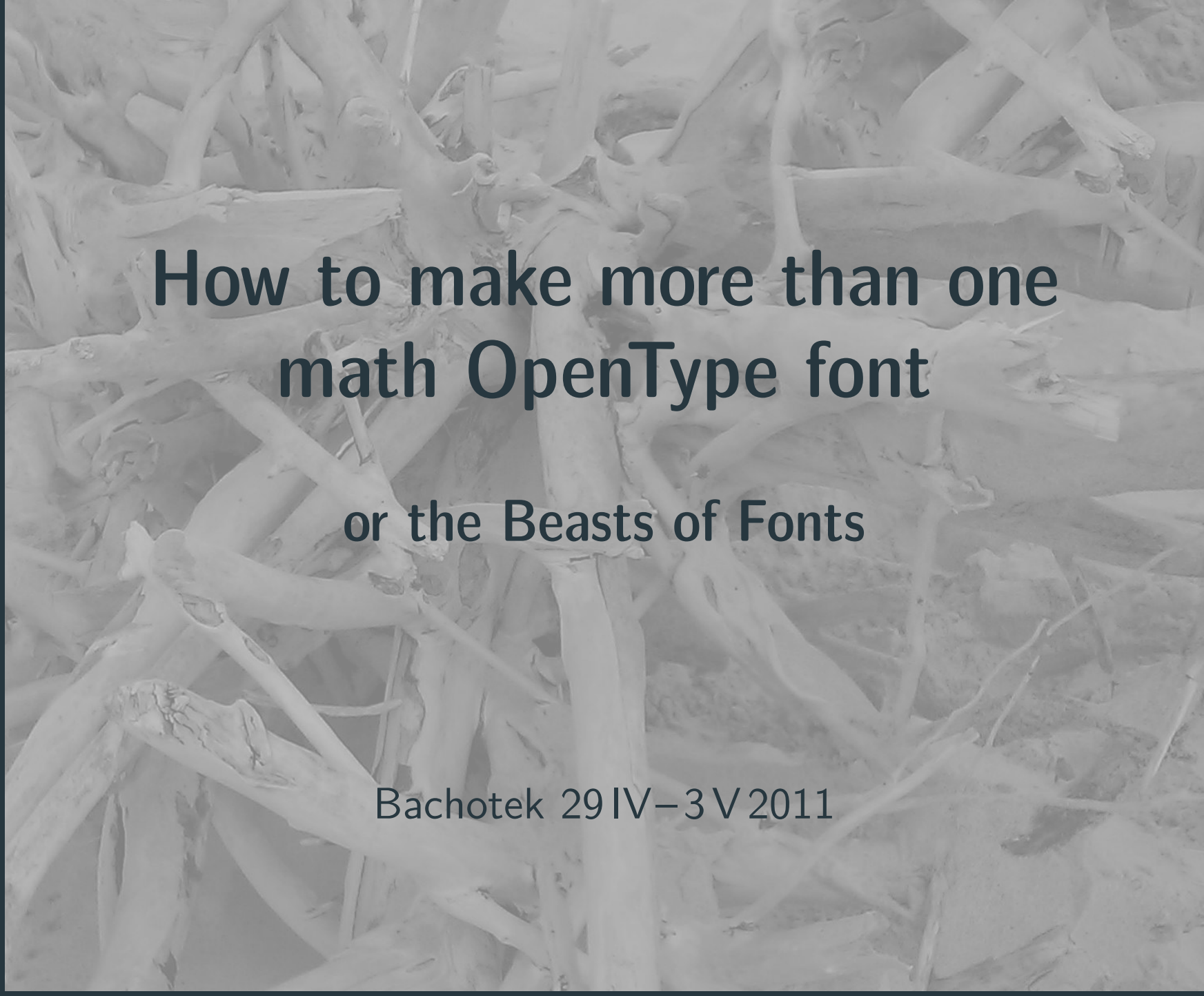


How to make more than one math OpenType font

or the Beasts of Fonts

Bachotek 29 IV – 3 V 2011

Bogusław Jackowski and Piotr Strzelczyk

It is widely known, that fonts are nasty beasts

It is widely known, that fonts are nasty beasts

. . . and that math fonts are nasty beasts to n^{th} power.

It is widely known, that fonts are nasty beasts

... and that math fonts are nasty beasts to n^{th} power.

Actually, fonts are not beasts at all, but there are a few Beasts that govern the Realm of Fonts.

It is widely known, that fonts are nasty beasts

... and that math fonts are nasty beasts to n^{th} power.

Actually, fonts are not beasts at all, but there are a few Beasts that govern the Realm of Fonts.

This is exactly the reason why since 2007, when Microsoft released their math-equipped Office with the math OpenType font Cambria, only two more math OpenType fonts were created: Asana (Apostolos Syropoulos) and XITS (Khaled Hosny), both stemming from the TEX clique.

THE BEAST OF ENTANGLEMENT

The fundamental question

The fundamental question

What is a what of a what? – Winnie the Pooh.

The fundamental question

What is a what of a what? – Winnie the Pooh.

It is extremely difficult to identify the nature of problems.

The fundamental question

What is a what of a what? – Winnie the Pooh.

It is extremely difficult to identify the nature of problems:

- which are the result of simple mistakes

The fundamental question

What is a what of a what? – Winnie the Pooh.

It is extremely difficult to identify the nature of problems:

- which are the result of simple mistakes,
- which are the result of idiosyncratic decisions

The fundamental question

What is a what of a what? – Winnie the Pooh.

It is extremely difficult to identify the nature of problems:

- which are the result of simple mistakes,
- which are the result of idiosyncratic decisions,
- which are the remnants of a former era

The fundamental question

What is a what of a what? – Winnie the Pooh.

It is extremely difficult to identify the nature of problems:

- which are the result of simple mistakes,
- which are the result of idiosyncratic decisions,
- which are the remnants of a former era,
- etc...

The fundamental question

What is a what of a what? – Winnie the Pooh.

It is extremely difficult to identify the nature of problems:

- which are the result of simple mistakes,
- which are the result of idiosyncratic decisions,
- which are the remnants of a former era,
- etc...

Of course, the identification is crucial to fix the problem.

An example – where is the math italic letter ‘h’?

An example – where is the math italic letter ‘h’?

Final Unicode Names List (ver. 6.0, 2010) reads:

[...]
U+1D452 MATHEMATICAL ITALIC SMALL E
U+1D453 MATHEMATICAL ITALIC SMALL F
U+1D454 MATHEMATICAL ITALIC SMALL G
U+1D456 MATHEMATICAL ITALIC SMALL I
U+1D457 MATHEMATICAL ITALIC SMALL J
U+1D458 MATHEMATICAL ITALIC SMALL K
[...]

An example – where is the math italic letter ‘h’?

Final Unicode Names List (ver. 6.0, 2010) reads:

[...]
U+1D452 MATHEMATICAL ITALIC SMALL E
U+1D453 MATHEMATICAL ITALIC SMALL F
U+1D454 MATHEMATICAL ITALIC SMALL G ⟵?
U+1D456 MATHEMATICAL ITALIC SMALL I
U+1D457 MATHEMATICAL ITALIC SMALL J
U+1D458 MATHEMATICAL ITALIC SMALL K
[...]

An example – where is the math italic letter ‘h’?

Final Unicode Names List (ver. 6.0, 2010) reads:

[...]
U+1D452 MATHEMATICAL ITALIC SMALL E
U+1D453 MATHEMATICAL ITALIC SMALL F
U+1D454 MATHEMATICAL ITALIC SMALL G ⟵?
U+1D456 MATHEMATICAL ITALIC SMALL I
U+1D457 MATHEMATICAL ITALIC SMALL J
U+1D458 MATHEMATICAL ITALIC SMALL K
[...]

The math italic letter ‘h’ has the Planck constant meaning permanently assigned:

U+210E PLANCK CONSTANT
= height, specific enthalpy,
simply a mathematical italic h;
this character’s name results from legacy usage

An example – where is the math italic letter ‘h’?

Final Unicode Names List (ver. 6.0, 2010) reads:

[...]
U+1D452 MATHEMATICAL ITALIC SMALL E
U+1D453 MATHEMATICAL ITALIC SMALL F
U+1D454 MATHEMATICAL ITALIC SMALL G ⟵?
U+1D456 MATHEMATICAL ITALIC SMALL I
U+1D457 MATHEMATICAL ITALIC SMALL J
U+1D458 MATHEMATICAL ITALIC SMALL K
[...]

The math italic letter ‘h’ has the Planck constant meaning permanently assigned:

U+210E PLANCK CONSTANT
= height, specific enthalpy,
simply a mathematical italic h;
this character’s name results from legacy usage

An example – where is the math italic letter ‘h’?

Final Unicode Names List (ver. 6.0, 2010) reads:

[...]
U+1D452 MATHEMATICAL ITALIC SMALL E
U+1D453 MATHEMATICAL ITALIC SMALL F
U+1D454 MATHEMATICAL ITALIC SMALL G ←?
U+1D456 MATHEMATICAL ITALIC SMALL I
U+1D457 MATHEMATICAL ITALIC SMALL J
U+1D458 MATHEMATICAL ITALIC SMALL K
[...]

The math italic letter ‘h’ has the Planck constant meaning permanently assigned:

U+210E PLANCK CONSTANT
= height, specific enthalpy,
simply a mathematical italic h;
this character’s name results from legacy usage

In this particular case, misconception, idiosyncrasy and relics coincide.


THE BEAST
OF MISCONCEPTION

Two basic misconceptions

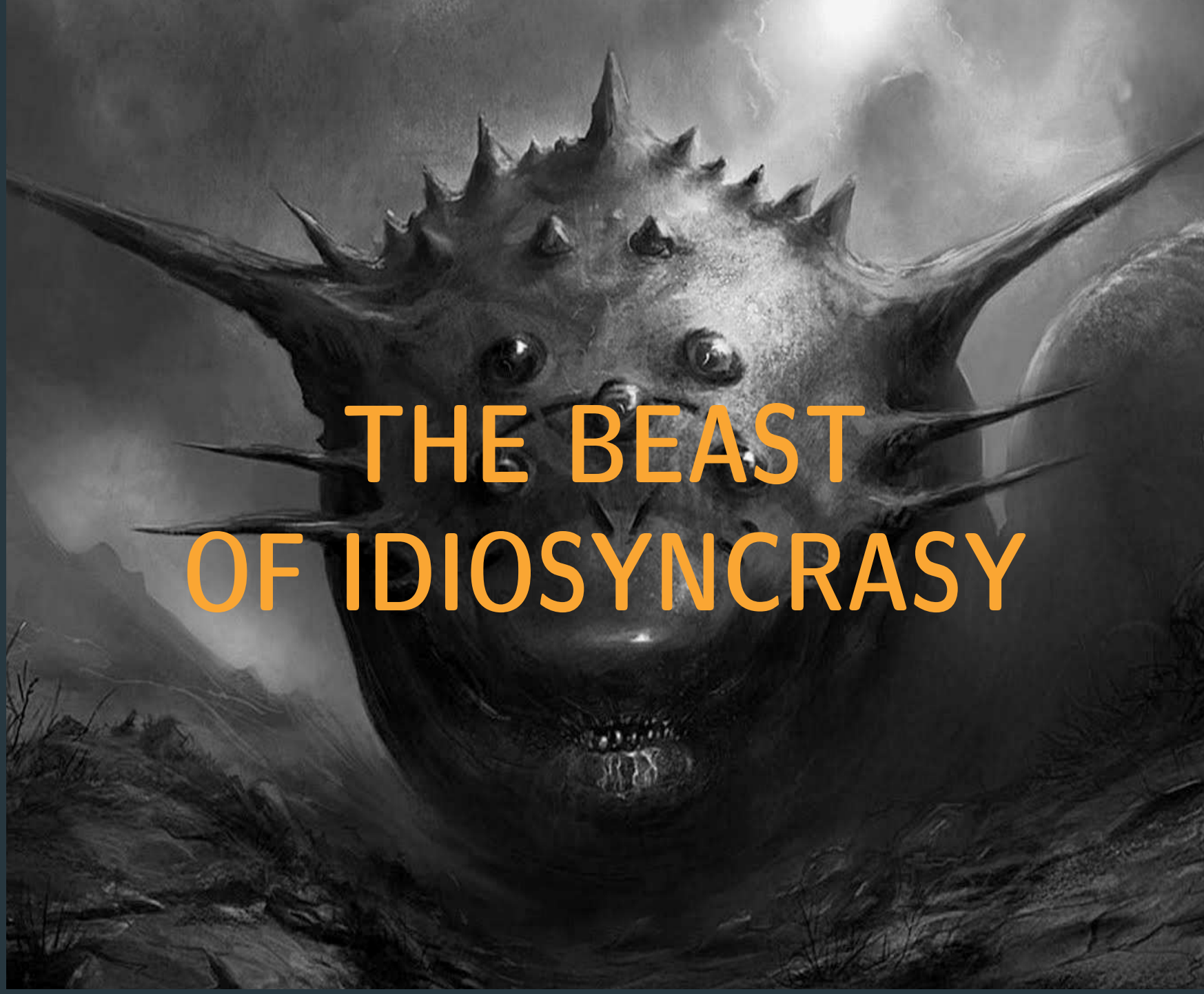
Two basic misconceptions

A contemporary font could be a smarter structure than a collection of fixed (predefined) shapes. It was an ingenious idea in Gutenberg's times. Today, instead of fancy yet obscure techniques involved in OpenType fonts (so called "features"), glyphs could be implemented as drawing programs. This would be especially beneficial in the case of math fonts.

Two basic misconceptions

A contemporary font could be a smarter structure than a collection of fixed (predefined) shapes. It was an ingenious idea in Gutenberg's times. Today, instead of fancy yet obscure techniques involved in OpenType fonts (so called "features"), glyphs could be implemented as drawing programs. This would be especially beneficial in the case of math fonts.

Also, the commonly adored Unicode often, in practice, turns out to be an obstacle. The idea of enumerating all entities used in various areas of human intellectual activities seems somewhat insane. Moreover, enumerating is static, while it is a dynamically adjustable protocol which is needed (cf. the success of the TCP/IP protocol).


THE BEAST
OF IDIOSYNCRASY

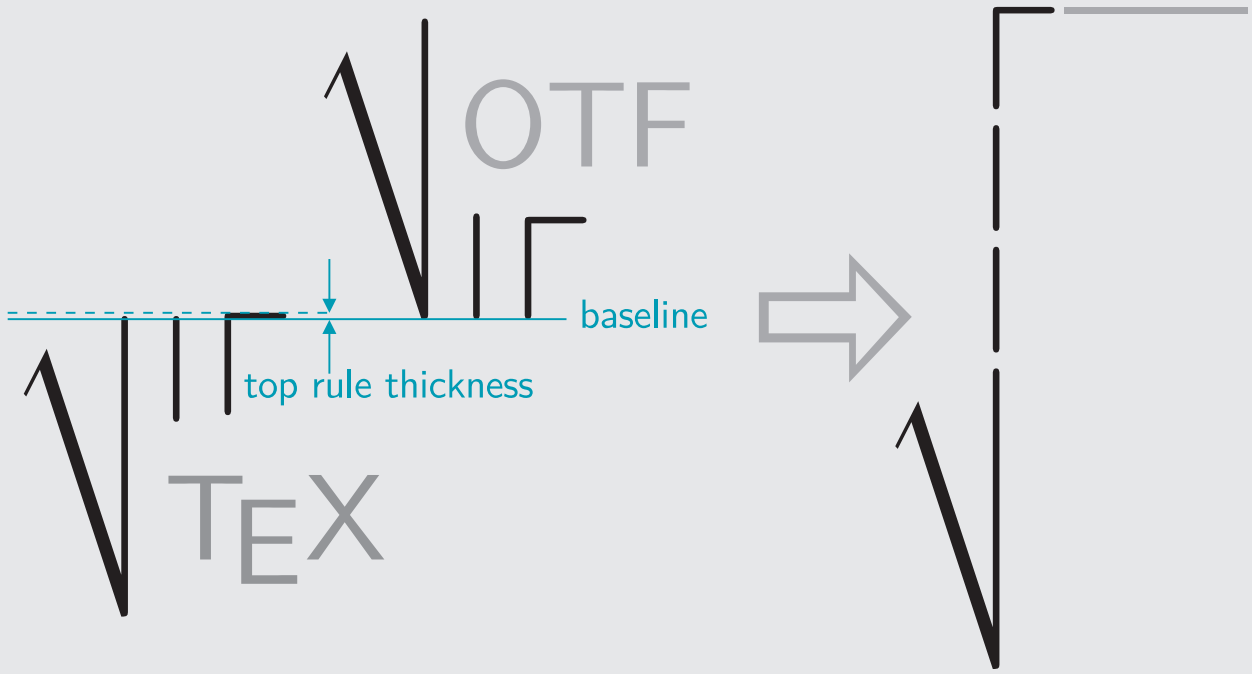
Let's do it somehow...

Let's do it somehow...

Tricky, peculiar solutions are vexingly frequent.

Let's do it somehow...

Tricky, peculiar solutions are vexingly frequent.



Mathematical glyphs, vertically assembled from smaller pieces, like e.g., the radical, must be placed above the baseline in OpenType fonts and below the baseline in traditional TEX fonts (more precisely, in TEX, the top element of the radical must slightly protrude above the baseline – the amount conveys the thickness of the radical's top rule).


THE BEAST
OF RELICS

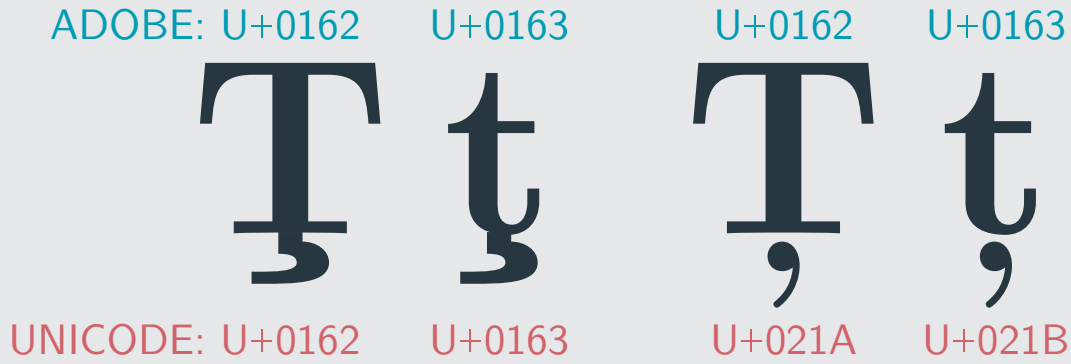
Everlasting relics

Everlasting relics

Relics are the most aggravating obstacles because they are permanent and there is no hope for their removal. On the contrary – the longer they stay, the deeper rooted they are.

Everlasting relics

Relics are the most aggravating obstacles because they are permanent and there is no hope for their removal. On the contrary – the longer they stay, the deeper rooted they are.



A symptomatic example is the case of glyphs *tcommaaccent* and *Tcommaaccent*. According to the Adobe Glyph List for New Fonts (2010), they are just aliases for *tcedilla* (U+0163) and *Tcedilla* (U+0162), respectively. The Unicode Standard is unequivocal here: the Latin small letter 't' with comma below should have the code U+021B, while the Latin capital letter 'T' with comma below – the code U+021A. After a few attempts to improve the situation, Adobe eventually gave up and reverted to the *status quo ante bellum*.

Everlasting relics

Relics are the most aggravating obstacles because they are permanent and there is no hope for their removal. On the contrary – the longer they stay, the deeper rooted they are.

Final Unicode Names List (ver. 6.0, 2010) reads:

FE18 PRESENTATION FORM FOR VERTICAL RIGHT WHITE LENTICULAR BRAKCET
% PRESENTATION FORM FOR VERTICAL RIGHT WHITE LENTICULAR BRACKET
∗ misspelling of "BRACKET" in character name is a known defect

1D0C5 BYZANTINE MUSICAL SYMBOL FHTORA SKLIRON CHROMA VASIS
% BYZANTINE MUSICAL SYMBOL FTHORA SKLIRON CHROMA VASIS
∗ misspelling of "FTHORA" in character name is a known defect

Everlasting relics

Relics are the most aggravating obstacles because they are permanent and there is no hope for their removal. On the contrary – the longer they stay, the deeper rooted they are.

Final Unicode Names List (ver. 6.0, 2010) reads:

FE18 PRESENTATION FORM FOR VERTICAL RIGHT WHITE LENTICULAR BRAKCET
% PRESENTATION FORM FOR VERTICAL RIGHT WHITE LENTICULAR BRACKET
* misspelling of "BRACKET" in character name is a known defect

1D0C5 BYZANTINE MUSICAL SYMBOL FHTORA SKLIRON CHROMA VASIS
% BYZANTINE MUSICAL SYMBOL FTHORA SKLIRON CHROMA VASIS
* misspelling of "FTHORA" in character name is a known defect

In this gloomy world

In this gloomy world,
in spite of vile machinations of the Beasts of Fonts

In this gloomy world,
in spite of vile machinations of the Beasts of Fonts,
Latin Modern Math OTF happily came to the world.

In this gloomy world,
in spite of vile machinations of the Beasts of Fonts,
Latin Modern Math OTF happily came to the world.

The creating of the Latin Modern Math OTF was not as perplexing as it might have been. It is the result of shared experience and efforts by Barbara Beeton, Hans Hagen, Taco Hoekwater, Khaled Hosny, Alan Jeffrey, Adam Twardoch, Ulrik Vieth, George Williams and others, which made our struggle against the insubordinate font matter significantly less painful.

Very many thanks!

**In this gloomy world,
in spite of vile machinations of the Beasts of Fonts,
Latin Modern Math OTF happily came to the world.**

The creating of the Latin Modern Math OTF was not as perplexing as it might have been. It is the result of shared experience and efforts by Barbara Beeton, Hans Hagen, Taco Hoekwater, Khaled Hosny, Alan Jeffrey, Adam Twardoch, Ulrik Vieth, George Williams and others, which made our struggle against the insubordinate font matter significantly less painful.

Very many thanks!

Of course, it is an "incunabulum" font, i.e., it is in its infancy and will be intensively developed in the nearest future.

LATIN MODERN MATH

THE DOMESTICATED MONSTER

The objective

The objective

It should be noted that none of the OpenType math fonts that appeared so far is perfect. In this number, of course, Latin Modern Math. Our aim, however, was not creating just one more math OpenType font. We wanted *to pave the way for next math fonts*, in particular, for selected TEX Gyre fonts.

The objective

It should be noted that none of the OpenType math fonts that appeared so far is perfect. In this number, of course, Latin Modern Math. Our aim, however, was not creating just one more math OpenType font. We wanted *to pave the way for next math fonts*, in particular, for selected TEX Gyre fonts.

We started from Latin Modern, i.e., Computer Modern math, as we believed that converting the TEX math structures into the OpenType TEX-based ones, should be simpler than creating the OpenType math structures from scratch.

The objective

It should be noted that none of the OpenType math fonts that appeared so far is perfect. In this number, of course, Latin Modern Math. Our aim, however, was not creating just one more math OpenType font. We wanted *to pave the way for next math fonts*, in particular, for selected TEX Gyre fonts.

We started from Latin Modern, i.e., Computer Modern math, as we believed that converting the TEX math structures into the OpenType TEX-based ones, should be simpler than creating the OpenType math structures from scratch.

We believe that, in general, it was an appropriate approach: first results we obtain after two months of coding. We underestimated a little bit the complexity arising from the heterogeneity of sources. But we drastically underestimated the time needed to spot the adequate glyph, its (uni)code and name (conforming to Adobe/Microsoft "standards").

The approach

The approach

Our starting point was a collection of several PostScript Type 1 font files.

The approach

Our starting point was a collection of several PostScript Type 1 font files, namely:

bbold10, bbold7, bbold5, eufb10, eufb7, eufb5,
eufm10, eufm7, eufm5, eusb10, eusb7, eusb5, eusm10, eusm7, eusm5,
lmbsy10, lmbsy7, lmbsy5, lmbx10, lmbx7, lmbx5, lmex10,
lmmi10, lmmi7, lmmi5, lmmib10, lmmib7, lmmib5, lmr10, lmr7, lmr5,
lmss12, lmss10, lmss8, lmssbo10, lmssbx10,
lmsso12, lmsso10, lmsso8, lmsy10, lmsy7, lmsy5,
lmtt12, lmtt10, lmtt8, msam10, msam7, msam5,
msbm10, msbm7, msbm5 – altogether 51 fonts.

The approach

Our starting point was a collection of several PostScript Type 1 font files, namely:

bbold10, bbold7, bbold5, eufb10, eufb7, eufb5,
eufm10, eufm7, eufm5, eusb10, eusb7, eusb5, eusm10, eusm7, eusm5,
lmbsy10, lmbsy7, lmbsy5, lmbx10, lmbx7, lmbx5, lmex10,
lmmi10, lmmi7, lmmi5, lmmib10, lmmib7, lmmib5, lmr10, lmr7, lmr5,
lmss12, lmss10, lmss8, lmssbo10, lmssbx10,
lmsso12, lmsso10, lmsso8, lmsy10, lmsy7, lmsy5,
lmtt12, lmtt10, lmtt8, msam10, msam7, msam5,
msbm10, msbm7, msbm5 – altogether 51 fonts.

All these fonts had to be crammed into one huge OTF file.
Actually, Latin Modern Math contains more than 4500 glyphs,
and many more are needed. Why is that so?!

The approach

According to the *Draft Unicode Technical Report #25* by Barbara Beeton, Asmus Freytag and Murray Sargent III, math font should contain the following groups of glyphs:

plain (upright, serifed)	Latin*, Greek and digits	lmr, lmmi (uprighted)
bold	Latin, Greek and digits	lmbx, lmmib (uprighted)
italic	Latin and Greek	lmmi
bold italic	Latin and Greek	lmmib
script (calligraphic)	Latin**	eusm (slanted)
bold script (calligraphic)	Latin**	eusb (slanted)
Fraktur	Latin	eufm
bold Fraktur	Latin	eufb
double-struck	Latin and digits	bbold (by Alan Jeffrey)
sans-serif	Latin and digits	lmss
sans-serif bold	Latin, Greek*** and digits	lmssbx
sans-serif italic	Latin	lmsso
sans-serif bold italic	Latin and Greek***	lmssbo
monospace	Latin and digits	lmtt

* plus basic diacritical characters
** caps only (like in CMs – temporarily?)
*** caps only (temporarily)

The approach

The process of generating a math OpenType font out of the aforementioned 51 PostScript Type 1 files comprised

The approach

The process of generating a math OpenType font out of the aforementioned 51 PostScript Type 1 files comprised:

- the preparation of a Python-based FontForge-geared engine (FFDKO) for converting PostScript Type 1 fonts into the OpenType format – an enhanced equivalent of the MakeOTF utility from Adobe Font Development Kit for OpenType

The approach

The process of generating a math OpenType font out of the aforementioned 51 PostScript Type 1 files comprised:

- the preparation of a Python-based FontForge-geared engine (FFDKO) for converting PostScript Type 1 fonts into the OpenType format – an enhanced equivalent of the MakeOTF utility from Adobe Font Development Kit for OpenType,
- converting of the PostScript Type 1 stuff into METATYPE 1 sources

The approach

The process of generating a math OpenType font out of the aforementioned 51 PostScript Type 1 files comprised:

- the preparation of a Python-based FontForge-geared engine (FFDKO) for converting PostScript Type 1 fonts into the OpenType format – an enhanced equivalent of the MakeOTF utility from Adobe Font Development Kit for OpenType,
- converting of the PostScript Type 1 stuff into METATYPE 1 sources,
- semi-manual adjustment of the sources (some characters were added, e.g., horizontally extendible braces, some removed, some modified, some extra data was computed, e.g., glyph axes, etc.)

The approach

The process of generating a math OpenType font out of the aforementioned 51 PostScript Type 1 files comprised:

- the preparation of a Python-based FontForge-geared engine (FFDKO) for converting PostScript Type 1 fonts into the OpenType format – an enhanced equivalent of the MakeOTF utility from Adobe Font Development Kit for OpenType,
- converting of the PostScript Type 1 stuff into METATYPE 1 sources,
- semi-manual adjustment of the sources (some characters were added, e.g., horizontally extendible braces, some removed, some modified, some extra data was computed, e.g., glyph axes, etc.),
- the generation of a single PostScript Type 1 font containing all relevant glyphs using METATYPE 1

The approach

The process of generating a math OpenType font out of the aforementioned 51 PostScript Type 1 files comprised:

- the preparation of a Python-based FontForge-geared engine (FFDKO) for converting PostScript Type 1 fonts into the OpenType format – an enhanced equivalent of the MakeOTF utility from Adobe Font Development Kit for OpenType,
- converting of the PostScript Type 1 stuff into METATYPE 1 sources,
- semi-manual adjustment of the sources (some characters were added, e.g., horizontally extendible braces, some removed, some modified, some extra data was computed, e.g., glyph axes, etc.),
- the generation of a single PostScript Type 1 font containing all relevant glyphs using METATYPE 1,
- semi-manual preparation of the relevant OTF math data for FFDKO, (the TEX metric data was extracted from TFMs using Python scripts)

The approach

The process of generating a math OpenType font out of the aforementioned 51 PostScript Type 1 files comprised:

- the preparation of a Python-based FontForge-geared engine (FFDKO) for converting PostScript Type 1 fonts into the OpenType format – an enhanced equivalent of the MakeOTF utility from Adobe Font Development Kit for OpenType,
- converting of the PostScript Type 1 stuff into METATYPE 1 sources,
- semi-manual adjustment of the sources (some characters were added, e.g., horizontally extendible braces, some removed, some modified, some extra data was computed, e.g., glyph axes, etc.),
- the generation of a single PostScript Type 1 font containing all relevant glyphs using METATYPE 1,
- semi-manual preparation of the relevant OTF math data for FFDKO, (the TEX metric data was extracted from TFMs using Python scripts),
- converting of the PostScript Type 1 font generated by METATYPE 1 into the OpenType format using FFDKO.

Future works

Future works

As was mentioned, Latin Modern Math OTF is an “incunabulum” font and definitely needs improvement.

Future works

As was mentioned, Latin Modern Math OTF is an “incunabulum” font and definitely needs improvement, in particular:

- enhancing the repertoire of glyphs: bold sans serif lowercase Greek, more math symbols (longer chains of vertically and horizontally extendible character, adding glyphs considered “canonical”, whatever would this mean, etc.)

Future works

As was mentioned, Latin Modern Math OTF is an "incunabulum" font and definitely needs improvement, in particular:

- enhancing the repertoire of glyphs: bold sans serif lowercase Greek, more math symbols (longer chains of vertically and horizontally extendible character, adding glyphs considered "canonical", whatever would this mean, etc.),
- checking the naming and coding scheme – this is a very tedious and time-consuming task

Future works

As was mentioned, Latin Modern Math OTF is an "incunabulum" font and definitely needs improvement, in particular:

- enhancing the repertoire of glyphs: bold sans serif lowercase Greek, more math symbols (longer chains of vertically and horizontally extendible character, adding glyphs considered "canonical", whatever would this mean, etc.),
- checking the naming and coding scheme – this is a very tedious and time-consuming task,
- complementing math data (TEX-based kerns, OpenType-specific kerns, improving the placement of axes, etc.)

Future works

As was mentioned, Latin Modern Math OTF is an "incunabulum" font and definitely needs improvement, in particular:

- enhancing the repertoire of glyphs: bold sans serif lowercase Greek, more math symbols (longer chains of vertically and horizontally extendible character, adding glyphs considered "canonical", whatever would this mean, etc.),
- checking the naming and coding scheme – this is a very tedious and time-consuming task,
- complementing math data (TEX-based kerns, OpenType-specific kerns, improving the placement of axes, etc.),
- adding selected OTF features (ssty, onum, etc.)

Future works

As was mentioned, Latin Modern Math OTF is an "incunabulum" font and definitely needs improvement, in particular:

- enhancing the repertoire of glyphs: bold sans serif lowercase Greek, more math symbols (longer chains of vertically and horizontally extendible character, adding glyphs considered "canonical", whatever would this mean, etc.),
- checking the naming and coding scheme – this is a very tedious and time-consuming task,
- complementing math data (TEX-based kerns, OpenType-specific kerns, improving the placement of axes, etc.),
- adding selected OTF features (ssty, onum, etc.),
- a question: which glyphs should have index forms?

Conclusions

Conclusions

The conclusive test of our approach will be the time needed to prepare other math fonts. We hope that once the Latin Modern Math OTF is released, a few months will suffice to make the next math font. We are going to add math to TEX Gyre Pagella first.

Conclusions

The conclusive test of our approach will be the time needed to prepare other math fonts. We hope that once the Latin Modern Math OTF is released, a few months will suffice to make the next math font. We are going to add math to TEX Gyre Pagella first. We hope for the success basing on the following factors:

- experience (especially, discernment in pitfalls).

Conclusions

The conclusive test of our approach will be the time needed to prepare other math fonts. We hope that once the Latin Modern Math OTF is released, a few months will suffice to make the next math font. We are going to add math to TEX Gyre Pagella first. We hope for the success basing on the following factors:

- experience (especially, discernment in pitfalls);
- Python scripts and other utilities (e.g., testing tools) are reusable to a large extent.

Conclusions

The conclusive test of our approach will be the time needed to prepare other math fonts. We hope that once the Latin Modern Math OTF is released, a few months will suffice to make the next math font. We are going to add math to TEX Gyre Pagella first. We hope for the success basing on the following factors:

- experience (especially, discernment in pitfalls);
- Python scripts and other utilities (e.g., testing tools) are reusable to a large extent;
- the data base of checked and verified glyphs names and unicode is fully reusable.

Conclusions

The conclusive test of our approach will be the time needed to prepare other math fonts. We hope that once the Latin Modern Math OTF is released, a few months will suffice to make the next math font. We are going to add math to TEX Gyre Pagella first. We hope for the success basing on the following factors:

- experience (especially, discernment in pitfalls);
- Python scripts and other utilities (e.g., testing tools) are reusable to a large extent;
- the data base of checked and verified glyphs names and unicode is fully reusable;
- math glyphs (operators, relational symbols, geometrical symbols) are to be created from scratch which seems to be the most laborious part of the job to be done – this does not mean, however, the most difficult part.

Conclusions

The conclusive test of our approach will be the time needed to prepare other math fonts. We hope that once the Latin Modern Math OTF is released, a few months will suffice to make the next math font. We are going to add math to TEX Gyre Pagella first. We hope for the success basing on the following factors:

- experience (especially, discernment in pitfalls);
- Python scripts and other utilities (e.g., testing tools) are reusable to a large extent;
- the data base of checked and verified glyphs names and unicode is fully reusable;
- math glyphs (operators, relational symbols, geometrical symbols) are to be created from scratch which seems to be the most laborious part of the job to be done – this does not mean, however, the most difficult part;
- does it make sense to produce TEX math fonts simultaneously with OTF ones (no chance for compatibility anyway)?

The OpenType math fonts project is supported
by TEX Users Groups, in particular, by the Czechoslovak
TEX Users Group CS TUG, the German-speaking TEX Users Group
DANTE e.V., the Polish TEX Users Group GUST,
the Dutch-speaking TEX Users Group NTG,
TUG India, UK-TUG, and – last but not least – TUG.

The effigies of the Font Beasts originate from
http://ascend-to-oblivion.blogspot.com/2010/04/wallpapers-creatures.html

The author of the opening (and closing) photo
Roots of Entanglement is Marek Ryćko

THANK YOU FOR YOUR ATTENTION

and

WELCOME TO NEXT BACHOTEXS