Open Source Could Create Better Election Software, Jobs

Tuesday, 14 July 2009 08:10 Jeremiah Gray

You may know the Federal Election Commission if you've ever been involved with national politics. The FEC is the regulatory agency that administers and enforces federal campaign finance laws; it's sort of like the IRS but specifically for congressional and senate campaigns. Any serious candidate for public office on a national level has to file campaign finance reports routinely or face administrative penalties. Along with campaigns themselves, political committees called PACs are also regulated by the FEC and must file regularly.

Since 2001, every PAC raising more than $50,000 in a year has been required to file electronically. This low threshold amount means that most political action committees must file their quarterly reports electronically and is specifically what led me to the FEC's electronic filing solution, FECFile, in the first place.

Last year, I was asked to help a political organization that needed to file its reports with the Federal Election Commission. I had previously helped them around their office where they run Ubuntu desktops and Debian servers, but I had never assisted with the reporting. So when a staff member called me asking for help with a program called FECFile, I had to look it up.

From the FEC's website:

"FECFile is a Windows based electronic report filing application. This easy to use software enables filers to record and track information required for reporting to the Commission and to securely submit this data electronically. The software contains a validation utility to perform checks on certain data fields prior to submission and a high-speed image generator to create a paper rendering of the report among other features."

The staff person at the PAC told me she was having problems finding files after exporting them, and since I figured she was running FECFile via WINE, I showed her how to access the WINE file system to find her files. It was a simple fix, and I didn't think much of it at the time. Several months later, I got another call that they were having trouble installing a required FECFile software update and that the FEC did not offer support for Linux-based systems.

I was able to install and test the software for them, but this time, I got to thinking. I thought about how backwards it is to offer only Windows-based .exes in an age of information exchange. Moreover, of all the entities that should be going out of their way to keep expenses low and satisfaction high, government agencies funded with public resources should be at the top of the list.

Wanting to find out more about how FECFile came to be, I reached out to Louis Levine, the senior vice president of IT at NGP Software in Washington DC, a top political fundraising and compliance software provider. Figuring, as an executive of an established software company, he may well prefer the proprietary offerings, Levine enthusiastically built the case for open sourcing the currently proprietary FEC software packages.

He shared with me a brief history of the software: FECFile was developed for the FEC and introduced in 2000 by a company called NIC Technologies (at that time known as SDR). In addition to FECFile, NIC Technologies supplied the FECFile "vendor tools" to the suite. These vendor tools are also closed source and Windows-only. One of these tools, the validator, causes additional headaches on Levine's more high-end servers; despite plans to upgrade the COBOL-based validator to C, the COBOL version remains. "Try running that on a server with >4GB of RAM," insists Levine, "it just doesn't work." To obfuscate the situation further, the validator was released as a .dll and as an .exe, but they use different code bases and return divergent results. As I learned more, it started to seem like FECFile may not even be manageable for people running Windows.

The quirks were not limited to the COBOL-based validator. Another tool, the view/print utility, uses a proprietary descriptor format to draw the actual filing forms. It's similar to something you would see in Crystal or another reporting program, but proprietary code drawing forms with a proprietary utility--sometimes with errors. A few years ago, they added a Print-to-PDF feature that usually works, but only from the command line, using a specific third-party generator. The final tool, an upload utility, also comes as an .exe with a GUI or as a .dll, which doesn't support multiple simultaneous uploads.

After a decade of experience in the election software business, Levine would doubtlessly prefer, at least, open source vendor tools.

"I can't imagine how many hours I could've saved had we access to the source code. Even if it were just the vendor tools, it would make such a difference. I can rattle off a number of improvements we would make, and that is just to the vendor toolset," Levine said, but then added, "As far as I know, the FEC owns the code and all the IP. They could do it. But I'm not holding my breath."

Hearing this sort of feedback from an industry expert suggests that through its reliance on the proprietary cathedral approach, the FEC not only prevents a natural economic ecosystem from developing, but it also obliges those already in its sphere of influence to commit resources to arcane and inarticulate practices.

Rather than committing public funds to delivering proprietary solutions that create new administrative overhead both in the government and the campaigns, the Federal Election Commission and other government

agencies could improve conditions for businesses and political organizations alike by adopting open sources and formats.
In this age of slumping economies and rising government liabilities, converting government reporting forms and software to open source projects seems like a fairly simple and obvious solution to reducing expenses and streamlining the filing process while creating fresh new opportunities for entrepreneurism.

Jeremiah Gray

www.jeremiahgray.com