The Design and Use of a Multiple-Alphabet Font with Ω

Yannis Haralambous
Atelier Fluxus Virus
187, rue Nationale,
59800 Lille, France.
Email: `yannis@pobox.com`

John Plaice
School of Computer Science and Engineering
University of New South Wales, Sydney 2052, Australia.
Email: `plaice@acm.org`

1998

Abstract

The Ω project aims to oer open and exible means for typesetting dierent scripts. By working at several dierent levels, it is possible to oer natural support for dierent languages and scripts, and strictly respect typographical traditions for each of them. This is illustrated with a large PostScript Type 1 font for the commonly used left-to-right non-cursive alphabets, called `omlgc` (Ω Latin-Greek-Cyrillic). This font, which more than covers the Unicode sections pertaining to those alphabets, as well as those of IPA, Armenian, Georgian and Tinagh (Berber), is builtvirtuallyout of smaller glyph banks. The Ω typesetting engine, based on that of TEX, is used to print documents using this font. The characters can be accessed either directly, or through the use of lters, called Ω Typesetting Processes (ΩTPs), which are applied to the input stream.

1 Introduction

Typesetting in dierent scripts and languages is a problem that is arguably solved today, either by software giants that adapt their products to what they consider to be local typesetting specications, or by individual users brewing their own limited, but practical, systems for their own needs.

However, typesetting in dierent scripts and languages*while still keeping the quality of traditional typography*and having an open system that can be adapted to any language and script, without loss of power or quality, is still an unachieved goal. The Ω project aims to solve precisely this problem.

The Ω project consists of the design of fonts for dierent languages and scripts, as well as of an engine that can be used for all possible situations.

As a result, in order to ensure eciency and openness, one can work at dierent levels, each one adapted to a specic aspect of multilingual typesetting. These levels correspond to methods used in the Ω system. They will be illustrated through the `omlgc` font, designed forcurrentlyLatin, Greek, Cyrillic, IPA, Armenian, Georgian and Tinagh.

This paper was typeset by the Ω engine, using fonts `omlgc` and `omarab`.

2 Ω Methods: an Overview

When developing an Ω typesetting convention for a given language, one can work at the following levels:

1. *The font level*. A font is a container of *glyphs* needed to typeset in a given language. These glyphs may or may not correspond to the graphical units of a script, whether these are called letters, ideograms

or otherwise; these glyphs may be parts of graphical units, combinations thereof, or new independent symbols.

The glyphs must be provided to the screen previewer and as well as to the printing engine. Ω itself is not concerned with these glyphs: as with TEX, Ω works only with metrics, not necessarily those for the fonts actually containing the glyphs.

The font level is the lowest development level, in the sense that glyphs are indivisible units that can be used in other higher level structures but cannot be dynamically modied by Ω itself.

2. *The virtual font level*. Once we have the glyphs we need, we combine them to form what is normally considered for a language to be a gramatically correct script entity (a ring accent alone is of no use, but å is part of the grammar for the Swedish language).

   A virtual font character is a combination of glyphs taken from several real fonts, or of other virtual font characters. In the å example above, the glyphs of the ring and of the letter a can actually be taken from entirely dierent fonts, in dierent formats (bitmap, PostScript, TrueType, etc.).

   Passing from real fonts to virtual fonts is essentially a matter of optimization of storage and memory: taking the seven Greek vowels, the three accents, two spirits, diaeresis, subscript iota and macron/breve diacritics (that makes 16 glyphs) we produce 336 (!) virtual glyphs. Describing a character in a virtual font is hundreds of times smaller than the PostScript code describing a hypothetical similar glyph.

   Virtual fonts are directly used by Ω: they can have up to 2^{16} positions and 2^{32} kerning pairs. The les created by Ω can be processed (previewed, printed) by utilities we have adapted; if the user has to use his/her own utilities which are not 16-bit clean, there is a tool to devirtualize the les, i.e. replace virtual fonts by the underlying real fonts, and make them as standard as TEX output les.

3. *The ΩTP level*. When Ω reads a document, it rst tokenizes it and expands commands and then forwards the data to the typesetting engine. Between these two steps we introduce an arbitrary number of lters, which we call Ω Typesetting Processes (ΩTPs). These are written in a Lex-like syntax, are loaded while reading the document, and can be activated and de-activated dynamically.

   A typical example for an ΩTP is contextual analysis of Arabic. Of course, this operation can also been done by macros, say using LATEX commands; but an ΩTP will do it much faster, it will avoid conicts with other LATEX commands, and it will be much easier to create and congure.

   Contextual analysis of Arabic is a typical example of a script property that is low level and that should not use *any* macros or other high-level structures. In the case of the Latin script one would not expect a user to constantly think of and ligatures and place them manually: it would be very bad strategy to use an ligature command; in Arabic this property is of the same nature, and hence should remain completely transparent.[1]

   But eciency should not only be limited to speed of typesetting: sometimes it is very important to also optimize the conguration and customization time and eort. A typical example is the management of encodings, whether input or output: by using a universal encodingwhich we call it *Unicode++*, since it is a superset of Unicodeeas intermediate step for our ΩTPs, every new input encoding requires only a foo → Unicode++ ΩTP and every new font encoding only a Unicode++ → foo one; these are signicantly easier to make than if one had to rewrite all processing steps, including contextual analysis and other bells and whistles.

4. *The hyphenation and sorting engines*. Hyphenation and sorting rules are grammatical properties of a language: they have nothing to do with typographical aspects, input methods, font encodings, etc. They only have to be performed on the Unicode++ level, i.e. the most abstract one. Once dened at this level, they can immediately be used with every input and output encoding. Ω hyphenation and sorting engines are still under development: for the time being, Ω does not sort, and hyphenation takes place as in TEX, using the (virtual) font encoding.

[1]And undoubtedly this would be the case if the *lingua franca* of computer science was Arabic or Urdu, and not English.

5. *The macro-command level.* LaTeX is a terric construction, featuring commands at multiple levels: TeX primitives, plain TeX commands, internal LaTeX commands, higher LaTeX commands and environments, and user-dened commands and environments. Our goal is to keep all script-dependent processing independent of the LaTeX macro level; because of this goal, every LaTeX package is usable in any script and language, and both Ω package and LaTeX package developers are able to work independently, while still producing mutually compatible software.

   By doing this, we also allow ourselves in the future to use a dierent typesetting engine than that found in TeX, without having to completely redo our system. The ΩTPs might need modication, but the basic structure would remain the same.

3 Level 1: Designing glyphs

For us, a glyph is simply a character from a PostScript, TrueType or Metafont font; so why call it such? Because we want to make the distinction between the images and the combinations of images, the latter of which Ω will use as characters for typesetting.

For our `omlgc` font, the glyphs are produced so that the glyphs for dierent alphabets look similar. With the exception of Tinagh, the history of the dierent alphabets making up this font is strongly interrelated; as such, it has been possible to design a single font with a characteristic feel. For Tinagh, the script has evolved less than the others, so we are doing creative experimentationonly time will tell if it is successful, i.e. native speakers accept it.

For example, the following two words Ըus Պալաաանեանի are Armenian. Although the traditional Armenian script does not look so rounded, it is considered common practice today to typeset books in this manner, and Armenians are not in the least suprised.

The `omlgc` font is a modern looking font. One question that might arise is what to do with classical versions of these alphabets. For ancient Greek, it is common to use modern typefaces, even though there are typefaces specically made for ancient Greek, such as Porson or Greek Sans Serif. Can this also be done for Coptic and Slavonic excerpts, without breaking the homogeneity of the surrounding modern font? We have tried to do this for Slavonicincluding not only letters and accents found in Unicode, but also all the other diacritics and symbols needed for Old Church Slavonicand our example looks like this: Ѿвѣща̀ є҆̓ і҆и҃съ и҆ речѐ є҆̓. The result is not particularly exiting to look at, but that is the point: it should t with the surrounding text, not look exotic.

Finally, a lot of work has gone into the design of IPA glyphs, which are problematic since many of them are basically font variations (italics, small caps, etc.) of other glyphs, thereby making it a nightmare to design fonts including IPA that have variations.

As we will see in the next section, a big part of the `omlgc` font is obtained by combining a limited number of glyphs. This corresponds to a grammatical reality: it is quite natural to assume that placing diacritics does not aect the shapes of either the base character or the diacritic itself. Often this is true, but there are times when typographical quality requires special shapes.

For example, one can imagine an h with a lower vertical stem, so that the accent of the Esperanto character ĥ is not excessively high. This is really a matter of taste, and we include these kinds of characters as variant forms in our font, and leave the decision whether to use them to the user.

Another similar example comes from the use of diaereses over the Greek vowels iota and upsilon, whether lowercase or uppercase; the distance between the two dots of the diaeresis depends on the letter underneath: compare ϊ, ϋ, Ϊ, Ϋ, while in the Latin script the diaeresis (Umlaut, tréma) always has the same width: ï, ö, , etc.

4 Level 2: Combining Glyphs into Virtual Fonts

A character of a virtual font can be a combination of characters of other fonts (whether real, in which case we actually have glyphs, or again virtual, in which case we have sub-combinations etc.), of black boxes of arbitrary height and width, and of PostScript code. (Since the latter feature is not implemented in all DVI drivers, one should refrain from using it, at least for the moment.)

We have chosen to work intensely with virtual fonts for two reasons: rst because by combining glyphs we can optimize space (and space management is crucial when you deal with 16-bit fonts), and second, to be able to use 16-bit fonts without re-writing all of the worlds drivers for TEXs DVI format (the underlying real fonts are 8-bit only, so that a devirtualized 16-bit font becomes 8-bit and can be processed by any decent DVI driver).

Virtual fonts are built using a (portable) Perl script, which reads the conguration le `omlgc.cfg`. This le contains one entry for each character of the virtual font. This line consists of (a) a 4-digit hexadecimal number, designating the characters position in the font, (b) an operator, (c) one or more character names, depending on the operator, (d) a certain number of optional operators and values.

4.0.1 The `N` operator.

The `N` stands for NAME, and simply means that the string following the operator is the (PostScript) name of one or more characters in some of the fonts loaded. For example,

```
03A5 N Upsilon
```

means that at position 03A5 of the virtual font, we have placed the Greek letter capital Upsilon ϒ. The PostScript names we have used try to be as standard as possible. However, most of the time there is no standard, so we just have to invent names.

The same glyph can be used for several font positions: for example, the Croatian Dze Đ has exactly the same shape as the Icelandic Eth Ð: we use the same glyph, and hence the same PostScript name. Nevertheless we try to optimize the use of glyphs so that one can typeset in one script without necessarily loading the PostScript fonts for other scripts: for example, although the capital Latin A has the same shape as the Greek capital Alpha, we use two dierent glyphs in two dierent PostScript fonts, so that when typesetting Greek one can avoid loading the Latin PostScript font. These considerations will be irrelevant when we will be able to use 16-bit monobloc PostScript fonts; for the moment, this is not part of the Adobe Type 1 font specication.

There are several options we can use: `#KRN`, `#KRNLEFT`, `#KRNRIGHT`, `#HOFFSET`, `#VOFFSET`.

The rst three concern kerning: we can state that a given character has the same kerning behavior as some other character, which we give by name. For example, ç will be kerned exactly like the letter c: everytime there is a kerning pair in the kernings le using c a new kerning pair will be dened, using ç instead of c (and if there is a c-c kerning pair, three new kerning pairs will be dened: ç-c, c-ç, ç-ç). Sometimes we kern a letter like some given letter on the right and like some other letter on the left: this is typically the case of ligatures: æ will be kerned as a on the left, and as e on the right; in that case, we use the `#KRNLEFT` and `#KRNRIGHT` operators:

```
00C6 N AE #KRNLEFT=A #KRNRIGHT=E
00E6 N ae #KRNLEFT=a #KRNRIGHT=e
0110 N Eth #KRN=D
```

The operators `#HOFFSET` and `#VOFFSET` will oset the glyph without aecting the box of the character.

4.0.2 The `XHAC` and `CHAC` operators.

These operators will place diacritics over letters, which are considered to have the same height as the lowercase letter x (x-height) or the one of an LGC uppercase letter (cap-height). The idea is that the height of letters can uctuate: a round letter, like o, is slighty higher than a at letter, like z, to counterbalance a well-known optical eect. The height of accents over these letters must be the same, even if they arent exactly of x-height, or cap-height: take for example ó and ú; if we would take the real height of these letters, the accent on the former would be slightly higher than the one on the latter.

How is this accent placed precisely? The Perl utility centers the bounding box of the accent over the bounding box of the letter, with a xed distance of EPSILON (a global value) between the lower boundary of the accent and either the x-height or cap-height of the font (those two are also global values provided with the utility).

Available options are `#KRN`, `#KRNLEFT`, `#KRNRIGHT`, `#LETTERLIKE`, `#ACCENTLIKE` and `#LETTERREVLIKE`.

The options #LETTERLIKE and #ACCENTLIKE allow us to use given glyphs, with the metrics of other glyphs. These options are extremely important in certain cases. A typical example is Vietnamese: the letter o with hook ơ is signicantly wider than the plain o, nevertheless accents have to be centered on the o part of the letter: , , , . This trick allows us to correctly place an accent on the vertical stem of a b or an h: , , ; to obtain this result, we simply ask the accent to be placed as if the letter was an l:

```
0603 CHAC b dot #LETTERLIKE=l
0623 CHAC h dot #LETTERLIKE=l
0627 CHAC h dieresis #LETTERLIKE=l
```

We have the same functionality with accents: using the #ACCENTLIKE operator we can place an accent as if it was some other accent. The typical example is again Vietnamese, where there are combined accents circumex + grave, circumex + acute, which have to be centered with respect to the middle axis of the circumex (and hence as if there were no acute or grave accent):

```
06D0 CHAC O circumflexacute #KERN=O #ACCENTLIKE=circumflex
06D1 XHAC o circumflexacute #KERN=o #ACCENTLIKE=circumflex
06D2 CHAC O circumflexgrave #KERN=O #ACCENTLIKE=circumflex
06D3 XHAC o circumflexgrave #KERN=o #ACCENTLIKE=circumflex
06D4 CHAC O circumflexhook #KERN=O #ACCENTLIKE=circumflex
06D5 XHAC o circumflexhook #KERN=o #ACCENTLIKE=circumflex
06D6 CHAC O circumflextilde #KERN=O #ACCENTLIKE=circumflex
06D7 XHAC o circumflextilde #KERN=o #ACCENTLIKE=circumflex
```

Another example is Slavonic, with letters such as ѭ, where the accent has to be placed on the right part of the ligature, as in ѩ́. This means that we should use the metrics of a given character, justied on the right of our box: this is the rôle of the #LETTERREVLIKE operator.

4.0.3 The ADJ operator.

This operator allows us to concatenate two characters, using a box with width equal to the sum of the widths of the two boxes, $\pm$ the eventual kerning between those characters, and height/depth the maximum height/depth of the two characters. We rst wanted to use that operator for the Croatian digraphs Lj, Nj, etc. but then decided that the whole idea of having code positions for these digraphs was so silly that we could very well do without. The operator nevertheless proved very useful for cases such as the Greek capital vowels with accent Ά, Έ, Ή, Ί, Ό, Ύ, Ώ.

This operator takes a special option, #MOVELEFT, which allows us to overide a kern between two characters and move the second one horizontally, together with its box.

5 Levels 3 and 4: ΩTPs and LaTeX Macros

Once the structure of fonts is well organized, we use ΩTPs for low level script- or language-dependent operations and macros for higher level operations. To give an idea of the power of ΩTPs, consider the following Latin transcription for Berber:

```
Tifinagh, d--tira timezwura n .imazighen.
Llant di tmurt--nnegh dat tira n ta.erabt d--tla.tinit.
Nnulfant--edd dat .imir n ugellid Masinisen. .Imazighen n
.imir--en, ttarun--tent ghefi.zra, degg .ifran, ghef .igduren,
maca tiggti ghef i.zekwan : ttarun fell--asen .isem n umettin,
d wi--t--ilan, d wayen yexdem di tudert--is akken ur t ttettun
.ina.tfaren.
```

This piece of text can be printed using the Latin script, as in:

> Tinaɣ, dtira timezwura n imaziɣen. Llant di tmurtnneɣ dat tira n taɛrabt dtlainit. Nnulfantedd dat imir n ugellid Masinisen. Imaziɣen n imiren, ttaruntent ɣera, degg ifran, ɣef igduren, maca tiggti ɣef iekwan : ttarun fellasen isem n umettin, d witilan, d wayen yexdem di tudertis akken ur t ttettun inafaren.

or in the original Tifinagh script, as in:

> X≶ΞΙ≶I·⁝, ΛX≶O· X≶ΣЖ≕:O· I ≶Σ·Ж≶⁝I. ⵂⵂ·IX Λ≶ XΣ:OXII⁝ Λ·X X≶O· I X·O·ΘX ΛXⵂ·ⴺ≶I≶X. II:ⵂΞΣ·IXΛΛ Λ·X ≶Σ≶O I :Xⵂⵂ≶Λ Σ·Ⲙ≶I≶ⲘI. ≶Σ·Ж≶⁝I I ≶Σ≶OI, XX·O:IXIX ⁝ΞΣ≶ⵋO·, ΛXX ≶ΞΣO·I, ⁝ΞΣ ≶XΛ:OI, Σ·G· X≶XXX≶ ⁝ΞΣ ≶ⵋ⇒≕·I : XX·O:I ΞΣⵂⵂ·ⲘI ≶ⲘΣ I :ΣXX≶I, Λ ≕≶X≶ⵂ·I, Λ ≕·ΣI Σ⁝⁝ΛΣ Λ≶ X:ΛOX≶Ⲙ ·⇒⇒I :O X XXXX:I ≶I·ႬΞΣ·OI.

In fact, Berber can also be written right-to-left in the Arabic script (font omarab):

> تيفيناغ، دـتيرا تيمزوّرا ن ئمازيغن. لانت دي تمورتـنّغ دات تيرا ن تاعرابت دـتلاطينيت. نّولفانتـدّ دات ئمير ن وقلّيد ماسينيسن. ئمازيغن ن ئميرـن، تّارونـتنت غفيزرا، دقّ ئفران، غف ئقدورن، ماشا تيقّتي غف يزكوان : تّارون فلّـاسن ئسم ن ومتّين، د ويـتـيلان، د واين يخدم دي تودرتـيس اكّن ور ت تّتّون ئناطفارن.

In the latter case, a font and direction change were necessary. Otherwise, only the output ΩTP needed to be changed for the three cases. Of course, LaTeX macros can be used to encapsulate these changes.

6 Discussion

As this document shows, the omlgc font is now usable for typesetting, using Ω, any language that uses the Latin, Greek, Cyrillic and Tifinagh alphabets, and will soon be ready for the other alphabets in this group.

The omlgc font is not the only multi-script font that has been developed. For example, PLAN 9 [1], Bitstream [2], and Windows [13] all have their own fonts, based on the Unicode encoding.

What does distinguish omlgc from the others is the emphasis that has been placed on typesetting. It is assumed right from the beginning that the set of font positions is much greater than the set of positions in, say, the Unicode definition.

This can best be visualized in Figures 3 and 4, given after the references. For the Greek page, we show the complete set of vowels, with all spirits and breathings, with the macron diacritic, used to denote a long vowel in classical Greek. None of these presentation forms appear in the Unicode specification---hence do not appear in the above fonts---but they are absolutely necessary for typesetting Greek.

Similarly, the Cyrillic page shows all the vowels, with both grave (old style) and acute (new style) accents to place emphasis. Once again, none of these forms are in the Unicode specification, but they are needed for typesetting texts in Russian.

But having these extra forms is of no cost, since they are all virtual characters (hence no extra memory), which can be accessed as needed in a text using the ΩTPs.

We complete the paper with four examples. We begin with three versions (roman, italic, bold) of a Greek text ΠΟΛΛΕΣ ΦΟΡΕΣ ΤΗΝ ΝΥΚΤΑ, by Andreas Embiricos. We follow with three pages of the omlgc font.

References

[1] C. A. Bigelow and K. Holmes. The design of a Unicode font. *Electronic Publishing* 6(3):289--305, 1993.

[2] Bitstream Inc. Bitstream Cyberbit. http://www.bitstream.com/cyberbit.htm.

[3] Y. Haralambous and J. Plaice. First applications of Ω: Greek, Arabic, Khmer, Poetica, ISO 10646/Unicode, etc. *TUGboat*, 15(3):344--352, 1994.

[4] Y. Haralambous and J. Plaice. Ω, a TEX extension including Unicode and featuring lex-like filtering processes. In Wlodek Bzyl and Tomasz Plata-Przechlewski, editors, *Proceedings of the European TEX Conference*, pages 153--166, Gdansk, Poland, 1994. GUST.

[5] Y. Haralambous and J. Plaice. ΩTimes and ΩHelvetica Fonts under Development: Step One. *TUGboat*, 17(3):126--146, 1996.

[6] Y. Haralambous and J. Plaice. A font for typesetting large mathematical symbols in Ω. *EuroTEX'98*, Saint-Malo, France, 29--31 March 1998.

[7] J. Plaice and Y. Haralambous. Typesetting French, German and English in Ω. *EuroTEX'98*, Saint-Malo, France, 29--31 March 1998.

[8] Y. Haralambous, J. Plaice, and J. Braams. Never again active characters! Ω-Babel. *TUGboat*, 16(4):418--427, 1995.

[9] Y. Haralambous, J. Plaice, and A. Picheral (trans.). Ω, une extension de TEX incluant UNICODE et des filtres de type Lex. *Cahiers GUTenberg*, 20:55--80, 1995. Translation of refereed conference publication [4].

[10] J. Plaice. Language-dependent ligatures. *TUGboat*, 14(3):271--274, 1993.

[11] J. Plaice. Progress in the Omega project. *TUGboat*, 15(3):320--324, 1994.

[12] J. Plaice and Y. Haralambous. The latest developments in Ω. *TUGboat*, 17(3):181--183, 1996.

[13] Microsoft Inc. True Type in Windows 95. `http://www.microsoft.com/truetype/tt/win95tt.htm`.

ΠΟΛΛΕΣ ΦΟΡΕΣ ΤΗΝ ΝΥΚΤΑ
᾿Ανδρέας ᾿Εμπειρῖκος

῞Οσοι ἀπὸ σᾶς γυρίζετε τὴν νύκτα μέσ᾿ στοὺς δρόμους, ἀμέριμνοι ἢ σκεπτικοί, τὴν ἄνοιξι, κατὰ τὴν ἐποχὴ τοῦ ᾿Επιταφίου Θρήνου, ἢ ἐκεῖ κοντὰ στὶς ὧρες τὶς χαρούμενες ποὺ ὁδηγοῦν στὴν θριαμβευτικὴν τὴν ἄνωσιν ποὺ πάει νὰ γίνῃ Πάσχα, πρὶν ἀκουσθοῦν οἱ ἀναστάσιμες καμπάνες, καί, ἀκόμη περισσότερο, τὶς νύκτες τοῦ καλοκαιριοῦ στοὺς δρόμους τοὺς ὀνειρικοὺς τοῦ σκοτεινοῦ Λονδίνου, στοὺς ἄλλους τοὺς πλατεῖς ἢ τοὺς στενούς, ποὺ ἐκτείνονται γύρω ἀπὸ τὸν Μόσχοβα στὴ Μόσχα, ἢ στὰς ὁδοὺς τῆς κάτασπρης ᾿Αθήνας, σέ δορυάλωτες στιγμὲς τῆς θλίψεως, ἢ σὲ ἀφρόεσσες στιγμὲς εὐδαιμονίας, ὅταν παράθυρα καὶ ἐξώφυλλα χαίνουν διάπλατα ἀνοικτὰ γιὰ νὰ δεχθοῦν δροσιὰ καὶ μῦρα, ὅσοι ἀπὸ σᾶς νύκτωρ γυρίζετε στοὺς δρόμους πανευτυχεῖς ποὺ ἐκσπερματίσατε, ἢ δυστυχεῖς ποὺ κάποια γυναῖκα δὲν ἔστερξε νὰ σᾶς δεχθῇ καὶ δὲν ἐστάθη, λίγο ἂν προσέξετε, θὰ ἀκούσετε πολλά, ὅσα στήν τύρβη τῆς ἡμέρας δύσκολον εἶναι νὰ ἀκουσθοῦν.

῞Οσοι ἀπὸ σᾶς γυρίζετε τὴν νύκτα μέσ᾿ στοὺς δρόμους, ἀμέριμνοι ἢ σκεπτικοί, τὴν ἄνοιξι, κατὰ τὴν ἐποχὴ τοῦ ᾿Επιταφίου Θρήνου, ἢ ἐκεῖ κοντὰ στὶς ὧρες τὶς χαρούμενες ποὺ ὁδηγοῦν στὴν θριαμβευτικὴν τὴν ἄνωσιν ποὺ πάει νὰ γίνῃ Πάσχα, πρὶν ἀκουσθοῦν οἱ ἀναστάσιμες καμπάνες, καί, ἀκόμη περισσότερο, τὶς νύκτες τοῦ καλοκαιριοῦ στοὺς δρόμους τοὺς ὀνειρικοὺς τοῦ σκοτεινοῦ Λονδίνου, στοὺς ἄλλους τοὺς πλατεῖς ἢ τοὺς στενούς, ποὺ ἐκτείνονται γύρω ἀπὸ τὸν Μόσχοβα στὴ Μόσχα, ἢ στὰς ὁδοὺς τῆς κάτασπρης ᾿Αθήνας, σέ δορυάλωτες στιγμὲς τῆς θλίψεως, ἢ σὲ ἀφρόεσσες στιγμὲς εὐδαιμονίας, ὅταν παράθυρα καὶ ἐξώφυλλα χαίνουν διάπλατα ἀνοικτὰ γιὰ νὰ δεχθοῦν δροσιὰ καὶ μῦρα, ὅσοι ἀπὸ σᾶς νύκτωρ γυρίζετε στοὺς δρόμους πανευτυχεῖς ποὺ ἐκσπερματίσατε, ἢ δυστυχεῖς ποὺ κάποια γυναῖκα δὲν ἔστερξε νὰ σᾶς δεχθῇ καὶ δὲν ἐστάθη, λίγο ἂν προσέξετε, θὰ ἀκούσετε πολλά, ὅσα στήν τύρβη τῆς ἡμέρας δύσκολον εἶναι νὰ ἀκουσθοῦν.

῞Οσοι ἀπὸ σᾶς γυρίζετε τὴν νύκτα μέσ᾿ στοὺς δρόμους, ἀμέριμνοι ἢ σκεπτικοί, τὴν ἄνοιξι, κατὰ τὴν ἐποχὴ τοῦ ᾿Επιταφίου Θρήνου, ἢ ἐκεῖ κοντὰ στὶς ὧρες τὶς χαρούμενες ποὺ ὁδηγοῦν στὴν θριαμβευτικὴν τὴν ἄνωσιν ποὺ πάει νὰ γίνῃ Πάσχα, πρὶν ἀκουσθοῦν οἱ ἀναστάσιμες καμπάνες, καί, ἀκόμη περισσότερο, τὶς νύκτες τοῦ καλοκαιριοῦ στοὺς δρόμους τοὺς ὀνειρικοὺς τοῦ σκοτεινοῦ Λονδίνου, στοὺς ἄλλους τοὺς πλατεῖς ἢ τοὺς στενούς, ποὺ ἐκτείνονται γύρω ἀπὸ τὸν Μόσχοβα στὴ Μόσχα, ἢ στὰς ὁδοὺς τῆς κάτασπρης ᾿Αθήνας, σέ δορυάλωτες στιγμὲς τῆς θλίψεως, ἢ σὲ ἀφρόεσσες στιγμὲς εὐδαιμονίας, ὅταν παράθυρα καὶ ἐξώφυλλα χαίνουν διάπλατα ἀνοικτὰ γιὰ νὰ δεχθοῦν δροσιὰ καὶ μῦρα, ὅσοι ἀπὸ σᾶς νύκτωρ γυρίζετε στοὺς δρόμους πανευτυχεῖς ποὺ ἐκσπερματίσατε, ἢ δυστυχεῖς ποὺ κάποια γυναῖκα δὲν ἔστερξε νὰ σᾶς δεχθῇ καὶ δὲν ἐστάθη, λίγο ἂν προσέξετε, θὰ ἀκούσετε πολλά, ὅσα στήν τύρβη τῆς ἡμέρας δύσκολον εἶναι νὰ ἀκουσθοῦν.

Figure 1: Roman, italic and bold versions of a Greek text.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0600	Ḁ	ḁ	Ḃ	ḃ	Ḅ	ḅ	Ḇ	ḇ	Ḉ	ḉ	Ḋ	ḋ	Ḍ	ḍ	Ḏ	ḏ
0610	Ḑ	ḑ	Ḓ	ḓ	Ḕ	ḕ	Ḗ	ḗ	Ḙ	ḙ	Ḛ	ḛ	Ḝ	ḝ	Ḟ	ḟ
0620	Ḡ	ḡ	Ḣ	ḣ	Ḥ	ḥ	Ḧ	ḧ	Ḩ	ḩ	Ḫ	ḫ	Ḭ	ḭ	Ḯ	ḯ
0630	Ḱ	ḱ	Ḳ	ḳ	Ḵ	ḵ	Ḷ	ḷ	Ḹ	ḹ	Ḻ	ḻ	Ḽ	ḽ	Ḿ	ḿ
0640	Ṁ	ṁ	Ṃ	ṃ	Ṅ	ṅ	Ṇ	ṇ	Ṉ	ṉ	Ṋ	ṋ	Ṍ	ṍ	Ṏ	ṏ
0650	Ṑ	ṑ	Ṓ	ṓ	Ṕ	ṕ	Ṗ	ṗ	Ṙ	ṙ	Ṛ	ṛ	Ṝ	ṝ	Ṟ	ṟ
0660	Ṡ	ṡ	Ṣ	ṣ	Ṥ	ṥ	Ṧ	ṧ	Ṩ	ṩ	Ṫ	ṫ	Ṭ	ṭ	Ṯ	ṯ
0670	Ṱ	ṱ	Ṳ	ṳ	Ṵ	ṵ	Ṷ	ṷ	Ṹ	ṹ	Ṻ	ṻ	Ṽ	ṽ	Ṿ	ṿ
0680	Ẁ	ẁ	Ẃ	ẃ	Ẅ	ẅ	Ẇ	ẇ	Ẉ	ẉ	Ẋ	ẋ	Ẍ	ẍ	Ẏ	ẏ
0690	Ẑ	ẑ	Ẓ	ẓ	Ẕ	ẕ	ẖ	ẗ	ẘ	ẙ	ẚ	ẛ				
06A0	Ạ	ạ	Ả	ả	Ấ	ấ	Ầ	ầ	Ẩ	ẩ	Ẫ	ẫ	Ậ	ậ	Ắ	ắ
06B0	Ằ	ằ	Ẳ	ẳ	Ẵ	ẵ	Ặ	ặ	Ẹ	ẹ	Ẻ	ẻ	Ẽ	ẽ	Ế	ế
06C0	Ề	ề	Ể	ể	Ễ	ễ	Ệ	ệ	Ỉ	ỉ	Ị	ị	Ọ	ọ	Ỏ	ỏ
06D0	Ố	ố	Ồ	ồ	Ổ	ổ	Ỗ	ỗ	Ộ	ộ	Ớ	ớ	Ờ	ờ	Ở	ở
06E0	Ỡ	ỡ	Ợ	ợ	Ụ	ụ	Ủ	ủ	Ứ	ứ	Ừ	ừ	Ử	ử	Ữ	ữ
06F0	Ự	ự	Ỳ	ỳ	Ỵ	ỵ	Ỷ	ỷ	Ỹ	ỹ						

Figure 2: Unicode Row 1E: Latin Extended Additional.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0800																
0810	ᾱ̔	ᾱ̓	ᾱ̀	ᾱ́	ᾱ͂	ᾱ̑	ᾱ̔̀	ᾱ̔́	ᾱ̔͂	ᾱ̔̑	ᾱ̓̀	ᾱ̓́	ᾱ̓͂	ᾱ̓̑		
0820	ᾱ̔ͅ	ᾱ̓ͅ	ᾱ̀ͅ	ᾱ́ͅ	ᾱ͂ͅ	ᾱ̑ͅ	ᾱ̔̀ͅ	ᾱ̔́ͅ	ᾱ̔͂ͅ	ᾱ̔̑ͅ	ᾱ̓̀ͅ	ᾱ̓́ͅ	ᾱ̓͂ͅ	ᾱ̓̑ͅ		
0830	ε̄̔	ε̄̓	ε̄̀	ε̄́	ε̄͂	ε̄̑	ε̄̔̀	ε̄̔́	ε̄̔͂	ε̄̔̑	ε̄̓̀	ε̄̓́	ε̄̓͂	ε̄̓̑		
0840	η̄̔	η̄̓	η̄̀	η̄́	η̄͂	η̄̑	η̄̔̀	η̄̔́	η̄̔͂	η̄̔̑	η̄̓̀	η̄̓́	η̄̓͂	η̄̓̑		
0850	ῃ̄̔	ῃ̄̓	ῃ̄̀	ῃ̄́	ῃ̄͂	ῃ̄̑	ῃ̄̔̀	ῃ̄̔́	ῃ̄̔͂	ῃ̄̔̑	ῃ̄̓̀	ῃ̄̓́	ῃ̄̓͂	ῃ̄̓̑		
0860	ῑ̔	ῑ̓	ῑ̀	ῑ́	ῑ͂	ῑ̑	ῑ̔̀	ῑ̔́	ῑ̔͂	ῑ̔̑	ῑ̓̀	ῑ̓́	ῑ̓͂	ῑ̓̑		
0870	ο̄̔	ο̄̓	ο̄̀	ο̄́	ο̄͂	ο̄̑	ο̄̔̀	ο̄̔́	ο̄̔͂	ο̄̔̑	ο̄̓̀	ο̄̓́	ο̄̓͂	ο̄̓̑		
0880	ῡ̔	ῡ̓	ῡ̀	ῡ́	ῡ͂	ῡ̑	ῡ̔̀	ῡ̔́	ῡ̔͂	ῡ̔̑	ῡ̓̀	ῡ̓́	ῡ̓͂	ῡ̓̑		
0890	ω̄̔	ω̄̓	ω̄̀	ω̄́	ω̄͂	ω̄̑	ω̄̔̀	ω̄̔́	ω̄̔͂	ω̄̔̑	ω̄̓̀	ω̄̓́	ω̄̓͂	ω̄̓̑		
08A0	ῳ̄̔	ῳ̄̓	ῳ̄̀	ῳ̄́	ῳ̄͂	ῳ̄̑	ῳ̄̔̀	ῳ̄̔́	ῳ̄̔͂	ῳ̄̔̑	ῳ̄̓̀	ῳ̄̓́	ῳ̄̓͂	ῳ̄̓̑		
08B0																
08C0																
08D0																
08E0																
08F0																

Figure 3: Greek vowels with macron, for long syllables (not in Unicode).

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0A00	Ғ	ғ	Ж̢	ж̢	З̢	з̢	К̢	к̢	Н̢	н̢	С̢	с̢	Х̢	х̢	Г̢	г̢
0A10	Ԓ	ԓ	Л̣	л̣	Н̣	н̣	Т̦	т̦	Ч̣	ч̣	А̄	а̄	О̄	о̄	О̦	о̦
0A20	У̓	у̓	У̀	у̀	У̃	у̃	У̂	у̂	Э̄	э̄	Э̇	э̇	Ю̄	ю̄	Я̄	я̄
0A30	І̢	і̢	І̇	і	Ѳ̄	ѳ̄	Ү̄	ү̄	А́	а́	Е́	е́	Ё́	ё́	Є́	є́
0A40	И́	и́	І́	і́	Ї́	ї́	О́	о́	У́	у́	Ы́	ы́	Ъ́	ъ́	Э́	э́
0A50	Ю́	ю́	Я́	я́	Ѣ́	ѣ́	Ѵ́	ѵ́	Ѧ́	ѧ́	Ѩ́	ѩ́	Ѫ́	ѫ́	Ѭ́	ѭ́
0A60	Ѥ́	ѥ́	Ѹ́	ѹ́	Ꙋ́	ꙋ́	Ꙗ́	ꙗ́	Ѡ́	ѡ́	Ф́	ф́	Ү́	ү́	Ӕ́	ӕ́
0A70	Ә́	ә́	Ѳ́	ѳ́	Ү́	ү́	Ђ́	ђ́								
0A80	А̀	Ꙗ̀	Ѧ̀	Є̀	Ѥ̀	Ѝ	І̀	Ф̀	Ѡ̀	Ю̀	Ѹ̀	Ѵ̀	Ѣ̀			
0A90	а̀	ꙗ̀	ѧ̀	є̀	ѥ̀	ѝ	і̀	о̀	ѡ̀	ю̀	ꙋ̀	ѵ̀	ѣ̀			
0AA0	А̑	Ꙗ̑	Ѧ̑	Є̑	Ѥ̑	И̑	І̑	Ф̑	Ѡ̑	Ю̑	Ѹ̑	Ѵ̑	Ѣ̑			
0AB0	а̑	ꙗ̑	ѧ̑	є̑	ѥ̑	и̑	і̑	ф̑	ѡ̑	ю̑	ѹ̑	ѵ̑	ѣ̑			
0AC0	А҆́	Ꙗ҆́	Ѧ҆́	Є҆́	Ѥ҆́	И҆́	І҆́	Ф҆́	Ѡ҆́	Ю҆́	Ѹ҆́	Ѵ҆́	Ѣ҆́			
0AD0	а҆́	ꙗ҆́	ѧ҆́	є҆́	ѥ҆́	и҆́	і҆́	ф҆́	ѡ҆́	ю҆́	ѹ҆́	ѵ҆́	ѣ҆́			
0AE0	А҆̀	Ꙗ҆̀	Ѧ҆̀	Є҆̀	Ѥ҆̀	И҆̀	І҆̀	Ф҆̀	Ѡ҆̀	Ю҆̀	Ѹ҆̀	Ѵ҆̀	Ѣ҆̀			
0AF0	а҆̀	ꙗ҆̀	ѧ҆̀	є҆̀	ѥ҆̀	и҆̀	і҆̀	ф҆̀	ѡ҆̀	ю҆̀	ѹ҆̀	ѵ҆̀	ѣ҆̀			

Figure 4: Accented Cyrillic characters, for emphasis (not in Unicode).