About this presentation

The report proper\(^1\) (mere facts) will be published in one of the forthcoming issues of the DANTE \TeXnische Komödie and, most probably, also in other LUG’s journals.

This presentation, although reporting on what was achieved, is totally different. It aims to bring forth those elements of the e-foundry’s projects which were difficult to place in a mere facts report.

I’ll try to show the achievements and hope that it transpires how much effort went into the project(s) . . .

\(^1\)For the previous report see B. Jackowski et al. in DTK, Heft 3, 2018

A tiny bit of history

Conceived and presented to LUG’s in 2015, the project officially started in 2017, though work began already in 2016.

Externally visible progress was being made until 2018, when it finally transpired that the long overdue revamping of MetaType 1, the e-foundry’s toolset, could not wait.

Work on the successor, Algotype started in 2018. In the midst of that effort Piotr Strzelczyk left the team. Luckily, Marek Ryćko agreed to step in. However, quite some time was lost.

Further loss of time was caused by the pandemic and, on top of if, Boguslaw Jackowski’s hospitalisation for both COVID-19 and a heart surgery.
Math symbol subsets for TG text fonts

The following four slides show samples of the extended repertoire. Over 850 mathematical, geometrical and technical symbols were selected and added.

Please note:
▶ shown fonts were made with Algotype, the new engine,
▶ samples compiled with LuaTeX and OTF fonts,
▶ shown are examples only for regular (top) and bold (bottom) variants,
▶ TG Adventor and TG Pagella were previously (before Algotype) extended and now “revised”,
▶ TG Schola and TG Termes are newly extended.
Backward compatible math style

The following four slides show samples of the ss10 feature—“backward compatible math style”—in action.

Please note:

- shown fonts were made with Algotype, the new engine,
- samples (formulas) compiled with LuaTeX and OTF fonts,
- TG Adventor and TG Pagella — revised (remade) from v. 2.501,
- TG Schola and TG Termes — newly made,
- lines marked ss10-, the default(!) are, hopefully, the better ones, but
- this is not to say that going ss10+ will not produce good results in text mode...

TG Adventor (revised)

| SS10- | f(x)=1/x | (x+1)(x-3) > 0 |
| SS10+ | f(x)=1/x | (x+1)(x-3) > 0 |

TG Pagella (revised)

| SS10- | f(x)=1/x | (x+1)(x-3) > 0 |
| SS10+ | f(x)=1/x | (x+1)(x-3) > 0 |
The anchor mechanism

The following five slides deal with and show samples of placing of accents using the “anchor” mechanism, i.e., the ccmp+mark+mkmk features\(^2\).

Please note:

- all shown fonts now made with Algotype,
- samples were compiled with Lua\TeX{} and OTF fonts,
- feature tables had to be prepared for all fonts and all shapes

---

\(^2\)see B. Jackowski et al. in DTK, Heft 3, 2018
Sample code for placing accents

Source:

L´ or L\char"030C % U+030C (caroncmb, caroncomb)
g, or g\char"0326 % U+0326 (uni0326, commaaccentcomb)
y, or y\char"0323 % U+0323 (uni0323, dotbelowcomb)
TG Termes (new in the pack)

Why Algotype?

From the above it should be obvious that proper tools were required for the work.

In the following slides I’ll try to illustrate the reasons for going from MetaType 1, the old tool chain, to Algotype, the new and shiny workshop.

Old MT1

The scheme of the old font engine

Old MT1 with reverse workflow

The scheme of the old font engine

Interim MT1(?), no reverse workflow

The scheme of the interim font engine

METAPOST font base
METAPOST source(s)
configuration files
PFB file
ENC and MAP files (for dvips)
AFM file
fixed AFM file

METAPOST
configuration files
EPS file
EPS file
...
auxiliary(OTI) file
Python

METAPOST
font base
METAPOST
source(s)
configuration
files
PFB file
TFM file
ENC and MAP
files (for dvips)
AFM file
fixed AFM file
OpenType font file

Algotype: simpler but no reverse workflow

The scheme of the current font engine

input data
(interim data ("frozen" font))
output data
(trendy fonts)

configuration and installation files
METAPOST sources files (splines, kerns, ligatures, font parameters etc.) plus additionally sample feature file and glyph order and alias data base (GOADB) file
OpenType font
PostScript Type 1 font
TrueType font
FontForge's spline font database (SFD)
XML, e.g., TTX's output
unified font object (UFO) format, etc.

information out, feedback in

Information out
The above problem is known since at least 10 years and not fixable by us. How do we disseminate such knowledge? How do we tell users, e.g., of the "backward compatible math style" ss10?

Feedback in
How do we learn about user needs or problems? Do they really need what we are doing? Is, e.g., a monospace font with math symbols for text editors really needed?

Education(?) is needed!
Feedback is needed!

Ten years after

An e-mail dated early March this year by a desperate user:

Missing character in LM Math

I'm having problems with the LM-Math font that I downloaded from your website. It seems that the small italic h is missing, see screenshot:

Could you please fix this?

5By "we" the GUST e-foundry team is meant.
Supporters

Support was received from (in no particular order):
▶ NTG,
▶ CS-TUG,
▶ CG (Context Group),
▶ DANTE e.V.,
▶ TUG,
▶ GUST (non-material),
▶ last but not least — individuals.

Thank you very, very much!

“Closing”?

Declared financial support came to end, but of course

\(closing \neq final\)

To be continued . . .