

A story of *kpfonts*: Reaching the limits of NFSS

Christophe Caignaert

Like a bird on the wire,
Like a drunk in a midnight choir,
I have tried, in my way, to be free.

LEONARD COHEN, *Bird on the wire*

One day, some years ago, I was with DANIEL FLIPO, the author of the *letrine* package and the French module of *babel*.

He reminded me that L^AT_EX is community software, and, if I don't find what I want, I have to write it! Without him, probably, *kpfonts* wouldn't exist.

Greetings to him...

1 Before *kpfonts*

1.1 I'm not a...

I have been a mathematics teacher in a high school in the north of France since 1980. My students are 19 or 20. I have been interested in computer science since the middle of the seventies.

I'm not a typographer and I'm not a T_EX, or L^AT_EX, expert. I'm unable to program in the T_EX language! Nothing fated me to become a font designer and package author... nothing at all!

1.2 First steps with computer typesetting

The first computer I bought was an Apple IIe. Then I began writing some papers with the Apple Writer¹ software, obviously text and not math documents...

Then I bought an HP personal computer with an 8 Mhz 80286 processor! I began writing some mathematics using ChiWriter,² shareware at that time.

Some years later, I used a student release of Scientific Word, Scientific Workplace,³ release 2.5. It was a private L^AT_EX editor with a limited wysiwyg formula editor. This was my first typesetting with good output. Scientific Word was good, but not very versatile, and month after month, I reached its limits. It was possible to insert any L^AT_EX command, but if it was unknown to SW, it would appear on the screen as a grey box. Some basic commands like `\sum\limits` in math resulted in a grey box for `\limits`. Because I'm never fully satisfied, I got more and more grey boxes in my documents with more and more (L^A)T_EX commands not interpreted

on screen. Therefore, I decided to forget it and I'm now using pdfT_EX.

Perhaps it seems foolish to you, but during these years I was working alone to discover this software. In my high school, most math teachers are, still at this moment, writing math by hand; some use a too-well-known word processor, and I'm alone in looking for better output quality... with L^AT_EX.

And you know it's not easy to discover L^AT_EX alone!

1.3 First interest in fonts

I have been interested in typography for a long time and I read that pdfT_EX can use TrueType fonts. I followed the article of DAMIR RAKITYANSKY⁴ to install my first *tff* fonts.

Thus, I discovered ligatures, kerning, metrics, virtual fonts: *pl*, *vpl*, *tfm*, *vf*, *fd*, *map* and *sty* files of the L^AT_EX font world, and also *tff*, *psb*, *afm* files coming from typography.

Because some users, mainly Windows users, never use a console or command line, I wrote a new paper, in French⁵ and in English,⁶ about installing *tff* fonts for pdfT_EX and step-by-step instructions for those using T_EXnicCenter. In addition, I built the necessary support files for many Windows *tff* fonts and free *tff* fonts available from a web site.⁷

I also made an artistic document for a local exhibition combining computer handwriting fonts with the meaning of the message, perhaps unfortunately for you, in French, called *Rendez-Vous*.⁸

Doing that work, I also discovered *the* font editor, *fontforge*, but also some other font editors: one of my friends works in typography and I used his professional computer during weekends.

I obviously discovered Bezier curves and the design of non-Metafont glyphs...

1.4 First steps of the future *kpfonts*

My first font was called *Christophe*; it was my first attempt to alter *Palladio* (the URW *Palatino*) as a challenge... for myself!

From the beginning, the principles were:

- very basic design, with a minimum number of Bezier curves,
- dynamic design with a marked diagonal force line from WSW to ENE.

⁴ <http://www.radamir.com/tex/ttf-tex.htm>

⁵ <http://c.caignaert.free.fr/Installer-Police-ttf.pdf>

⁶ <http://c.caignaert.free.fr/Install-ttf-Font.pdf>

⁷ <http://c.caignaert.free.fr/ttf.html>,

<http://c.caignaert.free.fr/ttf-english.html>

⁸ <http://c.caignaert.free.fr/Rendez-Vous.pdf>

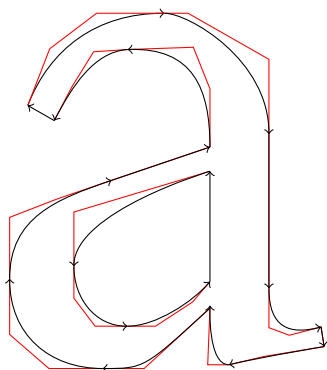
¹ http://en.wikipedia.org/wiki/Apple_Writer

² <http://en.wikipedia.org/wiki/ChiWriter>

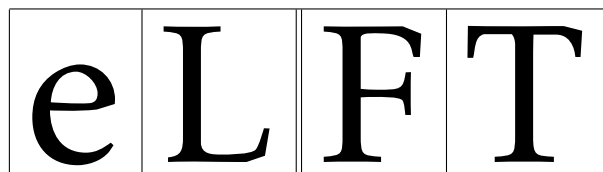
³ http://en.wikipedia.org/wiki/Scientific_WorkPlace

We can see here the roman upright *a* of both *kpfonts* and some other font packages, and the approximate corresponding set of Bezier curves.

Kpfonts	CM	Palatino	Utopia	Times
a	a	a	a	a



Next, you can see here the force line (sharp cut) and its symmetrical echo in *kpfonts*:



2 The development

I saw a beggar leaning on his wooden crutch,
he said to me, "You must not ask for so much."

And a pretty woman leaning in her darkened door,
she cried to me, "Hey, why not ask for more?"

LEONARD COHEN, *Bird on the wire*

2.1 Beginnings of the math set fonts

My first tests with math fonts was to use URW *Garamond* with the math symbols of *pxfonts*, the package I use at this moment in my documents. I called this *gxfnts*...

I discovered the global organisation of math fonts, with the main

- *operators*, like 0123 + - = $\Gamma\Delta$, and math operators like "sin",
- *letters* like *abc* $\alpha\beta\gamma$,
- *symbols*, the basic symbols, like $\rightarrow \mapsto \Rightarrow \exists$,
- *largesymbols*, the multi-size basic symbols, like

$$\Sigma \sum \int \int$$

and a lot of other things like AMS symbols, etc.

I also learned about the math alphabets, math delimiters. . .

I was impressed by the special tricks of DONALD KNUTH as

- long arrows made with minus sign and a regular arrow: '−' and '→' gives '→',
- long double arrows with equal sign and regular double arrow: '=' and '⇒' give '⇒',
- the use of the fake width and italic correction in math mode, width for subscript and italic correction for superscript,
- the famous *skewchar*, fake kerning to create the math accents: \tilde{a} .

It's like building the Golden Gate Bridge with three oz of spaghetti. . .

When the *gxfnts* package was in β -release, I sent a note to MICHEL BOVANI, the author of the *fourier* package, asking him his opinion.

Many thanks to him: he told me, with chosen words, it was *very bad*! And, even better, he told me *why*! For instance, the roman and greek letters of *gxfnts* were like cats and dogs. . .

Thus, I saw, at that moment, I had designed the Greek letters according to the design of the roman letters of *Christophe*. Even though the two projects were not linked at first, it was not so surprising: the same author and the same mood for design. . .

Therefore, it was obvious I had to combine these. . .

2.2 The *kpfonts* package

2.2.1 The 1.0 release

Then I decided to make a *full* package of fonts, i.e. needing only one `\usepackage` to run.

It was 2005/04/20, my fiftieth birthday. Often, many people think that your life is behind you at 50! And perhaps I had to prove I was not a has-been!

From that moment I decided to write a comprehensive package including

- the roman, sans serif and teletype fonts,
- all the symbols including AMS symbols, "not" symbols, et al.,
- calligraphic and script alphabets,
- a *frenchstyle* math option needing upright uppercase and greek letters .

At that time, I had:

- the normal and bold text fonts including small caps, from *Christophe*,
- the slanted greek letters, from *gxfnts*,

and my todo list was cluttered:

- the sans-serif and teletype fonts,

- the textcomp symbols,
- the symbols, large symbols, AMS symbols,
- the upright greeks, calligraphic, script, full mathbb and fraktur alphabets,
- reading the *fontinst* doc file carefully,
- fixing the font's math dimensions: I keep the math font dimensions as DONALD KNUTH had them, except for the position of a subscript with no superscript, lower in *kpfonts* than CM.

I didn't realize the great deal of work needed at that time. The next two years were the busiest of the story. If you look at the `readme.txt` file, you can find:

Release 1.0 2007/04/20

It was my 52nd birthday. For many years, the first new set of fonts designed for L^AT_EX. I was very anxious about the feedback.

Since the beginning, *kpfonts* has supported the *frenchstyle* option, with upright uppercase roman and lowercase greek letters in math mode. Even if, at that time, I had no idea about the future of *kpfonts*, from the beginning, I thought I would propose some options to customize the typesetting.

2.2.2 Old style options

At that time, my birthday was obviously very important, because the next line of `readme.txt` is

Release 1.1 2007/05/04 New 'oldstyle' option, and `\sqrt` bug fixed.

only *fifteen days* later!

I had built the *oldstyle* option during the two previous years and it was almost ready when I uploaded the 1.0 release...

In fact, I think it is a good thing to build a package but a better thing to build a *different* package. A large set of options to customize the typesetting will make the *difference*. This appeared little by little during the work, like an obvious element.

In France, we have a well-known collection of books called *La Pleïade* using a *Garamond* font set with the *ct* and *st* old ligatures and a long tail *Q*.

I decided, because I liked them, to offer these possibilities as an option, with *oldstyle* numbers as the default. Later, asked by German users, I built an *oldstylenums* option without the extra ligatures (in Release 2.1 2008/03/21).

Here you can see the design of the ligature forms compared to the standard forms, using the *light* option:

upright	oldstyle	upright	oldstyle
ct	ct	st	st
italic	oldstyle	italic	oldstyle
<i>ct</i>	<i>ct</i>	<i>st</i>	<i>st</i>

The font dimensions of the superscripts are altered with *oldstyle* numbers in math mode taking their design into account.

Because the T1 encoding is full, I had to find two slots for the new *ct* and *st* old ligatures. I chose to use the slots of two Icelandic letters. I had nothing against the Icelandic people or their language, but I had to make a decision... Obviously, *kpfonts* sends a warning in this case.

2.3 The *kpfonts* package, release 2

It was the first major evolution of the package:
Release 2.0 2008/01/01.

The new *f* ligatures, light fonts and very old style options appeared with this release. You can see that the second part of 2007 was a very intensive work period!

It was a new main number because, for me, the *light* option is *the major alteration* of *kpfonts*.

At that time, I thought, once again, that *kpfonts* was finished, except for the inevitable bug corrections...

2.3.1 Light fonts

In my opinion, too-bold fonts are in bad taste. Using the facilities of the font editors and a good deal of work, I built lighter fonts with the same metrics, corresponding to the *light* option.

It was necessary to design again the mathematical symbols: when you have a line that is .7 pt in 10 pt, the light font would be .5 pt.

You can see below the normal weight and light, upright and italic *a*:









upright	light	italic	light
a	a	<i>a</i>	<i>a</i>

What's more, it's not insignificant to save up to 20% toner when printing...

2.3.2 New *f* ligatures

A ligature is the way to combine two characters into one. The most common ligatures with T_EX are the *f* ligatures: *ff*, *fi*, *fl*, *ffi* and *ffl*.

There are different ways to design them. See examples below with the *fi* ligature:

URW Garamond	Kpfonts 1.x	Kpfonts	Palladio
			
URW Garamond	Kpfonts 1.x	Kpfonts	Palladio
			




At first, I made a bad choice, like a bridge, as with *Garamond* for instance. It was a bad choice relative to the design of the *f* of my fonts: the effect was not good because of the short terminal of its ascender. Thus, I decided to change it. It was necessary to change the design of the ascenders of these ligatures.

Note the old and new *fi* of *kpfonts* and the almost fake ligature in upright *URW Palladio* used by the *palatino*, *pxfonts*, and *mathpazo* packages.

2.3.3 Very old style options

In very old documents, instead of the round *s*, we find a long *f*, except at the end of the word. I couldn't find any package to typeset text and math with the long *f*. Then, I decided to built the necessary files and to offer these possibilities. It was done with *ReRelease 2.1* 2008/03/21.

For instance, here is *st* using italic shape and light fonts; you can see I also installed new ligatures:

Default	Old style	Very old style
		





At this moment, the idea to make a package with a large set of options to customize the typesetting was definitely established.

2.3.4 Large small capitals

It's interesting in a font package to have real small capitals and not fakes... From the beginning, I designed some small caps. In fact, I designed *very small* small caps, approximately as high as an *x*. I like it because they are different!

It's also not usual because in many cases, the small caps are fakes, scaled uppercase indeed. Don't forget that a fake seems not too bad if the scaling is not too strong, i.e. if the small caps are not too small! This is another of the reasons why small caps are usually rather large.

I decided then to work on a large small caps set of fonts. Indeed, the font editors are able to "blend" some fonts. Blending the existing *small* small caps and usual uppercase letters gives a good design to begin the work. It was done in *ReRelease 2.2* 2008/05/21, shown here using the *light* option.

lowercase	small cap	large small cap	uppercase
			

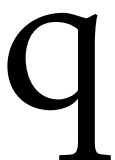



Thus, *kpfonts* has two sizes of small caps. It's very rare and even the very extensive OpenType font file doesn't allow for it!

As in the present article, I usually use small small caps for people's names and large small caps for acronyms.

2.3.5 The lowercase *q* record

Usually, with a given font set, you get four designs for a letter: upright and italic, normal and bold. If there are true small caps, you get also them in normal and bold, six designs in this case.

For the lowercase *q* in *kpfonts*, you get forty roman designs, and then more with the sans-serif and teletype fonts! Perhaps a record, even though there is no italic small caps *q*. Let's start with the default designs:

upright	bold	italic	bold
			

Light:

upright	bold	italic	bold
q	q	q	q

Small caps:

default	bold	large	bold
Q	Q	Q	Q

Light small caps:

default	bold	large	bold
Q	Q	Q	Q

Long tail small caps:

default	bold	large	bold
q	q	q	q

Long tail light small caps:

default	bold	large	bold
q	q	q	q

We get all these glyphs with the lowercase *q*!

2.3.6 No *f* ligatures

The option `nofligatures` appeared with `Release 2.3 2008/09/09`, requested by users who didn't like the ligatures.

With some packages, or modern T_EX-based engines, you can disable these ligatures but the result can be ugly:

Times <code>f{i}</code>	Utopia <code>f{i}</code>	Kpfonts <code>f{i}</code>	Kpfonts <code>nofligatures</code>
fi	fi	fi	fi

Times <code>f{i}</code>	Utopia <code>f{i}</code>	Kpfonts <code>f{i}</code>	Kpfonts <code>nofligatures</code>
<i>fi</i>	<i>fi</i>	<i>fi</i>	<i>fi</i>

And, don't forget it's worse at normal size! With upright Times, *f* and *i* seem incompatible, like cats and dogs, and with Utopia, the ascender of the *f* and the dot of the *i* are too close and don't fit together.

You can see that the result is not too bad with my fonts, but, I preferred to shorten the ascender of the *f* letter in this case. In my opinion, the look is better at normal size!

2.3.7 Slanted small caps

In the L^AT_EX new font selection scheme (NFSS), small caps and slanted (or italic) are *shapes*. The result is the impossibility of getting slanted small caps.

Installing slanted small caps, a new shape `scsl`, requires only some lines in the installation file used by `fontinst` program, and also some lines in the `sty` file. Here's an example:

Everybody, including TED SLANTED, can see it's better than JACK UPRIGHT does usually!

Slanted small caps also appeared with `Release 2.3 2008/09/09`. Later, a new option `easyscsl` allows you to fit together `\textsc` and `\textsl`. It's an option because, if you use `\textsc{\textsl{...}}` with other fonts, you get some edge effect. This option appeared with `Release 3.3 2010/04/20`, and sent a warning to the console. This point will be discussed in a later section.

2.3.8 Math fonts during this time

For some time now, we have been speaking about text fonts but math typesetting is also going on!

First, the `oldstylemath`, `veryoldstylemath` and `oldstylenumsmath` appeared at the same time as the text equivalent.

As of `Release 2.2`, you can use `narrowiints` option, For `\displaystyle\iiint dx, dy, dz`, let's see the output:

default	<i>narrowiints</i>
$\iiint dx dy dz$	$\iiint dx dy dz$

And with Release 2.3, the *partialup* option is added. For `\dfrac{\partial z}{\partial x}`, the output is:

default	<i>partialup</i>
$\frac{\partial z}{\partial x}$	$\frac{\partial z}{\partial x}$

2.4 The 3.0 release: new text kerning and math accents

2.4.1 New kerning

There were some inherited defaults in *kpfonts*, and, even at that time, we could see that the main problem was the kernings. One of the first lines of the `Readme` file is

Release 1.11 2007/06/03 Correct bad kernings of 'quote' symbols

It proves that, from the beginning, the kerning was a problem. Perhaps it's the biggest challenge for a beginner! The kerning by pairs is the way to tighten or spread two characters depending on their exact design. For instance, see *Ye* with and without kerning, here using the *light* option:

with	without
<i>Ye</i>	<i>Ye</i>

The font editors offer a lot of possibilities. One of these is automatic kerning. Usually you have to choose:

- the left and right characters to kern,
- the required space between two characters,
- the technique: minimum distance, average distance, average weight,
- the exceptions: numerals, lowercase-uppercase: in 'L^AT_EX', for instance, there is a kerning *T-e* but no kerning *a-T*...
- the equivalents, *o* and *ô* have often the same kerning...

These programs do their best but are regrettably not very good. And a beginner like me was too confident in their results. Even if, at the time, I corrected all the generated kernings by hand, I was too

confident about the basic results of the automatic kernings...

Some users protest rightly about incoherent kerning. I asked on *fctt*, the French version of *ctt*, and everybody thought new kernings would be a good thing although it can change the typesetting. I decided to work on it...

At the same time, subscript and superscript position, i.e. width and italic correction, of all the math alphabets were revisited. It's a very long hard job, with a large set of tests and much reinstallation of *kpfonts*. During these six months, I produced, with *fontinst* and batch files, at least 200 000 files...

It was available on CTAN as of Release 3.0 2009/03/03, and I thought now the work was not too far from being good fonts. Therefore, the new main number version.

See for instance the *Av* kerning in upright shape and *To* in italic with the 1.xx or 2.xx release versus the same with 3.xx (default fonts here).

before 3.0	3.0 and after	no kerning
<i>Av</i>	<i>Av</i>	<i>Av</i>

before 3.0	3.0 and after	no kerning
<i>To</i>	<i>To</i>	<i>To</i>

Scaled this much, the first may not appear to spread, but it's the case at normal size. That's the reason why the first kernings are too strong. Then, I was working on a screen... Thus I bought a laser printer and work now with printed tests!

2.4.2 New math accents and *widermath* option

Also with the 3.0 release, I installed new math accents such as `\widearc`, as in some other packages. Here are some examples:

<code>\widearc</code>	<code>\widearcarrow</code>
$\widearc{M_0 M_1}$	$\widearcarrow{M_0 M_1}$
<code>\wideparen</code>	<code>\widering</code>
$\wideparen{M_0 M_1}$	$\widering{M_0 M_1}$

You get also the new option *widermath*. The object is to provide slightly wider math typesetting, particularly for users working with 9 or 10 pt as the basic font size. Small sizes need proportionally bigger spaces...

2.4.3 *amsmath* options

Release 3.1 2009/05/20 offers the possibility to use the options of *amsmath* as options of *kpfonts*. This affects the basic and AMS math fonts and also the special math fonts of *kpfonts*...

These too-little-known options affect the default position of subscript in integral or summation symbols. To get more information, see the documentation of the AMS or *kpfonts* packages.

2.4.4 Sans-serif math versions

The last major evolution *kpfonts* was Release 3.2 2010/03/03 allowing math typesetting using sans-serif fonts. You can do it with a new option, *sfmath*, or with the new math versions *sf* and *boldsf*. Obviously, for full support, you also get both *rm* and *boldrm* math versions.

Some default symbols are serified, like \sum ; thus they have a new design, as you can see:

roman	sans-serif
$\sum_{p=0}^n$ $\sum_{p=0}^n$	$\sum_{p=0}^n$ $\sum_{p=0}^n$

In addition, I designed some sans-serif greek letters, uppercase and lowercase, slanted and upright:

roman	sans-serif
α β α β Γ Ψ Γ Ψ	α β α β Γ Ψ Γ Ψ

In case you are getting slightly sleepy reading this, let me explain exactly what it means. For instance, when you type $\backslash\alpha$, depending on the options and math version, you can get any of 12 different designs: normal or bold ($\times 2$); upright or slanted ($\times 2$); default or light roman; or sans-serif ($\times 3$)!

2.5 Special tricks

In fact, I don't like to have special tricks in a package, but I still use this possibility sometimes!

- To get the *veryoldstyle* *s*, usually at the end of a word, I use a classic fake ligature $s=$.

- narrowiints*

In the *kpfonts.sty* file, we find this code:

```
\re@DeclareMathSymbol{\iintop}{\mathop}{\largesymbolsA}{\narrowiints33}
```

where

- $\backslash\text{narrowiints}$ is 1 if the *narrowiints* option is selected, empty if not, and,
- the default $\backslash\text{iint}$ symbol is decimal 33 and the narrower one is decimal 133.

- Long tail Q is called:

- Qoldstyle* in the *afm* files and the *etx* files used by *fontinst* and,
- Q* in the *pdf* and *enc* files.

Thus, in *pdf* and *ps* output, it's *Q* and the search functions of *Acroread* and *Ghostscript* can find it in any case...

I use the same trick for the *veryoldstyle* long *s*.

3 Some examples

3.1 Text

I use the example of *testfont.tex* and the *L^AT_EX Companion*, slightly altered when using the *veryoldstyle* option.

3.1.1 Default

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æsop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æsop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

3.1.2 Options *oldstylenums* and *light* or *textlight*

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æsop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æsop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

3.1.3 Option *nofligatures*

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æsop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æsop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

3.1.4 Option *oldstyle*

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æsop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æsop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

3.1.5 Option *veryoldstyle* and *light* or *textlight*

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æfop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

For the price of £45, almost anything can be found floating in fields. ¡THE DAZED BROWN FOX QUICKLY GAVE 12345-67890 JUMPS! — ¿But aren't Kafka's Schloß and Æfop's Œuvres often naïve vis-à-vis the dæmonic phœnix's official rôle in fluffy soufflés?

3.1.6 Quiz

Exercise: find the minimal set of package options that are used in each of these cases. Except when using the *veryoldstyle* option, the source file is always the same, sometimes upright, sometimes italic.

1. A.QUEER says: making 29 **active** characters if *definitely nasty!*
2. A.QUEER says: making 29 **active** characters is *definitely nasty!*
3. A.QUEER says: making 29 **active** characters is *definitely nasty!*

4. A.QUEER says: making 29 **active** characters is *definitely nasty!*
5. A.QUEER says: making 29 **active** characters is *definitely nasty!*
6. A.QUEER says: making 29 **active** characters is *definitely nasty!*
7. A.QUEER says: making 29 **active** characters is *definitely nasty!*
8. A.QUEER says: making 29 **active** characters is *definitely nasty!*
9. A.QUEER says: making 29 **active** characters is *definitely nasty!*
10. A.QUEER says: making 29 **active** characters is *definitely nasty!*

Read the solution at the end of the article!

If you were very attentive, you can get 10 points!

3.2 Math

The figures on the following pages show math samples. These also use an example from the *L^AT_EX Companion*...

3.3 This document

This article uses only the *textlight* option. Obviously, in some parts, the options described are simulated using `\fontfamily`...

In the math examples, I use two *special tricks* to get the narrow `\iiint` and the upright `\partial` symbol.

In both the text and math examples, the output is scaled to the available line length.

Personal names are in default small caps, and acronyms are in large small caps.

4 The limits of NFSS

4.1 Non-existing features

Some features of *kpfonts* don't exist in the new font selection scheme:

- Two sizes of small caps:
 - The commands `\textothersc{...}` and `\otherscshape` allow you to use both sizes. They are often used in this document.
 - The option *largesmallcaps* changes the default small caps size. Then, you can use standard commands for large small caps!
- slanted small caps:
 - The following commands allow you to use the slanted small capitals:
 - `\textscsl{...}`
 - `\scslshape`
 - `\textotherscsl{...}`
 - `\otherscslshape`

1 Sample page of mathematical typesetting

First some large operators both in text: $\iiint_{\mathbb{Q}} f(x, y, z) dx dy dz$ and $\prod_{\gamma \in \Gamma_{\mathbb{C}}} \partial(\tilde{X}_{\gamma})$; and also on display:

$$\begin{aligned} \iiint_{\mathbb{Q}} f(w, x, y, z) dw dx dy dz &\leq \oint_{\partial \mathbb{Q}} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{\|x \oplus y\|} \right\} \right) \\ &\approx \bigsqcup_{\mathbb{Q} \in \bar{\mathbb{Q}}} \left[f^* \left(\frac{f(\mathbb{Q}(t))}{\sqrt{1-t^2}} \right) \right]_{t=\alpha}^{t=\vartheta} \end{aligned} \quad (1)$$

For x in the open interval $] -1, 1[$ the infinite sum in Equation (2) is convergent; however, this does not hold throughout the closed interval $[-1, 1]$.

$$(1-x)^{-k} = 1 + \sum_{j=1}^{\infty} (-1)^j \binom{k}{j} x^j \quad \text{for } k \in \mathbb{N}; k \neq 0. \quad (2)$$

Figure 1: Default

1 Sample page of mathematical typesetting

First some large operators both in text: $\iiint_{\mathbb{Q}} f(x, y, z) dx dy dz$ and $\prod_{\gamma \in \Gamma_{\mathbb{C}}} \partial(\tilde{X}_{\gamma})$; and also on display:

$$\begin{aligned} \iiint_{\mathbb{Q}} f(w, x, y, z) dw dx dy dz &\leq \oint_{\partial \mathbb{Q}} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{\|x \oplus y\|} \right\} \right) \\ &\approx \bigsqcup_{\mathbb{Q} \in \bar{\mathbb{Q}}} \left[f^* \left(\frac{f(\mathbb{Q}(t))}{\sqrt{1-t^2}} \right) \right]_{t=\alpha}^{t=\vartheta} \end{aligned} \quad (1)$$

For x in the open interval $] -1, 1[$ the infinite sum in Equation (2) is convergent; however, this does not hold throughout the closed interval $[-1, 1]$.

$$(1-x)^{-k} = 1 + \sum_{j=1}^{\infty} (-1)^j \binom{k}{j} x^j \quad \text{for } k \in \mathbb{N}; k \neq 0. \quad (2)$$

Figure 2: Options *lightmath* and *narrowiints*

1 Sample page of mathematical typesetting

First some large operators both in text: $\iiint_{\mathcal{Q}} f(x, y, z) dx dy dz$ and $\prod_{\gamma \in \Gamma_{\bar{c}}} \partial(\tilde{X}_{\gamma})$; and also on display:

$$\begin{aligned} \iiint_{\mathcal{Q}} f(w, x, y, z) dw dx dy dz &\leq \oint_{\partial \mathcal{Q}} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{\|x \oplus y\|} \right\} \right) \\ &\approx \bigsqcup_{\mathcal{Q} \in \tilde{\mathcal{Q}}} \left[f^* \left(\frac{f(\mathcal{Q}(t))}{\sqrt{1-t^2}} \right) \right]_{t=\alpha}^{t=\vartheta} \end{aligned} \quad (1)$$

For x in the open interval $] -1, 1[$ the infinite sum in Equation (2) is convergent; however, this does not hold throughout the closed interval $[-1, 1]$.

$$(1-x)^{-k} = 1 + \sum_{j=1}^{\infty} (-1)^j \binom{k}{j} x^j \quad \text{for } k \in \mathbb{N}; k \neq 0. \quad (2)$$

Figure 3: Options *nofligatures* and *uprightgreek*s

1 Sample page of mathematical typesetting

First some large operators both in text: $\iiint_{\mathcal{Q}} f(x, y, z) dx dy dz$ and $\prod_{\gamma \in \Gamma_{\bar{c}}} \partial(\tilde{X}_{\gamma})$; and also on display:

$$\begin{aligned} \iiint_{\mathcal{Q}} f(w, x, y, z) dw dx dy dz &\leq \oint_{\partial \mathcal{Q}} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{\|x \oplus y\|} \right\} \right) \\ &\approx \bigsqcup_{\mathcal{Q} \in \tilde{\mathcal{Q}}} \left[f^* \left(\frac{f(\mathcal{Q}(t))}{\sqrt{1-t^2}} \right) \right]_{t=\alpha}^{t=\vartheta} \end{aligned} \quad (1)$$

For x in the open interval $] -1, 1[$ the infinite sum in Equation (2) is convergent; however, this does not hold throughout the closed interval $[-1, 1]$.

$$(1-x)^{-k} = 1 + \sum_{j=1}^{\infty} (-1)^j \binom{k}{j} x^j \quad \text{for } k \in \mathbb{N}; k \neq 0. \quad (2)$$

Figure 4: Options *lightmath* and *partialup*

1 Sample page of mathematical typesetting

First some large operators both in text: $\iiint_{\mathbb{Q}} f(x, y, z) dx dy dz$ and $\prod_{\gamma \in \Gamma_{\mathbb{C}}} \partial(\tilde{X}_{\gamma})$; and also on display:

$$\begin{aligned} \iiint_{\mathbb{Q}} f(w, x, y, z) dw dx dy dz &\leq \oint_{\partial \mathbb{Q}} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{\|x \oplus y\|} \right\} \right) \\ &\approx \bigcup_{\mathbb{Q} \in \mathbb{Q}} \left[f^* \left(\frac{f(\mathbb{Q}(t))}{\sqrt{1-t^2}} \right) \right]_{t=\alpha}^{t=\vartheta} \end{aligned} \quad (1)$$

For x in the open interval $] -1, 1[$ the infinite sum in Equation (2) is convergent; however, this does not hold throughout the closed interval $[-1, 1]$.

$$(1-x)^{-k} = 1 + \sum_{j=1}^{\infty} (-1)^j \binom{k}{j} x^j \quad \text{for } k \in \mathbb{N}; k \neq 0. \quad (2)$$

Figure 5: Option *sfmathbb*

1 Sample page of mathematical typesetting

First some large operators both in text: $\iiint_{\mathbb{Q}} f(x, y, z) dx dy dz$ and $\prod_{\gamma \in \Gamma_{\mathbb{C}}} \partial(\tilde{X}_{\gamma})$; and also on display:

$$\begin{aligned} \iiint_{\mathbb{Q}} f(w, x, y, z) dw dx dy dz &\leq \oint_{\partial \mathbb{Q}} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{\|x \oplus y\|} \right\} \right) \\ &\approx \bigcup_{\mathbb{Q} \in \mathbb{Q}} \left[f^* \left(\frac{f(\mathbb{Q}(t))}{\sqrt{1-t^2}} \right) \right]_{t=\alpha}^{t=\vartheta} \end{aligned} \quad (1)$$

For x in the open interval $] -1, 1[$ the infinite sum in Equation (2) is convergent; however, this does not hold throughout the closed interval $[-1, 1]$.

$$(1-x)^{-k} = 1 + \sum_{j=1}^{\infty} (-1)^j \binom{k}{j} x^j \quad \text{for } k \in \mathbb{N}; k \neq 0. \quad (2)$$

Figure 6: Options *lightmath*, *fulloldstylenums* and *widernum*

1 Sample page of mathematical typesetting

First some large operators both in text: $\iiint_{\mathcal{Q}} f(x, y, z) dx dy dz$ and $\prod_{\gamma \in \Gamma_{\bar{c}}} \partial(\tilde{X}_{\gamma})$; and also on display:

$$\begin{aligned} \iiint_{\mathcal{Q}} f(w, x, y, z) dw dx dy dz &\leq \oint_{\partial \mathcal{Q}} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{\|x \oplus y\|} \right\} \right) \\ &\approx \bigcup_{\mathcal{Q} \in \bar{\mathcal{Q}}} \left[f^* \left(\frac{|\mathcal{Q}(t)|}{\sqrt{1-t^2}} \right) \right]_{t=\alpha}^{t=\beta} \end{aligned} \quad (1)$$

For x in the open interval $] -1, 1[$ the infinite sum in Equation (2) is convergent; however, this does not hold throughout the closed interval $[-1, 1]$.

$$(1-x)^{-k} = 1 + \sum_{j=1}^{\infty} (-1)^j \binom{k}{j} x^j \quad \text{for } k \in \mathbb{N}; k \neq 0. \quad (2)$$

Figure 7: Option *sfmath*

1 Sample page of mathematical typesetting

First some large operators both in text: $\iiint_{\mathcal{Q}} f(x, y, z) dx dy dz$ and $\prod_{\gamma \in \Gamma_{\bar{c}}} \partial(\tilde{X}_{\gamma})$; and also on display:

$$\begin{aligned} \iiint_{\mathcal{Q}} f(w, x, y, z) dw dx dy dz &\leq \oint_{\partial \mathcal{Q}} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{\|x \oplus y\|} \right\} \right) \\ &\approx \bigcup_{\mathcal{Q} \in \bar{\mathcal{Q}}} \left[f^* \left(\frac{|\mathcal{Q}(t)|}{\sqrt{1-t^2}} \right) \right]_{t=\alpha}^{t=\beta} \end{aligned} \quad (1)$$

For x in the open interval $] -1, 1[$ the infinite sum in Equation (2) is convergent; however, this does not hold throughout the closed interval $[-1, 1]$.

$$(1-x)^{-k} = 1 + \sum_{j=1}^{\infty} (-1)^j \binom{k}{j} x^j \quad \text{for } k \in \mathbb{N}; k \neq 0. \quad (2)$$

Figure 8: Options *sfmath*, *narrowiints*, *uprightgreek*s and *partialup*

- You can also use the *easyscs* option or the *slantsc* package to get slanted small caps as expected: `\textsc{\textsl{...}}`. But, you have to redefine these commands and the result can be disappointing!

```
\documentclass{minimal}
\usepackage{palatino}
\usepackage{slantsc}
\begin{document}
\textsl{\textsc{Hello}}
\end{document}
```

gives you 2 warnings... and the output is an upright “Hello” in *palatino*! This is the reason that the *easyscs* option of *kpfonts* gives an explicit warning.

- The option *largesmallcaps* also affects the default slanted small caps size.
- light variant fonts:
 - Because of the edge effects described below, there are no commands to switch between default and light fonts.

For instance, you have to redefine commands like `\textit`: therefore, you want the *italic* of the **actual** font, not the default!

- But the option *rmx* allows you to use these fonts without the usual

```
\fontfamily{...}\selectfont!
```

The corresponding table:

option	weight	<i>rmx</i>
<i>light</i>	m	l
	m	m
<i>light</i>	b/bx	sb/sbx
	b/bx	b/bx

4.2 Exponential number of files

4.2.1 Usual case

In most cases, when you have a font family, like *URW Garamond*, you get basically 4 fonts:

- upright or italic, and,
- normal or bold.

Thus

- 4 pairs *pf**b*/*af**m*, the design of characters and the metrics, and/or,
- 4 *ttf* or *otf* files including design and metrics.

(\LaTeX) doesn’t need the design of the characters, they need only the metrics, the *tfm* files to build the *dvi*. It’s one of the reasons why *dvi* files are small.

A *dvi* viewer or *dvips* (or equivalent) or *pdf \TeX* does need the actual characters to produce the final document, of course.

We describe briefly the chain of events to find the good metrics. Now, imagine you are \TeX (it’s easy if you try):

- You have an active family and encoding, for instance *jkp* and T1, i.e. default *kpfonts* family and Cork encoding,
- you read the `t1jkp.fd` file, for font definitions,
- you have an active shape and weight, for instance *it* and *n*, i.e. italic and normal weight,
- you read the line:


```
\DeclareFontShape{T1}{jkp}{m}{it}
      {<-> jkpmi8t}{}
      
```

 in the *fd* file. *jkpmi8t* is the needed *tfm* file, including metrics, ligatures and kerning.

Usual cases, like *Garamond*, require less than 50 files for an OT1 and T1 installation...

4.2.2 *kpfonts* case

For “hackers”, special use or curiosity, look at the rules to build the corresponding family names:

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

with the corresponding options:

l, x	light, <i>rmx</i>
k	<i>largesmallcaps</i>
f	<i>nofligatures</i>
osn, os, vos	<i>oldstylenums</i> , <i>oldstyle</i> , <i>veryoldstyle</i>

For the roman fonts, because we can choose between OT1 and T1 encoding, we have 72 families, excluding the TS1 ones for *textcomp*... The total number of families is 187 in the 3.12 release!

Most of the roman families have 15 *tfm* metrics:

- upright, italic, small caps, slanted, slanted small caps,
- each in normal, bold and bold extended...

But each *tfm* corresponds here to a *vf*, virtual font, file because there is no direct link between these *tfm* files and a *pf**b* file!

In the 3.12 release, we get

- 668 virtual font, *vf*, files,
- 858 tex font metric, *tfm*, files.

The *kpfonts* TDS tree has a total of 1,875 files...

I’m on my own and it’s impossible for me to be ensure with any probability that there is no bug! Indeed, writing this paper, I found one bug: *light* and *veryoldstyle* medium weight font were not *light* but the default!

4.2.3 About a new option

A new option, long tail Q without special ligatures *ct* and *st*, but with old style numbers or not, was requested by a user. This would mean 240 more *tfm* files, 240 more *vf* files and 24 more *fd* files, this just for T1 encoding and roman fonts.

For sans serif fonts, it's 96 *tfm*, 96 *vf* and 8 *fd* new files. No action and no more files for teletype fonts!

The number required by the OT1 encoding is the same, for a complete sum of 1,408 new files... Increasing the total number of *kpfonts*' files about 75%!

You see here the explicit exponential effect!

In fact, this option would not be hard to install (2 new *etx* files, the encoding files for *fontinst*, and some new lines in the installation file), but I don't agree with the request because there are already commands `\othertailQ` and `\othertailscq` to do the work...

4.2.4 Last way to be free

And if you want some options to choose freely:

- classic L^AT_EX *f* ligatures or not,
- *cl* ligature or not,
- *sl* ligature or not,
- oldstyle or lining numbers,
- round *s* or long *f*,
- long tail *Q* or classic *Q*,

if I'm not wrong it's about 20,000 files more...

The object of *kpfonts* is not to increase indefinitely the number of files on your hard disk!

The object of *kpfonts* is not to be in the Guinness book!

I don't think it's sensible to exceed 2,000 files in a package, even if it's possible!

To go further, to be free, I think somebody has to build some *otf* fonts using their advanced possibilities and has to use it running X_YL_AT_EX or Lua_TE_X, but that's another challenge...

Obviously, *otf* fonts will solve the above features problems without an exponential number of files, but won't easily solve these:

- small or large small caps;
- light or default fonts.

5 The end

Now I think the work is (almost) done and I'm proud of three things:

- the package runs mainly correctly,
- some people like the fonts and some people don't like them,
- some people like to customize their text and/or math typesetting using the set of options.

If everybody finds these three axioms are reasonable, you know what, I'm happy...

If I, if I have been unkind,
I hope that you can just let it go by.

LEONARD COHEN, *Bird on the wire*

◇ Christophe Caignaert
<http://ctan.org/pkg/kpfonts>

Solution to the quiz (p.168):

1. *lighhtext* and *veryoldstyle*
2. *oldstylelennums* and *largesmallcaps*
3. *easyscsl* and *oldstyle*
4. *lighhtext* and *nofligatures*
5. *lighhtext* and *oldstyle*
6. no options
7. *nofligatures*
8. *lighhtext*
9. *largesmallcaps*
10. *lighhtext*, *largesmallcaps*, *easyscsl* and *nofligatures*