

The New Computer Modern FontFamily version 5.2.0

Antonis Tsolomitis

June 18, 2024

Contents

1	Introduction	4
2	How to load the fonts	5
3	The Latin alphabet	5
3.1	Ligatures and stylistic alternatives in Latin	5
3.2	Oldstyle numbers	5
3.3	Old Italic	6
3.4	Diacritics Stacking	6
3.4.1	Coloring diacritics	7
4	Greek	7
4.1	Other character variants	8
4.2	Prosodic symbols	9
4.3	Archaic Greek writing	9
4.4	Aegean Numbers	10
4.5	Support for Papyrology	10
4.6	Support for Chemistry	11
5	Russian	11
6	Hebrew	12
7	Coptic and Epact Numbers	12
8	Cherokee	12
9	Canadian Aboriginal Syllabics	12
10	Devanagari	13
11	Medieval Latin and Uncial Greek	15

12 Braille	15
13 Ipa symbols	15
14 Bold Sans	17
15 Food Allergies	17
16 Unicode Math coverage and options	18
16.1 Optical sizes for more glyphs	19
16.2 Math Script	20
16.3 Blackboard Bold	20
16.4 Upright and extensible integrals	21
16.5 Additional and alternative characters in Math	22
16.6 Notes on Mathematics	22
17 The Medieval Latin and Uncial Greek glyph complement	23
18 The Aegean Numbers glyph complement	24

1 Introduction

The NewComputerModern FontFamily is a huge extension (“huge” in terms of the number of additional glyphs) of the `lm` fonts. It is not just a family adding random missing glyphs but it adds support for several more languages and shapes needed for academic (and not only) work. Currently it supports among others, Greek¹, Cyrillic², Hebrew, Cherokee and Coptic. Since it supports diacritics stacking the number of languages that use the Latin alphabet is greatly expanded. Diacritics stacking is also needed for Greek for papyrological work and this is also supported.

Version 4.0 adds to the classic design of computer modern new shapes for Latin and Greek, in particular it adds families for Medieval Latin and Uncial Greek matching in style to the main family.

In terms of weights and sizes, all of its shapes come in Regular, Book weights at 10 and 8 point sizes and in Bold at 10 points.

Mathematics is also supported in Regular and Book weights, currently providing a full coverage of the Unicode Math blocks (with a few more glyphs needed for Mathematics that Unicode has forgotten to encode).

What follows is a sequence of commands and results so as to show how to access all features of the fonts. Character tables are also included.

¹from Claudio Beccari’s Greek.

²from the `cmu` package.

2 How to load the fonts

The simpler way to load the fonts is through the `fontsetup` package. The command

```
\usepackage[default]{fontsetup}
```

will load the Book weight of the NewCM family, and

```
\usepackage[olddefault]{fontsetup}
```

will load the Regular weight.

Also notice that the fonts support the microtype package for fine typographic tuning. See the documentation of microtype for this.

3 The Latin alphabet

3.1 Ligatures and stylistic alternatives in Latin

The Serif font includes additional ligatures `fb ffb ffh ffj ffk fft fh fj ft fk` and the same with longs instead of `f` in the default `liga` table (in addition to the default `fi fl ffi ffl ff`). It also includes an alternative `k` (in the `cv01` table) and `ſp ch ck ct st il` in the `dlig` table. Finally it also includes “end” versions for the letters `a`, `e`, `m`, `n` and `r` in the `cv02` table. To access the alternative `k` load the relative font (here the Book weight) with

```
\setmainfont[CharacterVariant=1]{NewCM10-Book.otf}
```

To load the same font with the `dlig` table enabled use

```
\setmainfont[RawFeature=+dlig]{NewCM10-Book.otf}
```

and to load the font with endings variations use

```
\setmainfont[CharacterVariant=2]{NewCM10-Regular.otf}
```

Of course the above can be mixed separating the optional arguments with comma, or one can define a custom font say by using

```
\newfontfamily\myfont[<options to enable>]{NewCM10-Book.otf}
```

Book	k	a e m n r	sp ch ck ct st il
cv01	k		
cv02		a e m n r	
dlig			ſp ch ck ct st il

3.2 Oldstyle numbers

Typically oldstyle numbers are available in `onum` Lookup and with the `\textsc` if `fontsetup` is loaded. Also available they are with `\oldstylenums`. There are two series, one is with variable widths and one with fixed width for use in tables. The code

```
\oldstylenums{0123456789}\addfontfeatures{Numbers=Tabular}
```

```
\textsc{0123456789}
```

gives

```
0123456789
0123456789
```

An alternative design is also provided for the number 1 in cv06. The code

```
\oldstylenums{0123456789}\addfontfeatures{CharacterVariant=6}
\oldstylenums{0\textcolor{red}{1}23456789}
\addfontfeatures{CharacterVariant=6,Numbers=Tabular}
\oldstylenums{0\textcolor{red}{1}23456789}
```

gives

```
0123456789
0I23456789
0I23456789
```

3.3 Old Italic

The fonts also fully support the Old Italic Unicode block (U10300–U1032F) in the Sans font. For example, the slots U10307, U10310, U10312, U10314, U1031F and U1032F are $\mathfrak{M}\mathfrak{I}\mathfrak{Q}\mathfrak{X}\mathfrak{K}\mathfrak{E}$.

3.4 Diacritics Stacking

Diacritics—the full block U+0300 to U+036F—and diacritics stacking is supported. In the margin you can see an example of stacking on the letter “x” in Roman, Sans and Mono. If you need to enter these accents you can use the `\char` command or just copy-paste from the following line (from this pdf file or the provided source \TeX file):

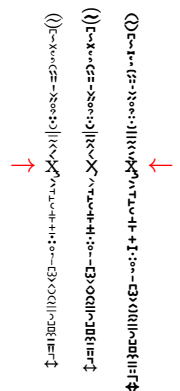
Some of the upper accents

\ / ^ ~ - - ˘ ˙ ˚ ˇ ˚ ˝

˘ ˙ ˚ ˛ ˜ ˝ ˞ ˟ ˠ ˡ ˢ ˣ ˤ ˥ ˦ ˧ ˨ ˩ ˪ ˫ ˬ ˭ ˮ ˯ ˰ ˱ ˲ ˳ ˴ ˵ ˶ ˷ ˸ ˹ ˺ ˻ ˼ ˽ ˾ ˿

Some of the lower accents

˘ ˙ ˚ ˛ ˜ ˝ ˞ ˟ ˠ ˡ ˢ ˣ ˤ ˥ ˦ ˧ ˨ ˩ ˪ ˫ ˬ ˭ ˮ ˯ ˰ ˱ ˲ ˳ ˴ ˵ ˶ ˷ ˸ ˹ ˺ ˻ ˼ ˽ ˾ ˿



Please note that stacking is by default supported with xetex. With luatex you have to add the option `Renderer=Harfbuzz`, say by `\addfontfeature{Renderer=Harfbuzz}`

Also notice that your text editor may not support stacking. The editor may show the accents one after the other, but the pdf produced by xetex or luatex will have the accents stacked.

3.4.1 Coloring diacritics

If one wants to use color for diacritics, different from the color of the base character this does not work with $\text{Xe}_{\text{L}}\text{A}_{\text{T}}\text{E}_{\text{X}}$ (the commands of the `color` package break the stacking mechanism). It works though with $\text{Lua}_{\text{L}}\text{A}_{\text{T}}\text{E}_{\text{X}}$ using the `luacolor` package. However, there is a problem when the base glyph and the first diacritic above exist in the font as a precomposed character. For example, this is the case with aacute (á) (U+00E1). Such characters are treated as one by Lua and they can not be colored with different colors. A work around is to place the empty character U+034F between “a” and acute (U+0301). So the following minimal example produces the result below:

```
\documentclass{article}
\usepackage[olddefault]{fontsetup}
\usepackage{luacolor}
\pagestyle{empty}
\newfontfamily{\ncmtest} [Renderer=Harfbuzz]{NewCM10-Regular.otf}
\definecolor{orange}{RGB}{255,191,0}
\definecolor{colorone}{RGB}{91,0,250}
\definecolor{colortwo}{RGB}{250,0,121}
\definecolor{colorthree}{RGB}{0,204,250}
\definecolor{colorfour}{RGB}{14,250,0}
\definecolor{colorfive}{RGB}{255,150,0}
\definecolor{colorgray}{gray}{0.8}
\newcommand{\emptydiacritic}{\char"034F}
\begin{document}
\Huge
{\ncmtest \color{colorgray}a\color{colorfour}\color{colortwo}%
\emptydiacritic\color{colorthree}\color{colorone}\color{colorfive}~}
\end{document}
```



4 Greek

The full Unicode Greek block is supported, which is

- U0370–U03FF for monotonic, where missing glyphs, such as Heta (Ϝ), Pamphilian digamma (ϝ) etc, have been added. For example, it is now possible to write $\beta\beta\lambda\iota$ instead of $\beta\beta\lambda\iota\circ$.
- U1F00–U1FFF for polytonic, and
- U10140–U1018F for ancient Greek numbers.

Θεώρημα 4.1 (Πυθαγόρειον) *Ἐν τοῖς ὀρθογωνίοις τριγώνοις τὸ ἀπὸ τῆς τῆν ὀρθὴν γωνίαν ὑποτείνουσας πλευρᾶς τετραγώνον ἴσον ἐστὶ τοῖς ἀπὸ τῶν τῆν ὀρθὴν γωνίαν περιεχουσῶν πλευρῶν τετραγώνοις.*

Small Caps is included (in Mono font too) and all polytonic accents of Greek. Ypogegrammeni is the default for all characters including Small Caps and prosgegrammeni is offered as an alternative shape in the `ss01` lookup table:

	ypogegrammeni	prosgegrammeni
regular	Ά Ή Ώ ΑΗΩ	Ά̣ Ή̣ Ώ̣ Α̣Η̣Ω̣
sans	Ά Ή Ώ ΑΗΩ	Ά̣ Ή̣ Ώ̣ Α̣Η̣Ω̣
mono	Ά Ή Ώ ΑΗΩ	Ά̣ Ή̣ Ώ̣ Α̣Η̣Ω̣

The prosgegrammeni alternates can be accessed with

```
\textprosgegrammeni{<text>}
```

or the

```
{\prosgegrammeni <text>}
```

of the `fontsetup` package.

4.1 Other character variants

Guillemots (left and right) have a different shape for Greek. For this to work the fonts must be loaded with the `cv04` character variant.

Compare the default guillemots: «» with Greek guillemots: «».

There is a serious problem with Unicode and the Greek anoteleia (U0387); the Greek semicolon. Unicode “thinks” that this character is the same with periodcentered (U00B7). This influences the way keyboards are configured by several vendors such as `xorg`. Anoteleia is a dot written at x-height and not at 1/2 the x-height as the periodcentered. Although Unicode recognizes the problem³, although they recognize that with their current standard you can not correctly write the Greek language, they refuse to fix it, justifying it by saying the magical words “backwards compatibility” (to a ...mistake, one could add).

`NewComputerModern` can not allow this, as it defies the purpose of its existence, which is to properly write every supported language. So enabling the `CharacterVariant 04 (cv04)` in addition to correct guillemots for Greek it maps periodcentered (produced by the keyboards (in Greek Linux keyboards by `AltGr+q`) to proper anoteleia.

It also fixes a long standing issue with the Greek apostrophe (') (U1FBD) which is not the same with quoteright (') (U2019). U1FBD named as “Greek Koronis” by Unicode is the proper character.

Another problem that has to do with quotes inside quotes. The internal quotes in Greek should be written with the characters `quotedblleft` and `quotedblbase`⁴. For example, this is correct for Greek

³personal communication

⁴Μανόλης Τριανταφυλλίδης, *Νεοελληνική Γραμματική της δημοτικής, Ανατύπωση της έκδοσης του ΟΕΕΒ 1941 με διορθώσεις, Θεσσαλονίκη 2002*, σελ. 66, ενότητα 133.

«άλφα “βήτα,,»

But the keyboards only produce quotesingle which is already mapped to apostrophe and it is difficult to remember the names “quotedblleft” and “quotedblbase”. So when enabling cv04 one can define the commands

```
\newcommand\leftgrquotes{\char"201C}
```

and

```
\newcommand\rightgrquotes{\char"201E}
```

for the rare case one needs quotes inside quotes. The `fontsetup` package does this automatically for Greek if the `xgreek` package has been loaded *before* the `fontsetup` package or when the language is set to Greek by, say, the Babel package. Otherwise, for non-Greek documents with small passages of Greek, the author may enable cv04 by creating a custom command such as

```
\newcommand\propergreek[CharacterVariant=4]{NewCM10-Book.otf}
```

A phrase with Greek quotes inside quotes, proper anoteleia, and proper apostrophe is

«φώναζε: “απ’ ἐξω την προπαίδεια,,»· σαν εκδίκηση ακουγόταν...

4.2 Prosodic symbols

In Greek philology and in linguistics it is often needed to stack accent-type symbols above letters, even if they are not vowels. Although rare in writings, it is for example valid to place dieresis over the consonants π, τ and χ of the nasal complexes μπ, ντ, γχ when it is necessary to show that these are pronounced, as written, voiceless, and not voiced. For example,

κομῆρέσα, αντιανῆτανῆτικός, ελεγῆτης

(see previous footnote). The fonts support this writing if a combining dieresis is placed after the letter to receive it. The combining dieresis is the character U0308. On Linux desktops this is easily entered pressing Alt+Control+u, release them, and type the sequence 0308 and space.

More than that, in linguistics, they need to combine several accents above Greek letters. All this stacking of accents is supported by the fonts. For example, one can write

ᾱ̃ ἔ̃ ῶ̄ ῶ̄ ῶ̄ ᾱ̃ ῶ̄ ῶ̄

by placing the combining accents from the Unicode block U0300–U0362 plus the usual Greek accents *after* the letter. So the above was typed as

α[~] ε[˘] ᾶ[˘] ᾶ[˘] ῶ[˘] ᾱ[˘] ῶ[˘] ῶ[˘]

4.3 Archaic Greek writing

The Sans Serif Regular font provides access to 6th century bce and 4th century bce Greek capitals in ss04 and ss03 lookups. The `fontsetup` package provides commands such as

where in the source we just typed the dot below (char U0323) after the glyph. This feature is supported for the 4th bce and 6th bce Greek in Sans:

Γ̣ Ε̣ Ω̣ Μ̣ Ε̣ Τ̣ Ρ̣ Ι̣ Α̣ Γ̣ Ε̣ Ω̣ Μ̣ Ε̣ Τ̣ Ρ̣ Ι̣ Α̣

4.6 Support for Chemistry

It happens often that Greek upright characters are needed in Chemistry. People often have trouble with this (and this is why packages such as `chemgreek` exist). If Greek keyboard is available then it is easy; you just type in Greek, say `β-glucan` to get “β-glucan”. But many writers do not have the Greek keyboard enabled, and they do not need to. Usually they type `β-glucan` but the result “β-glucan” is not satisfying. One can use the “up” versions typing `β-glucan` but still the result “β-glucan” looks more Math than Chemistry. To help with this, the `fontsetup` package provides commands such as `\chemAlpha`, `\chemalpha`, `\chemBeta`, `\chembeta`, etc. So this information essentially would only belong to the `fontsetup` documentation if it was not for kappa and rho. If we type in Greek κ-compound we get “κ-compound” which is not satisfying, as kappa is too cursive for this use. So the NewCM family provides an alternative kappa for this reason and this is how `\chemkappa` is defined in `fontsetup`:

```
\newcommand{\chemkappa}{\textrm{\char"03F0}}:
```

We write `\chemkappa-compound` and now get “κ-compound”.

(The `\textrm` command in the above definition is there to make the command work in math mode too.) Similar is the situation for `\chemrho` (ρ) and `\chemrhoalt` (ϱ).

5 Russian

Russian is supported using the glyphs from the `cmu` package but it has considerable improvements (for example, the quality of the bold sans (see below)).

Я помню чудное мгновенье:
 Передо мной явилась ты,
 Как мимолетное виденье,
 Как гений чистой красоты.
 (Пушкинъ)

Again, as in Greek there is a different kind of guillemots for Russian which are available in `CharacterVariant 3` (`cv03`). Compare:

Defaults guillemots: «» Russian guillemots: «» Greek guillemots: «»

Same is the situation with Russian emdash which is shorter than the default:

Default emdash: —
 Russian emdash: —

Next is a beautiful part of a poem in Marathi by तुकाराम (Tukaram) and its translation:

जें कां रंजलें गांजलें त्यांसि म्हणे जो आपुलें॥ १ ॥ तो चि साधु ओळखावा। देव तेथें चि जाणावा॥	Only the one who treats the downtrodden people equally is a sage ^a . One may sense the essence of god there.
	^a “The wise” of course, not the plant.

Devanagari Unicode letters (range U0900–U097F) are also available as variables (letters) and numbers in the Regular and Book Math fonts. They are available as usually in three weights in the Math fonts so that the color is balanced when in script size (eg in exponents or indices). For this to work a version of `fontsetup` package greater or equal to 1.8 with options `default` or `olddefault` loaded is needed. This is because Devanagari letters are not Math variables in Unicode standard and hence not supported currently as such by the `unicode-math` package. To show this possibility next is a theorem in Hindi (mixing with Greek):

प्रमेय (Πυθαγόρας (पिथागोरास)) अगर समकोण त्रिभुज के कर्ण की लंबाई को ‘अ’ और अन्य दो भुजाओं की लंबाई को ‘क’ और ‘ख’ कहते हैं, तो भुजाओं की लम्बाई के वर्गों की जोड़, कर्ण के वर्ग जितनी होती है, अर्थात् $अ^2 = क^2 + ख^2$ ।

However, if only Devanagari numbers are needed with the source using arabic numerals then one can use the Stylistic Set 04 of the Math font. So the command `\setmathfont[StylisticSet=4]{NewCMMath-Regular.otf}` with source:

```


$$\sum_{n=0}^{\infty} \frac{1}{n!} x^n = 1 + x + \frac{1}{2!} x^2 + \frac{1}{3!} x^3 + \frac{1}{4!} x^4 + \dots = e^x.$$


$$9! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot 8 \cdot 9.$$


$$2^{2^2}$$


```

will have the following effect:

$$\sum_{n=0}^{\infty} \frac{१}{n!} x^n = १ + x + \frac{१}{२!} x^2 + \frac{१}{३!} x^3 + \frac{१}{४!} x^4 + \dots = e^x.$$

$$९! = १ \cdot २ \cdot ३ \cdot ४ \cdot ५ \cdot ६ \cdot ७ \cdot ८ \cdot ९.$$

$$२^{२^२}$$

which when the math font is reset to use Arabic numbers with `\setmathfont{NewCMMath-Book.otf}` it gives:

$$\sum_{n=0}^{\infty} \frac{1}{n!} x^n = 1 + x + \frac{1}{2!} x^2 + \frac{1}{3!} x^3 + \frac{1}{4!} x^4 + \dots = e^x.$$

$$9! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot 8 \cdot 9.$$

$$2^{2^2}$$

11 Medieval Latin and Uncial Greek

The family includes new shapes for medieval and uncial Greek. The `fontsetup` package provides `\textuncial{<text>}` and `{\uncial text}` to use this shape. Let us write a sentence in this shape:

Τῆ ρΥΤΗΑΓΟΡΕΑΝ ΤῆΟΡΕΜ ΙΣ ΟΝΕ ΟΥ Τῆ ΜΟΣΤ ΙΜΠΟΡΤΑΝΤ
 ΤῆΟΡΕΜΣ ΙΝ ΜΑΤῆΜΑΤΙΣ.
 ΤΟ ΠΥΘΑΓΟΡΕΙΟ ΘΕΩΡῆΜΑ ΕΙΝΑΙ ΑΠΟ ΤΑ ΣΗΜΑΝΤΙΚΟΤΕΡΑ ΘΕ-
 ΩΡῆΜΑΤΑ ΤΩΝ ΜΑΘῆΜΑΤΙΚΩΝ.

12 Braille

Braille, both 6dot (uni2801–uni283F) as well as 8dit (uni2840–uni28FF) patterns are included in two versions. The Regular font provides the characters for sighted persons (such as teachers) so they can easily see which dots are on and which off. The Sans font contains the true Braille characters. I decided to have the sighted version in the Regular font since a blind person does not need the real Braille pattern, as those are produced by embossers. The Braille patterns here are meant as fonts to typeset text mainly for sighted persons.

	6dot	8dot
Regular version	⠠⠠⠠⠠⠠⠠	⠠⠠⠠⠠⠠⠠⠠⠠
Sans version	⠠⠠⠠⠠⠠⠠	⠠⠠⠠⠠⠠⠠

13 Ipa symbols

IPA symbols are included and following a suggestion of Huanyu Liu the kerning found in `tipa` package has been added here and further improved. Moreover the letters eth, eng, beta, theta and chi exists in IPA-style in the fonts and are accessible in the `ss05` lookup table since they are in a different design from the Latin and Greek letters. You can access this lookup table using the `\textipa` command of the `fontsetup` package.

	Non-IPA	IPA
Regular	ð ɳ β ϑ χ	ð ɳ β θ χ
Sans	ð ɳ β θ χ	ð ɳ β θ χ

I am grateful to निरंजन (Niranjan) for suggesting and testing all the IPA improvements that follow:

The joining of two characters such as \underline{ts} , \underline{dz} , \widehat{kp} , \widehat{tj} etc is also supported. The low tie is the character U035C and the upper tie is U0361. The \underline{ts} is produced by typing the sequence t then the low tie and then s. Similarly \widehat{kp} is produced by typing the sequence k then the upper tie and then p. For \underline{tj} and \underline{dz} the

fonts have a contextual chaining substitution table that uses a tie which prints lower so it does not touch *f* and *z*. In cases that one wants to show the tie as $x \underset{\sim}{y} z$ then one needs to enable the `cv5` character variant since the tie characters (U035C and U0361) are marks and not base glyphs. One can do that with a command such as

```
\newfontfamily{\showtie}[CharacterVariant=5]{NewCM10-Book.otf}
```

There is also an older practice that such sequences are joint into a ligature. This is not the modern way of writing but it seems than many people in the linguistics still prefer it. The fonts support this, if one enables the `lipa` table (local ipa) of the fonts. This can be done by adding the `RawFeature=+lipa` to the font specs when loading it and it is done automatically with the commands `\textoldipa{arg}` and `\oldipatext arg` of the `fontsetup` package.

For example, `\textoldipa{ts, tf}` produces *ts*, *tf*.

All other symbols of the `tipa` package are supported. Some examples are:

p̣, ˈ, ɫ, l̥, d̥, ʃ, ʂ, e̥, k̥, m̥, t̥, t̥, etc.

It is worth noting that all of the above is also available in the Mono family:

|*t̥ʂ*|*ɫ̥*|*ts*|*p̣*|*ˈ*|*ɫ̥*|*l̥*|*d̥*|*ʃ*|*ʂ*|*e̥*|*k̥*|*m̥*|*t̥*|*t̥*|

Another issue is that IPA used to use the Italic alternation of “g” even in the upright design. This restriction was lifted in 1949 as can be seen on page 3 in [IPAreview].

Even though the normal upright shape is accepted as an alternation, there are linguists who prefer the older (Italic) shape “g” and hence we are making it the default in the stylistic set dedicated for IPA (ss05).

Another such case is with the “a” shape. It has a different visual form in Italics “a” which resembles to a distinct symbol of IPA leading to a wrong transcription. In order to avoid this confusion, we have used the slanted version “a” of “a” in ss05. The other vowel symbol (ie, *ɑ* (U0251)) is also available in the fonts.

Next we give a real example of using IPA symbols. The quote below is from *A course in phonetics*, Ladefoged, Peter and Johnson, Keith, (2014), Cengage learning, pp. 285-286, and the `StylisticSet=05` has been enabled:

Other examples of this phenomenon have also been mentioned. We saw that before [ŋ], English does not distinguish between tense and lax vowels. Consequently, the vowel that occurs in, for example, sing has a quality between that of [i] and [ɪ]. Similarly, there is no distinction between tense and lax vowels before [ɹ]. The vowel in here in most forms of American English is also intermediate between [i] and [ɪ]. The principle of maximum perceptual separation also accounts for some of the differences between languages. French has two high rounded vowels, [u] as in *tout* [tu] “all,” and [y] as in *tu* [ty] “you.”

A recent addition for IPA is the support of the Unicode block U1D80–U1DBF which includes more letters with palatal and retroflex hook and more modifier small letters. Eg.

Upright	Italic
ɸ ɸ̟ ɸ̠ ɸ̡ ɸ̢ ɸ̣ ɸ̤ ɸ̥ ɸ̦ ɸ̧ ɸ̨ ɸ̩ ɸ̪ ɸ̫ ɸ̬ ɸ̭ ɸ̮ ɸ̯ ɸ̰ ɸ̱ ɸ̲ ɸ̳ ɸ̴ ɸ̵ ɸ̶ ɸ̷ ɸ̸ ɸ̹ ɸ̺ ɸ̻ ɸ̼ ɸ̽ ɸ̾ ɸ̿	<i>ɸ ɸ̟ ɸ̠ ɸ̡ ɸ̢ ɸ̣ ɸ̤ ɸ̥ ɸ̦ ɸ̧ ɸ̨ ɸ̩ ɸ̪ ɸ̫ ɸ̬ ɸ̭ ɸ̮ ɸ̯ ɸ̰ ɸ̱ ɸ̲ ɸ̳ ɸ̴ ɸ̵ ɸ̶ ɸ̷ ɸ̸ ɸ̹ ɸ̺ ɸ̻ ɸ̼ ɸ̽ ɸ̾ ɸ̿</i>
Sans Upright	Sans Oblique
ɸ ɸ̟ ɸ̠ ɸ̡ ɸ̢ ɸ̣ ɸ̤ ɸ̥ ɸ̦ ɸ̧ ɸ̨ ɸ̩ ɸ̪ ɸ̫ ɸ̬ ɸ̭ ɸ̮ ɸ̯ ɸ̰ ɸ̱ ɸ̲ ɸ̳ ɸ̴ ɸ̵ ɸ̶ ɸ̷ ɸ̸ ɸ̹ ɸ̺ ɸ̻ ɸ̼ ɸ̽ ɸ̾ ɸ̿	<i>ɸ ɸ̟ ɸ̠ ɸ̡ ɸ̢ ɸ̣ ɸ̤ ɸ̥ ɸ̦ ɸ̧ ɸ̨ ɸ̩ ɸ̪ ɸ̫ ɸ̬ ɸ̭ ɸ̮ ɸ̯ ɸ̰ ɸ̱ ɸ̲ ɸ̳ ɸ̴ ɸ̵ ɸ̶ ɸ̷ ɸ̸ ɸ̹ ɸ̺ ɸ̻ ɸ̼ ɸ̽ ɸ̾ ɸ̿</i>















14 Bold Sans

lm fonts and cmu fonts do not contain a properly made BoldSans. Their BoldSans is a stroke-extension of the Sans with rounded corners. NewCM fixes that and provides a true BoldSans:

LM	NewCM
X E	X E Я Д Л

15 Food Allergies

Food allergy symbols have long been proposed to be accepted to Unicode standard but there has not been any progress up to now. The Sans 10 Regular and Book include standard allergy symbols in the Private Use Area. Each glyph is named after what it represents. For example, the soya symbol is named “soya” so copying the symbol from a pdf file and pasting elsewhere you will get its name, that is “soya”. The symbols are in positions U+E033 to U+E040 and can be accessed using the commands of the next table.

<code>\char"E033</code>	 CRUSTACEANS	<code>\char"E034</code>	 EGGS
<code>\char"E035</code>	 GLUTEN	<code>\char"E036</code>	 FISH
<code>\char"E037</code>	 LUPIN	<code>\char"E038</code>	 MILK
<code>\char"E039</code>	 MOLLUSCS	<code>\char"E03A</code>	 MUSTARD
<code>\char"E03B</code>	 PEANUT	<code>\char"E03C</code>	 SESAME
<code>\char"E03D</code>	 SOYA	<code>\char"E03E</code>	 TREENUTS
<code>\char"E03F</code>	 CELERY	<code>\char"E040</code>	 SO ₂

16 Unicode Math coverage and options

NewCM provides full Unicode math support, that is all Mathematics Unicode Slots presented in <http://www.unicode.org/charts/> in both Math weights, Regular and Book. These blocks are:

Mathematical Symbols

Arrows (uni2190–uni21FF)

Supplemental Arrows-A (uni27F0–uni27FF)

Supplemental Arrows-B (uni2900–uni297F)

Supplemental Arrows-C (u1F800–u1F8FF)

Additional Arrows (uni2B00–uni2BFF)

Miscellaneous Symbols and Arrows (uni2B00–uni2BFF)

Mathematical Alphanumeric Symbols

(u1D400–u1D7FF)
Arabic Mathematical Alphabetic Symbols
(u1EE00–u1EEFF)
Letterlike Symbols (uni2100–uni214F)

Mathematical Operators

(uni2200–uni22FF)
Basic operators: Plus, Factorial
(uni0000–uni007F)
Division, Multiplication
(uni0080–uni00FF)
Supplemental Mathematical Operators
(uni2A00–uni2AFF)
Miscellaneous Mathematical Symbols-A
(uni27C0–uni27EF)
Miscellaneous Mathematical Symbols-B
(uni2980–uni29FF)
Floors and Ceilings (uni2308–uni230B)
Invisible Operators (uni2061–uni2064)

Geometric Shapes (uni25A0–25FF)

Additional Shapes (uni2B00–uni2BFF)
Box Drawing (uni2500–uni257F)
Block Elements (uni2580–uni259F)
Geometric Shapes Extended (u1F780–u1F7FF)

Unfortunately, the `unicode-math` package does not provide commands currently for the hundreds of extra glyphs that have been added in order to fully cover the above Unicode ranges. The user can consult the Unicode charts at the above link and access the required glyph with `\char"#` where `#` is the Unicode number of the slot the glyph belongs to.

For example, `\char"2BDA` will give the Hygeia symbol (uni2BDA) the Rod of Asclepius as shown above (grayed and scaled $\times 8$). The glyph that appeared in TUGboat (see [AT]), being more realistic will be moved to a new font in the future with ornaments.



16.1 Optical sizes for more glyphs

So far the fonts provided optical sizes for 1st and 2nd order exponents for letters. This was not true though for binary operators and some symbols commonly used in mathematics. For example, the `\perp` symbol (\perp) often appears in 2nd order exponents and then it appeared very thin. Some printers could even hardly print its thin lines. Now, such glyphs plus several binary operators, such as $+$,

myself) got used to it and find it difficult to feel their non-matching design. Moreover, the new design can not be metrically equivalent with the past so there will be slight changes in the older documents if re-run. With this in mind and the fact that the new design should be considered “beta”, as there are characters that need improvement, I decided bring back to default the AMS design and keep the new design for new documents for the StylisticSet 3.

The AMS blackboard bold are:

ABCDEFGHIJKLMNOPQRSTUVWXYZ
 abcdefghijklmnopqrstuvwxyz
 0123456789 $\pi\gamma\Gamma\Pi\Sigma$ *Ddeij*

To access the new ones needs to load the math font enabling the `ss03` stylistic set using for example

`\setmathfont [StylisticSet=3]{NewCMMath-Book.otf}` Then the above blackboard bold design changes to

ABCDEFGHIJKLMNOPQRSTUVWXYZ
 abcdefghijklmnopqrstuvwxyz
 0123456789 $\pi\gamma\Gamma\Pi\Sigma$ *Ddeij*

If using the latest `fontsetup` (to be released soon) then you can choose the NewCM blackboard bold with the option `newcmbb`.

16.4 Upright and extensible integrals

The Math fonts (both Regular and Book weights) include upright integrals in the `ss02` StylisticSet. Use with

`\setmathfont [StylisticSet=2]{NewCMMath-Book.otf}`
 or
`\setmathfont [StylisticSet=2]{NewCMMath-Regular.otf}`

or use the `upint` option of the `fontsetup` package with

`\usepackage [upint, default]{fontsetup}`

for the Book weight, or

`\usepackage [upint, olddefault]{fontsetup}`

for the regular weight.

Moreover, extensible integrals are supported by the fonts but *NOT* by the Unicode TeX engines. The following code is a trick so that extensible integrals can be constructed using Lua \LaTeX . The result is shown at the end of the article. What the code below does, is that it defines the slot `uni222B` (integral) as a delimiter. And then this is extended as a delimiter with the mechanism that the font provides.

```

\documentclass{article}
\usepackage[default]{fontsetup}
\begin{document}
$
\Uleft\Udelimiter 0 0 "222B
\begin{pmatrix}
1\2\3\4\5\6\7\8\9
\end{pmatrix}
\Uright.
$
\end{document}

```

$$\left(\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array} \right)$$

16.5 Additional and alternative characters in Math

The Math fonts provide the character `\varnothing` (\emptyset), as an alternative to `\emptyset` (\emptyset), through Character Variant `cv01`. The `fontsetup` package provides the option ‘`varnothing`’ to easily switch to the alternative character.

It also provides four more symbols that correspond to the commands

`\nrightrightarrows` (\rightrightarrows) `\nleftleftarrows` (\nleftleftarrows)

and

`\twoheadhookrightarrow` (\twoheadhookrightarrow) `\twoheadhookleftarrow` (\twoheadhookleftarrow)

and supported by the `default` and `olddefault` options of the `fontsetup` package. These symbols are not in the Unicode Standard and so they are added in the Private Area of the fonts.

16.6 Notes on Mathematics

Extensible tildes and hats produce different results with $X_{\Gamma}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ and $\text{L}^{\text{u}}\text{a}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ because these unicode engines treat differently the width of characters. In particular, $X_{\Gamma}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ handles italic correction as part of the character width but $\text{L}^{\text{u}}\text{a}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ does this only if a character follows. This affects how extensible accents like `\widehat` or `\widetilde` select the proper size. With $X_{\Gamma}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, `\widetilde{Y}` `\widetilde{X}` will give the expected result $\widetilde{Y}\widetilde{X}$; but with $\text{L}^{\text{u}}\text{a}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ the letter Y is narrower (and its italic correction is not taken into account if it stands alone) and gets the plain tilde size, as in \tilde{Y} . To bypass this with $\text{L}^{\text{u}}\text{a}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ one has to use a zero width character after a letter such as Y so that $\text{L}^{\text{u}}\text{a}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ takes into account the italic correction of Y . For example with

```
\widetilde{Y\Uchar"200D}
```

the result will be correct (U200D is the “Zero Width Joiner” in Unicode).

I thank Ulrike Fischer for this solution.

17 The Medieval Latin and Uncial Greek glyph complement

Table 1: NewCMUncial10-Book.otf

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Basic Latin																
U+0020-002F		!	"	#	\$	%	&	'	()	*	+	,	-	.	-
U+0030-003F	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	-
U+0040-004F	⓪	Ⓐ	Ⓑ	Ⓒ	Ⓓ	Ⓔ	Ⓕ	Ⓖ	Ⓗ	Ⓘ	⓫	⓬	Ⓜ	Ⓝ	Ⓞ	
U+0050-005F	Ⓟ	Ⓠ	Ⓡ	Ⓢ	Ⓣ	Ⓤ	Ⓥ	Ⓦ	Ⓧ	Ⓨ	Ⓩ	-	-	-	-	-
U+0060-006F	-	ⓐ	ⓑ	ⓒ	ⓓ	ⓔ	ⓕ	ⓖ	ⓗ	ⓘ	ⓙ	ⓓ	ⓔ	ⓕ	ⓖ	ⓗ
U+0070-007F	ⓓ	ⓔ	ⓕ	ⓖ	ⓗ	ⓘ	ⓙ	ⓓ	ⓔ	ⓕ	ⓖ	ⓗ	ⓘ	ⓙ	ⓓ	ⓔ
Latin-1 Supplement																
U+00A0-00AF	-	-	-	-	-	-	-	-	-	¨	-	-	-	-	-	-
U+00B0-00BF	-	-	-	-	-	-	-	·	-	-	-	-	-	-	-	-
Greek and Coptic																
U+0370-037F	-	-	-	-	´	,	-	-	-	-	-	-	-	-	-	;
U+0380-038F	-	-	-	-	´	ˆ	λ	·	€	Η	Ι	-	Ο	-	Υ	Ω
U+0390-039F	ϊ	λ	β	Γ	Δ	€	Z	H	Θ	I	K	Λ	M	N	Ξ	O
U+03A0-03AF	Π	P	-	C	T	Υ	Φ	X	Ψ	Ω	İ	ÿ	λ	ε	η	ι
U+03B0-03BF	ÿ	λ	β	Γ	Δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
U+03C0-03CF	π	ρ	σ	τ	υ	φ	χ	ψ	ω	ı	ÿ	ο	τ	ω	-	
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Total number of glyphs shown from NewCMUncial10-Book.otf: 160

18 The Aegean Numbers glyph complement

<code>\aegeanseparator</code>	ι	<code>\aegeaneighthundred</code>	⊞
<code>\aegeanseparatordot</code>	·	<code>\aegeanninehundred</code>	⊞⊞
<code>\aegeancheckmark</code>	×	<code>\aegeanonethousand</code>	⊕
<code>\aegeanone</code>	ι	<code>\aegeantwothousand</code>	⊕⊕
<code>\aegeantwo</code>	″	<code>\aegeanthreethousand</code>	⊕⊕⊕
<code>\aegeanthree</code>	‴	<code>\aegeanfourthousand</code>	⊕⊕⊕⊕
<code>\aegeanfour</code>	⦶	<code>\aegeanfivethousand</code>	⊕⊕⊕⊕⊕
<code>\aegeanfive</code>	⦶⦶	<code>\aegeansixthousand</code>	⊕⊕⊕⊕⊕⊕
<code>\aegeansix</code>	⦶⦶⦶	<code>\aegeanseventhousand</code>	⊕⊕⊕⊕⊕⊕⊕
<code>\aegeanseven</code>	⦶⦶⦶⦶	<code>\aegeaneightthousand</code>	⊕⊕⊕⊕⊕⊕⊕⊕
<code>\aegeaneight</code>	⦶⦶⦶⦶⦶	<code>\aegeanninethousand</code>	⊕⊕⊕⊕⊕⊕⊕⊕⊕
<code>\aegeanine</code>	⦶⦶⦶⦶⦶⦶	<code>\aegeantenthousand</code>	⊕
<code>\aegeanten</code>	-	<code>\aegeantwentythousand</code>	⊕⊕
<code>\aegeantwenty</code>	=	<code>\aegeanthirtythousand</code>	⊕⊕⊕
<code>\aegeanthirty</code>	≡	<code>\aegeanfourtythousand</code>	⊕⊕⊕⊕
<code>\aegeanfourty</code>	==	<code>\aegeanfiftythousand</code>	⊕⊕⊕⊕⊕
<code>\aegeanfifty</code>	≡≡	<code>\aegeansixtythousand</code>	⊕⊕⊕⊕⊕⊕
<code>\aegeansixty</code>	≡≡≡	<code>\aegeanseventythousand</code>	⊕⊕⊕⊕⊕⊕⊕
<code>\aegeanseventy</code>	≡≡≡≡	<code>\aegeaneightythousand</code>	⊕⊕⊕⊕⊕⊕⊕⊕
<code>\aegeaneighty</code>	≡≡≡≡≡	<code>\aegeanninetythousand</code>	⊕⊕⊕⊕⊕⊕⊕⊕⊕
<code>\aegeanninety</code>	≡≡≡≡≡≡	<code>\aegeanweightbaseunit</code>	⊞⊞
<code>\aegeanonehundred</code>	°	<code>\aegeanweightfirstsubunit</code>	⊞
<code>\aegeantwohundred</code>	°°	<code>\aegeanweightsecondsubunit</code>	⊞⊞
<code>\aegeanthreehundred</code>	°°°	<code>\aegeanweightthirdsubunit</code>	⊞⊞⊞
<code>\aegeanfourhundred</code>	°°°°	<code>\aegeanweightfourthsubunit</code>	⊞⊞⊞⊞
<code>\aegeanfivehundred</code>	°°°°°	<code>\aegeandrymeasurefirstsubunit</code>	⊞
<code>\aegeansixhundred</code>	°°°°°°	<code>\aegeanliquidmeasurefirstsubunit</code>	⊞
<code>\aegeansevenhundred</code>	°°°°°°°	<code>\aegeansecondsubunit</code>	⊞
		<code>\aegeanthirdsubunit</code>	⊞

References

- [AT] Antonis Tsolomitis, *The NewComputerModern font family*, TUGboat Vol. 42, No. 1, 2021.
- [IPArevis] Council actions on revisions of the IPA, Phonetic Representation: b) Revision of the IPA, Journal of the International Phonetic Association, Volume 23, Issue 1, June 1993, pp. 32–34.