

Kp-Fonts

The JOHANNES KEPLER project

Release 2.0

CHRISTOPHE CAIGNAERT
c.caignaert@free.fr

This 2.0 release is the major evolution of *kpfonts*, the Kepler's project package. We get :



- the *light* option with lighter math and text fonts and the same metrics;
- the *oldstyle* and *veryoldstyle* family options for typesetting respectively with oldstyle numbers and like in 1700.

1. Kp-Fonts

1.1. What is Kp-Fonts?

Kp-Fonts provides a full set of fonts for LaTeX typesetting, including roman, sans-serif et fixed-width fonts, as well as a set of mathematics symbol fonts with, regular and bold, all the common symbols and all those of the AMS and more...

The typical feature of these fonts is to have a basic and dynamic shape. For instance, look at the roman upright "a" glyphs:

Kp-Fonts	Kp-light	CM	Palatino	Utopia	Times
					

This LaTeX package est distributed with a GPL licence.

Kp-Fonts doesn't require any other font package and is fully compatible with *amsmath* and *textcomp*. Kp-fonts loads these two packages.

*You haven't to load **amsmath** and **textcomp** packages*

To use it, you just have to add `\usepackage{kpfonts}` in your document preamble, if necessary with the options described below.

Besides, the aim is to get a good compatibility with different sets of math font packages.

1.2. Text fonts

- The encodings `τ1`, `or1` et `rs1` are fully available, except for the upper-case symbol `\SS` in teletype fonts;
- Roman fonts are available with upright, small caps, italic, slanted and oldstyle shape, all with regular, bold and bold extended weight;
- Sans-serif fonts are available with upright, small caps and slanted shape, all with regular, bold and bold extended weight;
- Teletypes fonts are available with upright and slanted shape, all with regular and bold weight;

*The default weight is **bold**, not **bold extended**.*

1.3. Math fonts

- The `\boldmath` command is fully installed and Kp-Fonts **doesn't need** "poor man bold" glyphs;
- Kp-Fonts provides upright and slanted greeks: $\alpha\beta\gamma\Gamma\Delta$ and $\alpha\beta\gamma\Gamma\Delta$;
- `\mathbb` provides $\mathbb{A}\mathbb{B}\mathbb{C}$, characters designed from capital upright roman;
- `\mathcal` provides $\mathcal{A}\mathcal{B}\mathcal{C}$, slightly altered fonts from `cm`;
- With `\mathscr`, you get $\mathscr{A}\mathscr{B}\mathscr{C}$, without any extra package; these letters are designed from traditional fonts;
- `\mathfrak` provides $\mathfrak{A}\mathfrak{B}\mathfrak{C}\mathfrak{a}\mathfrak{b}\mathfrak{c}$, still from traditional fonts, altered to be more readable.

2. Options

2.1. Global options

nomath: Kp-Fonts doesn't modify default mathematical fonts;

notext: Kp-Fonts doesn't modify default text fonts;

nosf: Kp-Fonts doesn't modify default sans serif fonts (text and math);

nott: Kp-Fonts doesn't modify default fixed width fonts (text and math);

onlyrm: equivalent to the last two options;

light: then, you use the light version of the fonts. The metrics are the same. The display is not very good, but the print is fine if you like light fonts;

fulloldstyle: equivalent to both `oldstyle` and `oldstylemath`;

fullveryoldstyle: equivalent to both `veryoldstyle` and `veryoldstylemath`.

With both the two first options, Kp-Fonts does nothing...

This text is typesetted with default fonts, and below you can see an example of the light fonts set, upright and italic, both regular and bold :

While the high-level font commands are intended for use in a document, the low-level commands...

While the high-level font commands are intended for use in a document, the low-level commands...

While the high-level font commands are intended for use in a document, the low-level commands...

While the high-level font commands are intended for use in a document, the low-level commands...

2.2. Text fonts options

These options affect only text fonts.

oldstyle: provides here "Q", and oldstyle numbers by default.

With roman and sans-serif fonts, you get the old ligatures "ſt" and "ſt". Oldstyle also provides the small capital "Q".

A new command `\classicstylenums` allows you to get usual numbers in roman fonts.

Some examples:

- Queer font, queer actual aesthetic!
- *Queer font, queer actual aesthetic!*
- **Queer font, queer actual aesthetic!**
- **QUEER FONT, QUEER ACTUAL AESTHETIC!**
- *Queer font, queer actual aesthetic!*
- **QUEER FONT, QUEER ACTUAL AESTHETIC!**

veryoldstyle: Same as the `oldstyle` option except the round "s" is replaced with the long "f".

You can get the round "s" using the ligature "s=", often used at the end of the word.

Example: `\textit{costs=}` gives "cofts"

rmx then , you get six weights, with the corresponding fonts:

light	l	medium in light mode
medium	m	medium in default mode
semi-bold	sb	bold in light mode
bold	b	bold in default mode
semi-bold extended	sbx	bold extended in light mode
bold extended	bx	bold extended in default mode

In this case, the *light* option affects only math fonts.

This option is not my favorite because the default and light families are cousins but not sisters.

This last option affects only roman fonts.

2.3. Greek letters in math mode

uprightgreeks: *lowercase* greeks are upright, $\alpha\beta\gamma$;

slantedGreeks: *Uppercase* greeks are slanted, $\Gamma\Delta$.

2.4. Math font options

nomathscript: Kp-Fonts doesn't install the `\mathscr` command; you need it if you use `\mathrsfs` package;

mathcalasscript: swaps the `\mathcal` and `\mathscr` fonts;

classicReIm: the `\Re` and `\Im` commands produce \Re and \Im . In default of, Kp-Fonts provides \Re and \Im ;

uprightRoman: the *Uppercase* romans are upright.

frenchstyle: equivalent to both the options `uprightRoman` et `uprightgreeks`, uppercase romans and lowercase greeks are upright, usual French style when typesetting mathematics; lowercase romans remain slanted;

upright : synonymous with the previous option;

oldstylemath: provides the oldstyle numbers in default and changes the `\mathrm`, `\mathsf` and `\mathtt` fonts; they run as text fonts with oldstyle option;

veryoldstylemath : same as `oldstylemath` except the round "s" is replaced with the long "f".

Note that, when using the `oldstylemath`, `veryoldstylemath`, `fulloldstyle` or `fullveryoldstyle` options, in math mode, as the numbers 3, 4, 5, 7 et 9 have a depth, superscripts are moved up.

3. Use

3.1. Greek letters

We have described above the two options that alter the default greek letters. Otherwise, you can get the other greek letters using `\otheralpha`, `\otherGamma`. The result depends on the used options.

What's more, for uppercase, you can use `\varGamma` synonymous with the previous.

Finally, the `\alphas1`, `\alphaup`, `\Gammaaup` or `\Gammaas1` commands always give you the letter you want, whatever the chosen options...

3.2. Standard symbols

All the standard symbols are present, including all the symbols of the `amssymb` package, all with regular and bold weight.

For instance: the standard symbols \leq \bullet \pm or AMS symbols \Rightarrow \subsetneq \leqslant

and in bold: \leq \bullet \pm or \Rightarrow \subsetneq \leqslant .

Obviously, you get the usual constructions, both regular and bold: \mapsto \rightarrow , \mapsto \rightarrow

3.3. Extra symbols

Kp-Fonts provides a lot of other symbols and a lot of negative symbols not displayed here.

<code>\mappedfrom</code>	\leftarrow	<code>\longmappedfrom</code>	\longleftarrow
<code>\Mapsto</code>	\Rightarrow	<code>\Longmapsto</code>	\Longrightarrow
<code>\Mappedfrom</code>	\Leftarrow	<code>\Longmappedfrom</code>	\Longleftarrow
<code>\mmapsto</code>	\Rrightarrow	<code>\longmmapsto</code>	\Rrightarrow
<code>\mmappedfrom</code>	\Leftarrow	<code>\longmmappedfrom</code>	\Longleftarrow
<code>\Mmapsto</code>	\Rrightarrow	<code>\Longmmapsto</code>	\Longrightarrow
<code>\Mmappedfrom</code>	\Leftarrow	<code>\Longmmappedfrom</code>	\Longleftarrow

<code>\dashleftarrow</code>	\dashleftarrow	<code>\dashrightarrow</code>	\dashrightarrow
<code>\dashleftrightharrow</code>	\dashleftrightharrow	<code>\leftsquigarrow</code>	\leftsquigarrow
<code>\Nearrow</code>	\nearrow	<code>\Searrow</code>	\searrow
<code>\Nwarrow</code>	\nwarrow	<code>\Swarrow</code>	\swarrow
<code>\varepsilon\emptyset</code>	\emptyset		
<code>\leadsto</code>	\leadsto	<code>\leadsto</code>	\leadsto

You can combine these last two symbols:

`\leadsto\leadsto\leadsto\leadsto` give $\leadsto\leadsto\leadsto\leadsto$

<code>\boxright</code>	\boxrightarrow	<code>\Diamondright</code>	\Diamondrightarrow
<code>\circlearrowright</code>	\circlearrowright	<code>\boxleft</code>	\boxleftarrow
<code>\Diamondleft</code>	\Diamondleftarrow	<code>\circleleft</code>	\circleleftarrow
<code>\boxdotright</code>	\boxrightarrow	<code>\Diamonddotright</code>	\Diamondrightarrow
<code>\circleddotright</code>	\circledrightarrow	<code>\boxdotleft</code>	\boxleftarrow
<code>\Diamonddotleft</code>	\Diamondleftarrow	<code>\circleddotleft</code>	\circledleftarrow
<code>\boxRight</code>	\boxrightarrow	<code>\boxLeft</code>	\boxleftarrow
<code>\boxdotRight</code>	\boxrightarrow	<code>\boxdotLeft</code>	\boxleftarrow
<code>\DiamondRight</code>	\Diamondrightarrow	<code>\DiamondLeft</code>	\Diamondleftarrow
<code>\DiamonddotRight</code>	\Diamondrightarrow	<code>\DiamonddotLeft</code>	\Diamondleftarrow

<code>\multimap</code>	\multimap	<code>\multimapinv</code>	\multimapinv
<code>\multimapboth</code>	\multimapboth	<code>\multimapdot</code>	\multimapdot
<code>\multimapdotinv</code>	\multimapdotinv	<code>\multimapdotboth</code>	\multimapdotboth
<code>\multimapdotbothA</code>	\multimapdotbothA	<code>\multimapdotbothB</code>	\multimapdotbothB
<code>\multimapbothvert</code>	\multimapbothvert	<code>\multimapdotbothvert</code>	\multimapdotbothvert
<code>\multimapdotbothAvert</code>	\multimapdotbothAvert	<code>\multimapdotbothBvert</code>	\multimapdotbothBvert

<code>\Wr</code>	\approx	<code>\sqcupplus</code>	\boxplus	<code>\sqcapplus</code>	\boxcap
<code>\medcirc</code>	\bigcirc	<code>\medbullet</code>	\bullet	<code>\doteq</code>	\doteq
<code>\VDash</code>	\Vdash	<code>\VvDash</code>	\VdashV	<code>\cong</code>	\cong
<code>\preceqq</code>	\preceq	<code>\succeqq</code>	\succeq	<code>\coloneqq</code>	\coloneqq
<code>\varparallel</code>	\parallel	<code>\varparallelinv</code>	\parallel	<code>\colonapprox</code>	\approx
<code>\colonsim</code>	\sim	<code>\Colonapprox</code>	\approx	<code>\Colonsim</code>	\sim
<code>\eqqcolon</code>	\equiv	<code>\coloneq</code>	\equiv	<code>\eqcolon</code>	\equiv
<code>\Coloneqq</code>	\equiv	<code>\Eqqcolon</code>	\equiv	<code>\invamp</code>	\otimes
<code>\Diamonddot</code>	\diamond	<code>\Diamond</code>	\diamond	<code>\Diamondblack</code>	\blacklozenge
<code>\strictif</code>	\rightarrow	<code>\strictfi</code>	\leftarrow	<code>\strictiff</code>	\rightarrow
<code>\circledless</code>	\oslash	<code>\circledgtr</code>	\oslash	<code>\circledwedge</code>	\oslash
<code>\circledvee</code>	\oslash	<code>\circledbar</code>	\oslash	<code>\circledbslash</code>	\oslash

<code>\lJoin</code>	\Join	<code>\rJoin</code>	\Join	<code>\Join</code>	\Join
<code>\openJoin</code>	\times	<code>\lrtimes</code>	\Join	<code>\opentimes</code>	\times
<code>\Lbag</code>	$\{$	<code>\Rbag</code>	$\}$	<code>\nplus</code>	\oplus
<code>\Top</code>	\top	<code>\Bot</code>	\bot	<code>\Perp</code>	\perp
<code>\boxast</code>	\boxtimes	<code>\boxbslash</code>	\boxtimes	<code>\boxbar</code>	\boxtimes
<code>\boxslash</code>	\boxtimes	<code>\lambdaslash</code>	λ	<code>\lambdabar</code>	λ
<code>\varclubsuit</code>	\clubsuit	<code>\vardiamondsuit</code>	\diamond	<code>\varheartsuit</code>	\heartsuit
<code>\varspadesuit</code>	\spadesuit	<code>\llbracket</code>	\llbracket	<code>\rrbracket</code>	\rrbracket

`\lbag`, `\rbag`, `\llbracket` and `\rrbracket` are vertically extensive.

3.4. Variant integrate symbols

When we write a primitive, often the result is not very attractive because the function is too far from the integrate symbol.

As you can see here:

$$\int f(t)dt$$

Kp-Fonts provides variant commands to avoid this. The first is the `\varint` command and you get:

$$\int f(t)dt$$

It is up to you to choose whichever you prefer!

Obviously, this command is not convenient for computing an integral...
 You can also use `\D{...}`, the integrate symbol "d" command with best spacing:

$$\text{\displaystyle\varint f(t)\D{t}} \quad \text{gives} \quad \int f(t)dt$$

With the *frenchstyle* option, you get an upright "d", like above.

Others variant commands are:

`\variint`, `\variiint`, `\variiiiint` et `\varidotsint`.

3.5. New extensive symbols

First, the `\widehat` et `\widetilde` commands have been extended:



































$\widetilde{t\grave{il}de}$ $\widehat{chapeau}$

Finally, some new symbols:

<code>\bignplus</code>	\bigoplus	\biguplus	<code>\bigsqcupplus</code>	\bigsqcupplus	\bigsqcupplus
<code>\bigsqcapplus</code>	\bigsqcapplus	\bigsqcapplus	<code>\bigsqcap</code>	\bigsqcap	\bigsqcap
<code>\varprod</code>	\varprod	\varprod			

3.6. More integrate symbols

There are many unusual integrate symbols :

<code>\oiint</code>			<code>\ointctrclockwise</code>		
<code>\ointclockwise</code>			<code>\sqint</code>		
<code>\idotsint</code>			<code>\oiint</code>		
<code>\varointctrclockwise</code>			<code>\varointclockwise</code>		
<code>\fint</code>			<code>\ointctrclockwise</code>		
<code>\varoiintclockwise</code>			<code>\oiintclockwise</code>		
<code>\varoiintctrclockwise</code>			<code>\oiintctrclockwise</code>		
<code>\varoiintctrclockwise</code>			<code>\sqiint</code>		
<code>\sqiint</code>					

4. Installation

The tree provides a standard TDS. You have to copy all the files in one of your texmf trees first and then update your data base files.

Now, you have to deal with the .map file. If you have a web2c distribution, just run updmap.

```
updmap --enable Map=kpfonts.map
```

If it is not running, look for all the psfonts.map and pdftex.map. Then, append all the kpfonts.map lines into each of them and save them ... in the same directory!

With MikTeX (-> 2.4), it's easy to do the same thing with the files:

texmf/pdftex/config/psfonts.map, or,

texmf/dvips/config/psfonts.map, according to your use.

5. Some extra points

5.1. The JOHANNES KEPLER-project text families

family	series	shape	<i>rm</i> , <i>sf</i> or <i>tt</i> default with options:
jkp	m,b,bx	n,it,sc,sl	
jkpos	m,b,bx	n,it,sc,sl	<i>oldstyle</i> or <i>fulloldstyle</i>
jkpvos	m,b,bx	n,it,sc,sl	<i>veryoldstyle</i> or <i>fullveryoldstyle</i>
jkpl	m,b,bx	n,it,sc,sl	<i>light</i>
jkplos	m,b,bx	n,it,sc,sl	<i>light</i> and (<i>oldstyle</i> or <i>fulloldstyle</i>)
jkplvos	m,b,bx	n,it,sc,sl	<i>light</i> and (<i>veryoldstyle</i> or <i>fullveryoldstyle</i>)
jkpx	l,m,sb,sbx,b,bx	n,it,sc,sl	<i>rmx</i>
jkpxos	l,m,sb,sbx,b,bx	n,it,sc,sl	<i>rmx</i> and (<i>oldstyle</i> or <i>fulloldstyle</i>)
jkpxvos	l,m,sb,sbx,b,bx	n,it,sc,sl	<i>rmx</i> and (<i>veryoldstyle</i> or <i>fullveryoldstyle</i>)
jkpss	m,b,bx	n,sc,sl	
jkpssos	m,b,bx	n,sc,sl	<i>oldstyle</i> or <i>fulloldstyle</i>
jkpssvos	m,b,bx	n,sc,sl	<i>veryoldstyle</i> or <i>fullveryoldstyle</i>
jkptt	m,b	n,sl	
jkpttos	m,b	n,sl	<i>oldstyle</i> or <i>fulloldstyle</i>
jkpttvos	m,b	n,sl	<i>veryoldstyle</i> or <i>fullveryoldstyle</i>

If you want, or if you have to use the low-level commands!

Note that all the families support OT1, T1 and TS1 encodings.

5.2. Metrics and compatibility

- The *light* option does not change any metric;
- The *oldstyle* family options change only the metrics of the numbers;
- The *veryoldstyle* family options change also the metric of the lower-case 's' and the height of the superscripts in math mode;
- There is a full compatibility of the options, except if it is a nonsense, like both the *nomath* and *frenchstyle* options;
- In case of conflict between *oldstyle* and *veryoldstyle* family options, the *oldstyle* option is ignored.

5.3. Displaying and printing

Often, some display and printing problems exist...

The main reason is an automatic hinting. I'm not a professional typograph and I can't do better; besides, there are some printing problems with old releases of *Acroread*® using *Windows*®.

Note that the printing is better using *Ghostscript-Ghostview*®...

With the *light* option, the print is better than display!

5.4. My favorite options

- For text-only document, I use the *light* and *oldstyle* options;
- and for text and math document, I use the *light* and *frenchstyle* options, but I'm french!

5.5. JOHANNES KEPLER 1571-1630

KEPLER was forced, due to the counter Reformation and because he was a Lutheran, to move to Prague to work with the renowned Danish astronomer, TYCHO BRAHE. Using the data that TYCHO had collected, KEPLER discovered the first two laws of planetary motion (1609). And what is just as important about this work is that it is the first published account wherein a scientist documents shows how he has coped with the multitude of imperfect data to forge a theory of surpassing accuracy" (O. GINGERICH in forward to JOHANNES KEPLER *New Astronomy* translated by W. DONAHUE, Cambridge Univ Press, 1992), in other words a fundamental law of nature. Today we call this the scientific method.

From nasa website

<http://kepler.nasa.gov/johannes/>

5.6. Remark

Note that *Kepler*® is a registered font name supplied by *Adobe*™. The Kp-Fonts have nothing to do with those.

5.7. Thanks

Many thanks to

- NICOLAS BOULENGUEZ (Tests);
- MICHEL BOVANI (Fonts);
- DANIEL FLIPO (Latex);
- SOURAYA MUHIDINE (Translation reviewing)
- PETER ROSENBERG (URW)