

Kp-fonts: OpenType version

Daniel FLIPO

6th October 2022

This bundle provides OpenType versions of Type1 Kp-fonts designed by Christophe Caignaert. See `Kpfonts-Doc-English.pdf` for the full documentation of the original Type1 fonts.

It is usable only with LuaTeX or XeTeX engines; it consists of sixteen Text OpenType fonts, a Roman family **KpRoman** in eight shapes and weights—*Regular, Italic, Bold, BoldItalic, Light, LightItalic, Semibold, SemiboldItalic*—, a Sans-Serif family **KpSans** and a TypeWriter family **KpMono**, each of them in four shapes and weights—*Regular, Italic, Bold* and *BoldItalic*— and five maths OpenType fonts **KpMath-Regular**, **KpMath-Bold**, **KpMath-Light**, **KpMath-Semibold** and **KpMath-Sans**.

KpRoman and **KpSans** families have small caps available in two sizes (SmallCaps and PetiteCaps), upper and lowercase digits (0123456789), ancient ligatures *ct*, *st* and *Q* a long-tailed capital *Q*. Superior and inferior digits and letters have been added to the OpenType **KpRoman** and **KpSans** fonts for footnotes' calls and abbreviations 1st, 2nd...

The support of text fonts covers Latin and Latin Extended A (U+0020 to U+017F) but neither IPA nor Greek nor Cyrillic alphabets.

For all maths fonts, Latin and Greek letters are available in Upright and Italic shapes, in Bold and Regular weights: α β Γ Δ , α β Γ Δ , α β Γ Δ , α β Γ Δ .

Blackboard Bold capitals are available in two shapes, Serif and Sans: `\mathbb{ABC}` prints either \mathbb{ABC} (default) or \mathbb{ABC} (option `[Style=bbsans]`) Commands `\mathcal{ABC}` and `\mathscr{ABC}` print respectively \mathcal{ABC} and \mathscr{ABC} while `\mathfrak{ABCabc}` prints \mathfrak{ABCabc} .

File `unimath-kpfonts.pdf` shows the full list of Unicode maths symbols provided by Kp-fonts, compared with other common maths fonts. More symbols, specific to Kp-fonts, are listed in section 3.2.

A style file `kpfont-otf.sty` is provided to load Kp-fonts easily. It is derived from `kpfont.sty` but options differ.

Please beware of the *experimental* status of the current version (0.45).

All fonts are covered by OFL licence, style file and documentation are under LPPL-1.3 licence.

1 Loading kpfonts-otf.sty

For users of the original kpfonts.sty package, the easiest way to try the OpenType version is to load kpfonts-otf.sty:

```
\usepackage[ options ]{kpfonts-otf}
```

this loads unicode-math, fontspec and realscripts, and defines KpRoman (Regular or Light depending on options), KpSans and KpMono as Text fonts, KpMath (Regular or Light depending on options) as maths fonts.

kpfonts-otf.sty also defines all symbols available in latexsym and amssymb under the same names¹ and a bunch of Kp-fonts specifics symbols.

1.1 Global options for both text and maths

light: switches to *light* fonts, metrics are unchanged; *light* fonts might not look perfect on screen but they print fine.

Please compare *regular* (left) and *light* fonts (right):

Normal or light? Just a matter of taste. $E = mc^2$	Normal or light? Just a matter of taste. $E = mc^2$
--	--

<i>Normal or light? Just a matter of taste</i>	<i>Normal or light? Just a matter of taste</i>
--	--

Normal or light? Just a matter of taste	Normal or light? Just a matter of taste
--	--

<i>Normal or light? Just a matter of taste</i>	<i>Normal or light? Just a matter of taste</i>
---	---

nomath: load neither unicode-math nor KpMath fonts; useful for documents without maths, or to choose other maths fonts.

notext: do not change any Text font, use the defaults.

nosf: do not change Sans-Serif Text fonts, use the defaults.

nott: do not change Typewriter Text fonts, use the defaults.

onlyrm: equivalent to the last two combined.

fulloldstyle: equivalent to options oldstyle and oldstylemath.

fulloldstylenums: equivalent to options oldstylenums and oldstylenumsmath.

1.2 Options for text fonts only

lighttext: switches to *light* Text fonts.

Two more weights are provided by kpfonts-otf.sty: with the *lighttext* (or *light*) option, *Semibold* and *Extrabold* vs *Light* and *Semibold* without it. These weights are available through \ltseries, \sbseries and \ebseries commands to be used in a group or alternatively through one argument commands \textlt{ }, \textsb{ } and \texteb{ }.

{\sbseries\itshape Foo} or \textsb{\textit{Foo}} print *Foo*.

1. Unicode names often differ from AMS names.

oldstylenums: provides lowercase digits as a default.

To get uppercase digits locally: `{\addfontfeature{Numbers=Lining} 123}`.

Examples, upright, italic, bold and bolditalic:

— 0123456789!
— 0123456789!
— 0123456789!
— 0123456789!

oldstyle: provides lowercase digits as a default, long-tailed Q (Quebec) and (for Roman and Sans-Serif fonts only) old style ligatures ‘Œ’ and ‘Œ’.

Examples:

— *Quest* for an attractive font!
— *Quest* for an attractive font!
— **Quest** for an attractive font!
— QUEST FOR AN ATTRACTIVE FONT!
— *Quest* for an attractive font!
— QUEST FOR AN ATTRACTIVE FONT!

veryoldstyle: same as option **oldstyle** but the round ‘s’ is replaced by the long one ‘ſ’, unless it ends a word (then it remains ‘s’)² and ancient ligatures fi, fl, ft are activated. Coding `s=` prints a round ‘s’ anytime; in most cases this coding is not necessary with LuaTeX, f.i. `\textit{some of Highlands’ mysterious castles...}` will print *some of Highlands’ mysterious castles...* which is correct; with XeTeX an = sign must be added at the end of `mysterious=` to get a round ‘s’ there : final ‘s’ followed by a punctuation sign is never turned into ‘ſ’, when followed by a space it is a trickier, see option **longs** below.

longs inhibits the transformation of an ‘s’ followed by a space into ‘ſ’. This option, available only with LuaTeX, is silently switched on by **veryoldstyle**; it is used explicitly in the present document to deal with the examples of long ‘s’ shown in the preceding item.

largesmallcaps: prints larger SMALL CAPS than the default ones (PETITE CAPS).

altfligs : prints alternative shapes for ligatures fi, fl, ffi, ffl instead of fi, fl, ffi, ffl.

germandbls : \SS prints SS instead of ß (capital *Eszett*), ditto for small/petite caps.

eurosym : replaces the Euro character (€) by the official symbol (€) (available in regular, italic, bold and bolditalic).

harfbuzz : switches `Renderer=Harfbuzz` for HBLuaTeX engine; up to version 0.34, this renderer was silently activated but seldom useful.

1.3 Options for maths fonts *only*

lightmath: uses *light* maths fonts.

2. The OpenType `calt` feature is used to catch ‘s’ ending a word.

bbsans: command `\mathbb` prints Sans-Serif Blackboard Bold capitals with Serif fonts too: compare $\mathbb{C}, \mathbb{K}, \mathbb{N}, \mathbb{Q}, \mathbb{R}, \mathbb{Z}$, with $\mathbb{C}, \mathbb{K}, \mathbb{N}, \mathbb{Q}, \mathbb{R}, \mathbb{Z}$ (default).

frenchstyle: Latin uppercase letters and all Greek letters are printed upright, only lowercase Latin letters are printed in italics; this follows the French typographic usage.

oldstylenumsmath: prints lowercase digits in maths (default is uppercase).

narrowiints: prints condensed repeated integrals :

$\int\!\!\!\int$ and $\int\!\!\!\int\!\!\!\int$ instead of \iint and \iiint (default).

partialup: the `\partial` symbol is printed upright ∂ instead of ∂ .

fancyReIm: commands `\Re` and `\Im` print \Re and \Im respectively instead of \mathbb{R} and \mathbb{I} .

tight: horizontal spaces tightened in maths mode (same settings as `fourier-otf`).

noDcommand: do not define `\D` to avoid incompatibilities with other packages.

Please note that the **mathcal** option has been deleted: commands `\mathcal{ABC}` and `\mathscr{ABC}` now print ABC and \mathscr{ABC} respectively when `kpfonts-otf.sty` is loaded.

2 Another way to load Kp-fonts

Loading Kp-fonts through `kpfonts-otf.sty` offers only a limited choice of options; the standard commands `\setmainfont`, `\setsansfont`, `\setmonofont`, `\setmathfont` offer much more flexibility.

On the other hand, `kpfonts-otf.sty` defines a lot of useful commands to access AMS and specific Kp-fonts symbols. Loading `kpfonts-otf` with the `symbols` option enables to get all these commands defined without loading any font:

```
\usepackage[symbols]{kpfonts-otf}
```

Please note that `unicode-math`³ (and `fontspec`) *are loaded* by this procedure, no need to do it again, unless specific options are required, then `unicode-math` has to be loaded *before* `kpfonts-otf`, f.i.:

```
\usepackage[math-style=ISO,bold-style=upright]{unicode-math}
\usepackage[symbols]{kpfonts-otf}
```

Then, it is up to the user to load Kp-fonts with whatever option he/she likes using commands

```
\set...font{font}[options].
```

For documents requiring no maths fonts, loading `fontspec` and using the `\set...font` commands is enough, no need to load `kpfonts-otf` at all.

3. A careful reading of both manuals `unicode-math.pdf` and `fontspec.pdf` (available in all TeX distributions) is required in order to take full advantage of these packages.

2.1 Options for Text fonts

Here are the options available for Text Kp-fonts:

Numbers=Lowercase to get lowercase digits 1,2,3 instead of 1,2,3; the default is **Numbers=Lining**.

SmallCapsFeatures = {Letters=SmallCaps} the `\textsc{}` command will print larger SMALL CAPS than the default PETITE CAPS.

The default setting⁴, is **SmallCapsFeatures = {Letters=PetiteCaps}**.

Ligatures=TeX (default) ' !` ?` -- --- print respectively ' i ÿ - —.

Ligatures=Common (default) automatic ligatures ff ffi ffl fi fl.

StylisticSet=1 provides an alternative for glyphs ffi ffl fi fl (ff is unchanged).

Ligatures=Required : adds ft and tt ligatures.

Ligatures=Rare adds ct and st ligatures.

Style=Swash to get the long-tailed capital Q: Queen, also in small caps (both sizes): QUEEN and QUEEN.

Style=Historic replaces any instance of 's' by the long variant f. It is still possible to get a round 's' coding it as 's=' specially at end of words; see options `veryoldstyle` and `longs` p. 3 for more details.

Ligatures=Historic switches specific ligatures for the long f: fi, fl, ft.

StylisticSet=2 : \SS prints SS instead of ß (capital *Eszett*), ditto for small/petite caps.

StylisticSet=3 : replaces the Euro character (€) by the official symbol (€) (available in regular, italics, bold and bolditalic).

Options may be are chosen for each font, say:

```
\setmonofont{KpMono}[Numbers=Lowercase,Style=Historic]
```

but can also be shared by different typefaces:

```
\defaultfontfeatures+[KpRoman,KpSans,KpMono]{Numbers=Lowercase}
\defaultfontfeatures+[KpRoman,KpSans]{%
  Ligatures = Rare,
  Style = Swash,
  SmallCapsFeatures = {Letters=PetiteCaps},
}
\setmainfont{KpRoman}
\setsansfont{KpSans}
\setmonofont{KpMono}
```

Notes : 1. `\setmonofont{KpMono}`, `\setsansfont{KpSans}`, `\setmainfont{KpRoman}` rely on files `KpMono.fontspec`, `KpSans.fontspec` and `KpRoman.fontspec` installed by Kpfonts.

4. Changed in v0.37 to match the original kpfonts package.

2. Note the + ending `\defaultfontfeatures+` : options are *added*, not overwriting any other (including those of `fontspec.cfg`).

3. Options can be gathered: `Ligatures={Rare, Historic}` (with braces) is equivalent to `Ligatures=Rare` and `Ligatures=Historic`.

4. These options can also be switched on and off *locally* using `\addfontfeatures` inside a group, f.i. to print lowercase digits 1234567890 with a font loaded with option `Numbers=Lining`:

```
{\addfontfeatures{Numbers=Lowercase}1234567890}
```

Actually, a shortcut is available in this case: `\oldstylenums{1234567890}`.

5. With the **KpRoman**, it is possible to define two more weights *Light* and *Semibold* borrowed from **KpLight**:

```
\newfontfamily\KpLight{\KpLight}[<same options as KpRoman>]
\newcommand*\ltseries{\KpLight}
\newcommand*\sbseries{\KpLight\bfseries}
\DeclareTextFontCommand{\textlt}{\ltseries}
\DeclareTextFontCommand{\textsb}{\sbseries}
```

These weights are then available through `\ltseries`, `\sbseries` commands to be used in a group or alternatively through one argument commands `\textlt{}` and `\textsb{}`.

With the **KpLight**, weights *Semibold* and *Extrabold* can be defined similarly.

2.2 Options for maths fonts

The following options can be passed either to `unicode-math`⁵ or to `\setmathfont{}`:

math-style = ISO, TeX (default), french, upright;

bold-style = ISO, TeX (default), upright;

partial = upright (default italic);

nabla = italic (default upright);

Seven ‘Style Variants’ are available with Kp-fonts, here are the first three:

Style=mathcal (+ss01) commands `\mathcal{}` and `\mathscr{}` print *ABC* instead of *ℳℳℳ* (default), see note below;

Style=bbsans (+ss02) `\mathbb{}` prints Sans-Serif Blackboard bold capitals *ℳℳℳ* for Serif maths fonts **KpMath-Regular** and **KpMath-Light** instead of *ℳℳℳ* ;

Style=narrowiints (+ss03) provides condensed repeated integrals: *∫∫∫* instead of *∫∫∫* (default).

Note: if you want commands `\mathcal{ABC}` and `\mathscr{ABC}` to print *ABC* and *ℳℳℳ* respectively, you can use `unicode-math`’s option range this way:

```
\setmathfont{KpMath-Regular}[options]
\setmathfont{KpMath-Regular}[range={cal,bfcal},StylisticSet=1]
```

5. See the manual `unicode-math.pdf`.

Both lines are mandatory: the first one loads **KpMath** as usual while the second one modifies `\mathcal{}` command's output.

The next four tables present the other Style Variants available:

Table 1 – Style=leqslant (+ss04)

Command	Default	Variant
<code>\leq</code>	\leq	\leqslant
<code>\geq</code>	\geq	\geqslant
<code>\nleq</code>	\nleq	\nleqslant
<code>\ngeq</code>	\ngeq	\ngeqslant
<code>\leqq</code>	\leqq	\leqslant
<code>\geqq</code>	\geqq	\geqslant
<code>\eqless</code>	\lessgtr	\lessgtr
<code>\eqgtr</code>	\gtrless	\gtrless
<code>\lesseqgtr</code>	\lesseqgtr	\lesseqgtr
<code>\gtreqless</code>	\gtreqless	\gtreqless
<code>\lesseqqgtr</code>	\lesseqqgtr	\lesseqqgtr
<code>\gtreqqless</code>	\gtreqqless	\gtreqqless

Table 2 – Style=smaller (+ss05)

Command	Default	Variant
<code>\mid</code>	\mid	\mid
<code>\nmid</code>	\nmid	\nmid
<code>\parallel</code>	\parallel	\parallel
<code>\nparallel</code>	\nparallel	\nparallel
<code>\parallelslant</code>	\parallel	\parallel
<code>\nparallelslant</code>	\nparallel	\nparallel

Table 3 – Style=subsetneq (+ss06)

Command	Default	Variant
<code>\subsetneq</code>	\subsetneq	\subsetneq
<code>\supsetneq</code>	\supsetneq	\supsetneq
<code>\subsetneqq</code>	\subsetneqq	\subsetneqq
<code>\supsetneqq</code>	\supsetneqq	\supsetneqq

Table 4 – Style=parallelslant (+ss07)

Command	Default	Variant
<code>\parallel</code>	\parallel	\parallel
<code>\nparallel</code>	\nparallel	\nparallel
<code>\shortparallel</code>	\parallel	\parallel
<code>\nshortparallel</code>	\nparallel	\nparallel

Example: switching styles 4 (leqslant) and 6 (subsetneq) can be achieved coding either `\setmathfont{KpMath-Regular.otf}[StylisticSet={4,6}]` or `\setmathfont{KpMath-Regular.otf}[Style={leqslant,subsetneq}]` but this second syntax is available only if `kpfonts-otf.sty` has been loaded (eventually with the `symbols` option).

Table 5 on the following page shows the available ‘Glyphs Variants’:

Example: with `\setmathfont{KpMath-Regular.otf}[CharacterVariant={3,6}]` commands `\epsilon` and `\phi` print ϵ and ϕ instead of ϵ and ϕ . The same is true of course for all shapes and and weights (upright, bold, bolditalic, sans-serif, etc.): f.i. with `math-style=french`, `\epsilon` and `\phi` print ϵ and ϕ (upright).

Note about `\hbar`: `unicode-math` defines `\hbar` as `\hslash` (U+210F) while `amsmath` provides two different glyphs (italic h with horizontal or diagonal stroke). `kpfonts-otf` now follows `unicode-math`; the italic h with horizontal stroke can be printed using `\hslash` or `\hbar` together with character variant `cv01` or with `\mithbar` (replacement for AMS’ command `\hbar`).

Table 5 – Glyphs Variants

	Default		Variant		Command	
cv00	\Re	\mathbb{I}	\Re	\mathbb{I}	$\backslash\mathrm{Re}$	$\backslash\mathrm{Im}$
cv01	\hbar		\hbar		$\backslash\mathrm{hslash}$	or $\backslash\mathrm{hbar}$
cv02	\emptyset		\emptyset		$\backslash\mathrm{emptyset}$	
cv03	ϵ		ϵ		$\backslash\mathrm{epsilon}$	
cv04	κ		κ		$\backslash\mathrm{kappa}$	
cv05	π		ϖ		$\backslash\mathrm{pi}$	
cv06	ϕ		φ		$\backslash\mathrm{phi}$	
cv07	ρ		ρ		$\backslash\mathrm{rho}$	
cv08	σ		ς		$\backslash\mathrm{sigma}$	
cv09	θ		ϑ		$\backslash\mathrm{theta}$	
cv10	Θ		Θ		$\backslash\mathrm{Theta}$	

3 Specific commands defined in `kpfonts-otf.sty`

3.1 Integrals

Kp-fonts offers variants for integral symbols suitable for indefinite integrals, they are coded as `\varint`, `\variint`, `\variiint`, `\variiiiint` and `\varidotsint`. Compare $\int f(t) dt$ and $\int f(t) dt$ and also

$$\int f(t) dt \quad \text{and} \quad \int f(t) dt$$

`\D{...}` prints an upright ‘d’ and improves kernings around the differential element: `\displaystyle\varint f(t)\D{t}` prints $\int f(t) dt$.

3.2 Specific maths symbols

The next tables present symbols unavailable as Unicode characters, they are coded in Kp-fonts’ private zone.

$\backslash\mathrm{mmapsto}$	\mapsto	$\backslash\mathrm{longmmapsto}$	\longmapsto
$\backslash\mathrm{mmapsfrom}$	\mapsfrom	$\backslash\mathrm{longmmapsfrom}$	\longmapsfrom
$\backslash\mathrm{Mmapsto}$	\Mmapsto	$\backslash\mathrm{Longmmapsto}$	\Longmmapsto
$\backslash\mathrm{Mmapsfrom}$	\Mmapsfrom	$\backslash\mathrm{Longmmapsfrom}$	\Longmmapsfrom
$\backslash\mathrm{leftrightrightarrow}$	\leftrightarrow	$\backslash\mathrm{leadsto}$	\leadsto
$\backslash\mathrm{boxright}$	\boxrightarrow	$\backslash\mathrm{boxleft}$	\boxleftarrow
$\backslash\mathrm{circleright}$	\circrightarrow	$\backslash\mathrm{circleleft}$	\circleftarrow
$\backslash\mathrm{Diamondright}$	\diamondrightarrow	$\backslash\mathrm{Diamondleft}$	\diamondleftarrow
$\backslash\mathrm{boxdotright}$	\boxdotrightarrow	$\backslash\mathrm{boxdotleft}$	\boxdotleftarrow
$\backslash\mathrm{circledotright}$	\circledrightarrow	$\backslash\mathrm{circledotleft}$	\circledleftarrow
$\backslash\mathrm{Diamonddotright}$	\diamondrightarrow	$\backslash\mathrm{Diamonddotleft}$	\diamondleftarrow

$\backslash\boxRight$	\Rightarrow	$\backslash\boxLeft$	\Leftarrow
$\backslash\boxdotRight$	\Rightarrow	$\backslash\boxdotLeft$	\Leftarrow
$\backslash\DiamondRight$	\Leftrightarrow	$\backslash\DiamondLeft$	\Leftrightarrow
$\backslash\DiamonddotRight$	\Leftrightarrow	$\backslash\DiamonddotLeft$	\Leftrightarrow
$\backslash\multimapdot$	\multimap	$\backslash\multimapdotinv$	\multimap
$\backslash\multimapdotboth$	\multimap		
$\backslash\multimapbothvert$	\multimap	$\backslash\multimapdotbothvert$	\multimap
$\backslash\multimapdotbothAvert$	\multimap	$\backslash\multimapdotbothBvert$	\multimap
$\backslash\capplus$	\capplus	$\backslash\squcupplus$	\sqcupplus
$\backslash\parallelslant$	\parallel	$\backslash\colonsim$	\sim
$\backslash\parallelbackslant$	\parallel	$\backslash\Colonsim$	\sim
$\backslash\eqqColon$	\equiv	$\backslash\Colondash$	\dashv
$\backslash\strictif$	\rightarrow	$\backslash\strictfi$	\leftarrow
$\backslash\circledvee$	\circledvee	$\backslash\circledwedge$	\circledwedge
$\backslash\openJoin$	\times	$\backslash\opentimes$	\times
$\backslash\lambdash$	λ	$\backslash\lambdabar$	λ
		$\backslash\Wdash$	\Wdash
$\backslash\idotsint$	\int		
$\backslash\ointclockwise$	\oint	$\backslash\varointctrackwise$	\oint
$\backslash\oiintclockwise$	\oiint	$\backslash\oiintctrackwise$	\oiint
$\backslash\varoiintclockwise$	\varoiint	$\backslash\varoiintctrackwise$	\varoiint
$\backslash\oiiiiintclockwise$	\oiiiiint	$\backslash\oiiiiintctrackwise$	\oiiiiint
$\backslash\varoiiiiintclockwise$	\varoiiiiint	$\backslash\varoiiiiintctrackwise$	\varoiiiiint
$\backslash\squint$	\sint	$\backslash\squiiint$	\sint

The full list of Unicode symbols available with Kp-fonts is shown in file `unimath-kpfonts.pdf`.

3.3 Wide accents

— `\widehat` and `\widetilde`

\hat{x} \widehat{xx} \widehat{xxx} \widehat{xxxx} \widehat{xxxxx} \widehat{xxxxxx} \tilde{x} \widetilde{xx} \widetilde{xxx} \widetilde{xxxx} \widetilde{xxxxx} \widetilde{xxxxxx}

— `\overline` and `\underline`

$$\overline{x} \quad \overline{xy} \quad \overline{xyz} \quad \overline{A \cup B} \quad \overline{A \cup (B \cap C) \cup D} \quad \underline{m+n+p}$$

— `\wideoverbar`, `\widecheck` and `\widebreve`

$$\overline{\overline{x}} \quad \overline{\overline{xy}} \quad \overline{\overline{xyz}} \quad \check{x} \quad \check{\overline{xxxx}} \quad \check{\overline{xxxxxxxx}} \quad \check{x} \quad \check{\overline{xxxx}} \quad \check{\overline{xxxxxxxx}}$$

— `\overparen` and `\underparen`

$$\widehat{x} \quad \widehat{xy} \quad \widehat{xyz} \quad \widehat{\overset{\circ}{A \cup B}} \quad \widehat{\overset{\circ}{A \cup (B \cap C) \cup D}} \quad \widehat{\overset{2}{x+y}} \quad \widehat{\overset{26}{a+b+\dots+z}}$$

$$\underline{x} \quad \underline{xz} \quad \underline{xyz} \quad \underline{\overset{2}{x+z}} \quad \underline{\overset{26}{a+b+\dots+z}}$$

— `\overbrace` and `\underbrace`

$$\overbrace{a} \quad \overbrace{ab} \quad \overbrace{abc} \quad \overbrace{abcd} \quad \overbrace{abcde} \quad \overbrace{a+b+c}^3 \quad \overbrace{a+b+\dots+z}^{26}$$

$$\underbrace{a} \quad \underbrace{ab} \quad \underbrace{abc} \quad \underbrace{abcd} \quad \underbrace{abcde} \quad \underbrace{a+b+c}_3 \quad \underbrace{a+b+\dots+z}_{26}$$

— `\overrightarrow` and `\overleftarrow`

$$\overrightarrow{v} \quad \overrightarrow{M} \quad \overrightarrow{vv} \quad \overrightarrow{AB} \quad \overrightarrow{ABC} \quad \overrightarrow{ABCD} \quad \overrightarrow{ABCDEFGH}.$$

$$\overleftarrow{v} \quad \overleftarrow{M} \quad \overleftarrow{vv} \quad \overleftarrow{AB} \quad \overleftarrow{ABC} \quad \overleftarrow{ABCD} \quad \overleftarrow{ABCDEFGH}$$

— Enfin `\widearc` and `\widearcarrow` (or `\overrightarrowarc`)

$$\widearc{AMB} \quad \widearcarrow{AMB}$$

3.4 Maths Versions

Different versions of the **KpMath** fonts may be defined in the document's preamble:

```
\setmathfont{KpMath-Regular.otf}[version=base, options ]
```

```
\setmathfont{KpMath-Bold.otf}[version=bold, options ]
```

```
\setmathfont{KpMath-Semibold.otf}[version=semibold, options ]
```

```
\setmathfont{KpMath-Sans.otf}[version=sans, options ]
```

```
\setmathfont{KpMath-Light.otf}[version=light, options ]
```

then, it is easy to switch from one version to another one with `\mathversion{name}`.

Example ⁶ :

```
\setmathfont{KpMath-Regular.otf}[Style=leqslant, CharacterVariant=3]
\setmathfont{KpMath-Bold.otf}[version=bold,
    Style=leqslant, CharacterVariant=3]
\setmathfont{KpMath-Sans.otf}[version=sans,
    Style=leqslant, CharacterVariant=3]
```

Here is the same equation in three versions, normal, bold and Sans-Serif:

$$\mathbb{E}_i(N_i) = \sum_{n \geq 1} P_i(N_i \geq n) = \frac{\varepsilon_i}{1 - \varepsilon_i} < +\infty$$

`\mathversion{bold}`

$$\mathbb{E}_i(N_i) = \sum_{n \geqslant 1} P_i(N_i \geqslant n) = \frac{\varepsilon_i}{1 - \varepsilon_i} < +\infty$$

`\mathversion{sans}`

$$\mathbb{E}_i(N_i) = \sum_{n \geqslant 1} P_i(N_i \geqslant n) = \frac{\varepsilon_i}{1 - \varepsilon_i} < +\infty$$

6. Option `CharacterVariant=3` changes ϵ into ε .