The `combofont` package

Ulrike Fischer[*]

May 27, 2017

1 Status: EXPERIMENTAL

This is a EXPERIMENTAL package.

It can disappear without notice e. g. if the luaotfload changes so that it no longer work, or if luatex changes, or if fontspec includes the code.

It is also possible that syntax and commands change in a incompatible way. So if you use it in a production environment: **You have been warned**

2 Introduction

In version 2.7. luaotfload supports combining characters from multiple fonts into a single virtualized one.

That means that one can build a font that takes e.g. the capital letters from a sans serif font and the lowercase letters from a serif font. Or a font that pulls in missing greek or cyrillic glyphs from another font.

The methods pulls in *only* glyphs. It is not suitable for every imaginable font combination – some drawbacks are described below – and one should use it with care. Nevertheless it is a quite neat extension of the tools to manipulate fonts.

The main problem with the examples in the luaotfload manual is that it creates fonts of a fix size. This means that they don't respond to command like `\large` or `\footnotesize`.

After trying around a bit and then asking a question (https://tex.stackexchange.com/questions/371647/call-a-luatex-combo-font-through-nfss) I got from David Carlisle the idea to use a `size`-Funktion which one define with `\DeclareSizeFunction` to inject the needed code to size the combo-font in a nfss-`\DeclareFontShape`-command.

[*] fischer@troubleshooting-tex.de

combofont.sty is the result.

3 Using combo fonts

To be able to use a combo font with standard LaTeX font commands you have to do two things:

1. Setup and describe the building of the combo font with `\setupcombofont`
2. Write `nfss`-declarations

3.1 Setup the combo font

`\setupcombofont``{`⟨*name*⟩`}{`⟨*comma list of basefonts*⟩`}{`⟨*comma list of ranges and code-points*⟩`}`

{⟨*name*⟩**}** is the name of the font. It should be some unique ascii-string without spaces. If you intent to define lots of fonts it would be a good idea to think about a sensible naming sheme. In the example here I simply used `combotest-regular` and `combotest-bold`.

{⟨*<comma list of basefonts>*⟩**}** This should be a list of font declarations you want to use to build your combo font

```
{
 {file:lmroman10-regular.otf:\combodefaultfeat} at #1pt,
 {file:lmsans10-regular.otf} at \fpeval{#1/10*15}pt,
 {file:cmunrm.otf}           at #1pt
}
```

Important points are:

Order of the fonts The first font is the main font which will receive the glyphs. So think carefully which font is should be and setup its font features correctly. combofont defines as a helper command `\combodefaultfeat` which sets `mode=node;script=latn;language=DFLT;+tlig;`.

Size declaration The font description should end with a size declaration line `at #1pt`. When processing the font `#1` will be replaced by the current font size. As you can see in the second font you can do calculations.

4 Call through nfss?

– Some Text with Capital Words! Eh bien, mon prince. Gênes et Lueques ne sont plus que des apanages, des поместья, de la famille Buonaparte? Some Text with Capital Words! Eh bien, mon prince. Gênes et Lueques ne sont plus que des apanages, des поместья, de la famille Buonaparte?

Some Text with Capital Words! Eh bien, mon prince. Gênes et Lueques ne sont plus que des apanages, des поместья, de la famille Buonaparte?

Some Text with Capital Words! Eh bien, mon prince. Gênes et Lueques ne sont plus que des apanages, des поместья, de la famille Buonaparte? Some Text with Capital Words! Eh bien, mon prince. Gênes et Lueques ne sont plus que des apanages, des поместья, de la famille Buonaparte?

Some Text with Capital Words fi Eh bien, mon prince. Gênes et Lueques ne sont plus que des apanages, des поместья, de la famille Buonaparte êßä