An up-to-date look at free software and its makers

PROJECTS ON THE MOVE

Free software covers such a diverse range of utilities, applications, and other assorted projects that it can be hard to find the perfect tool. We pick the best of the bunch. This month we cover Yast2 for Debian, KDE 4, Bluetooth headsets, and the Debian project. **BY MARTIN LOSCHWITZ**

Novell GPL'd the Yast2 configuration tool shortly after acquiring Suse. The Yast2 sources were freely accessible prior to this change, but the license restricted use and distribution of the program. These license restrictions prevented legal full downloads of the Suse media for a long time.

Yast4Debian

Novell has stated that it intends to make Yast2 a universal standard. According to Novell, distributors should modify the Yast source code to match their own products. The first major project to tackle this task was Debian. The Debian distribution does not have its own central configuration tool, and creating a comprehensive tool like Yast2 from scratch would have involved a lot of work. The Yast4Debian project is currently porting the Yast2 version supplied with Suse 9.1, although an upgrade to the Suse 9.2 version is imminent.

Yast2 comprises the Libyutil library, a program core, and a number of additional modules. The core is simply an interpreter for Yast2's own programming language, YCP, which combines various elements of other common scripting languages and is used to create new modules. Both the Yast2 libraries and the core were written in C++ and pre-configured with autotools to support building on any Linux system.

Porting the modules will be the biggest challenge. User interfaces such as the Yast2 Ncurses text interface and the Yast2 Qt GUI can be built on other Linux systems without any trouble. Unfortunately, porting the other components is far less trivial. There is even a module for Yast2 icons, and there are many interdependencies between modules that do not appear to make much sense. For example, to compile the partitioning module, you need the Autoyast module. Autoyast adds other dependencies, such as the X11 configuration module, which in turn requires the X11 mouse and X11 keyboard modules. Also, Autoyast depends on the Yast Online Update module and the Bootloader module.

Under the Hood

The task of modifying the highly Suse-specific infrastructure to match the tar-

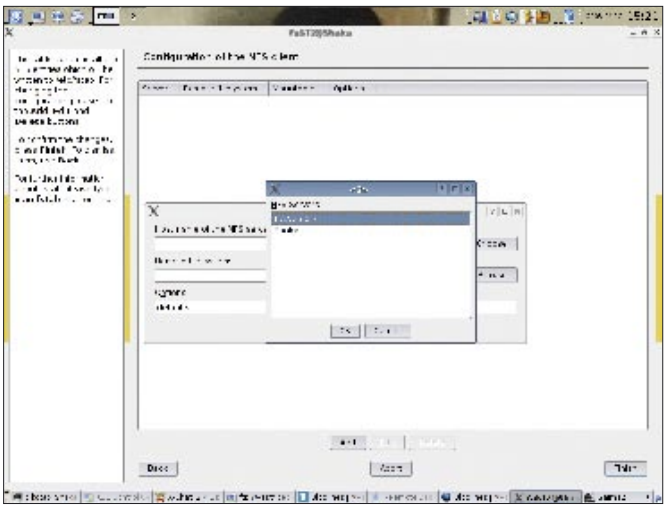

Figure 1: The Yast4Debian project brings Suse's popular Yast configuration tool to Debian.

get distribution is even more complex than compiling Yast2 and the accompanying modules. In Debian's case, package management is a major challenge. A Yast2 module communicates with the RPM database, but Debian uses the Debian package format. On a more positive note, other Yast2 modules needed far less modification; for example, because Debian and Suse both use *shadow*, the developers were able to adopt the user management module more or less unchanged.

The Yast4Debian developers can point to the first fruits of their labor: besides the user configuration module, the partitioner now also works without any trouble. But there is still much to do, and the imminent move to the Yast2 version from the current Suse distribution will add considerably to the workload. If you are interested in getting involved with the porting work, check out the project homepage at [1] for details.

Qt 4 and KDE 4

The KDE development cycle is traditionally modeled on that of the underlying graphics library, Qt. For example, a development branch of KDE 4.0 was opened just after Qt 4.0 was released. KDE 4.0 is designed to leverage the features the new Qt version offers, while at the same time removing a few design weaknesses that affected KDE 3.

The gap between the release dates makes it obvious that some major changes to KDE's Qt underpinnings must have taken place. Version 4.0 of the graphics library was released June 27; the previous version dates back almost three years to October 21, 2001.

One major change to Qt affects the license rather than the code. Qt users originally only had the option of selecting a free license on Linux, but with Qt 4, the manufacturer Trolltech introduced the GPL license for proprietary programs by Windows developers in addition to the *commercial license*. Some programmers are even talking about porting KDE to Windows – now that the legal issues have been removed, it is just a question of technology.

Qt 4 also adds value for end users. The library now supports 64-bit architectures, and this means considerable performance benefits.

The new engine, which goes by the name of Arthur, also deserves a mention. It allows users to add painting and drawing functions to Qt applications in a simple way. Arthur has a big speed advantage over the QPainter tool provided by Qt 3, and it interfaces with the GUIs of any operating systems that support Qt. This means that the developer no longer needs to think about the specific characteristics of a user interface. The new Scribe font engine also removes the font problems that affected Qt 3.

All in all, Qt 4 gives KDE programmers a powerful base for the next generation desktop. The project Wiki [2] for the recently opened KDE 4 branch already has an impressive list of suggested features. Among other things, the developers plan icon rendering and optical updates of the individual KDE components. KDE will also improve the user experience by using Qt's own access methods for all icons and buttons, thus providing a standardized look.

The Plasma [3] project will be looking to revamp the KDE 4 look. According to plans, it will integrate various KDE elements, including a completely reworked version of the Control Center, the Kicker, and the Kdesktop, which will act as the root window that contains all other KDE windows. On the design side, Superkaramba – an optional component previously – will allow programmers to enhance more or less any aspect of the desktop with a minimum of effort, giving users a Mac OS X style task bar (Figure 2). As a result, the developers are aiming for a modern and "completely new desktop design." A KDE 3 compatibility library will provide a bridgehead between KDE 4 and its predecessor.

The KDE programmers have not yet set a completion date. Of course, the KDE people are not the only ones really interested in the success of their development activities: a few weeks ago, Trolltech hired KDE programmer Aaron Seigo to allow Aaron to concentrate on KDE development. Trolltech seems to be looking to make the best known Qt application an object of prestige.

Bluetooth Headsets on Linux

Bluetooth headsets are a useful thing. In the car, these cheap earphones can replace a more expensive hands-free system, and they work with any Bluetooth devices that support appropriate audio profiles. A driver for the Alsa Sound System means that at least a few headsets can be used for listening to music on Linux home PCs. This said, the Alsa Bluetooth project has still not been formally accepted into the mainstream Alsa developer branch. Besides the source code and an installation guide, the *snd-bt-sco* module is available from the project homepage at [4].

The Alsa Bluetooth extension uses the A2DP (Advanced Audio Distribution Profile) protocol to support stereo sound, although this has been restricted to stereo reception thus far. Stereo transmissions have failed, as the current headset hardware does not support this function. However, new Bluetooth headsets that don't have this problem are under development, and the *snd-bt-sco* driver already has the necessary functionality.

Bug Squashing Party

There has been a bustle of activity at the developer branch of Debian in the wake of the Debian Sarge release. For example, the future version of the distribution has moved to GCC version 4.0. The migration pushed the number of release-criti-

ADVERTISEMENT

cal bug reports up to almost 600, and this has left Debian developers facing the task of removing the bugs.

To get up to speed, Frank Lichtenheld initiated a bug squashing party at the start of August. A bug squashing party consists of developers meeting on an IRC channel to hunt down the bugs in their own and other project teams' packages. The administrators loosen up the rules for package uploads to allow anyone taking part to upload their patches directly. The group bug hunt led to the removal of 32 release critical errors, letting a number of packages previously considered *unstable* make their way to *testing*.

While development work on Sarge was in its final stages, bug squashing parties took place more or less once a week, helping the new version to take the final release hurdle – in fact, group bug hunting has often led to visible results in the past.

The Future of the Debian CD Tool

The Sarge release saw the Debian software collection grow to encompass 15,500 packages. If you do not have a fast Internet connection at install time, you need DVDs or CDs with the software. The complete collection fills a dual-layer DVD or 14 CD Roms. With an Internet connection, you simply need the first CD and can download the rest.

To ensure that the first CD does its job well, it needs to contain a full set of vital packages. Additionally, the dependencies are not permitted to refer to any other media. The Debian CD Tool [5] ensures that the medium complies with these conditions, but there are two things to worry about. For one, the source code is more or less unintelligible, as various developers have adapted it to the changing conditions at Debian over the years. Programmers who wish to use the CD Tools to create their own sets for special requirements first have to dig down into the code and attempt to understand the programming.

Figure 2: Superkaramba enhances more or less any aspect of the KDE desktop. The Plasma project will give KDE 4 an attractive design.

At the Debian Developer Conference (DebConf) in Helsinki, Finland, a group of programmers centered around Steve McIntyre got together to discuss ways of simplifying the CD tool. Source code restructuring is the central issue; at the same time, the team will be looking to update and improve automated dependency checking. Steve McIntyre has announced a number of goals on the Debian-Devel-Announce mailing list [6].

But the CD Tool is facing an even bigger problem. To find out which packages it should put where on the 14 CD set, the tool draws on the results of the Popularity Contest. This is a kind of hit list of Debian packages. Debian users can set up the *popularity-contest* package to send a list of the packages installed on their systems to the popularity contest server once a week. The statistics this produces are evaluated to decide the package order on the Debian CDs.

Unfortunately, the number of users who transmit package lists is dwindling, which makes the data less reliable. In a posting to the Debian-Devel mailing list [7], Petter Reinholdtsen warned of the imminent demise of the package list. Without the list, the CD Tool is useless.

That's all folks...

… for this month at least, but we do have one request: if you can recommend a program that you would like to see featured in *Projects on the Move*, why not mail me with your suggestion [8]? I look forward to your comments! ■

INFO

[1] YaST4Debian: *http://yast4debian.alioth.debian.org/*

[2] KDE 4 targets: *http://wiki.kde.org/KDE+4+Goals*

[3] Plasma project: *http://plasma.kde.org/*

[4] Bluetooth Alsa: *http://bluetooth-alsa.sourceforge.net/*

[5] Debian CD Tool: *http://www.debian.org/CD/*

[6] Steve McIntyre on the Debian CD Tool: *http://lists.debian.org/debian-devel-announce/2005/07/msg00005.html*

[7] Peter Reinholdtsen on the popularity contest: *http://lists.debian.org/debian-devel/2005/07/msg01269.html*

[8] Tips and suggestions: *projects@linux-magazine.com*