So what is Defoma anyway?

Angus Lees <gus@debian.org>

Debian Miniconf – 12*th January* 2004

The Problem

UNIX has no central system for managing fonts, so each project has developed its own:

- X11
- Fontconfig (GNOME, KDE)
- TeX
- Ghostscript

“But it works for me”

Apparently some people don’t speak American and *need* to install new fonts before they can do anything at all.

(The principle author, Yasuhiro Take, and most contributors to Defoma are Japanese.)

. . . and It's Getting Worse

Two words:

Desktop Revolution

Debian Needs

- Has to cope with all font formats and all applications
- Can't require (non-trivial) changes to existing code
- Easy for developers to use

Defoma in a Nutshell

Two parts to Defoma

1. Registering fonts and their attributes (“hints”)

2. Invoking *application scripts* when new fonts are added or removed

   Each application script is expected to generate the appropriate configuration for that application.

Fonts 101

Formats

Vector:

- PostScript Type1 (`.pfa`, `.pfb`)
- TrueType
- CID
- METAFONT (`.mf` – TeX)

Bitmap:

- PostScript Type3
- `.pk`, `.bdf`, `.pcf`

Some Font Attributes

Foundry Who made the font (Adobe, URW, Bitstream)

Family Name of a related group of fonts (Helvetica, Times)

Style Variations within a family (serif, sans serif, typewriter, SMALL CAPS)

Weight “Boldness” (light, **bold**, **black**)

Width “Stretchedness” (condensed/expanded)

Spacing Variable or fixed

Slant (Upright, *oblique*, *italic*)

Encoding Which index is what glyph (StandardEncoding)

Design Size (9pt, 12pt)

Defoma Fonts

```
category type1
begin /usr/share/fonts/type1/gsfonts/a010013l.pfb
  AFM = /usr/share/fonts/type1/gsfonts/a010013l.afm
  FontName = URWGothicL-Book
  Charset = ISO8859-1
  Family = URW_Gothic_L
  GeneralFamily = Gothic
  Weight = Light
  Width = Variable
  Shape = NoSerif Upright
  Priority = 20
  X-FontName = -urw-urw_gothic_l-book-r-normal--0-0-0-0-p-0-
iso8859-1
  Alias = AvantGarde-Book
  Foundry = URW
end
```

```
category truetype
begin /usr/share/fonts/truetype/ttf-bitstream-vera/Vera.ttf
  Family = Bitstream-Vera-Sans
  FontName = Bitstream-Vera-Sans
  Encoding = Unicode
  Location = English
  Charset = ISO8859-1 ISO8859-9 ISO8859-15 ISO10646-1
  UniCharset = ISO8859-1 ISO8859-9 ISO8859-15
  GeneralFamily = SansSerif
  Weight = Medium
  Width = Variable
  Shape = NoSerif Upright
  Foundry = Bitstream
  Priority = 20
end
```

defoma-hints

Outputs a suitable hints file to STDOUT.

```
defoma-hints [-c] [--no-question] category fonts...

defoma-hints type1 /path/*.pfb > myfonts.hints
```

Packaging Details

```
FILE=/etc/defoma/hints/pkg.hints

install hints file into $FILE

postinst:
 test $1 = configure && \
   defoma-font reregister-all $FILE

prerm:
 test $1 = remove && \
   defoma-font purge-all $FILE
```

. . . or use **dh_installdefoma**(1)

dfontmgr

defoma-hints(1) and **defoma-font**(1) are good for adding large numbers of fonts.

Users can also install/remove/query fonts using the **dfontmgr**(1) GUI tool.

psfontmgr

defoma-psfont-installer(1) registers printer fonts so Defoma applications know about them. The easiest way to use it is to feed it a suitable PPD file.

`psfontmgr` also maintains a category of available PostScript fonts and their most similar X11 font—useful for applications that need to provide “print previews”.

Defoma Applications

A Defoma application script is a Perl file that has certain functions invoked when fonts are added or removed.

The principle tasks of the *app script* is to:–

1. Convert the fonts into whatever naming scheme the application uses (including aliases)

2. Choose the highest priority font providing each name, probably through a defoma “ID cache”

3. Configure the application to use those fonts

Substituted Fonts

Many applications assume certain fonts exist. For example, ghostscript needs the 14 base PostScript fonts, fontconfig assumes “serif”, “sans-serif” and “monospace” exists.

Using a “subst cache”, Defoma can choose the “best” font to provide a particular ID.

In practice, most application scripts would treat substituted fonts the same as an aliased font.

Substitution Rule Example

```
# must be on one line - wrapped here for clarity
monospace
   --Width,* Fixed
   --GeneralFamily,2 Typewriter
   --Shape Upright
   --Weight Medium
```

“**Must** be Fixed width, it’s more important to be a Typewriter font and less important to be Upright and Medium weight”

Example: fontconfig

```
@ACCEPT_CATEGORIES = qw(type1 truetype cid);

package fontconfig;

sub main {
  my $cmd = shift;
  if ($cmd eq ’init’) {
    init();
  } elsif ($cmd eq ’register’) {
    return register(@_);
  } elsif ($cmd eq ’unregister’) {
    return unregister(@_);
  } elsif ($cmd eq ’do-install-real’) {
    return do_install_real(@_);
  } elsif ($cmd eq ’do-remove-real’) {
    return do_remove_real(@_);
  } elsif ($cmd eq ’term’) {
    return term(@_);
  }
  0;
}

*truetype  = \&main;
*type1     = \&main;
*cid       = \&main;
```

```
sub init {
  $Id ||= defoma_id_open_cache() or return 1;
  $Sb ||= defoma_subst_open(rulename => ’fontconfig’,
                            threshold => 70,
                            idobject => $Id) or return 1;

  return 0;
}
```

```perl
sub register {
  my $font = shift;
  my $hints = parse_hints_start(@_);

  return 1 unless $hints->{FontName};

  my $priority = $hints->{Priority} || 20;
  my ($fontname) = split / +/, $hints->{FontName};
  my @alias = split / +/, $hints->{Alias} if $hints->{Alias};

  defoma_id_register($Id, type => 'real', font => $font,
       id => $fontname, priority => $priority,
       hints => join(' ', @_));

  foreach my $alias (@alias) {
    defoma_id_register($Id, type => 'alias', font => $font,
         id => $alias, priority => $priority,
         origin => $fontname);
  }

  defoma_subst_register($Sb, $font, $fontname);

  return 0;
}
```

```perl
sub unregister {
  my $font = shift;
  defoma_subst_unregister($Sb, $font);
  defoma_id_unregister($Id, type => ’alias’, font => $font);
  defoma_id_unregister($Id, type => ’real’, font => $font);
  return 0;
}
```

Other Functions

`do_install_real()` symlinks the font below `/var/lib/defoma/fontconfig.d/`

`do_remove_real()` removes that symlink

`term()` writes a suitable `fonts.conf` which:–

- points fontconfig to `/var/lib/defoma/fontconfig.d/`
- generates `<alias>` entries by querying the ID cache for registered font aliases and any substituted fonts
- runs `fc-cache` to rebuild the fontconfig cache

More Information

```
aptitude install defoma-doc
```

Read `developers.html`, followed by `defoma-script.html` if writing an app script.