Kp-Fonts

The Johannes Kepler project

Release 3.33

Christophe Caignaert (inactive)

c.caignaert@free.fr

Save up your toner and the environment, use the "light" option,
it's 20% toner less!

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
a	a	a	a	a	a

The project is to provide a large set of options to customize your math or text typesetting.

This LaTeX package is distributed with a gpl licence.

Kp-Fonts doesn't require any other font package and is fully compatible with amsmath and textcomp with the full option. Kp-fonts loads these two packages except if you use relative options.

You haven't to load amsmath and textcomp packages

You can use the options of amsmath a option of kpfonts, except cmex10.

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 т1, oт1 et тs1 are fully available, except for the upper-case symbol \SS in teletype fonts; if you use some accents, you have to work with т1 encoding, oт1 is present for historical compatibility;

- Roman fonts are available with upright, small caps (usual and large), italic, slanted, small caps slanted (usual and large) and oldstyle shape, all with regular, bold and bold extended weight;

- Sans-serif fonts are available with upright and slanted, small caps upright and slanted (usual and large) shape, and oldstyle 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 $\mathbb{ABC}$, characters designed from capital upright roman;

- \mathcal provides $\mathcal{ABC}$, slighty altered fonts from cm;

- With \mathscr, you get $\mathscr{ABC}$, without any extra package; these letters are designed from traditional fonts;

- \mathfrak provides $\mathfrak{ABCabc}$, still from traditional fonts, altered to be more readable: read $\mathfrak{CTAN}$ and *CTAN* in `www.ctan.org`!

- See the \mathupright, or \mathup, alphabet in the *sf* math versions section.

2. Options

2.1. Main global options

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

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...***

fulloldstylenums: equivalent to both *oldstylenums*

and *oldstylenumsmath*;

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

fullveryoldstyle: equivalent to both *veryoldstyle*

and *veryoldstylemath*.

2.2.   Other 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;

noamsmath: Kp-Fonts doesn't load `amsmath` package;

notextcomp: Kp-Fonts doesn't load `textcomp` package.

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

2.3.   Text fonts options

These options affect only text fonts.

lighttext : "light" fonts are used in text mode.

oldstylenums: provides here oldstyle numbers by default.

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

Some examples, upright and italic, medium and bold:

- 0123456789!
- *0123456789!*
- **0123456789!**
- ***0123456789!***

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

With roman and sans-serif fonts, you get the old ligatures "ct" 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 "ſ".

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

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

Obviously, there is no long "ſ" in smallcaps shape!

rmx then , you get six weights, with the correponding 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, the *lighttext* option is ignored.

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

This option affects only roman fonts.

largesmallcaps gives larger small capitals than default:

LARGE and SMALL (default) small capitals!

easyscsl allows you to fit together the commands \textsl and \textsc to get slanted small caps.

Attention,

- this option needs the package ifthen,
- to fit together these commands with other fonts produces some errors...

nofligatures provides a typeseting without the ff, fi, fl, ffi and ffl ligatures.

The letter *f* has a new design to get a good aspect.

This option has no effect with *oldstyle* or *veryoldstyle* ones.

You get for instance, "A final effort" instead of, "A final effort".

And also, *"A final effort"* instead of, *"A final effort"*.

2.4.   New text commands

\textscsl{...} and **\scslshape:** provide slanted small caps.

SLANTED SMALL CAPS

Obviously, \textsc{\textsl{...}} *has not the same effect without using the option* easyscsl*!*

This shape is not usual in TEX!

\otherscshape, \textothersc{...} get the *other* small caps (default or large) roman or sans-serif.

For instance, you swap between

SMALL CAPS TEXT   *and*   SMALL CAPS TEXT

\otherscslshape, \textotherscsl{...} get the *other* slanted small caps (default or large) roman or sans-serif.

For instance, you swap between

Small caps text   *and*   Small caps text

\othertailQ provides the other design of the uppercase letter "Q", with small or large tail.

For instance, you swap between

Question   *and*   Question

\othertailscq, \othertailscslq are the same commands for the lowercase small capitals "Q".

For instance, you swap between

Question   *and*   Question

*The "other" commands change the **size** of the small caps
and the "othertail" commands change the **design** of the letters "Q"*

In some cases, the section headers for instance, the "other" commands have to be protected: `"\protect\...other..."`!

2.5.   Math modes

There are six math modes relative to the command `\mathversion`.
For instance,

`\mathversion{sf}$\left(a+b\right)^2$`   gives: $(a + b)^2$

normal: Default mode corresponding to the used options.

bold: Bold default mode.

sf: The math mode using the sans-serif slanted fonts.

$$\sum_{n=1}^{+\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

boldsf: Corresponding bold mode.

$$\sum_{n=1}^{+\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

rm: The usual math mode using the italic roman fonts.

$$\sum_{n=1}^{+\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

boldrm: Corresponding bold mode.

$$\sum_{n=1}^{+\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

> With the *sf* math versions, the option *widermath* is ignored, as the option *lightmath*, or, *light* in math typesetting!

We have also to note there is a new math alphabet \mathupright or \mathup. It's equivalent to

- \mathrm with the *rm* math versions, and,
- \mathsf with the *sf* math versions.

In another way, it's provide the upright alphabet corresponding to math letters.

2.6. Math typesetting options

lighttext : "light" fonts are used in math mode.

sfmath: the default math mode using the sans-serif slanted fonts, default and *bold*.

Then, you can use the package *bm* with sans-serif math typesetting;

sfmathbb: in all cases, the \mathbb font is sans-serif: $\mathbb{C\,K\,N\,Q\,R\,Z}$;

rmmathbb: in all cases, the \mathbb font is roman: $\mathbb{C\,K\,N\,Q\,R\,Z}$;

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 $\Re e$ and $\Im m$;

uprightRoman: the *Uppercase* romans are upright.

frenchstyle: equivalent to both the options *uprightRoman*

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

upright: synonymous with the previous option;

oldstylenumsmath: provides the oldstyle numbers in default and changes the `\mathrm`, `\mathsf` and `\mathtt` fonts; they run as text fonts with *oldstylenums* 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 "ſ";

narrowiints: provides narrower multiple integral symbols:

$$\iint \text{ and } \iiint \quad \text{instead of} \quad \iint \text{ and } \iiint$$

partialup: provides upright design of the `\partial` symbol:

$$\partial \quad \text{instead of} \quad \partial$$

You got also the absolute commands `\partialup` and `\partialsl`...

widermath: with wider space between mathematic letters;

noDcommand: for compatibility with some other package, kpfont doesn't load the command `\D`.

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

2.7. Position of subscripts and superscripts

In math mode, just about with `amsmath`, the next options adjust the position of subscripts and superscripts.

a/ Integral symbols

The default option is **nointlimits**.

`\int_a^bf(t)`	**intlimits**	**nointlimits**	**fullintlimits**
textstyle	$\int_a^b f(t)$	$\int_a^b f(t)$	$\int\limits_a^b f(t)$
displaystyle	$\int_a^b f(t)$	$\int_a^b f(t)$	$\int\limits_a^b f(t)$

b/ Sums and products

The default option is **sumlimits**.

`\sum_{i=1}^nu_n`	**sumlimits**	**nosumlimits**	**fullsumlimits**
textstyle	$\sum_{i=1}^{n} u_n$	$\sum_{i=1}^{n} u_n$	$\sum\limits_{i=1}^{n} u_n$
displaystyle	$\sum\limits_{i=1}^{n} u_n$	$\sum_{i=1}^{n} u_n$	$\sum\limits_{i=1}^{n} u_n$

2.8.   Greek letters in math mode, options

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

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

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 commands like
`\otheralpha` or `\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 `\Gammasl` 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 $\twoheadrightarrow$ $\subsetneq$ $\leqslant$
and in bold: $\boldsymbol{\leq}$ $\boldsymbol{\bullet}$ $\boldsymbol{\pm}$ or $\boldsymbol{\twoheadrightarrow}$ $\boldsymbol{\subsetneq}$ $\boldsymbol{\leqslant}$.
Obviously, you get the usual constructions, both regular and bold: $\mapsto\longmapsto$,
$\boldsymbol{\mapsto\longmapsto}$

3.3.   Extra symbols

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

\mappedfrom	↤	\longmappedfrom	⟻
\Mapsto	⇸	\Longmapsto	⟹
\Mappedfrom	⇷	\Longmappedfrom	⟸
\mmapsto	⊢→	\longmmapsto	⊢⟶
\mmappedfrom	←⊣	\longmmappedfrom	⟵⊣
\Mmapsto	�⇒	\Longmmapsto	⊩⟹
\Mmappedfrom	⇐⊣	\Longmmappedfrom	⟸⊣

\dashleftarrow	⇠	\dashrightarrow	⇢
\dashleftrightarrow	⇠⇢	\leftsquigarrow	⬳
\Nearrow	⤢	\Searrow	⬂
\Nwarrow	⬃	\Swarrow	⤡
\varemptyset	∅		
\leadstoext	∼	\leadsto	⤳

You can combine these last two symbols:
\leadstoext\leadstoext\leadstoext\leadsto give ⤳⤳⤳⤳

\boxright	□→	\Diamondright	◇→
\circleright	○→	\boxleft	←□
\Diamondleft	←◇	\circleleft	←○
\boxdotright	⊡→	\Diamonddotright	◇→
\circleddotright	⊙→	\boxdotleft	←⊡
\Diamonddotleft	←◇	\circleddotleft	←⊙
\boxRight	□⇒	\boxLeft	⇐□
\boxdotRight	⊡⇒	\boxdotLeft	⇐⊡
\DiamondRight	◇⇒	\DiamondLeft	⇐◇
\DiamonddotRight	◇⇒	\DiamonddotLeft	⇐◇

\multimap	⊸	\multimapinv	⟜
\multimapboth	⧟	\multimapdot	⊶
\multimapdotinv	⟜	\multimapdotboth	●—●
\multimapdotbothA	○—●	\multimapdotbothB	●—○
\multimapbothvert	⸙	\multimapdotbothvert	⸙
\multimapdotbothAvert	⸙	\multimapdotbothBvert	⸙

\Wr	⚹	\sqcupplus	⊎	\sqcapplus	⊞
\medcirc	○	\medbullet	●	\doteq	≐
\VDash	⊫	\VvDash	⊪	\cong	≅
\preceqq	⪯	\succeqq	⪰	\coloneqq	≔
\varparallel	⫽	\varparallelinv	⫽	\colonapprox	≈
\colonsim	:∼	\Colonapprox	∷≈	\Colonsim	∷∼

\eqqcolon	=:	\coloneq	:−	\eqcolon	−:
\Coloneqq	∷=	\Eqqcolon	=∷	\invamp	⅋
\Diamonddot	◇	\Diamond	◇	\Diamondblack	◆
\strictif	⊰	\strictfi	⊱	\strictiff	⊰⊱
\circledless	⧀	\circledgtr	⧁	\circledwedge	⧀
\circledvee	⟁	\circledbar	⦶	\circledbslash	⦸

\lJoin	⋉	\rJoin	⋊	\Join	⋈
\openJoin	⨯	\lrtimes	⋈	\opentimes	⨯
\Lbag	⟆	\Rbag	⟅	\nplus	⨥
\Top	⊤	\Bot	⊥	\Perp	⊥
\boxast	⊡	\boxbslash	⧅	\boxbar	▣
\boxslash	⧄	\lambdaslash	ƛ	\lambdabar	ƛ
\varclubsuit	♣	\vardiamondsuit	◆	\varheartsuit	♥
\varspadesuit	♤	\llbracket	⟦	\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:

$$\int f(t)\,\mathrm{d}t$$

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

$$\int\!f(t)\,\mathrm{d}t$$

It is up to you to choose whichever you prefer!

11

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

`\displaystyle\varint f(t)\D{t}`   gives $\displaystyle\int f(t)\,dt$

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

Others variant commands are:
`\variint, \variiint, \variiiint et \varidotsint`.

3.5.   New extensive symbols

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

$$\widetilde{tilde} \qquad \widehat{chapeau}$$

You get also th extensive `\widearc`, `\widearcarrow` (ou `\wide0arc`), `\wideparen` and `\widering`:

$$\overarc{arc} \qquad \overrightarrow{arrow} \qquad \wideparen{paren} \qquad \widering{RING}$$

This last command makes an error with the option `noamsmath`
Finally, some new symbols:

`\bignplus`	⊞	⊞	`\bigsqcupplus`	⊎	⊎
`\bigsqcapplus`	⊞	⊞	`\bigsqcap`	⊓	⊓
`\varprod`	✕	✕			

3.6.   More integrate symbols

There are many unusual integrate symbols:

\oiint	∯	∯	\ointctrclockwise	∮	∮
\ointclockwise	∮	∮	\sqint	⨖	⨖
\idotsint	∫⋯∫	∫⋯∫	\oiiint	∰	∰
\varointctrclockwise	∮	∮	\varointclockwise	∮	∮
\fint	⨍	⨍	\oiintctrclockwise	∯	∯
\varoiintclockwise	∯	∯	\oiintclockwise	∯	∯
\varoiintctrclockwise	∯	∯	\oiiintctrclockwise	∰	∰
\varoiiintctrclockwise	∰	∰	\sqiint	⨖	⨖
\sqiiint	⨗	⨗			

4. Installation

- With MikTeX, install the package as described here:

  `http://docs.miktex.org/manual/pkgmgt.html#id562117`

- With other distribution, or to install manualy with MikTeX, follow these instructions:

  1. The tree provides a standard TDS. You have to copy all the files in one of your local `texmf` trees first and then update your data base files.
  2. Now, you have to deal with the `.map` file.
     - If you have a web2c distribution, just run updmap:
       `updmap --enable Map=kpfonts.map`
       and/or, this time as *root*:
       `updmap-sys --enable Map=kpfonts.map`
     - With MikTeX, follow the instructions of the manual:
       `http://docs.miktex.org/manual/advanced.html#psfonts`

5. Some extra points

5.1. The Johannes Kepler-project text families

If you want, or if you have to use the low-level commands, the names of the families are:

roman	jkp[l,x][k][f][osn,os,vos]
sans serif	jkpss[k][f][osn,os,vos]
teletype	jkptt[osn,os,vos]

with the relative options:

l, x	light, rmx
k	largesmallcaps
f	nofligatures
osn, os, vos	oldstylenums, oldstyle, veryoldstyle

Obviously, you can't have **f** and **os** or **vos**.
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 *oldstylenums* or *oldstyle* options do not change any horizontal metric;

- The *veryoldstyle* family options change the metrics of the lowercase '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 *oldstylenums*, *oldstyle* and *veryoldstyle* family options, the lighter options are 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. Abstract

You get almost all the features of *kpfonts* in one page with the file

kpfonts-abstract.pdf

5.5. Searching a word

With the *veryoldstyle* options, the browsers don't find the rare ligatures **si, sl, st** and the **sacute**, **scaron** and **scedilla**.

Don't use this option if you want to search these words!

5.6. My favorite options

- For text-only document, I use the *light* and *oldstyle* options;

- and for text and math document, I use the *light*, *frenchstyle*, *narrowiints* and *partialup* options, but I'm french!

5.7. 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 foreword 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/

5.8. Remark

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

5.9. Thanks

Many thanks to

- Nicolas Boulenguez (Tests);

- Michel Bovani (Fonts);

- Daniel Flipo (LaTeX);

- Souraya Muhidine (Translation reviewing)

- Peter Rosenberg (urw)

- Christian Tellechea (package *xstring*)

- and the contributors of `comp.text.tex` and `fr.comp.text.tex`