

ABOUT THE 9.16 DISTRIBUTION

RICHARD KOCH

1. INTRODUCTION

(Note: The 9.16 sources have an error which breaks compiling on Snow Leopard and PPC. To fix this error, open `trio/triop.h`, got to line 407, and comment out the entire line, which reads `TRIO_CONST trio_pointer_t trio_get_argument TRIO_PROTO((trio_pointer_t ref));`);

The Ghostscript 9.16 install package should work on Leopard and higher systems, for both Intel and PPC. This standalone package will eventually become part of MacTeX-2015.

Ghostscript 9.16 was released on March 19, 2015. The source code comes from <http://www.ghostscript.com>. (Ghostscript fonts used to be provided separately at <http://sourceforge.net/projects/gs-fonts/>, but are now part of the source distribution.)

Recent versions of Ghostscript contained compressed copies of the fonts inside the binary, but this year we compiled with a flag making the fonts available separately in `/usr/local/share/ghostscript/9.16/Resource/Fonts`.

I'd like users to test this release before it is used in MacTeX-2015. My web page also contains earlier versions of Ghostscript. Installing one of these over Ghostscript-9.16 works fine and returns to the earlier version in case of trouble with this version.

2. WHAT IS INSTALLED

Ghostscript binaries are installed in `/usr/local/bin`. Ghostscript support files are installed in `/usr/local/share/ghostscript/9.16` and man pages are placed in `/usr/local/share/man`.

3. THE BINARIES

The install package contains six binaries, `gs-X11`, `gs-noX11`, `gs-X11-64Bit`, `gs-noX11-64Bit`, `gs-X11-Yosemite`, and `gs-noX11-Yosemite`. The first two are universal binaries with 32 bit

Date: April 18, 2015.

code for Intel and 32 bit code for PPC. The second two are universal binaries with Intel 32 bit and 64 bit code. The final two binaries are 64 bit binaries compiled on Yosemite.

On Leopard, the gs binaries are installed. On Snow Leopard through Mavericks, the gs-64Bit binaries are installed. On Yosemite, the Yosemite binaries are installed. Binaries are then renamed to gs-X11 and gs-noX11. Finally the installer defines a symbolic link gs pointing to either gs-X11 or gs-noX11 depending on whether X11 is installed.

At the end of installation, all machines will contain gs-X11 and gs-noX11 and a symbolic link gs to the appropriate binary. Notice that a user who does not have X11 but later installs it can reset the gs link and get a ghostscript with X11 support.

4. PROVIDING ACCESS TO TEX FONT FILES

In 2012, David Arnold reported that MetaPost can create illustrations containing computer modern fonts. If the output is later processed in TeX + DVI mode, these fonts need to be accessed by Ghostscript. Previous versions of Ghostscript in MacTeX substituted default postscript fonts for these missing computer modern fonts.

Bruno Voisin studied this problem extensively and provided a natural and elegant solution to the problem. The solution involves adding two files named “Fontmap” and “Fontmap.cmr” to /usr/local/share/ghostscript/9.16/Resource/Init and adding symbolic links to the pfb files in TeX to /usr/local/share/ghostscript/9.16/Resource/Font. For example, a typical symbolic link is

```
blex.pfb->/Library/TeX/Root/texmf-dist/fonts/type1/public/belleek/blex.pfb
```

Note that this link passes through the /Library/TeX/Root link provided by the TeX Dist data structure, and thus causes Ghostscript to use fonts in the currently active TeX distribution. This method isn’t entirely foolproof because TeX Live might change the location of important pfb files. But the Ghostscript package in MacTeX will be adjusted for the latest release of TeX Live, so the links should satisfy most users.

5. PROVIDING ACCESS TO JAPANESE FONTS

In 2015, KUROKI Yusuke and Munehiro Yamamoto contacted us asking for modifications to support Japanese fonts in Ghostscript. Ghostscript 9.16 has these modifications. Bruno Voison was helpful with this project. The changes for Japanese fonts are:

- By default, recent versions of Ghostscript compile fonts and initialization files into the executable as a “rom” file system. Since additional fonts and initialization files are required for Japan, we compile ghostscript with a flag to turn off this feature:

```
./configure --disable-compile-inits
make
sudo make install
```

- The empty “cidfmap” file in `/usr/local/share/ghostscript/9.16/Resource/Init` is replaced by a file containing

```
/Ryumin-Light /HiraMinProN-W3 ;
/GothicBBB-Medium /HiraKakuProN-W3 ;
/FutoMinA101-Bold /HiraMinProN-W6 ;
/FutoGoB101-Bold /HiraKakuProN-W6 ;
/Jun101-Light /HiraMaruProN-W4 ;
```

- Sixteen fonts similar to HiraKakuPro-W3 are added to `/usr/local/share/ghostscript/9.16/Resource/CIDFont`.
- A large number, 1428, of small font files are placed in `/usr/local/share/ghostscript/9.16/Resource/Font`, each similar to

```
%%!PS-Adobe-3.0 Resource-Font
%% DocumentNeededResources: 78-EUC-H (CMap)
%% IncludeResource: 78-EUC-H (CMap)
%% BeginResource: Font (FutoGoB101-Bold-78-EUC-H)
(FutoGoB101-Bold-78-EUC-H)
(78-EUC-H) /CMap findresource
[(FutoGoB101-Bold) /CIDFont findresource]
composefont
pop
%% EndResource
%% EOF
```

- The full `Fontmap.cmr` from the Ghostscript sources is installed in `/usr/local/share/ghostscript/9.16/Resource/Init`, which already contains a similar file `Fontmap.(GS)` due to the first item in this list. This was also done for the Arnold-Voisin modifications described in the previous section. Checking shows that the `Fontmap.cmr` files for the two modifications are the same.
- The file “Fontmap” in this `Init` folder is replaced by a file containing

```
(Fontmap.GS) .runlibfile
(Fontmap.cmr) .runlibfile
```