

Conversion of T_EX fonts into Type 1 format

Szabó Péter

`<pts@inf.bme.hu>`

September 2001

presented at EuroT_EX 2001

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
T _E Xtrace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 1 of 12

Home Page

Go Back

Full Screen

Close

Quit

✂ Motivation ✂

The problem: Acrobat Reader renders most fonts slowly and unreadably ugly on screen. This is because most $\text{T}_{\text{E}}\text{X}$ fonts can be included into PDF files only as high resolution raster (bitmap) images, and Acrobat Reader shows such images slowly and inaccurately.

- ☞ example: nice, with Type 1 font
- ☞ example: ugly, with bitmap font generated by METAFONT

Solution: convert the offending $\text{T}_{\text{E}}\text{X}$ fonts into something that Acrobat Reader can display well, and make `pdftex` embed the converted fonts into the PDF file. The best candidate for this font format is Adobe Type 1.

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
T_EXtrace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 2 of 12

Home Page

Go Back

Full Screen

Close | Quit

✧ Bitmap and vector fonts ✧

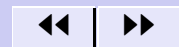
Bitmap fonts:

- ☞ easy and fast to render
- ☞ device specific
- ☞ resolution specific
- ☞ generated, not post-processable
- ☞ print nicely iff DPI is large
- ☞ separate, small screen version
- ☞ examples: \TeX .pk, \TeX .gf, X11 BDF

Vector outline fonts:

- ☞ need complicated rendering
- ☞ device independent
- ☞ scalable, transformable
- ☞ editable
- ☞ always print nicely
- ☞ hard to read on screen
- ☞ examples: Type 1, OpenType, TrueType, \TeX .mf

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
\TeX trace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 3 of 12

Home Page

Go Back

Full Screen

Close | Quit

✂ Format comparison ✂

Type 1 files (.pfa, .pfb):

- ↻ vector outline format
- ↻ work with `tex` and `pdftex`
- ↻ appeared with PostScript (1985)
- ↻ supported by most OSs and DTP software
- ↻ 3rd order (cubic) Bézier curves
- ↻ filled regions only
- ↻ embedded into PDF verbatim, the PDF-viewer renders them
- ↻ rendered nicely in Acrobat

METAFONT source files (.mf):

- ↻ vector outline format
- ↻ work with `tex` and `pdftex`
- ↻ appeared with T_EX (1983)
- ↻ only METAFONT understands them (not even METAPOST)
- ↻ 3rd order (cubic) Bézier curves
- ↻ filled and stroked regions etc.
- ↻ already rendered before embedded to PDF
- ↻ displayed ugly in Acrobat

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
T _E Xtrace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 4 of 12

Home Page

Go Back

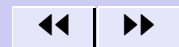
Full Screen

Close | Quit

✧ Possible solutions ✧

- ✧ avoid conversion, use existing fonts and glue them together with *virtual fonts* (.vf). Example: AE (CM→EC). Drawback: missing glyphs, missing design sizes.
- ✧ *design new* fonts parallelly in both formats. Example: METATYPE1 (at EuroT_EX 2001). Drawback: cannot convert existing fonts.
- ✧ *modify* METAFONT to output vector fonts. Drawback: too much effort.
- ✧ *post-process* METAPOST's output. Example: MetaFog. Drawbacks: METAPOST can't understand all .mf files, Type 1 requires special contours without intersections etc., needs human intervention.
- ✧ *trace* METAFONT's bitmap output. Example: T_EXtrace. Drawback: minor quality loss.
- ✧ a glyphwise *mixture* of these

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
T _E Xtrace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



✧ T_EXtrace ✧

T_EXtrace is a collection of scripts for UNIX that convert *any* T_EX font into a Type 1 outline font immediately suitable for use with T_EX. The documents using these fonts cannot be visually distinguished from those using the originals, moreover PDF documents show up quickly and nicely in Acrobat Reader.



<http://textrace.sf.net/>

The operation of T_EXtrace:

1. calls dvips and gs to render all the 256 glyphs in high (≥ 7000 DPI) resolution
2. calls autotrace to convert each bitmap to outlines
3. makes syntactical corrections, positions glyphs to their origin and assembles them to a Type 1 .pfb file

Motivation

Bitmap and vector ...

Format comparison

Possible solutions

T_EXtrace

Benefits

Screen shot

Problems remaining

Quality comparison

The ideal solution

Conclusion



Page 6 of 12

Home Page

Go Back

Full Screen

Close

Quit

✧ Benefits ✧

- ☞ accepts *any* \TeX font (.mf, .pk, Type 1, TrueType etc.)
- ☞ operates completely *automatically*, finishes one file in ≈ 20 minutes without asking any questions
- ☞ is *free*, licensed under the GNU GPL
- ☞ produces *portable Type 1 output*, which avoid bugs in many third-party software

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
\TeX trace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 7 of 12

Home Page

Go Back

Full Screen

Close | Quit

✧ Screen shot ✧

```

> autotrace tmp/tmp_char.pbm -filter-iterations 9 -background-color FFFFFFFF
> gs -d_n=252 -dNODISPLAY -q tmp/gstderr.ps trace2.ps
> autotrace tmp/tmp_char.pbm -filter-iterations 9 -background-color FFFFFFFF
> gs -d_n=253 -dNODISPLAY -q tmp/gstderr.ps trace2.ps
> autotrace tmp/tmp_char.pbm -filter-iterations 9 -background-color FFFFFFFF
> gs -d_n=254 -dNODISPLAY -q tmp/gstderr.ps trace2.ps
> autotrace tmp/tmp_char.pbm -filter-iterations 9 -background-color FFFFFFFF
> gs -d_n=255 -dNODISPLAY -q tmp/gstderr.ps trace2.ps
> perl ./t1d2gsx.pl --256
255 CharBBox entries.
> perl -x typelfix.pl tmp/tmp_gen1.gsx pfb: gcti1000.pfb --pack --dump-spaces=no
--debug-warnings --chk-insize=no
infile tmp/tmp_gen1.gsx
insize 370868
will preserve comments
ok internal interp
ok internal poss
packed 257 chars
writing PFB
outfile gcti1000.pfb
outside 101784
check: OK /tmp/_t1f_2936_1: /TeX-ecti1000
done
~$ _

```

Motivation

Bitmap and vector ...

Format comparison

Possible solutions

TeXtrace

Benefits

Screen shot

Problems remaining

Quality comparison

The ideal solution

Conclusion



Page 8 of 12

Home Page

Go Back

Full Screen

Close

Quit

✧ Problems remaining ✧

- ☞ huge font files (3.15 ×)
- ☞ no hinting in output
- ☞ AutoTrace mis-recongizes some corners
- ☞ METAFONT fails for some fonts
- ☞ limited portability (needs UNIX, teTeX, bash, perl, gs)
- ☞ written for experts
- ☞ no Unicode support, limited to 256 characters
- ☞ doesn't convert metrics (but original .tfm is OK for TeX)

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
TeXtrace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 9 of 12

Home Page

Go Back

Full Screen

Close

Quit

✧ Quality comparison ✧

Original METAFONT fonts:

- ☞ printed perfectly (≥ 600 DPI)
- ☞ displayed nicely in `xdvi`
- ☞ displayed awful in `gs`
- ☞ displayed tolerably in `gv -an`
- ☞ displayed awful in `acroread`

Fonts converted by $\text{T}_{\text{E}}\text{X}$ trace:

- ☞ printed nicely (≥ 300 DPI)
- ☞ displayed nicely in `xdvi`
- ☞ displayed awful in `gs`
- ☞ displayed nicely in `gv -an`
- ☞ displayed nicely in `acroread`

Reasons:

- ☞ Acrobat Reader hates bitmap fonts
- ☞ screen resolution (≤ 100 DPI) is too small, but antialiasing solves the problem
- ☞ hinting without antialiasing is not enough for Type 1 or METAFONT.

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
$\text{T}_{\text{E}}\text{X}$ trace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 10 of 12

Home Page

Go Back

Full Screen

Close | Quit

✂ The ideal solution ✂

1. *merge* METAFONT and METAPOST
 - ☞ allow both bitmap a PostScript output
 - ☞ allow elliptical pens, METAFONT pictures etc.
 - ☞ support all METAFONT language dirty tricks
 2. *post-process* the PostScript output (better than MetaFog)
 - ☞ convert strokes (etc.) to fills
 - ☞ remove contour overlaps
 - ☞ reorganize touching contours
 - ☞ do other small modifications for Type 1 compliance
 3. *guess* most *hinting* information automatically (really hard)
 4. implement an *effective human interface* to modify hinting
 5. convert metrics, kerning, ligatures, support Unicode
- Of course, none of the steps have been implemented yet.

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
TeXtrace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 11 of 12

Home Page

Go Back

Full Screen

Close | Quit

✧ Conclusion ✧

- ☞ T_EXtrace already works, is automatic and free
- ☞ T_EXtrace can be used by anyone who understands the problem, but can't solve it by him/herself
- ☞ professional quality requires a font expert (preferably a font designer) much time.
- ☞ the ideal, almost automatic solution requires too much resources to implement
- ☞ Not even mayor T_EX fonts are expected to be freely available in high quality Type 1 for years. Until this happens, T_EXtrace is the best alternative.

Motivation
Bitmap and vector ...
Format comparison
Possible solutions
T _E Xtrace
Benefits
Screen shot
Problems remaining
Quality comparison
The ideal solution
Conclusion



Page 12 of 12

Home Page

Go Back

Full Screen

Close | Quit