

Kp-Fonts

The JOHANNES KEPLER project

CHRISTOPHE CAIGNAERT
c.caignaert@free.fr

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 "a" glyph:

Kp-Fonts	CM	Palatino	Utopia	Times
a	a	a	a	a

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 uprights 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** the `\bm` command;
- 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.

With both these options, Kp-Fonts does nothing...

2.2. Text fonts options

nosf : Kp-Fonts doesn't modify default sans serif fonts;

nott : Kp-Fonts doesn't modify default fixed width fonts.

These two options affect only text fonts.

oldstyle : Roman font provides here "Q" and the old ligatures "Œ" and "Œ", and also, oldstyle numbers by default.

A new command `\classicstylenums` allows you to get usual numbers. This option affects only roman font.

2.3. Greek letters in math mode

uprightgreek : *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;

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 **uprightgreek**, uppercase romans and lowercase greeks are upright, usual French style when typesetting mathematics; lowercase romans remain slanted;

upright : synonymous with the previous option.

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 `\alphasl`, `\alphaup`, `\Gammaup` or `\Gammaasl` 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 \mapsto , \mapsto

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>	\Rightarrow	<code>\Longmmapsto</code>	\Longrightarrow
<code>\Mappedfrom</code>	\Leftarrow	<code>\Longmmappedfrom</code>	\Longleftarrow
<code>\dashleftarrow</code>	\dashleftarrow	<code>\dashrightarrow</code>	\dashrightarrow
<code>\dashleftrightharpoon</code>	\longleftrightarrow	<code>\leftsquigarrow</code>	\leftrightsquigarrow
<code>\Nearrow</code>	\nearrow	<code>\Searrow</code>	\searrow
<code>\Nwarrow</code>	\nwarrow	<code>\Swarrow</code>	\swarrow
<code>\leadstoext</code>	\leadsto	<code>\leadsto</code>	\leadsto

You can combine these last two symbols:

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

<code>\boxright</code>	\squarerightarrow	<code>\Diamondright</code>	\diamondrightarrow
<code>\circcleright</code>	\circrightarrow	<code>\boxleft</code>	\squareleftarrow
<code>\Diamondleft</code>	\diamondleftarrow	<code>\circleleft</code>	\circleftarrow
<code>\boxdotright</code>	\squarerightarrow	<code>\Diamondddotright</code>	\diamondrightarrow
<code>\circleddotright</code>	\circrightarrow	<code>\boxdotleft</code>	\squareleftarrow
<code>\Diamondddotleft</code>	\diamondleftarrow	<code>\circleddotleft</code>	\circleftarrow
<code>\boxRight</code>	\squareRightarrow	<code>\boxLeft</code>	\squareLeftarrow
<code>\boxdotRight</code>	\squareRightarrow	<code>\boxdotLeft</code>	\squareLeftarrow
<code>\DiamondRight</code>	\diamondRightarrow	<code>\DiamondLeft</code>	\diamondLeftarrow
<code>\DiamondddotRight</code>	\diamondRightarrow	<code>\DiamondddotLeft</code>	\diamondLeftarrow
<code>\multimap</code>	\multimap	<code>\multimapinv</code>	\multimap
<code>\multimapboth</code>	\multimap	<code>\multimapdot</code>	\multimap
<code>\multimapdotinv</code>	\multimap	<code>\multimapdotboth</code>	\multimap
<code>\multimapdotbothA</code>	\multimap	<code>\multimapdotbothB</code>	\multimap
<code>\multimapbothvert</code>	\multimap	<code>\multimapdotbothvert</code>	\multimap
<code>\multimapdotbothAvert</code>	\multimap	<code>\multimapdotbothBvert</code>	\multimap

<code>\Wr</code>	\Re	<code>\sqcupplus</code>	\boxplus	<code>\sqcappplus</code>	\boxplus
<code>\medcirc</code>	\bigcirc	<code>\medbullet</code>	\bullet	<code>\doteq</code>	\doteq
<code>\VDash</code>	\Vdash	<code>\VvDash</code>	\Vdash	<code>\cong</code>	\cong
<code>\preceq</code>	\preceq	<code>\succeq</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>	\rightarrow	<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>	\ltimes	<code>\rJoin</code>	\ltimes	<code>\Join</code>	\ltimes
<code>\openJoin</code>	\times	<code>\lrtimes</code>	\ltimes	<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>	\diamondsuit	<code>\varheartsuit</code>	\heartsuit
<code>\varspadesuit</code>	\spadesuit	<code>\llbracket</code>	\llbracket	<code>\rrbracket</code>	\rrbracket

`\lbag`, `\rbag`, `\llbracket` et `\rrbracket` sont verticalement extensibles.

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...

Both display and text style exist.

Others variant commands are:

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

3.5. New extensive symbols

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

\widetilde{tild} $\widehat{chapeau}$

Finally, some new symbols:

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

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 `udpmap`.

```
udpmap --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. 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*®...

5.2. Further outlook

If you ask for ... light and semi-bold fonts.

Otherwise, if you need one extra symbol, contact me, I will do my best.

5.3. 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 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/>

5.4. Remark

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

5.5. Thanks

Many thanks to

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