

Managing Gnome Fonts

BETTER FONTS AHEAD

Linux's text display still suffers from legacy issues, but it is well on the way to recovering. This article explains how Fontconfig, X, and Gnome work together for a brighter future. **BY OLIVER FROMMEL**

Despite the amazing progress Linux has made over the last few years, many users still complain of ugly fonts on the Linux desktop. The reasons for this are not always easy to find, as many software components combine to put what a user perceives as letters on the computer screen.

A variety of different programs handle different tasks in different scenarios. For example, the software components responsible for displaying text in a text-based console are completely different

from the ones used by the X system. And even X has many different approaches to handling fonts.

For example, applications can handle fonts themselves, from parsing the font file to actually rendering the characters. Other programs use the legacy (and trusted) X library; newer applications use Gnome libraries. This explains why programs running side by side on a Linux desktop display fonts in a completely different way. And the fact that font support is under active development, with the focus on internationalization, makes the situation even more complicated.

The X Legacy

The X Window system normally handles font management and rendering on the desktop, with KDE and Gnome building on this foundation. The current XFree86 and the alternative X.org can handle most common font formats, as they use the Fontconfig system by the FreeDesktop project [1]. Fontconfig parses font files and gives the X Window system rasterized characters if necessary. It uses

FreeType [2] to render the ubiquitous TrueType fonts (see the box titled "Rendering and Rasterizing"). The Xft-Font library simply passes this data on, whereas it was completely in charge of font rendering before Fontconfig was introduced.

Gnome mainly uses these components to handle fonts in applications. It also provides the Pango library, which mainly handles the layout of text blocks in Unicode character sets.

The introduction of Fontconfig solved a lot of font-related problems in Linux. Applications that use the library benefit from being able to use any fonts installed on a system. For example, the current version of Gimp can finally use TrueType fonts directly after you install Fontconfig on your system. And Mozilla also renders fonts better now that the developers have started to rely on Fontconfig.

Setting Up Fontconfig

Gnome programs that use the libraries I just mentioned have an easier job of font handling by using a standardized mech-



Figure 1: The Font Preferences dialog box in Gnome configures standard fonts.

anism to access any fonts you have installed.

Fontconfig also has a lot of advantages for users and administrators. The configuration files reside in `/etc/fonts`. The XML formatted `fonts.conf` file defines the paths to the font files and common alias names, such as *Sans* for a variety of serif-less fonts. Fonts typically reside in the directories `/usr/share/fonts` and `/usr/X11R6/lib/X11/fonts`. Listing 1 shows you the configuration file section that handles this.

The last entry adds the `.fonts` directory to each user's home directory. You can store font files here to make them available to any applications that use Fontconfig.

To remove the need to list fonts for each application, and to crawl the directories, Fontconfig stores font information in cache files. The following would install a new Truetype font called *ACID.TTF* in the private font directory:

```
cd
cp ACID.TTF .fonts
cd .fonts
fc-cache
```

Running the `fc-cache` command against the font directory updates the cache file in the current directory, assuming that X is running. You can now use the *Acid-Bath* font in any Gnome application.

Configuring Fonts

The Gnome desktop has a program for configuring standard fonts; users with Ubuntu will find this tool in *Computer* | *Desktop Settings* | *Font*, for example. This is where you store fonts for applications, the desktop, window titles, and terminal windows. The window also has a function for setting the font display quality, for example *Best Form*.

The *Details* button takes you to another window with more granular

Listing 1: `/etc/fonts/fonts.conf`

```
01 <dir>/usr/share/fonts</dir>
02 <dir>/usr/X11R6/lib/X11/
   fonts/Type1</dir>
03 <dir>/usr/local/share/
   fonts</dir>
04 <dir>~/.fonts</dir>
```

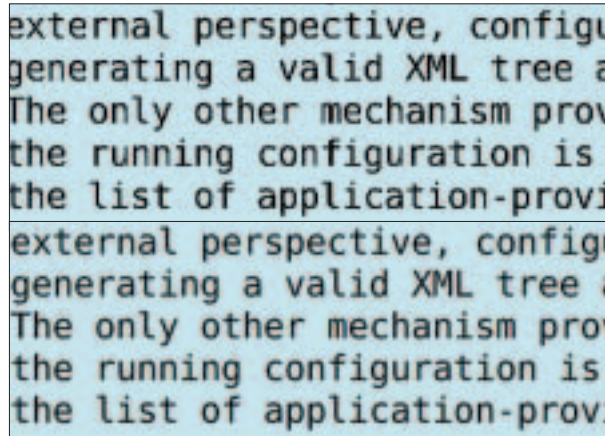


Figure 2: Gnome fonts without (top) and with anti-aliasing.

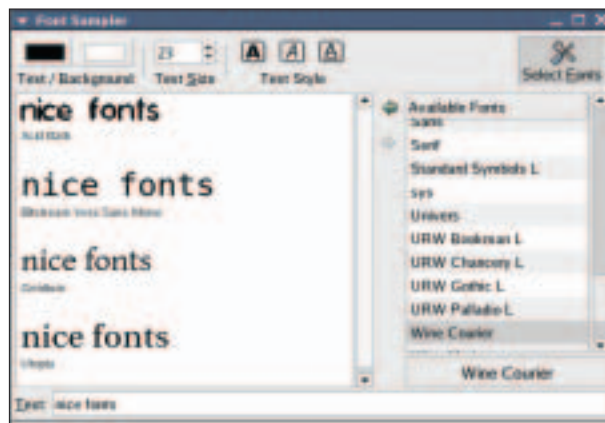


Figure 3: The Gnome Font Sampler shows user-definable sample text in the selected Gnome fonts.

controls, such as anti-aliasing. Figure 2 shows you the difference.

Browsing Font Samples

Gnome does not have a program for listing fonts, but Nautilus will do the trick, if you type `fonts:` for *File* | *Open location*. However, this only shows the system fonts; it does not take the `~/.fonts`

Rendering and Rasterizing

The conversion of data to a format suitable for displaying on screen is often referred to as *rendering* – no matter which application it concerns. For example, web browsers render HTML source code by displaying a page on screen based on that code. Graphics programs use a 3D scene description to render an image. When font programs render fonts, they refer to the mathematical description of a character to extrapolate the individual dots for the character, which they then display on screen. As the dots that make up a character are organized in a table or raster, this is also referred to as rasterizing

directory into consideration. Also, the program does not give you a font sample when you double click a font. Within the Nautilus window, you can right click on a font to specify the *Application font* for Gnome.

The Gnome Font Sampler [4] gives a visual impression of the installed fonts. It parses all the installed fonts and shows a sample of each. You will need the Gnome developer libraries, including Pango, Freetype, and many others to build the program. Start by configuring the source code using the `./configure` command, and then build the program by typing `make`.

If you do not want to install the Gnome Font Sampler, you can simply run it from the build directly, `./src/gfontssampler`, after completing the build.

Or you can type `make install` to install the program in your system directories. The Font Sampler quickly gives you a visual overview of the fonts you have installed.

Variety is the Spice of Life

The current font architecture gives Gnome a powerful system for high-quality fonts in various common formats and all languages.

By using this system, applications could achieve a uniform look and feel, but it is probably too much to expect every single free software author to re-write their source code for purely cosmetic reasons. ■

INFO

- [1] Fontconfig: <http://www.fontconfig.org>
- [2] Freetype: <http://www.freetype.org>
- [3] Pango: <http://www.pango.org>
- [4] Gnome Font Sampler: <http://linuxadvocate.org/projects/gfontssampler>