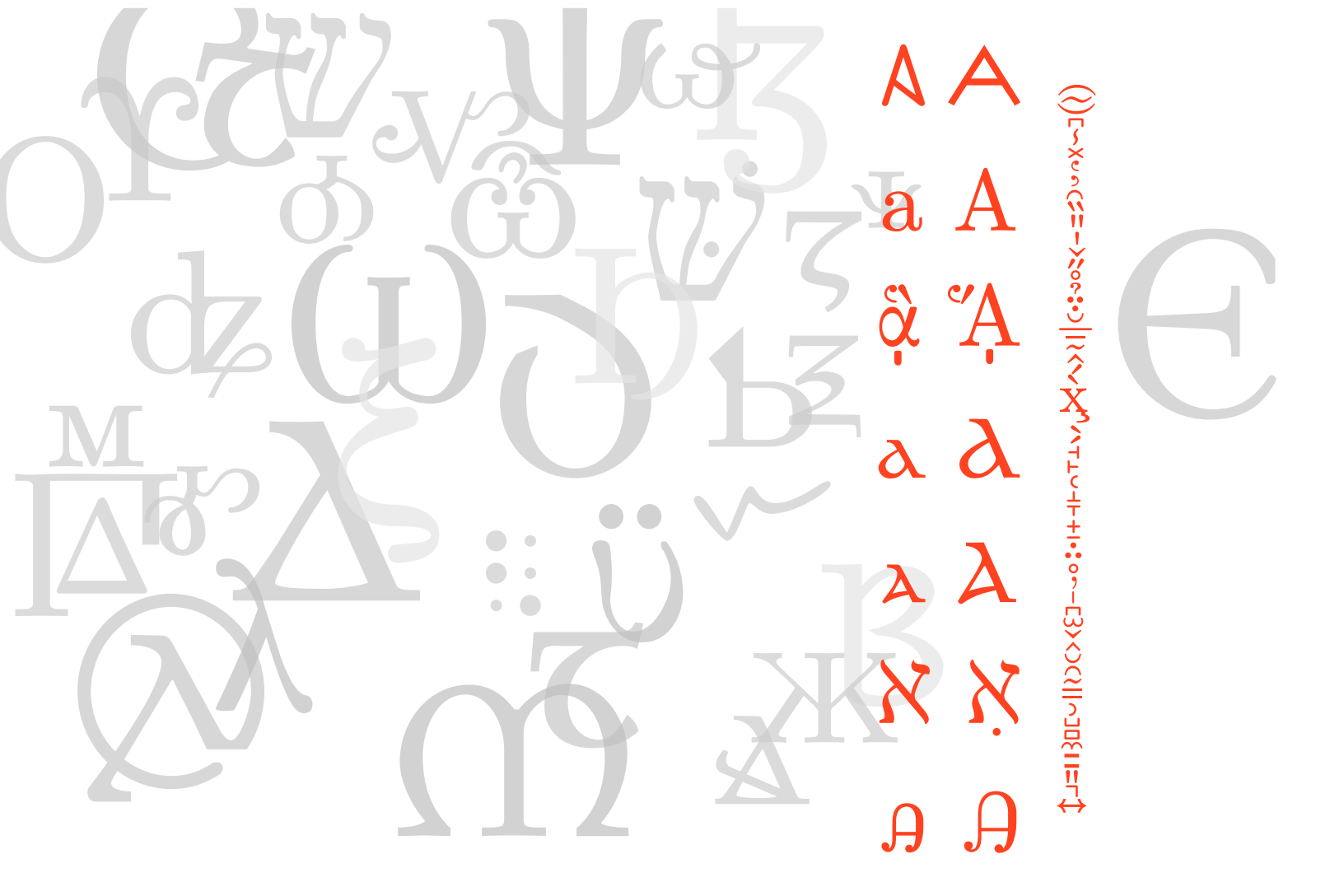
The NewComputerModern FontFamily

Antonis Tsolomitis • University of the Aegean • Department of Mathematics



The New Computer Modern FontFamily version 7.0.1

Antonis Tsolomitis

January 19, 2025

Contents

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, Greek[1], Cyrillic[2], Devanagari, Hebrew, Coptic, Cherokee and Canadian Aboriginal. 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.

[1] from Claudio Beccari's Greek.
[2] from the `cmu` package.

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.

IMPORTANT: If you want to provide patches for the fonts please contact me before you create them. The fonts evolve quickly and you may not have the latest development version and your patches may not apply if created for the published version.

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,

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

will load the Regular weight, and

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

will load the Sans Serif NewCM family.

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 îl in the dlig table. Finally it also inludes "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
0123456789
0123456789

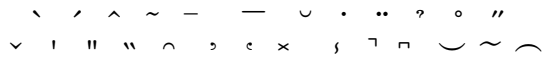
3.3 Old Italic

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 𐌇𐌐𐌒𐌔𐌟𐌯.

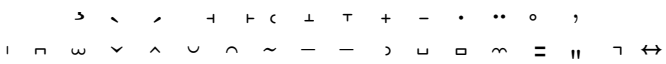
3.4 Diacritics Stacking

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



Some of the lower accents



Please note that stacking is by default supported with xetex. With luatex you have to add the option `Renderer=Harfbuzz`, say by
`\addfontfeatures{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 XeLaTeX (the commands of the `color` package break the stacking mechanism). It works though with LuaLaTeX 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 colorized 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 (Ͱ), Pamphilian digamma (ͷ) etc, have been added. For example, it is now possible to write

  βιϐλίο instead of βιβλίο.

  In order to auto select this conversion for middle beta and theta the StylisticSets ss07 must be enabled with, say,

  `\addfontfeatures{StylisticSet=7}`,

  but for the Sans and Mono fonts `StylisticSet=6` is also needed, with

  `\addfontfeatures{StylisticSet=6,StylisticSet=7}`.

  To disable this feature you can do

  `\addfontfeature{RawFeature={-ss07}}`

  or

  `\addfontfeature{RawFeature={-ss06},RawFeature={-ss07}}`

  for the Sans and Mono families.

Source	βιβλίο	θυμήθηκα
ss07 enabled	βιϐλίο	ϑυμήθηκα
ss06 and ss07 enabled for Sans	**βιϐλίο**	**ϑυμήθηκα**
ss06 and ss07 disabled	βιβλίο, **βιβλίο**	ϑυμήθηκα, **θυμήθηκα**

- U1F00–U1FFF for polytonic, and
- U10140–U1018F for ancient Greek numbers.

Θεώρημα 4.1 (Πυθαγόρειον) *Ἐν τοῖς ὀρθογωνίοις τριγώνοις τὸ ἀπὸ τῆς τὴν ὀρθὴν γωνίαν ὑποτεινούσης πλευρᾶς τετράγωνον ἴσον ἐστὶ τοῖς ἀπὸ τῶν τὴν 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	ᾼ ῌ ῼ ᾲῂῲ	ᾈ ᾘ ᾨ ᾀᾐᾠ
sans	ᾼ ῌ ῼ ᾲῂῲ	ᾈ ᾘ ᾨ ᾀᾐᾠ
mono	ᾼ ῌ ῼ ᾲῂῲ	ᾈ ᾘ ᾨ ᾀᾐᾠ

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 problem[3], althought 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

«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

[3] personal communication

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

`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᾽ έξω 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 κ of the nasal complexes μπ, ντ, γκ when it is necessary to show that these are pronounced, as written, voiceless, and not voiced. For example,

κομπρέσα, αντιανταντικός, ελεγκτής

(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

α̃ ε̃ Ă Ā Ω̆ ά Ẍ ẗ

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

```
α̃  ε̃  Ă  Ā  Ω̆  ά  Ẍ  ΐ
```

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

`\textivbce{}`, `\ivbce`, `\textvibce{}` and `\vibce`

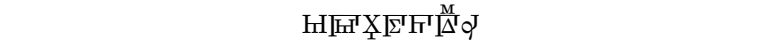
6th century bce:

ΜΗΔΕΙΣ ΑΓΕΩΜΕΤΡΗΤΟΣ ΕΙΣΙΤΩ

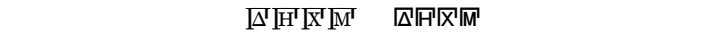
4th century bce:

ΜΗΔΕΙϹ ΑΓΕΩΜΕΤΡΗΤΟϹ ΕΙϹΙΤΩ

Moreover, all fonts (except Mono & Math) support Ancient Greek Numerals (the full Unicode block of Greek digits U10140–U1018E is supported), with most symbols designed from scratch (and did not exist in C. Beccari's original fonts). A few of the new symbols:



The four numerals that already existed in this range (that is U10144–U10147) in Beccari's fonts have been altered to a new design matching the style of cm but also provide some Ancient Greek flair. The new designs in Serifed and SansSerifed are:



The `fontsetup` package provides commands for all of the above symbols. The commands follow the Unicode name of each slot (minus the "Greek Acrophonic"). So the Unicode slot U1014F named "Greek Acrophonic Attic Five Staters" can be accessed with the command `\atticfivestaters` and it gives 𐅯; and the slot u10182 named "Greek Kyathos Base Sign" can be accessed with the command `\greekkyathosbasesign` and it gives 𐆒.

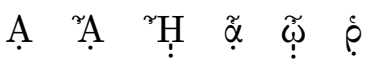
4.4 Aegean Numbers

Aegean numbers are supported in the Sans fonts and their slots are defined in `fontsetup` package using commands of the form `\aegeanXXXX` where `XXXX` is the Unicode name of the character (without spaces). A few examples are:

and the whole table of Aegean Numbers with the commands to access the glyphs is shown on page 30.

4.5 Support for Papyrology

Papyrology needs to declare that a glyph is missing from the papyrus or the papyrus is worn at this point and the papyrologist adds the missing glyph but it is not clear from the papyrus. This is done by adding a dot below the glyph and it is supported for all Greek glyphs in the upright fonts monotonic or polytonic:



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:

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 `$\beta$-glucan` but the result "β-glucan" is not satisfying. One can use the "up" versions typing `$\upbeta$-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: —

6 Hebrew

The Hebrew blocks U0590–U05FF and Hebrew Presentation forms UFB1D–UFB4F are fully covered. and A few letters from Hebrew:

אבגדהוׁשׁשׂלצ

7 Coptic and Epact Numbers

The Coptic language is fully supported. This covers the Coptic blocks in the Greek and Coptic Unicode block (U03E2–U03EF), the full Coptic Unicode block (U2C80–U2CFF) and the Coptic Epact Numbers (U102E0–U102FF). A few letters from Coptic and Epact numbers follow:

ⲗⲟⲅⲟⲥ ⲛ̀ⲁⲓⲅⲩⲡⲧⲓⲟⲥ ؾ ؽ ؼ ؿ ـ

8 Cherokee

Both Unicode blocks U13A0–13FF and UAB70–UABBF for Cherokee are supported. The samples below were kindly provided by Sedi Eastwood:

Sample 1:

EGᏢCᎯGᏏᏂᏍᏫᎯ TᏙᏢᏫᎯ ᎬᏟᏢᏟᎯᏟᎵᏂᏍᏫᎯ TᏙᏢᏫᎯ	Live/exist in a manner that there is never a reason or purpose to let go of one another. (gvjalijvdijadayohisdi ijehesdi gvtsalitsvditsadayohisdi itsehesdi)

Sample 2:

ᏚGᏏᏲG4ᏫᎯ ᏚᏟᎵᏲG4ᏫᎯ	Value the existence of one another. (dejadageyusesdi detsadageyusesdi)

Sample 3:

iᏢᏫᏲᎯ ᏚGᏏᏰᏎ4ᏫᎯ iᏢᏫᏲᎯ ᏚᏟᎵᏰᏎ4ᏫᎯ	Think of one another as sacred or hold the existence of others sacred. (vlisgedi dejadayelvsesdi)

9 Canadian Aboriginal Syllabics

Canadian Aboriginal Syllabics are supported in the sans font. The full Unicode blocks are covered, which are U1400–U167F, U18B0–U18F5, and U11AB0–U11ABF.

Some examples in Blackfoot Siksiká:

Blackfoot-speaking real people: ᖹᐟᒧᐧᒣᐤ Siksiká: ᓵᓴᐧᖾ
Siksikáí'powahsin: ᓵᖿᓵᖿ¶ᐟᑲᕮᐦᓵᐤ In bold: ᓵᖿᓵᖿ¶ᐟᑲᕮᐦᓵᐤ
Niitsipowahsin: ᖹᐨᓵᑲᕮᐦᓵᐤ In oblique: ᖹᐨᓵᑲᕮᐦᓵᐤ

Some more in Cree:

ᛘᐊᐧᑯ ᐅᒪ ᐊᓵᓂᐞᒧᓂᑲᐣ ᑭᒥᓇᐧᐊᐧ᙮
ᐞᓂᓂᐊᐧ᙮ ᒪᓂᑐᐸ ᑲᐊᔾᑎᐧᕽ ᐁᐊᐧᑲᓂᐧ᙮ ᐅᑭ
ᑲᓇᑎ"ᑕᑎᐧ᙮ ᐊᓵᓂᐞᐊᑐᑎᑭᐞᓂᓂᐣ° ᑲᑭᒥᓂ"ᑎᐧ᙮
ᐅᒧᓂ°ᕽ ᐊᑲᐧᓂ ᒥᑕ"ᑐᒥᑕᓇ° ᐊᑎᑭ⁺ ᐊᖫ⁺
ᐁᐊᑐᑎᖾᐧ᙮ ᐅᑕ ᒪᓂᑐᐸᕽ

The masonry for the wall through time has been given to people of Manitoba by the international union of bricklayers and allied craftsmen local one Manitoba to commemorate 100 years of service, by Manitoba masonry contractors and by Manitoba masonry suppliers.

10 Devanagari

Devanagari script is supported for the serifed font in Regular (10pt/8pt), Book (10pt/8pt), and Bold (10pt). The fonts support Hindi (as the default), Sanskrit, Marathi and Nepali Languages. The optional arguments for the `fontspec` font-selection mechanism must include
`Script=Devanagari, Language=XXXX` where `XXXX` must be replaced with one of `Hindi`, `Sanskrit`, `Marathi`, `Nepali`. If the `Language` parameter is not set then the default is `Hindi`. For LuaLaTeX the parameter `Renderer=Harfbuzz` must also be included.

So if say Marathi is needed as the default font document then one can use the following:

```
\usepackage{fontspec}
\setmainfont[Script=Devanagari, Language=Marathi,%
Renderer=Harfbuzz]{NewCM10Devanagari-Book.otf}
```

The Devanagari fonts were developed with the help of निरंजन (Niranjan) whose name appears in the copyright section of the fonts and I also thank him for providing the samples below. It should also be noted that the design is original and based on old handwritten books. In old civilizations, such as the Indian one, it is only natural that the design be affected by how the handwritten books look. Ink creates "drop"-like serifs on paper that absorbs it, and the tool used to write also affects the look of the script. All these were taken into account. A Sanskrit sample from बृहदारण्यकोपनिषद् (bṛhadāraṇyakopaniṣad) follows:

ॐ पूर्णमदः पूर्णमिदं, पूर्णात्पूर्णमुदच्यते।
पूर्णस्य पूर्णमादाय पूर्णमेवावशिष्यते॥

That[a] is complete;
this[b] too is complete.
From one complete comes the other. Taking out
one complete from the other too results in a complete.

[a] the outer world
[b] the inner world

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 $अ^२ = क^२ + ख^२$।

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:

```
$$\sum_{n=0}^\infty \frac1{n!}x^n=1+x+\frac{1}{2!}x^2
   +\frac1{3!}x^3+\frac1{4!}x^4+\ldots=e^x.$$
$$9!=1\cdot2\cdot3\cdot4\cdot5\cdot6\cdot7\cdot8\cdot9.$$
$$2^{2^2}$$
```

will have the following effect:

$$\sum_{n=०}^{\infty} \frac{१}{n!}x^n = १ + x + \frac{१}{२!}x^२ + \frac{१}{३!}x^३ + \frac{१}{४!}x^४ + ... = e^x.$$

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

২২২

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

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

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

$$2^{2^2}$$

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:

> Ƭhe ƿythagorean Ƭheorem is one of the most important theorems in Mathematics.
> Το Πγθαγορειο Θεωρημα ειναι απο τα σημαντικοτερα θεωρηματα των Μαθηματικων.

Medieval Latin and Uncial Greek fonts will give a lot of "missing slot" warnings if the microtype package is loaded. To suppress these warnings use

```
\DeclareMicrotypeAlias{NewCMUncial10-Book.otf}{TU-empty}
```

and similarly for the other NewCMUncial fonts.

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 ŋ β ϑ χ	ð ŋ β θ χ
Sans	**ð ŋ β θ χ**	**ð ŋ 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 t͡s, d͡z, k͡p, t͡ʃ etc is also supported. The low tie is the character U035C and the upper tie is U0361. The t͡s is produced by typing the sequence t then the low tie and then s. Similarly k͡p is produced by typing the sequence k then the upper tie and then p. For t͡ʃ and d͡ʒ the fonts have a contextual chaining substitution table that uses a tie which prints lower so it does not touch ʃ and ʒ. In cases that one wants to show the tie as x͜y͡z 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, tʃ}` produces ʦ, ʧ.

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

pʼ, ˈ, ɬ, l̪, ɖ, ɟ, ʂ, e̞, k̟, m̪, t̪, t̪, etc.

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

|t͜s|t͡ʃ|ʦ|pʼ|ˈ|ɬ|l̪|ɖ|ɟ|ʂ|e̞|k̟|m̪|t̪|t̪|

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 [IPArev].

Even though the normal upright shape is accepted as an alternation, there are linguists who prefer the older (Italic) shape "ɡ" 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* [t̪u] “all,” and [y] as in *tu* [t̪y] “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
ᶹ ᶂ ᶈ ᶉ ᶑ ᶌ ᶏ ᶚ ᶙ ᶻ ᵑ ᶞ ᶟ ᶿ	*ᶹ ᶂ ᶈ ᶉ ᶑ ᶌ ᶏ ᶚ ᶙ ᶻ ᵑ ᶞ ᶟ ᶿ*
Sans Upright	Sans Oblique
ᶹ ᶂ ᶈ ᶉ ᶑ ᶌ ᶏ ᶚ ᶙ ᶻ ᵑ ᶞ ᶟ ᶿ	*ᶹ ᶂ ᶈ ᶉ ᶑ ᶌ ᶏ ᶚ ᶙ ᶻ ᵑ ᶞ ᶟ ᶿ*

14 Currency Symbols

Support for currency symbols block U20A0–U20C0 in all text fonts (except uncials).

Roman: ₠₡₢₣₤₥₦₧₨₩₪₫€₭₮₯₰₱₲₳₴₵₶₷₸₹₺₻₼₽₾₿⃀

Sans: ₠₡₢₣₤₥₦₧₨₩₪₫€₭₮₯₰₱₲₳₴₵₶₷₸₹₺₻₼₽₾₿⃀

Mono: ₠₡₢₣₤₥₦₧₨₩₪₫€₭₮₯₰₱₲₳₴₵₶₷₸₹₺₻₼₽₾₿⃀

Interesting is the design of Spanish Peseta in the Mono family (seen enlarged in the margin) were we have to fit essentially three letters in the width of the Mono font.

Especially for the Rupee sign (U20B9) the upright serifed non-Devanagari fonts have the “latinized” version, but to avoid changing fonts they also provide the Devanagari version in CharacterVariant 7. So the code

```
\begin{LARGE}
\begin{center}
  100\,₹ \hspace{2cm} {\addfontfeatures{CharacterVariant=7}100\,₹}
\end{center}
\end{LARGE}
```

gives

100 ₹ 100 ₹

15 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
	

16 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.

\char"E033	 CRUSTACEANS	\char"E034	 EGGS
\char"E035	 GLUTEN	\char"E036	 FISH
\char"E037	 LUPIN	\char"E038	 MILK
\char"E039	 MOLLUSCS	\char"E03A	 MUSTARD

\char"E03B	 PEANUT	\char"E03C	 SESAME
\char"E03D	 SOYA	\char"E03E	 TREENUTS
\char"E03F	 CELERY	\char"E040	 SO_2

17 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 the Math weights, Regular, Book and Bold. 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 hundrends 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 ×8). The glyph that appeared in TUGboat (see [AT]), being more realistic will be moved to a new font in the future with ornaments.

17.1 Bold Math

A complete math font, such as NewCM, contains all alphabetic characters in bold too. These characters are typically accessed using the `\mathbf` command. However, this is not true for bold versions of symbols. This creates difficulties when a user has some math in chapter or section titles, or when a user wants to create a poster with colored background. In the later case the Regular and Book weights look too light (especially with dark backgrounds) and one is in need of a real Bold Math font, that has everything in Bold.

NewCM, starting from version 6.0.0 provides an independent Bold Math font for the first time for a Computer Modern font family. Let us compare:

Book inline: $\sqrt[3]{x+y\pm 1} = \sum_{n=1}^{\infty} \int_K f_n(x)\,dx$ and the same in display

$$\sqrt[3]{x+y\pm 1} = \sum_{n=1}^{\infty} \int_K f_n(x)\,dx.$$

Bold inline: $\sqrt[3]{\boldsymbol{x+y\pm 1}} = \sum_{\boldsymbol{n=1}}^{\boldsymbol{\infty}} \int_{\boldsymbol{K}} \boldsymbol{f_n(x)\,dx}$ and the same in display

$$\sqrt[3]{\boldsymbol{x+y\pm 1}} = \sum_{\boldsymbol{n=1}}^{\boldsymbol{\infty}} \int_{\boldsymbol{K}} \boldsymbol{f_n(x)\,dx.}$$

Bold Math with colors inverted:

$$\sqrt[3]{\boldsymbol{x+y\pm 1}} = \sum_{\boldsymbol{n=1}}^{\boldsymbol{\infty}} \int_{\boldsymbol{K}} \boldsymbol{f_n(x)\,dx.}$$

In order to use the Bold font for chapter and section titles, when using the `default` or `olddefault` options of the `fontsetup` package, change the math version to bold with `\mathversion{bold}` *before* the commands for chapter and section and switch back to normal with `\mathversion{normal}` *afterwards*.

The `unicode-math` package, according to it's documentation has still some troubles with "versions" and the `range` options. These seem to affect at least the calligraphic and script math alphabets. In normal version for example the commands `\symcal` and `\symscr` work as expected, but when one switches to the bold version the `\symscr` fails. In this case one can use `\symbfscr`: $\boldsymbol{\mathcal{A}}$ $\boldsymbol{\mathscr{A}}$ (which was `$\symcal A$ $\symbfscr A$`).

17.2 Sans Serif Math

As of version 7.0.0, the family includes a full-featured Math Sans font. Up to now such a font did not exist in the CM family (although partial solutions existed), and it posed a serious problem for scientific writing especially in the preparation of presentations. The font supplied with NewCM covers all Unicode math slots but it also provides some new features. The lowercase Latin alphabet has been completely re-worked so that it really looks as it should when used for Math variables. The letters are

$$|a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z|$$

In Large

$$\mathsf{|a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z|}$$

Moreover the calligraphic capitals have been re-worked to match better with the sans serif style (see subsection 17.4 for how to select them)

$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

Same goes for the `\mathbb/\symbb` capital letters. They have been adjusted to match the Sans design, both in style and weight:

$$\mathbb{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

It is also worth noticing that the SansSerifed and the Serifed letters have swapped slots in the font. So if using the MathSans font and you write for example `$\symsf{ABCDabcd}$` you will get the Serifed version(!): ABCDabcd. This choice facilitates converting a document with Serifed fonts using Sans for emphasis or differentiation to keep these characteristics when changed to Sans.

One can see the Sans Serif Math font in action in the provided file `testmath-newcm.pdf` which comes form the $\mathcal{AMS}$-LaTeX bundle (here the logo is written in NewCMSans).

17.3 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 $+$, $-$, $\pm$, $\dot{+}$ etc are now provided in optical sizes, so that expressions such as

$$+^{+^+} \quad \perp^{\perp^\perp} \quad *^{*^*} \quad \frac{|P_{F^\perp}(K)|}{e^{(x+y)^*}}$$

appear on screen and print properly on printers. Zoom or print and compare with `latinmodern-math` font:

$$+^{+^+} \quad \perp^{\perp^\perp} \quad *^{*^*} \quad \frac{|P_{F^\perp}(K)|}{e^{(x+y)^*}}$$

17.4 Math Script

Calligraphic letters are accessed as usual with `\mathcal` or `\symcal`, producing

$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

However, mathematicians often need a second level of "scriptness". The fonts provide an alternative calligraphic, a script design at StylisticSet 1. For this to work one has to re-set the math font using
`\setmathfont[StylisticSet=1]{NewCMMath-Book.otf}`
(or the Regular version). So the following code

```
$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$
\setmathfont[StylisticSet=1]{NewCMMath-Book.otf}
$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$
$$
\mscra\mscrb\mscrc\mscrd\mscre\mscrf\mscrg\mscrh\mscri\mscrj
\mscrk\mscrl\mscrm\mscrn\mscro\mscrp\mscrq\mscrr\mscrs\mscrt
\mscru\mscrv\mscrw\mscrx\mscry\mscrz
$$
\setmathfont{NewCMMath-Book.otf}
$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$
```

produces

$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

$$\mathcal{abcdefghijklmnopqrstuvwxyz}$$

$$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

17.5 Math Kerning

Math kerning has been added to all NewCM Math fonts. This feature greatly improves Math typesetting, especially for the calligraphic letters but for regular letters as well, such as, Y or Γ.

Latin Modern	NewCM-Regular
$F_x\ Y_x\ \mathcal{N}_x\ \mathcal{J}_x\ \mathcal{G}_x\ \Gamma_x$	$F_x\ Y_x\ \mathcal{N}_x\ \mathcal{J}_x\ \mathcal{G}_x\ \Gamma_x$

This works with XeLaTeX but does not seem to work with LuaLaTeX. On the contrary now, kerning in math mode seems to work with LuaLaTeX but not with XeLaTeX! The next table shows the effect of kerning in math mode with calligraphic capital letters and capital Latin italic letters with period and comma (look closely). This table is compiled with LuaLaTeX.

Before	Current (with Lua)
$\mathcal{H}.^{\mathcal{H},^{\mathcal{H},}}\ H.H.^{H,H,^{H,H,}}$	$\mathcal{H}.^{\mathcal{H},^{\mathcal{H},}}\ H.H.^{H,H,^{H,H,}}$

Kerning in math mode does not work with XeLaTeX as said above. With LuaLaTeX we notice that it works with calligraphic capitals but not with Latin italic capitals. The reason looks to be the fact that these letters have italic correction enabled in the fonts (as they should). Italic correction breaks the

application of the kern. Now if italic correction is removed, by say `$H\/.$`, then the engine does not see the characters `H` and period as consecutive characters to apply the kern, and the kern is lost.

The code of the table above is as follows:

```
\begin{tabular}{c|c}
 Before & Current (with Lua)\\ \hline
  {\setmathfont{latinmodern-math.otf}\Large
${\symcal H.}^{{\symcal H,}^{\symcal H,}}$
${H\/.}{H.}^{{H\/,}{H,}^{H\/,H,}}$}
 & {\setmathfont{NewCMMath-Regular.otf}\Large
${\symcal H.}^{{\symcal H,}^{\symcal H,}}$
 ${H\/.}{H.}^{{H\/,}{H,}^{H\/,H,}}$}
\end{tabular}
```

We conclude this subsection realizing that there is no single engine (among XeLaTeX and LuaLaTeX) that supports all features.

17.6 Blackboard Bold

The fonts contain as default the AMS blackboard bold. These are:

$$\mathbb{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

$$\mathbb{abcdefghijklmnopqrstuvwxyz}$$

$$\mathbb{0123456789}\ \mathbb{\pi\gamma\Gamma\Pi}\sum\mathbb{Ddeij}$$

They also contain a blackboard bold that matches the design of Computer Modern but respecting the fact that most users have been used for a long period to the AMS bb design. Compare the default

$$\mathrm{R}\in\mathbb{R}\qquad \mathrm{Q}\in\mathbb{Q}$$

with the new design

$$\mathrm{R}\in\mathbb{R}\qquad \mathrm{Q}\in\mathbb{Q}$$

To access this design one 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

$$\mathbb{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

$$\mathbb{abcdefghijklmnopqrstuvwxyz}$$

$$\mathbb{0123456789}\ \mathbb{\pi\gamma\Gamma\Pi}\sum\mathbb{Ddeij}$$

If using the latest `fontsetup` then you can choose the NewCM blackboard bold with the option `newcmbb`.

17.6.1 Small Caps for Blackboard Bold

It happens one to need Blackboard Bold in Small Caps. For example, this may arise when running heads are in small caps and contain a Blackboard Bold symbol. Consider for example the case when a chapter is named "The Lebesgue measure on $\mathbb{R}$". Then the running head in small caps will look awkward:

THE LEBESGUE MEASURE ON $\mathbb{R}$

In such cases, a small caps version of the number sets is needed. NewCM provides the needed glyphs in `ss05`. So setting
`\setmathfont[StylisticSet=5]{NewCMMath-Book.otf}`
the above running head becomes:

THE LEBESGUE MEASURE ON $\mathbb{R}$

These glyphs are provided in the CM style (instead of the AMS bb) by enabling `ss03` too. So the command
`\setmathfont[StylisticSet=3,StylisticSet=5]{NewCMMath-Book.otf}`
will produce:

THE LEBESGUE MEASURE ON $\mathbb{R}$

The same is true for the Regular and Bold Math fonts.

17.7 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 LuaLaTeX. 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}
```

$$\int \begin{pmatrix} 1\\2\\3\\4\\5\\6\\7\\8\\9 \end{pmatrix}$$

17.8 Additional and alternative characters in Math

17.8.1 Alternative symbols

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

17.8.2 Additional symbols

It also provides four more arrows that correspond to the commands

`\nrightrightarrows` ($\not\rightrightarrows$) `\nleftleftarrows` ($\not\leftleftarrows$)

and

`\twoheadhookrightarrow` ($\hookrightarrow$) `\twoheadhookleftarrow` ($\hookleftarrow$)

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.

17.8.3 Additional operators

An operator for convolution of functions seems to have long been forgotten from both Unicode and the TEX world. The `default` and `olddefault` options of the `fontsetup` package define a new command `\convolution` which behaves just like the `\sum` and `\int` operators. The convolution of N functions in inline mode: $\mathop{\ast}_{n=1}^{N} f_n$ and the same in display mode:

$$\mathop{\ast}_{n=1}^{N} f_n.$$

17.8.4 "Smoother" changing radicals

One more radical size has been added that improves the way radical sizes change. Compare the previous state

$$\sqrt{q} \quad \sqrt{q_1^2} \quad \sqrt{q_1^{2^3}} \quad \sqrt{\frac{1}{2}}$$

with the new one

$$\sqrt{q}\quad\sqrt{q_1^2}\quad\sqrt{q_1^{2^3}}\quad\sqrt{\frac{1}{2}}.$$

17.9 Notes on Mathematics

Extensible tildes and hats produce different results with XeLaTeX and LuaLaTeX because these unicode engines treat differently the width of characters. In particular, XeLaTeX handles italic correction as part of the character width but LuaLaTeX does this only if a character follows. This affects how extensible accents like `\widehat` or `\widetilde` select the proper size. With XeLaTeX, `\widetilde{Y} \widetilde{X}` will give the expected result $\widetilde{Y}\widetilde{X}$; but with LuaLaTeX 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 LuaLaTeX one has to use a zero width character after a letter such as Y so that LuaLaTeX 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.

18 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		!	"	#	$	%	ℭ	'	(	)	*	+	,	-	.	-
U+0030 - 003F	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	-
U+0040 - 004F	ⓐ	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
U+0050 - 005F	P	Q	R	S	T	U	V	W	X	Y	Z	-	-	-	-	-
U+0060 - 006F	-	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
U+0070 - 007F	p	q	r	s	t	u	v	w	x	y	z	-	-	-	-	-
Latin-1 Supplement																
U+00A0 - 00AF	-	-	-	-	-	-	-	-	¨	-	-	-	-	-	-	-
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Table 1: NewCMUncial10-Book.otf *cont.*

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+00B0 - 00BF	-	-	-	-	-	-	-	·	-	-	-	-	-	-	-	-

Greek and Coptic

U+0370 - 037F	-	-	-	-	ʹ	͵	-	-	-	-	-	-	-	-	;	-
U+0380 - 038F	-	-	-	-	΄	΅	Ά	·	Έ	Η	Ι	-	Ο	-	Υ	Ω
U+0390 - 039F	ϊ	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
U+03A0 - 03AF	Π	Ρ	-	C	Τ	Υ	Φ	Χ	Ψ	ω	Ϊ	Ϋ	α	ε	η	ι
U+03B0 - 03BF	ΰ	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
U+03C0 - 03CF	π	ρ	ς	c	τ	υ	φ	χ	ψ	ω	ϊ	ϋ	ο	υ	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

19 The Aegean Numbers glyph complement

\aegeanseparator	𐄁	\aegeaneighthundred	𐄈
\aegeanseparatordot	𐄀	\aegeanninehundred	𐄉
\aegeancheckmark	×	\aegeanonethousand	𐄊
\aegeanone	𐄇	\aegeantwothousand	𐄋
\aegeantwo	𐄇𐄇	\aegeanthreethousand	𐄌
\aegeanthree	𐄇𐄇𐄇	\aegeanfourthousand	𐄍
\aegeanfour	𐄇𐄇𐄇𐄇	\aegeanfivethousand	𐄎
\aegeanfive	𐄏	\aegeansixthousand	𐄏
\aegeansix	𐄏𐄇	\aegeanseventhousand	𐄐
\aegeanseven	𐄏𐄇𐄇	\aegeaneightthousand	𐄑
\aegeaneight	𐄏𐄇𐄇𐄇	\aegeanninethousand	𐄒
\aegeanine	𐄏𐄇𐄇𐄇𐄇	\aegeantenthousand	𐄓
\aegeanten	-	\aegeantwentythousand	𐄔
\aegeantwenty	=	\aegeanthirtythousand	𐄕
\aegeanthirty	≡	\aegeanfourtythousand	𐄖
\aegeanfourty	==	\aegeanfiftythousand	𐄗
\aegeanfifty	≡=	\aegeansixtythousand	𐄘
\aegeansixty	≡≡	\aegeanseventythousand	𐄙
\aegeanseventy	≡≡	\aegeaneightythousand	𐄚
\aegeaneighty	≡≡	\aegeanninetythousand	𐄛
\aegeanninety	≡≡≡	\aegeanweightbaseunit	𐄜
\aegeanonehundred	∘	\aegeanweightfirstsubunit	𐄝
\aegeantwohundred	∘∘	\aegeanweightsecondsubunit	𐄞
\aegeanthreehundred	∘∘∘	\aegeanweightthirdsubunit	𐄟
\aegeanfourhundred	⁘	\aegeanweightfourthsubunit	𐄠
\aegeanfivehundred	⁘∘	\aegeandrymeasurefirstsubunit	𐄡
\aegeansixhundred	⁘∘∘	\aegeanliquidmeasurefirstsubunit	𐄢
\aegeansevenhundred	⁘⁘	\aegeansecondsubunit	𐄣
		\aegeanthirdsubunit	𐄤

References

[AT] Antonis Tsolomitis, *The NewComputerModern font family*, TUGboat Vol. 42, No. 1, 2021.

[IPArev] 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.