Quick as a Gnash

Adobe spent ten years making Flash, but it took Rob Savoye 1/10th that time. **Paul Hudson** speaks to a coding maestro...

Not just anyone can reverse engineer the internet's most popular proprietary format. Then again, not just anyone can code on GCC, GDB, OLPC, Cygwin, DejaGnu and Linux while finding time to do hurricane-relief work in New Orleans. But Rob Savoye did all that and more, and his free rebuild of Adobe's Flash Player has added open codecs so that the web has the chance of a truly free future.

Linux Format: So just as a brief intro, could you give our readers a brief history of your work in the free software world?
Rob Savoye: I started working on free software in 1987 on the GCC project... 20 years now! I worked on GCC and GDB for a hardware company who couldn't afford expensive compilers from Sun Microsystems. So free software helped us in those days. So I worked on GCC and GDB for a couple of years, and what happened was that in the late 80s we got so many requests for paid support that we started Cygnus Support in 1989. I worked for Cygnus Support almost up to the end when it was sold to Red Hat around 2001. So I've pretty much only done free software for 20 years now. I've done a lot of work on GCC and GDB, I wrote DejaGnu, I wrote libgloss, I worked on newlib, I started the Cygwin project, and I designed the eCos operating system. I've worked on a lot of Linux stuff since 91, and I've spent years consulting to Nasa and others on GCC and Linux kernel work. Then I started the Gnash project as a user interface layer for set-top boxes.

LXF: Wait, so you went from GDB to Gnash? That's a big jump!
RS: A friend of mine, John Gilmore, invested in a lot of little companies. And so he calls me up one day and says, "hey, I know some guys who want to do a Flash-based user interface for their stereo," and I'm like, "great! I'm into digital music, this sounds fun." And so, I searched the internet and saw what was out there for free software players, and there weren't that many. So I took the technology out there and added all this ActionScript stuff and finished the user interface for the stereo. It was a six-month project and I never really thought about it – it was a one-off project, right? Then a couple of months later, John calls me back and says, "hey, how would you like to turn that Flash player thing into a plugin for Firefox?" I'm like, "John, I don't do graphics, and besides, I'm packing to go to New Orleans for Katrina relief." So when I got back from Katrina five months later I was broke, so I called up John and said "yeah, a Flash player plugin for Firefox sounds great!" And so I did it. And instantly we got a huge amount of publicity just because it worked. It didn't have YouTube support at the time, but for a lot of us free software folks it was important,

and because of my background in the GCC team, Gnash runs on 64-bit machines, it runs on multiple processors – way, way, way more than the Adobe version – and I suddenly realised that there was this real need for free Flash technology, and that scope of the project has really expanded recently into a lot of work in patent-free codecs. But that was it – Gnash is a like a lot of other projects in that it wasn't a decision initially, it just happened. And then I realised, good projects find *you*.

LXF: So you are still involved with GCC or GDB?
KK: Slightly. I did a lot of the GCC work for the OLPC project. I did all the compiler optimisations for the Geode GX chip. There aren't that many freelance GCC hackers around! I'm really into embedded systems, so I'm one of the guys who helped add all the cross-compiling support into GCC and even did the first Linux port to 68K a long time ago. And then what happened was that I was working on Gnash – funded by myself and John Gilmore mostly, and often not funded at all – and then what happened was that one day I got a call from Bob Young (ex-CEO of Red Hat), and so Bob's like "I've got this little press business now, and we want to really start getting into digital media more. We really need free software and I know with some more money you guys could pay your developers, you could work full time, you can increase your pace of development so that my company could use this stuff." So we decided to do it. So Bob has been sponsoring us for a year and a half. And we realised recently that that was not quite sufficient as our scope expanded into what we're calling open media as opposed to Gnash – that's only a part of it. So we set up a non-profit foundation called Open Media Now (**openmedianow.org**), Bob and John are both on the board of directors, and we're talking to a lot of companies who have never had Flash support for their systems. If you use MIPS-based hardware, SH-based hardware, weird ARM chips, 64-bit PowerPC, then we're it. We have PS3 support! We're on Itanium, we run on Solaris, we run on a lot of interesting operating systems. We're really into cross-platform, and Adobe just doesn't care about multiple platforms.

ON FINDING A PROJECT

> **"Gnash wasn't a decision initially, it just happened. And then I realised, good projects find *you*."**

LXF: What was the point where you decided that Gnash wasn't enough, and that you needed to push for open media?
KK: So what happened is this little thing called "software patents". When you use MP3 or FLV, they're proprietary. And although we use FFMPEG and Gstreamer – we actually support all these codecs – we can't distribute Gnash that way. So I got buried in bug reports about YouTube not working, and it's infuriating. Even when we did the OLPC "give one, get one" program, I was buried in bug reports saying, "I thought YouTube worked on this thing?" But of course the OLPC project cannot legally redistribute the codecs. So what happened was that I was getting buried in bug reports, so we suddenly realised that the issue we were trying to solve wasn't just reverse engineering Flash players, the problem was the media. So we're doing a lot of work to open that up – Gnash fully supports patent-free codecs such as Ogg Vorbis and Theora, and Dirac and stuff. But the problem was bigger, and we realised we needed to build an entire infrastructure, because the free codecs have all been out there for years, but one project supports one but the server doesn't… it's just not there. And you need a seamless infrastructure if you're going to make media free. Commercial companies don't care about this sort of stuff because they license it, but most of our friends are in the free software world, so we want to support the Fedoras, the Ubuntus, the Debians, the BSDs, and all those people. Wikipedia is about to put up a new video sharing site that's all Ogg Vorbis and Theora-based, and they need a server for it. We're working right now on Flash-based video conferencing, and that's server-side – and it's all proprietary formats. But if you have the server and you have the client, then you can change the codec and nobody cares.