

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

PROJECTS ON THE MOVE

Split-ups, disputes, and flame wars repeatedly ripple through the open source community. So far, forks have benefitted the community, as illustrated by the reunion of Compiz and Beryl in the Compiz Fusion project. **BY CARSTEN SCHNOBER**

The open source community is a virtual community in which opposites meet: large enterprises that earn money with free software on the one side and idealists who view the free software development model as a new paradigm for society on the other. In between lies developers with different motivations and diverse cultural, social, and geographical backgrounds.

The truly heterogeneous composition of the free developer community is both a boon and a hindrance. On one hand, it means great flexibility because you are likely to find an expert who can write patches for exotic hardware or a freelance translator who will translate software into a language that a commercial enterprise might ignore. On the other hand, different mentalities often lead to different opinions, and in many cases,

these differences escalate into disputes (a.k.a. flame wars).

Once a flame war starts between the developers of a software project, something often happens that is unique to open source programming – a fork, in which a group or a person quits working with the other developers, points the program code in a completely different direction, and thus launches a new project. Open source licenses are the legal basis for forks. The source code of free software is nobody's property and is available without restriction to anyone who is interested in it – both to the

members of the project and to previously uninvolved developers. Nobody can stop you picking up the code and launching a new project based on it, even if the math of efficiency would suggest that stakeholders would be better off cooperating.

An essay called "Homesteading the Noosphere" by Eric S. Raymond provided an early definition of a fork: "The most important characteristic of a fork is that it spawns competing projects that cannot later exchange code, splitting the potential developer community." [1]

To Fork or Not to Fork?

History shows that just a few large-scale projects have actually forked, indicating that developers are aware of the drawbacks. In most cases, stakeholders reach a workable compromise





Figure 1: 3D desktops bring all kinds of useful and attractive optical effects to the desktop.

and developers can continue to join their efforts. This explains why the results of previous forks are not as dramatic as Raymond anticipated.

In 1997, a group of 25 developers left the GCC project [2] and founded EGCS (Experimental/Enhanced GNU Compiler System) [3]. EGCS provided a catchall for various minor forks that implemented features the GCC maintainers were unwilling to accept into the official GCC branch. The effects of the fork were positive for the most part, and in 1999, the Free Software Foundation, at the same time the home of GCC development, made EGCS the official GCC because of its more effective development work, thus officially closing the divide.

In 2003, the Linux community was concerned with the imminent fork of XFree86 [4], the graphics server for Linux and other Unix-style operating systems for which there was no alternative at the time. In this case, it was licensing rather than technical issues that led to differences of opinion, and the fork finally occurred in 2004. The fork did not weaken development in this case, either, because nearly all of the XFree86 developers decided to move to the X.org [5] replacement project. X.org seamlessly took over as the new standard graphics server for all major Linux distributions shortly after.

The latest example of a fork has also shown that the open source community really does try to find a meaningful compromise to focus its efforts. I'm referring to the 3D desktops Beryl [6] (see Figure 1) and Compiz [7], which was mainly based on an initiative by Swedish devel-

oper David Revemann. Compiz was the earlier of the two projects and aimed to add transparency and other optical effects to Unix and especially to the Linux desktop. Compiz is mainly funded by Novell, and it is Novell that decides which patches and plugins to accept into the Compiz tree.

Because of a resistance to integrate new plugins, a number of Compiz developers

launched the Compiz fork Beryl last year. The developers accused Novell of keeping development under lock and key, and they set out to orient their work on the open source bazaar-type model and make the project a home for what they held to be critical features.

Although the Beryl developers had announced that they would be designing their code to support easy integration with Compiz wherever possible, it became apparent that the projects were drifting apart after the first few releases. Incompatibilities resulted, and it wasn't long until graphics adapters started to favor one of the 3D desktops. At the same time, the number of features that forced users to choose either Beryl or Compiz started to increase – two competing projects moving in different directions looked likely.

The developers of the 3D desktops have now come to the conclusion that working on two projects is inefficient, and they have moved toward reunification. At the same time, developers have founded a new project, Compiz Fusion [8], to merge Beryl and Compiz extras. The extras are extensions that the Compiz maintainer does not consider to be critical or particularly beneficial with respect to usability, particularly plugins and the CompizConfig Settings Manager [9], a tool for convenient 3D desktop configuration. Examples of Compiz Fusion extra plugins include animated iconizing, expanding of windows, or snowflakes on the 3D desktop.

Despite the reconciliation, unity between the two forks is unlikely in the near future. Instead, Novell program-

mers will continue to work mainly on Compiz, whereas Compiz Fusion represents a community-based extension. Contributions to Compiz Fusion from the Beryl branch will only involve those components that do not depend on the window manager and are thus easily reconciled with the Compiz window manager.

An initial developer version of Compiz Fusion has been released, with a more stable version due to follow. This compromise gives rise to hope that Compiz and Compiz Fusion will be included with an increasing number of Linux distributions in the future and that the 3D desktop might even become the standard choice. Although this goal is the intention of many major distributions, it is still risky to deploy Compiz as the standard window manager because of difficulties with several graphics cards.

Curse and Blessing

As Raymond feared, forks hamper the efficiency of projects. Development of free software relies on a variety of participants, including programmers, translators, manual writers, and users who are prepared to test the software and suggest improvements. When a team splits up, development will be affected. This said, the ability to fork a project is one of the basic freedoms of open source software and its licenses. A fork can be a last resort if vastly divergent interests make continuing collaboration impossible. In the case of the GCC project and Compiz, the competing objectives were new features vs. stability. ■

INFO

- [1] "Homesteading the Noosphere" by Eric S. Raymond, <http://www.catb.org/~esr/writings/cathedral-bazaar/homesteading>
- [2] GNU Compiler Collection: <http://gcc.gnu.org>
- [3] EGCS: <http://en.wikipedia.org/wiki/EGCS#EGCS>
- [4] XFree86: <http://www.xfree86.org>
- [5] X.org: <http://www.x.org>
- [6] Beryl: <http://www.beryl-project.org>
- [7] Compiz: <http://www.compiz.org>
- [8] Compiz Fusion: <http://www.compiz-fusion.org>
- [9] CompizConfig Settings Manager: <http://wiki.compiz-fusion.org/CCSM>