



Building Community Around the Google Summer of Code *by Pedro Giffuni*

FreeBSD has been a mentoring organization in each Google Summer of Code, GSoC for short, a program where Google sponsors college students to work on open-source projects. This is meant to be a great way to get students involved in software development and at the same time help fund development in communities that are usually short of resources.

In FreeBSD's case, we have been really fortunate. Not many projects get considered year after year as we have been. Part of our success comes, no doubt, from having a great project that is hugely influential and innovative. Another part of our success has come from having a consistent administrator behind it, namely Gavin Atkinson, and an amazing group of mentors who have been available to work with students.

I will admit that I became a FreeBSD committer after helping a student in his GSoC project. I was not the mentor; I just helped. The real mentor behind it thought I could be a good committer; and now, more than 1,300 commits later, and with my occasional screw-ups, I hope he still feels the same. The student didn't get a commit bit, but he did get a new job and eventually became a major contributor to a related open-source project, so I think the experience was pretty satisfactory for everyone involved. I have been a mentor for the Google Summer of Code for the last three years, and this year I became involved with GSoC administration.

So Is It Really Worth It or Are We Just Ripping Off Money from Google?

It is definitely worth it, but probably not in the way most people think. Code from students rarely makes it into FreeBSD's main code repository. FreeBSD has become a hugely complex project and achieving the code quality to just bring in new code is not easy. The Bugzilla database, if you are brave enough to look at it, has many open issues with patches that appear to fix them, but if we committed them blindly, we would open many more holes.

The GSoC brings students to work with a

largely deployed and tested codebase and tasks tend to be challenging. As a FreeBSD developer, I have learned many things that are not taught in books, and in this sense, contributing to the project involves learning and growing as a professional.

Mentoring Projects

The key to any GSoC project is mentoring: we can't take on any project without mentors. As things go, we also don't have many trivial tasks for students to do, so having a mentor who thoroughly knows the existing codebase is essential.

I have been lucky enough to teach at the university level, and one thing that is similar in a GSoC project is that you learn to teach/mentor as you go along. As in real life, students don't always know what they say they know. As in real life, students and mentors don't always understand each other, and this is especially true if they don't see each other's faces. Projects can fail, but failures can be useful on their own.

My first project failed, and while I think that it was not really my fault, I still think we should not be afraid of failure. Both of us learned something in the process, and even when the student didn't agree he was not performing well, he later reapplied to another GSoC with us, so it was hardly a trauma he wouldn't recover from. On my side, I just keep learning new things every time I mentor.

The key behind a successful project is not really the code; the key is answering questions like: Is the student understanding and learning something? Is the student finding value from getting involved in the community? And the occasional, Wouldn't it be interesting if we were to try ...?

Of course, the code might eventually come

about, and there is always a good chance someone else will find it useful. I have personally taken code from a GSoC developed in another project and imported it, after a lot of hard work, into FreeBSD. Eventually I reverted it and fixed it again and reimplemented it, but ... well... if I had had to write the code from scratch, it wouldn't have happened at all.

We also try to introduce students into a culture: we have them use the same tools we use, we make heavy use of version control, code is reviewed, and interaction with other community members encouraged. It is disappointing to see how some other projects, which I won't detail here, finish a GSoC without a status report and no traces of the code, good or bad, that was developed as part of it.

Another thing we have noticed, and I find particularly interesting, is that when other projects, possibly but not necessarily a BSD variant, don't get selected for the GSoC, we may get an influx of developers with experience from those projects. This is great, as we introduce some important variety to our own development process: I would like it if we started working on common projects with other BSDs, but also with related codebase *Illumos*, or variants like *Debian kFreeBSD*, *UbuntuBSD*, or even different codebases like *Apache Cloudstack*. We surely haven't exploited all the possibilities.

How We Choose Them

Mentors are always the key: we have a steady influx of students each year, but developers don't just appear spontaneously. The selection process involves mentors voting for their favorite proposals, but ultimately we depend on finding mentors for the projects.

As we have made our way through the GSoC program, we have improved concepts like early mentorship, that is, getting potential students to discuss their proposals publicly and discussing them with possible mentors before the proposal is made. We do have a preference for students who have experience in the FreeBSD community and people who already have some experience in version control, be it subversion, git, or even perforce. It is also a good sign if the student has already used FreeBSD and knows how to rebuild the kernel or has already submitted a patch.

FreeBSD is, of course, rather particular in that it

is a complete operating system: while it is not uncommon to find projects that affect only the kernel or only userland, it wouldn't be unexpected for a project to work on multiple areas to achieve its results. While we won't discard any general purpose project, we generally do try to favor projects that people can't do elsewhere, and so any work related to FreeBSD-specific technologies like *capsicum*, *netgraph*, *netmap*, *bhyve*, *jails*, or even somewhat less specific technologies like ZFS and DTrace are likely to catch our attention.

As of the time of this writing, we have received 36 proposals, and we accepted 15. We were conservative but I hope we caught the best of what our users want to find in FreeBSD.

Graduating from the GSoC

We have had students who participated multiple times and they have found it a great experience, which is ultimately what defines whether a project is fully successful or not. Recently Google has decided that students can only participate twice in the program. This is understandable and fair, and while we did lose a very interesting project due to it, we just have to remember that we are not here for the money; we are here because we like it. When there are exceptional projects, we will eventually find other ways to sponsor them.

Getting students involved in conferences and interacting with other developers, or even helping them get in touch with companies that use FreeBSD, is vital for the community and is a process that we are still learning.

HUGE thanks, Google!

Pedro Giffuni is a Mechanical Engineer from the Universidad Nacional de Colombia. He learned his way around computers when he was around 14 and trapped between the early days of Microsoft Basic in a Tandy Coco and an Intellivision. Pedro's first experience with FreeBSD was in 1997 when he installed 2.1.5-Release by downloading a floppy in his school in Bogotá from a then "fast" 19.2K modem connection. By then he knew how to program very well in Pascal and Fortran, which he would later discover to be basically useless. After much more fun than pain, Pedro has not needed to try other UNIX-like variants since then.

WRITE FOR US!



Contact Jim Maurer
with your article ideas.
(jmaurer@freebsdjournal.com)