Fonts, Fonts, and more Fonts!

EuroTeX 2001

Tom Kacvinsky

2001/09/25

Overview

- Font formats
- OS and non-TeX appliction support
- Which TeX applications use which font formats
- Caveats

Font Formats

- Type 0 (Composite)
- Type 1 and MM extensions
- Type 2
- Type 3
- Types 9, 10, 11, 32 (CIDkeyed)
- Type 14 (Chameleon)
- Type 42
- True Type
- OpenType

Type 1 and MM fonts

- Uses cubic Bezier curves to describe glyph outlines.
- Depends on PS interpreter or font rasterizer to ra sterizerthe font.
- “Private” data is encrypted, but if font is decrypted, it is mostly ASCII (there is some binary data -- the glyph outlines and subroutines).

Type 1 and MM fonts (cont.)

- Multiple Master (MM) fonts are based on Type 1, but allow for design axes (optical size and weight, for instance). The “final” font that is used is an instance of the font where the contribution of each of the design axes is fixed.
- For more information on Type 1 and MM fonts, see Adobe’s documentation available at http://partners.adobe.com.

Type 2 (CFF) fonts

- Type 1 on steroids.
- Binary data, rather than ASCII data.
- Not encrypted.
- Table based (data is looked up in a table using indices).
- Operators used for creating glyph outlines have been changed for realization of smaller glyph definitions.

Type 2 (CFF) fonts (cont.)

- Requires font raterizer or Level 3 interpreter for rasterization.
- Used in OpenType fonts.
- To a PS interpreter, treated the same as a Type 1 font, except the FontType key's value is 2, not 1.

Type 3 fonts

- In general, require a PS interpreter for rasterization.
- More flexible than Type 1 fonts, as they use the full power of a PS interpreter; however, certain rules must be obeyed.
- Can be bitmap, or outline, or a combination of both bitmap and outline.

Type 14 fonts

- Undocumented format proprietary to Adobe.

Type 42 fonts

- PS wrapper around a True Type font.
- See discussion of True Type fonts for more information.
- Type 42 fonts require a PS interpreter which understands the Type 42 format (and hence, has a True Type rasterizer). Most level 2 and all level 3 interpreters understand Type 42 fonts.

Type 0 fonts

- A composite font -- a font composed of base fonts (Types 1, 2, 3, 14, and 42), other Type 0 fonts, and CID keyed fonts.
- Organized hierarchically. The top level font is called the root font. Fonts below the root font are called descendant fonts, and the Type 0 font above a descendant font is called a parent font.

Type 0 fonts (cont.)

- Special rules are applied when the PS operator show and its variants are used when the current font is a Type 0 font.
- Composite fonts can use other composite fonts for descendent fonts, up to 5 levels deep.
- One application of composite fonts is synthesis of caps and small caps fonts.

CID keyed fonts

- CID stands for Character Identifier
- Type 9 fonts use Type 1 glyph outlines, Type 10 fonts build glyphs in a fashion similar to how Type 3 fonts build glyphs, Type 11 fonts uses Type 42 (True Type) glyph outlines, and Type 32 uses cached glyph information.
- Type 9 and Type 11 CID keyed fonts are the most common CID keyed fonts.

CID keyed fonts (cont.)

- CID keyed fonts are a combination of ASCII data and tables whose data is referenced by offsets derived from the CID.
- Character codes are mapped to CIDs by a specialPS Resource called a Cmap (character map).

True Type fonts

- Quadratic Bezier curves are used rather than cubic Bezier curves.
- Jointly developed by Apple and Microsoft (!) as a competitor to Type 1 fonts.
- Table based.
- Binary data.
- Glyph data is referenced by offsets derived from a glyph index.

True Type fonts (cont.)

- Glyph indices are obtained from a cmap table in the font. That is, the cmap table is used to map character codes to glyph indices.
- A Type 42 font is PS wrapper around the data of the True Type font. The data is stored in an array called sfnts. The elements of the sfnts array are hexadecimal encoded strings containing a portion of the TT data

True Type fonts (cont.)

- There are patents associated with interpreting True Type fonts. Apple holds the patents.

OpenType fonts

- Format jointly developed by Adobe and Microsoft.
- Two types of glyph data: CFF and True Type.
- Like True Type fonts -- table based, binary data.
- Extra layout information is stored in tables in the font. New layout tables can be designed and included in the

OpenType fonts (cont.)

- May require (1) a typesetting system that resolves OT layout tables to standard OS/interpreter layout routines, and (2) a True Type/CFF rasterizer. InDesign is an application that can handle OT fonts.

OS and non-TeX application support

- MacOS 8.5 -- 9.x
- MacOS X
- Windows
- UNIX/Linux
- Non-TeX applications

MacOS 8.5 -- 9.x

- OS itself handles True Type fonts.
- ATM is required for CID keyed and Type 1/MM support.
- True Type fonts are stored in a screen suicase, with TT data stored in the sfnt resource.
- Type 1/MM data is split between a screen suitcase and an LWFN file.

MacOS 8.5 -- 9.x (cont.)

- Type 9 CID keyed fonts are stored in one of two formats: a data fork file that contains the CID keyed font itself, with ATM providing the CMap data, or in a screen suitcase in which the CID keyed font is stored in a special table. In the latter case, ATM is still needed for rasterization needs.
- Not sure about other CID keyed font types

MacOS X

- True Type support built in.
- Haven't experimented, but apparently, Type 1 support is also built in. Perhaps this is inherited from NeXTStep?
- Ships with some high quality CJK fonts. Are they OpenType? According to press releases, they are. I haven't confirmed this.

MacOS X (cont.)

- OS X gets rid of the screen suitcase. The data previously stored in the screen suitcase (the resource fork) is now stored in a special format in what was previously called the data fork (and what Windows and UNIX users have always called a file).

Windows

- True Type is built into Windows 9X. OpenType support is built in, but only for True Type glyph data OT fonts.
- OpenType support for both CFF and TT fonts is built into Windows 2000 and its variants (Microsoft licensed code from Adobe for the CFF rasterizer).
- W2K also has built in support for Type 1 fonts.

Windows (cont.)

- ATM is necessary for Type 1 support on Windows 9X and NT.
- Be careful! Windows NT will convert Type 1 fonts to TT format unless ATM is installed.
- Not sure about CID keyed support for Windows.

Linux/UNIX

- Type 1 rasterizer built into most modern UNIX X window font servers..
- The same is true for TT support.
- On Linux, one can get TT support using xfstt. The TT rasterizer that is used is FreeType 1.x.
- There are versions of font servers for XFree 86 that use FreeType 2, so that they support Type 1 fonts.

Linux/UNIX (cont.)

- FreeType is one font rasterizer for UNIX/Linux (and other systems, as well). It supports OT/TT/Type 1/CFF/CID/MM fonts.
- T1Lib is another font rasterizer for UNIX/Linux, but it only supports Type 1 fonts (not sure about MM support).

Non-TeX applications

- InDesign was the first application to ship with OT font support.
- Photoshop was Adobe's next application to ship with OT support.
- Illustrator is the next application scheduled to get OT support.
- Each uses CoolType, Adobes font engine.
- Most applications get their font support from the OS.

TeX applications

- dvips/odvips
- pdftex
- dvipdfm
- xdvi

dvips/odvips

- dvips can use Type 1, Type 3 and Type 42 fonts. Subsetting is supported for Type 1 fonts only. Other font formats (3, 42) need to have the font fully embedded.
- odvips can use what dvips uses, plus CID keyed fonts.

pdftex

- pdftex supports Type 1, TT, and OpenType fonts (either TT or CFF glyph outlines).
- pdftex can subset TT and Type 1 fonts.
- For OpenType fonts with TT glyph data, pdftex can subset glyph data.
- For OpenType/CFF fonts, the CFF data is placed in the PDF file unaltered. No subsetting occurs.

pdftex (cont.)

- Type 3 fonts are supported, but only by conversion of PK files to bitmap Type 3 format.

dvipdfm

- Supports Type 1 fonts. I do not know if it supports TT fonts.

xdvi

- There is a version of xdvi that uses T1Lib for Type 1 font support. If FreeType were used instead, TT font support could also be added.

Caveats

- CID keyed font support (or large CJK TT font) for TeX (pdftex, tex and other variants, but not Omega) applications is difficult. The general idea is to split the font into several smaller Type 1 font (think of ttf2pt1) and then to use the smaller Type 1 files.

Caveats (cont.)

- MM support is also available, but one has to create instances before hand. This is usually done by using Eddie Kohler's MM tools.
- Encoding issues -- most OSs have some character codes that the cannot display. So one has to be careful in what encodings they use.

Caveats (cont.)

- Be careful with subsetting!

The Good, The Bad and the Ugly

- The Good: OpenType
- The Bad: Apple's Patents
- The Ugly: bitmap Type 3 fonts