First Conference on Freely Redistributable Software: A Physicist's Report

Brian Gough

to be published in Computers in Physics *May/June issue*

Introduction

Just over ten years from the beginnings of the Free Software movement and the publication of the "GNU Manifesto" nearly 200 people from 14 countries around the world converged on the "First Conference on Freely Redistributable Software" near MIT this February.

They came to discuss "Free Software" – software which is written to be shared by everyone – and to hear about its past, present and future from famous hackers such as Richard Stallman, the founder of the GNU Project, and Linus Torvalds, the author of the Linux kernel.

As a computational physicist at Fermilab I use GNU tools, like gcc, and Linux in my work every day. I went along to this conference to learn more about free software and the people behind it. I hoped to see what the future might hold in store for those of us working with computers in physics.

The main events of the 3-day conference were the plenary talks, including the keynote addresses of Linus Torvalds and Richard Stallman. The plenary speakers covered everything from the use of Linux in operating systems research and high-tech medicine to the economic and legal aspects of free software. Speakers from Europe discussed the limits of network bandwidth on international ftp sites, the distribution of free software in former-Yugoslavia and the internationalisation of GNU programs to different languages. The remaining time was devoted to tutorials on free software. These were

Figure 1. Discussions about free software during a coffee break

Figure 2. Philippe Defert of CERN describing the ASIS installation server

certainly "authoritative", with classes led by Richard Stallman (on advanced Emacs and Gcc), Don Libes (the author of "Expect"), Michael Bushnell (one of the designers of the GNU HURD), Phil Hughes (the publisher of Linux Journal) and perl-wizard Tom Christiansen among others.

Plenary Talks

Among the plenary talks, physics was well-represented by Philippe Defert of CERN. He described the problem of keeping hundreds of software packages up-to-date on the thousands of workstations in use at CERN. Anyone who has tried to bring just one commercial workstation up to a level ready for doing real scientific work (and keep it that way), with emacs, gcc, TeX, etc... will know that this might occupy a whole army of system administrators!

CERN is trying to completely automate the installation of useful freely-available Unix packages on any workstation. The project (which is still under development) is called ASIS (pronounced "as is"). ASIS stands for "Applications Software Installation Server". Anyone familiar with the use of "distributions" and "package managers" in Linux will recognise ASIS as their natural generalisation to arbitrary machines and arbitrary flavors of Unix.

By performing the initial compilations centrally, under the control of package "experts", any unnecessary duplication of effort by individual system administrators should be avoided. To speed up the unpacking of tar files, compilation and installation procedure ASIS uses a perl-based system which can automate these mundane operations. A Tcl/Tk graphical user interface then allows packages to be selected and installed automatically across the network from an ASIS sever.

Although still under development ASIS is already in use at CERN, with over 400 packages supported on 7 different operating systems. Like another idea which grew out of small beginnings at CERN (the World Wide Web) ASIS has good "scaling properties". There is no technical reason why the servers at CERN could not be mirrored at ftp sites around the world. To speculate on the possibilities, just as everyone now uses the World Wide Web, will system administrators of the future install ready-to-run copies of emacs and gcc from their nearest ASIS server, mirrored from CERN? Maybe.

Figure 3. Linus Torvalds waits to give his keynote address

Linus Torvalds' keynote address

In his keynote address, Linus Torvalds talked about why free software is better than proprietary software and the status of Linux development efforts.

He first focused on the area where the advantages of free software are greatest. This is for software that we depend on every day, that we want to be able to understand and trust, and for software that – if it breaks – we need to be able to fix quickly.

He used an analogy like this: If you need to rely on your car to get to work every day, would you want to buy a type of car where the manufacturer locked the hood closed to prevent you (or anyone except the authorised dealer) repairing the engine? and what if the dealer, whenever the car needed minor repairs, always told people to buy a new car because "the problems had been fixed in the latest model"?

Most people would agree that the freedom to get a car repaired wherever we want, whether at the dealer or an independent mechanic, is a good thing. Imagine being legally required to return only to the original manufacturer for all repairs, for the whole life of the car. This would no doubt be good for the authorised car dealers (who could raise their prices) but bad for consumers.

Linus Torvalds then pointed out that when we buy expensive and complex commercial software, such as Unix operating systems, without source code, we are in this situation of being forced to return to the manufacturer to fix every problem. With free software people have more choice.

To understand this it is important to remember that the word "Free" refers to "freedom" and not "zero price". It means the right to have access to source code to programs (not just the executables), in order to be able to fix, modify and enhance them, and share the results with others. In this case the opposite of "free" is "proprietary" or "restricted", not "expensive". The original ideas of free software, and why "free" refers to "freedom" and not "zero price", are explained in Richard Stallman's "GNU Manifesto".

Linus Torvalds said that the reliability of Linux can also be attributed to it being free software: when bugs are found they can be fixed and the patches made available. When bugs are found in proprietary software, you can do nothing except report them and wait. In many cases, the software company may recommend that you buy their latest version.

Figure 4. Richard Stallman giving a seminar on gcc

In the case of commercial versions of Unix, which are distributed as proprietary executables without source code, users cannot fix the system if it has a bug, study how it works, extend it (perhaps to interface to a piece of lab equipment), give copies to colleagues or install it on other machines without purchasing expensive additional licenses. Now that there are alternatives, it is not surprising that many people are choosing to use free systems like Linux.

Richard Stallman's keynote address

As the final speaker at the conference, Richard Stallman began by announcing that one of the goals of the GNU project had been achieved. It is now possible to have a completely free computing environment on the PC platform, using a GNU/Linux system – the combination of Linux kernel and GNU software. The operating system kernel, libraries, compilers, utilities and programs are all available in source code form.

However, he emphasised that in this success there is also a danger: people have begun to promote GNU/Linux and free software purely on their technical advantages over commercial products – forgetting about the more important philosophy of freedom and cooperation which made this possible. GNU and Linux would not be a success, in Stallman's view, if they just ended up being used by commercial software companies as a cheap platform for their proprietary products. This is already beginning to happen, he warned, and it could bring back all the problems with proprietary software that free software was meant to avoid.

Richard Stallman described the events which led him to start the GNU Project and work on software which could be shared by everyone, free from proprietary restrictions on distributing source code. He recalled the early days of his computing career. This was in a time when all software was available as source code i.e. free (in the sense of freedom). With source code, software could be modified, improved and shared, and he remembered how much more productive this was. It meant that problems could actually be fixed!

With the commercialisation of software he began to encounter a dog-eat-dog

world of proprietary programs, restrictive licenses, non-disclosure agreements and software patents. With proprietary software, bugs would often never get fixed (since companies would not release their source code to users).

Around 1983 he decided that he would rather cooperate with others by sharing source code, than write programs with these commercial restrictions. With free software he thought people would still have the chance to work in the more productive way he had known. This would need free compilers, libraries, tools and eventually free operating systems – and thus the GNU Project was begun, to create them. As many physicists are now beginning to discover, Stallman was right and the free GNU/Linux environment is much more productive than its commercial alternatives.

Conclusions

As for the future of free software in physics, I can only expect its use to grow. The availability of source code, low cost and absence of commercial licensing conditions already make it attractive for any scientific project on a limited budget. The possibility of contributing to software development in a way which benefits our community, by adding new features to the source code, also fits well with the spirit of research. Advantages like these will no doubt mean that in years to come more free software like GNU and Linux will be running on more and more computers in physics.

Further Information

The ASIS Project
http://wwwcn.cern.ch/dci/asis
The GNU Manifesto
ftp://prep.ai.mit.edu:/pub/gnu/GNUinfo/GNU
Unofficial GNU Information
http://www.cs.pdx.edu/ trent/gnu/

Copies of the Conference Proceedings are available at a price of $25 each (+$10 shipping for orders outside the US) from

Free Software Foundation
59 Temple Place
Suite 330
Boston, MA 02111
Fax +1 617 542-2652

Photographs © Ulrik Dickow