The philokalia package

Apostolos Syropoulos
366, 28th October Str.
GR-671 33 Xanthi, HELLAS
Email:asyropoulos@yahoo.com

2007/01/25 updated on 2017/09/20

Abstract

This document describes the functionality of the philokalia package, which has been designed to ease the use of the Philokalia-Regular OpenType font with XeLaTeX, as well as the OpenType features of this font.

1 Introduction

The philokalia package has been designed to ease the use of the Philokalia-Regular (henceforward it will specified simply as the font) OpenType font with XeLaTeX. The package provides two options: `global` and `local`. When the package is used with the `global` option the font is made the main font of the entire document. Also, this option provides support to typeset titling capitals. When the package is used with the `local` option, which is the default option, it provides the commands `\textphlk` and `\phkl` to enable typesetting of short passages. The first command takes one argument which is typeset using the font. The second command makes the font the main font of the current (local) scope. The complete package provides the following XeLaTeX files:

driver	produce the documentation
philokalia	the package itself
TUpkl	The Philokalia font shapes

2 About The Font

The font started as a project to digitize the typeface that was used to typeset the Philokalia books. For information regarding these books, the reader should visit either the following URL:

`http://orthodoxwiki.org/Philokalia`

or enter the word *philokalia* in the Amazon.com search box, for information about various editions of the books. The project was carried out by Apostolos Syropoulos and Ioannis Gamvets. Initially, the project was part of Ioannis Gamvetas's diploma thesis, but the resulting OpenType font is not part of this thesis work and it was developed by this author. The goal of the original thesis work was to develop Type 1 fonts and tools to typeset portions of the Philokalia books with Omega. Since, the original thesis work has not been completed yet (!), the idea of releasing an OpenType font emerged quite naturally. The font contains only Greek letters and it can be used to typeset any Greek text. However, since the shapes of the letters are not that obvious, here is a "translation"-table:

α	β	γ	δ	ε
ζ	η	ϑ	ι	κ
λ	μ	ν	ξ	ο
π	ρ	σ	ς	τ
υ	φ	χ	ψ	ω

One interesting feature of this font is that accents are placed *after* capital letters (see the sample below). Another interesting feature of the font is the great number of ligatures it includes. In fact, it includes more than 40 *historical* ligatures plus two *contextual* ligatures. When typing and these letters are not part of a word, then one gets the contextual ligature ϗ, which is the Greek ampersand. The same ligature can be obtained if we substitute *iota* with *iota with varia*. The table below shows all the historical ligatures included in the font:

ϫ		αὶ		ερ		γδ		κτ	
μτ		ϖ		ϑ		φ		τῆ	
ϗ		καὶ		σ		δῆ		ὁ	
ὅ		ὁ̓		ϰ		ϰ́		οὺ	
ὃ		ὃ̓		ὅ		αι		αῖ	
αί		αὶ		αἰ		αύ		ϙ	
εῖ		δξ		δ̓ξ		εί		εὶ	
εῖ		ὶω		ὶω		ὶῶ		ὶώ	
ωυ		ωύ		ϛ		∧	-	M̊	1M
μ̊	1	ϛ	1						

Notice that in order to get the symbols ∧ and ϛ one has to type a tonos and then a hyphen or the digit one, respectively. The symbols ∧ and ϛ have been used by the ancient Greek mathematician Diophantus to denote the subtraction operator[1] and an unknown quantity (similar to the x we use today in simple equations like $x + x^2 = 3$). Also, the symbols M̊ and μ̊ have been used by Diophantus to denote the unit (i.e., the number one). These symbol were not in the original Philokalia font, but they have been included here for reasons of completeness. The followin commands are provided for people who only want to typeset these symbols: `\dsubop`, `\dUnit`, `\dunit`, and `\dunknown`.

The font includes titling capitals that are accessible through the titling OpenType feature, which, however, is implicitly available. In particular, the command `\textinit`, which has two arguments, the first being the first letter of a word and the second being the rest of the word, can be used to typeset a drop capital. The following short passage from Aristotle's treatise *The Poetics* was typeset with the font so as to demonstrate the various features of it.

Περὶ ποιητικῆς αὐτῆς τε καὶ τῶν εἰδῶν αὐτῆς, ἥν τινα δύναμιν ἕκαστον ἔχει, καὶ πῶς δεῖ συνίστασθαι τοὺς μύθους εἰ μέλλει καλῶς ἕξειν ἡ ποίησις, ἔτι δὲ ἐκ πόσων καὶ ποίων ἐστὶ μορίων, ὁμοίως δὲ καὶ περὶ τῶν ἄλλων ὅσα τῆς αὐτῆς ἐστι μεθόδου, λέγωμεν ἀρξάμενοι κατὰ φύσιν πρῶτον ἀπὸ τῶν πρώτων.

Ἐποποιία δὴ καὶ ἡ τῆς τραγῳδίας ποίησις ἔτι δὲ κωμῳδία καὶ ἡ διθυραμβοποιητικὴ καὶ τῆς αὐλητικῆς ἡ πλείστη καὶ κιθαριστικῆς πᾶσαι τυγχάνουσιν ὖσαι μιμήσεις τὸ σύνολον· διαφέρουσι δὲ ἀλλήλων τρισίν, ἢ γὰρ τῷ ἐν ἑτέροις μιμεῖσθαι ἢ τῷ ἕτερα ἢ τῷ ἑτέρως καὶ μὴ τὸν αὐτὸν τρόπον.

Ὥσπερ γὰρ καὶ χρώμασι καὶ σχήμασι πολλὰ μιμοῦνταί τινες ἀπεικάζοντες (οἱ μὲν [20] διὰ τέχνης οἱ δὲ διὰ συνηθείας), ἕτεροι δὲ διὰ τῆς φωνῆς, οὕτω καὶ ταῖς εἰρημέναις τέχναις ἅπασαι μὲν ποιοῦνται τὴν μίμησιν ἐν ῥυθμῷ καὶ λόγῳ καὶ ἁρμονίᾳ, τούτοις δ' ἢ χωρὶς ἢ μεμιγμένοις· οἷον ἁρμονίᾳ μὲν καὶ ῥυθμῷ χρώμεναι μόνον ἥ τε αὐλητικὴ καὶ ἡ κιθαριστικὴ καὶ εἴ τινες [25] ἕτεραι τυγχάνουσιν ὖσαι τοιαῦται τὴν δύναμιν, οἷον ἡ τῶν συρίγγων, αὐτῷ δὲ τῷ ῥυθμῷ [μιμοῦνται] χωρὶς ἁρμονίας ἡ τῶν ὀρχηστῶν (καὶ γὰρ οὗτοι διὰ τῶν σχηματιζομένων ῥυθμῶν μιμοῦνται καὶ ἤθη καὶ πάθη καὶ πράξεις)·

[1] Diophantus did not explicitly specified negative numbers in his writings, although he was aware of them.

The first line of the code that was used to typeset the previous passage follows:

```
\setlanguage{ancientgreek}\textinit{Π}{ }          …
```

Notice that here we enable the `ancientgreek` hyphenation patterns to allows XeTeX to correctly hyphenate the text.

3 The package philokalia

First of all, we have to load a number of packages that are necessary for the correct use of the font. In addition, we use the lettrine package to typeset the titling capitals.

```
⟨*philokalia⟩
\RequirePackage{xltxtra}
\RequirePackage{lettrine}
\newsavebox{\Pb@x}
```

Unfortunately, we cannot use the lettrine package as it stands. We need to sligthly modify two macros. In particular, the macro `\Lettrine@height`, which computes the height of the titling capital, has to compute the height of the letter `x` of a font in order to carry out its computation. Since the font does not include this character, we had to modify the code so as to comp ute the height of the letter instead. Similarly, we had to modify the macro `\LettrineFont` because it was designed with the assumption that the main font of the document has the `X` character, which, again, is not valid for our case.

```
\def\Lettrine@height{%
   \@tempdima=\baselineskip
   \setlength{\L@height}{\theL@lines\@tempdima}%
   \ifnum\theL@lines>1
     \addtolength{\L@height}{-\@tempdima}%
   \fi
   \sbox{\L@tbox}{\LettrineTextFont "03B1}%%% Modified here
   \addtolength{\L@height}{\ht\L@tbox}%
   \addtolength{\L@height}{\L@oversize\L@height}}
\renewcommand*{\LettrineFont}{%
   \Lettrine@height
   \sbox{\L@tbox}{\LettrineFontHook\fontsize{\L@height}{\L@height}%
                  \selectfont "0391}%%% Modified here
   \@tempcntb=\ht\L@tbox
   \@tempcnta=\L@height
   \multiply\@tempcnta by 100
   \divide\@tempcntb by 100
   \divide\@tempcnta by \@tempcntb
   \advance\@tempcnta by -9999
   \ifnum\@tempcnta>0
     \def\@tempa{1.\the\@tempcnta}%
   \else
     \def\@tempa{1}%
   \fi
   \LettrineFontHook
   \fontsize{\@tempa\L@height}{\@tempa\L@height}%
   \selectfont
}
```

Now, we have to define the two options: global and local (the default one). In the first case, we define the command `\textinit` that is used to typeset drop capitals from the titling capitals included in the font. By default, the `titl` feature of the font is not enabled as this would mean that all paragraphs would start with these really huge titling capitals. Also, the font is made the default font for the entire document. Notice

that we use a box variable to store the letter that will appear as a drop capital. This is necessary in order to have in the box the titling capital and not the ordinary capital letter, or else the `\lettrine` command will fail to correctly compute the height and width of the letter.

```
\DeclareOption{global}{%
   \renewcommand{\rmdefault}{plk}%
   \DeclareFontFamily{TU}{plktitl}{}%
   \DeclareFontShape{TU}{plktitl}{m}{n}%
    {<-> "[Philokalia-Regular]/ICU:script=grek,+titl"}{}%
   \newcommand{\textinit}[2]{%
       \savebox{\Pb@x}{\usefont{TU}{plktitl}{m}{n} #1}%
       \lettrine[lines=3]{\usebox{\Pb@x}}{#2}}%
}
```

When the package is used with the default option, it provides two commands to typeset short passages of text. As is evident from the code that follows, the first macro is actually a definition and the second a command that can be used to typeset its argument with the font. Users should use the definition with care.

```
\DeclareOption{local}{%
   \def\phkl{\fontfamily{plk}\selectfont}%
   \newcommand{\textphlk}[1]{{\phkl #1}}%
}
\ExecuteOptions{local}
\ProcessOptions
```

The following commands are glyph access commands for the archaic mathematical symbols included in the font. They produce the symbols $\wedge$, $\mathring{\mathrm{M}}$, μ°, and $\mathfrak{z}$, respectively.

```
\def\dsubop{{\fontfamily{plk}\selectfont\char"018B}}
\def\dUnit{{\fontfamily{plk}\selectfont\char"018C}}
\def\dunit{{\fontfamily{plk}\selectfont\char"018D}}
\def\dunknown{{\fontfamily{plk}\selectfont\char"018E}}
```

⟨/philokalia⟩

4 The Font Definition File

Since there is only one font shape, there is not much work to do: we just need to specify the available font properties. We use the standard font encoding defined by package fontspec. In this version we use `TU` font encoding instead of `EU1`, which was predefined by the same package.

⟨*TUplk⟩

```
\DeclareFontFamily{TU}{plk}{}
\DeclareFontShape{TU}{plk}{m}{n}%
     {<-> "[Philokalia-Regular]/ICU:script=grek,+hlig,+clig:mapping=tex-text"}{
}
```

And then we define the various substitutions:

```
\DeclareFontShape{TU}{plk}{m}{sl}{<-> ssub * plk/m/n}{}
\DeclareFontShape{TU}{plk}{m}{it}{<-> ssub * plk/m/sl}{}
\DeclareFontShape{TU}{plk}{m}{sc}{<-> ssub * plk/m/n}{}
\DeclareFontShape{TU}{plk}{b}{n}{<-> ssub * plk/m/n}{}
\DeclareFontShape{TU}{plk}{b}{sl}{<-> ssub * plk/m/n}{}
\DeclareFontShape{TU}{plk}{b}{it}{<-> ssub * plk/m/n}{}
\DeclareFontShape{TU}{plk}{bx}{n}{<-> ssub * plk/b/n}{}
\DeclareFontShape{TU}{plk}{bx}{it}{<-> ssub * plk/b/sl}{}
\DeclareFontShape{TU}{plk}{bx}{sl}{<-> ssub * plk/b/sl}{}
```

⟨/TUplk⟩

Acknowledgements

First of all, I thank Ioannis Gamvetas for his collaboration back in the years 2002-2003. Also, I would like to thank George Williams, the designer of FontForge, the free font-editor, for all his valuable help, Adam Twardoch for explaining to me how to implement the contextual ligature, and Dimitrios Philippou for his comments and suggestions. Also, I thank Ioannis Vamvakas for letting me know that the previous version does not work with the latest version of package fontspec. Last, but certainly not least, I thank my son for his help!