

GNU.FREE... A Free Software Odyssey

by Jason Kitcat

I stared at the email in disbelief, I really couldn't believe it. I'd merely been chasing some copyright assignment papers that had gone AWOL in the Free Software Foundation and now here was an email from rms (Richard Stallman)! My heart was leaping as I read rms ask if I wanted my software to become part of the GNU Project, the ultimate Free Software umbrella project. Of course, at this point I had no idea how hands-on rms was, nor how much work it would take for my software to become fully GNU – I was just thrilled by the glamour and recognition it brought to my efforts.

GNU.FREE had started merely as an idea which needed an acronym when I was studying a joint degree in Computer Science and Management Science at the University of Warwick. My interests had led me to do a final year project which addressed the impact of the Information Revolution on political life. One area that I had flagged up in my initial reading was the continual trend towards (and the obligatory predictions by pundits) the voting process being technologised with the help of phones, computers and the Internet. However when I looked into the details of the existing systems I was extremely disappointed by their poor security, usability and their exclusively proprietary nature. Let me rephrase – I was shocked and really angry – how could these commercial outfits with smooth talking websites so easily gain the blind trust of governments?

I won't give a lecture on all the issues surrounding electronic democracy as I could easily fill this whole magazine with my thoughts on the matter. But I did find it surprising that nobody was actually pointing out the risks of using closed systems in voting, and few people are even discussing it today. I strongly felt that the Free Software paradigm was the best and more importantly, the right way to develop a voting system for several reasons. Free Software was a better use of tax payer's money, it would improve security and trust in the voting process and it prevented the government from being locked into proprietary offerings for decades – a nation should have control of its technological destiny.

Thus apart from writing a huge dissertation I very ambitiously (and naively) set about creating a Free Software Internet Voting system that I would come to call FREE (Free Referenda & Elections Electronically), a name which I was quite pleased with, having struggled with more letter combinations than I care to admit.

After seven months of solitary development on my small home network of two Macs and a PC I had created a working system and in the process learnt a huge amount about Java and security. There was still much to be

done but the project was submitted and to my delight I was given an extremely high mark and a joint prize for best project for the year.

Before I'd even received a mark I had begun to package the software up and put it online. I'd had to wait until the code had been submitted for marking before I could share it, otherwise there could be complications with regards to plagiarism. From 20th March 2000 FREE was available online, and as I continued to develop it new releases were pumped online. In all, by my count, 13 production releases have been published since that first day. In the course of those releases I've surmounted a huge variety of technical problems, added a wide variety of technical features which required skills ranging from XML to SQL to Blowfish encryption. I've wrestled with bizarre network bugs which have had me leafing through screens and screens of encrypted packet dumps trying to figure out the problems and I've spent a huge amount on Java-related books dealing with threading, performance tuning, security, AWT graphics and God knows what else!

The coding has at times been frustrating and mind numbing but in the end it's always been extremely satisfying. But the reality is that the proportion of time I've spent on programming has gone down as the project has progressed. Let me explain why...

Initially I'd primarily focussed on the software but as the odd email began to trickle in and I looked at other projects to gain from their experiences I realised the need to focus on other aspects of the project – a successful Free Software development needed more than good code. At the same time electronic voting was becoming a hot topic, especially after Florida 2000. Several companies were creating a lot of noise over their products, governments were squawking wildly and I was trying to keep the Free Software agenda in the debate. I contributed and networked as much as I could while also killing myself trying to start a small Internet consultancy with two other partners.

I met some great people on and off-line, rms was always extremely precise but helpful and a lot of friends were very supportive of the project. I managed to get onto the editorial board of "The Bell", a small newsletter dedicated to electronic voting and later onto the OASIS standards committee for Internet Voting which was being chaired by the UK's Office of the e-Envoy. I had already made some contacts in the Office due to the little debacle of the Gateway's unfriendliness to non-Microsoft users – the information I gleaned in that episode was splashed in this worthy organ and The Register. They knew me and so I could make some trouble, and contribute my experience, to the standard where previously commercial interests and well-meaning but inexperienced (as far as electronic voting was concerned) civil servants were the only ones working on what I felt to be a key, but flawed, development.

So of course, inevitably, priorities altered. From trying to just get a summary of my final year dissertation online I was writing articles for our website and magazines publicising my views on Free Software, electronic voting and democracy as a whole. I refocused and called the 'entity' dealing with all this The FREE e-democracy Project and gave it a set of goals and business cards.

The FREE e-democracy Project's goals

Software Development Aims

- Provide a secure and private system
- Create scalable and reliable software
- Offer a non-commercial, non-partisan voting alternative
- Use the GPL to create an open system that Internet users will trust
- Release a system that can be used to support the growth of effective democracy anywhere in the world

Project Aims

- Develop a leading electronic voting system
- Advocate the Free Software paradigm
- Evangelise the use of technology to strengthen democracy within a holistic understanding of the current malaise i.e. Internet voting alone isn't going to solve turnout problems

I was armed and ready to go into battle, a battle which bizarrely still hasn't come. I'd based my activities on the assumption that there would be a large public debate on 'electrifying' the voting process. Disturbingly the debate hasn't really happened in the USA, so I'm trying to provoke it with my limited resources here in the UK but it's a tough call.

I strongly felt that the position I voiced was bolstered by having developed a viable alternative for voting in national elections that ran on Free Software. So I was keen to ensure the development maintained momentum and thus my credibility in any debate that could be prodded into life.

I continued working incredibly hard pumping out releases. I responded to all emails as fully as possible as soon as possible. But volunteers and helpers weren't rolling in. Why not? I'd setup all the SourceForge tools, I had discussion forums and mailing lists where regular updates were mailed. I was even seeing thousands upon thousands of downloads in the logs. But nothing.

I got involved in FreeDevelopers which was an ambitious idea for a co-operative Free Software company. They had a need for an evoting system

however it got all very complicated and political very quickly, so I kept my distance (I'm still in discussion with a spin-off called GNU Herds). But rms had informed me that to become dubbed a GNU package I would need to not only improve my documentation but convert all the user-interface code from Swing to AWT as there was no Free Software Swing support. I didn't have the skills to do this and didn't think I would get the skills soon – GUI programming in Java is, in my opinion, one of the hardest aspects of the language. I put word out on various lists including the FreeDevelopers lists and was joyed to get a FreeDevelopers volunteer from India (a Mr C.V Rajagopal) do all the conversion work with a minimum of fuss.

It was in fact a very weird experience. I didn't know this guy, he was very business-like but after the code was submitted and it worked I never heard from him again. Without him FREE would never have become the GNU.FREE package it is now. In the meantime I was slaving on making the software as accessible as possible... I wrote swathes of documentation, installation notes and ensured that the automated JavaDoc files stayed current. I'd always been keen on good code readability, but now I was taking it very seriously trying to stick to the GNU and Sun guidelines while adding very copious comments. Since those efforts I've had no major code contributions. Oh the irony.

The site was up-to-date, the code was humming – it was by now a suite of several programs, installers, optional bits and a huge wodge of documentation files, I was keeping as much profile as possible in news sources and mailing lists. But code contributions were minimal, I got the occasional very helpful and serious analysis of various issues which did help develop the software but a large proportion of the emails were rants. "You shouldn't be doing this", "don't support the current powers, we need a revolution" and so on. It wasn't so much the opinions but the fact that people would email me out of the blue with extremely strongly worded opinions that often were based on a, at best, cursory examination of the project's website.

Somehow I felt obliged to respond to these emails with equal care as those emails from people having trouble using the system. But it ground me down... Although the download statistics were encouraging nobody was actually contacting me to let me know that they were using the software successfully. All I got were the rants, the problems, the pleas for help on school projects (which I replied to and go no thanks for) and attempts to co-opt my efforts into other people's causes and interests.

I'd somehow imagined at first that the whole collaborative development mojo would happen if the right tools were there. As my own experience of developing online communities grew through work done for clients I realised that I needed to be more proactive. But the apathy, the

negativity really wore me down. Especially because I was working damn hard on the code.

But now, with the benefit of hindsight, I can pinpoint several errors. Professor Ed Steinmueller, from SPRU at the University of Sussex, has pointed out the importance of the initial 'bid' in the open source world – that's what he classifies most 'projects' on SourceForge as, bids for collaboration. But my bid was non-existent as I'd been developing for seven months in isolation and when I did go public I hadn't really made an effective attempt to pull others in because I wasn't really hooked into the right places or even aware how hard you needed to work to pull others in.

I think I needed to come to terms with the truth that most Free Software *is* developed to 'scratch an itch' and the reality is that heavy-duty Internet Voting isn't going to scratch many people's itches. It's also a tough area to get into, a lot of the development is very far from being sexy or hot when compared to file-sharing, multimedia or web tools. No, I didn't have a hot ticket bid on my hands.

The grind was getting to me, but I gritted my teeth and kept developing the software. It had been dubbed a GNU project on 8th February 2001 and I was pushing on with GNU.FREE no matter what others did – I believed it was an important project to do, and a few others shared this common belief even if they couldn't directly contribute. What was most shocking was how disorganised the GNU project actually was. They had some great resources on the web but when something needed doing they were so dependent on volunteers that many things got dropped or delayed. So while I got increased legitimacy, wider distribution and some useful contacts from GNU, I didn't get any significant support in the actual business of building the code.

26th January 2002 marked the last release of GNU.FREE, though I didn't know it at the time. It was a mammoth release which dealt with some very complex problems by using an XML configuration system. The key issue had been supporting various voting and counting systems such as Single Transferable Vote or First Past the Post. Looking at the next version filled me with dread as implementing many of these voting systems was proving extremely difficult, the number of variations was astounding. So were some security issues that I wanted to nail down.

Consequently while I discussed the issues with some people and did some research into potential solutions I was also getting more embroiled in other areas of electronic voting. The standards work with the Office of the e-Envoy was highlighting the massive complexity of maintaining voting security and anonymity, as were reports coming out of trials in the UK and USA. US experts such as Bruce Schneier and Rebecca Mercuri were

saying some very thought provoking things. These issues were all rolling around in my brain along with the remaining development issues I had difficulty solving for GNU.FREE. Even with the best intentions in the world some simple installation mistakes could fundamentally compromise our system. How hard would it be for something to go wrong? Was it realistic to find a way to prevent all these potential screw-ups with code?

The final straw was a book by Greg Palast, "The best democracy money can buy", which in one chapter explained how the Republicans had stolen tens of thousands of votes in Florida. The details are arduous, but in essence a sub-contractor used to clean the electoral roll was the cover for the illegitimate removal of a large number of Democrat voters. This was shocking enough, but the fact the Palast had to come all the way to the UK to get the story published in the independently funded BBC and Guardian newspaper truly drove me crazy.

The reality is that technology is a fantastic smokescreen behind which fraudulent plans can be hatched. Technology can't solve the inherent problems that most people aren't comfortable with computers, they have trouble enough upgrading their copy of Internet Explorer. Yet such inexperienced users, elections administrators, were buying and even advocating computerised voting systems. Who knew what tricks politicians, suppliers and consultants could be pulling? The potential for fraud is huge and that's before you look at the actual technical failings of the systems. I realised that I couldn't go on building a system that was going to be open to abuse, no matter what I did. Making electronic voting sufficiently secure would be nigh on impossible, especially when you had to keep it anonymous and auditable. It's a very, very hard technical proposition – get rid of the requirement for anonymity and it becomes tricky but not outrageous. But that we can't do, not if we want to keep a healthy and trusted voting process. So the very development activity I thought might help my credibility in the anticipated debate I now regarded as a liability.

Despite all this I was still emotionally very tied to the continued 'life' of the software's development, so it took quite some time before I actually crystallised my thoughts into a formal announcement. Finally, after much thought and many long walks, I felt secure and confident in my decision. So I sat down, loaded BBEdit and posted the following to the project's website:

25-10-2002 FREE project policy change...

From my experience of designing and developing GNU.FREE over the past three years it has become clear that creating an Internet Voting system sufficiently secure, reliable and anonymous is extremely difficult, if not impossible. As Bruce Schneier points out "a secure Internet voting system is theoretically possible, but it

would be the first secure networked application **ever created** in the history of computers."

I've spent much of my limited time and energy trying to persuade people (and the UK government in particular) that a Free Software voting system is the **only realistic way** a trustworthy voting system could be created. But they are culturally set in their ways and always need to deal with companies, no matter how fragile their security actually is. Governments don't even have the skills to assess the reliability of the people let alone the technologies these companies sell.

However the more I have coded, researched, discussed and read the more I've realised that rather than encourage, in a way, the use of electronic voting techniques (even if I only advocate the use of Free Software) I'm much better off focussing on the dangers all such technologies present to processes such as voting. The US have a bold voice on this issue in Rebecca Mercuri. The UK has no voice on these issues and certainly the world as a whole needs more educated (if I may say so) voices on the use of technology in democratic processes.

Thus I have halted development of GNU.FREE but it remains online and I still support the concept of Free Software in e-government. If organisations want to use GNU.FREE I'm happy to advise on the issues but I'll be focussing elsewhere because, as Mr.Schneier points out, "building a secure Internet-based voting system is a very hard problem, harder than all the other computer security problems we've attempted and failed at. I believe that the risks to democracy are too great to attempt it." And this guy eats crypto for breakfast.

So, that was that. Now I had to dig in and try to catalyse the debate. As far as I'm concerned that marked the end of my career running a Free Software project. But there was one more thing... I'd failed to inform my copyright protectors and arch-supporters, the Free Software Foundation.

It took less than a month before I was once again staring at an email in disbelief and wonder. The bleach white of the screen pixelated into a blur as I leaned back and appreciated rms's typically concise email.

From: Richard Stallman <rms@gnu.org>
To: Jason Kitcat <jleep@free-project.org>
Subject: Ceasing work on GNU.FREE
Date Sent: Sun, Nov 10, 2002 06:07pm

I support your decision to stop, and I think that your history in the project should give you a powerful platform for opposing the use of risky online voting.

Nonetheless, you ought to have discussed it with the GNU Project before making a public announcement.

Oops. Well I'd messed up, but it seemed like a good omen that rms agreed with my views. Now I just had to get everyone else to agree with my arguments, I switched applications and began to plan a new article... "Why Electronic Voting could be democracy's last gasp."

Epilogue

I'm still Head of Production at my small Internet Consultancy, Swing Digital and the FREE e-democracy Project continues its role in trying to stir a debate around e-government and e-voting. In the meantime I'm doing a MSc at the Science & Technology Policy Research Unit (SPRU), University of Sussex where I'm exploring issues of technology and innovation management, particularly in relation to e-government.