

# Everything we want to know about Font Resources

Chris Rowley

Open University, UK

c dot a dot rowley (at) open dot ac dot uk

## Abstract

A brief discussion of Font Resources: subsystems that know how to ‘visualize’ a string of (Unicode) characters. We mention existing Font Resources, desired capabilities for  $\LaTeX$ , and questions for further study.

## 1 Introduction

This article is based on my introduction to a discussion of font resources and their use that took place towards the end of the TUG 2006 conference. By that time, a lot of useful detailed information (much of it correct!) about modern font resources had been provided, both formally and informally, to me by many delegates (whilst others, naturally, assumed that we all knew everything before they started talking).

I therefore decided to focus on more general ideas about the philosophy behind the new technology and how this fits with the classical approach of  $\TeX$ -based typesetters.

The bibliography points to a few of the many relevant web resources.

## 2 $\LaTeX$ -related work

There are many specific questions to be asked and decisions to be made before building a  $\LaTeX$  [4] system for automated selection from the available choices offered by modern font resources, due to the large range of variants within what was once a single font. We shall also need to extend the font-changing commands; work on glyph selection in math mode has been started already. Most of these ideas are being pioneered by Will Robertson in his work with  $X\TeX$  [8], particularly the `fontspec` [3] package.

## 3 What is a Font Resource ?

For current purposes I shall use the following informal answer. A FONT RESOURCE is a subsystem that ‘knows how to visualize a string of (Unicode) characters’. Note that this is not quite the same as saying that a font resource is itself capable of rendering the character string; thus, for example, a  $\TeX$  `.tfm` file is a Font Resource although it contains only metric information, not rendering information.

So is that all a Font Resource can do? And must it be able to visualise *any* text string? These are two questions that do not need precise answers, I pose them only to get you thinking!

An important extra property of any Font Resource that can be really useful to a typesetter (either human or software) is flexibility and the major advantage of modern fonts is that they have a lot more abilities than classical fonts. Equally important for flexible Font Resources is that they should be self-aware: they need to know about these abilities and how much flexibility they provide.

## 4 What can we ask of a Font Resource ?

A flexible Font Resource needs to be self-aware so that the font selection system of a typesetter (such as  $\LaTeX$ ’s NFSS) can ask it about its abilities. Typical modern fonts can be queried in this way by typesetting software such as  $X\TeX$  (although the interface is not very intuitive), so it will be possible to extend the NFSS to exploit their added flexibility.

## 5 Exemplary types of Font Resource

This is not a classification of types of Font Resource but merely some more-or-less mythical examples of the range of possibilities that are now, or maybe soon, available.

**simple** just stacks aligned glyphs in one writing direction (maybe ‘returns the advance width’)

**$\TeX$  tfm** ... adds (a fixed set of)

- kerns
- ligatures (mandatory, aesthetic, Knuthian)
- italic corrections

**AAT/OpenType** ... adds (the possibility of)

- (importantly) script plus language
- (usefully) choice of ligatures
- (for fun fonts) choice of swashes, etc.
- ... many other things (see Apple Advanced Typography [1] and the Microsoft/Adobe version [5])

**FreeType 2** ... use various font resource types [2]

**$\Omega_2$**  ... adds many more features (see the article by Yannis Haralambous in these proceedings)

**ParaType** ... adds line-breaking and justification (perhaps?)

## 6 The discussion

At this point some well-known gurus were dragged onto the platform to answer questions and provoke further discussion. We started off with some rhetorical questions that allowed the panelists to remove many of my misunderstandings. Particular thanks are due to Yannis Haralambous, Taco Hoekwater, Jonathan Kew & Arthur Reutenauer.

The questions:

- What is a Font Resource?
- Should there be a clear interface between Font Resource and the Typesetter?

The discussion ranged far and wide, drifting into political as well as technical areas. Here are some highlights.

- Clarification of the various parts of current font technology and their relationship to aspects of micro-formatting — including paragraph making!
- Font Resources may do a lot ... and the typesetter may need control too; BUT there is *no* clear division between them.
- Currently, middleware is important for typesetting, especially for complex scripts: examples are Pango [6] and Uniscribe [7].
- IMPORTANT: a modern typesetter should be able to use whatever Font Resources are available and not need specialised formats such as `.tfm` files.

## 7 The questions continue

Here are some further questions which are still open. I hope they will stimulate further discoveries leading to articles in *TUGboat*.

### 7.1 Questions for a Font Resource

- What can be asked about a visualisation?
- What can be asked about the abilities?
- How practical is it to query the tables in a modern Font Resource?
  - Is it inefficient?
  - Can all information be extracted?

### 7.2 Handling deficiencies in a Font Resource

How should a typesetting system best handle a modern font resource that does not explicitly provide all the information required by a typesetter (such as accurate vertical metrics or italic corrections)? Some possibilities, assuming that the extra information can be calculated or found elsewhere:

- Add to existing tables using standard methods: do these exist?
- Add extra tables in the Font Resource: is this feasible?
- Produce external tables that enhance/override internals: is this feasible?

Such activities also raise legal and moral questions about the licencing terms of modern font resources: should there be a distinction between the rendering information (the glyphs themselves), the metric information and the ‘use information’ (i.e., whether to use ligatures and other features).

Finally some questions for (or demands of) font designers:

- What tables are needed for high quality typesetting?
- How should the glyphs, metrics and other information be enhanced/extended/corrected?

## References

- [1] <http://www.apple.com/macosx/features/fontbook>
- [2] <http://www.freetype.org>
- [3] <http://www.ctan.org/tex-archive/macros/xetex/latex/fontspec>
- [4] <http://www.latex-project.org>
- [5] <http://www.adobe.com/uk/type/opentype>
- [6] <http://www.pango.org>
- [7] <http://www.microsoft.com/typography/developers/uniscribe>
- [8] <http://scripts.sil.org/xetex>