: `./mf2outline.py --encoding=unicode ffmr10`

4 The Fetamont Faces

Fetamont comes in 36 different faces, including script faces and condensed faces.

The file name of every face begins with the prefix `ffm`, which stands for «*f*ree typeface *f*eta*m*ont». The suffixes normally contain a symbol for the weight: `l` for light, `r` for regular, `b` for bold and `h` for heavy. The number at the end stands for the optical size (e.g. 10 pt). Depending on the face, the suffix is made of additional symbols:

Upright				Oblique			
	r8	b8	h8		o8	bo8	ho8
	r9	b9	h9		o9	bo9	ho9
l10	r10	b10	h10	lo10	o10	bo10	ho10
Condensed Upright				Condensed Oblique			
lc10	c10			lco10	co10		
		bc40				bco40	
Ultracondensed Upright				Ultracondensed Oblique			
lq10				lqo10			
Script Upright				Script Oblique			
lw10	w10	bw10	hw10	lwo10	wo10	bwo10	hwo10

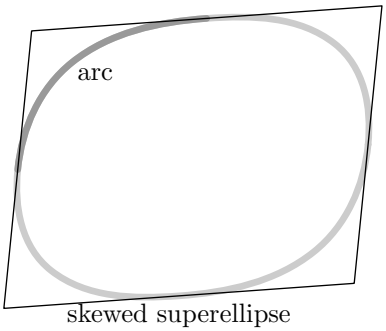
The number of possible faces is theoretically endless. Anyone wishing to design a new face for Fetamont can do so by just redefining the parameters of `ffmr10.mf` , saving the file under a new name and compiling this file with MF2OUTLINE.

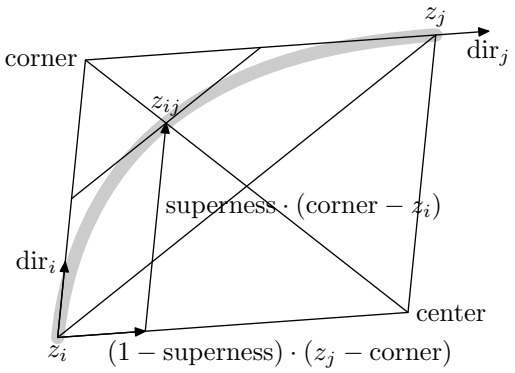
5 Special Techniques

Fetamont uses some special techniques. The following subsections will document these techniques.

5.1 Arc Constructions

Practically all curved paths in *Fetamont* are made out of so-called *arcs*. An arc is a kind of a quarter of a skewed superellipse. The skew is only needed if the arcs have to look randomized like in the script style of fetamont.





In order to draw such an arc, the user defines the starting points z_i, the starting direction dir_i, the ending point z_j, the ending direction dir_j and a so-called *superness*. The macro $\mathbf{arc}(z_i, \mathrm{dir}_i, z_j, \mathrm{dir}_j)$ then defines the path as follows:

- Compute the point z_{ij}, which is at center + superness · (corner − center) in vector terms. So if e.g. superness = 0.8, z_{ij} is reached after travelling 80 % of the straight path from corner to center. One can see easily, that z_{ij} can also be computed by

$$z_{ij} = z_i + \text{superness} \cdot (\text{corner} - z_i) + (1 - \text{superness}) \cdot (z_j - \text{corner})$$

- Now make a nice curve, that leaves z_i in the direction dir_i, passes z_{ij} in the direction $z_j - z_i$ and ends in z_j heading for the direction dir_j.

Here is the METAFONT translation of this construction report:

```
vardef arc(expr zi,diri,zj,dirj) =
 zi{diri}...
 begingroup
  save corner,zij;
  pair corner,zij;
  corner=zi+whatever*diri=zj+whatever*dirj;
  zij=zi+superness*(corner-zi)+(1-superness)*(zj-corner);
  zij
 endgroup{zj-zi}
 ...zj{dirj}
enddef;
```

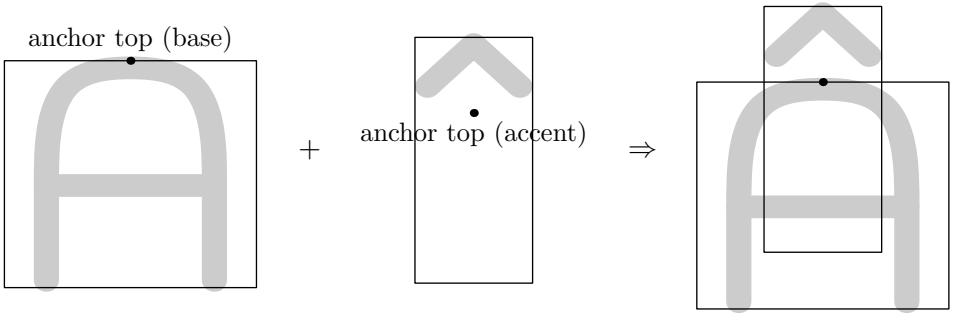
Everything in between `begingroup` and `endgroup` is just the computation of z_{ij}.

Note that Donald E. Knuth used a little different approach to draw randomized arcs for his «crazy shapes» of the Logo typeface.

5.2 Combined Characters

In order to draw accented and other combined characters, it is helpful to use *anchors*. The concept of anchors is common in type design outside of the METAFONT/-POST world. However, anchors rarely have been seen in METAFONT/-POST up to now.

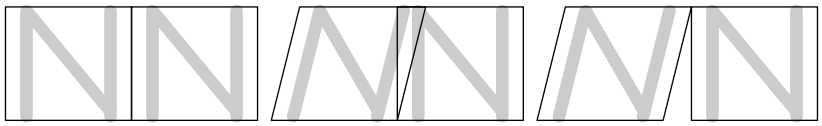
The idea is easy: Put an anchor at a given point in a base glyph and in the accent glyph; then overlay the two glyphs such that the anchors coincide, producing the pre-composed accented character.



Normally there are several kind of anchors needed. E.g. «Â» and «Ą» need two different anchors and so do «Ĺ» and «Ł». Fetamont needs five kind of anchors: «top», «topright», «bot» and «cedilla».

5.3 Italic Corrections

Letter spacing is unproblematic if two upright letters are combined, like «NN». But if the first letter is italic, the letters will get too close (like «*N*N») and need additional space (like «*N*N»). This additional space is called *italic correction*.



D. E. Knuth has already defined an italic correction for the letter «T», because this is the last letter of the logos METAFONT and METAPOST. As for the *Computer Modern* typeface he found `italcorr ht#*slant+.5u#` to be a suitable italic correction. However, this is not a perfect idea because the italic correction should tend to 0 (and not `.5u#`) when the slant tends to 0. Hence, every character in Fetamont different to «T» has an italic correction proportional to the slant and the letter height. E.g. the letter «A» has an italic correction of `.8ht#*slant`.

5.4 Randomize Feature

Normally, the randomization of the script faces has a fixed seed. However, for the OpenType versions of the script faces I have additionaly included five variants with random seeds. LuaTEX can access these variants via the Randomize feature.

Eben schließt in sanfter Ruh
Lämpel seine Kirche zu;
Und mit Buch und Notenheften
Nach besorgten Amtsgeschäften,
Lenkt er freudig seine Schritte
Zu der heimatlichen Hütte,
Zündet er sein Pfeifchen an.

The text shown above is the product of the following source:

```
\documentclass[11pt]{article}
\usepackage{fontspec}
\setmainfont[Letters=Random]{Fetamont Script}
\begin{document}
\noindent \textbf{Eben schließt in sanfter Ruh}\\
Lämpel seine Kirche zu;\\ Und mit Buch und Notenheften\\
Nach besorgten Amtsgeschäften,\\ Lenkt er freudig seine Schritte\\
Zu der heimatlichen Hütte,\\ Zündet er sein Pfeifchen an.
\end{document}
```

References

[Jackowski01] Bogusław Jackowski, Janusz M. Nowacki, and Piotr Strzelczyk. *METATYPE1: A METAPOST-based engine for generating Type 1 fonts*. `ntg.nl/eurotex/JackowskiMT.pdf`, 2001

[Romer17] Linus Romer. *The Fetamont Package*. 2017