redhat

Issue #3 January 2005

RED HAT MAGAZINE

FROM TEACHER TO CRUSADER: CONFESSIONS OF AN OPEN SOURCE EDUCATOR

by Jeff Elkner

Introduction

Ever since I heard of the GPL while a graduate student more than 10 years ago I've been a crusader for software freedom. I am motivated by a desire to live in a world in which democracy and social justice are the birthrights of all people, and convinced that the best way to bring this about is for people to have direct control over the economic and social forces that effect them. Since the world around us is increasingly dominated by information technology and the software that runs it, democracy and software freedom go hand in hand.

My crusade began in a high school mathematics/computer science classroom, in which it is still largely rooted. Over the past decade it has been steadily reaching out into the community, where it has followed the Internet and the World Wide Web beyond the frontiers of local community into the world at large. In this article I will describe my journey, sharing highlights of where I've been, details of where I am now, and a few thoughts on where I'm going.

From GCC to GNU/Linux

My first encounter with free software was the GNU C compiler that CS students were using on the VAX systems at Bowie State University. It was 1994 and I was in a Master's program in Computer Science. I found out from one of my classmates that the same compiler and editor we were using in school could be used at home, together with a free operating system kernel called Linux. I was soon experimenting with the Yggdrasil and then Slackware distributions of GNU/Linux.

While running GCC was what brought me to free software in the first place, it was the important social philosophy behind it that kept me there. As soon as I learned about the GNU project and the Free Software Foundation, I made up my mind to use free software whenever I could.

I was teaching math at Forestville High School in Prince George's County, and had already begun building a Computer Science program at the school. I had no budget for this project, but I did have the blessing of my principal. So I borrowed a pick up truck from one of the school custodians and headed over to the discarded equipment warehouse at Goddard Space Flight Center with a few of my students, and together we filled up the back of the pick up truck with old computer parts. Soon we had a small computer lab made up of a hodgepodge of computers running a new GNU/Linux distribution: Red Hat Linux 1.0.

I learned an important lesson at this point that was to guide all of the work I have done since: a technology education program which involves students in as many aspects of the program as possible creates both the best learning experience for the students and makes it possible to achieve things which would otherwise not be possible. Using

free software plays an especially important role in this kind of a model. Since all of the software I use is free, the school system has licenses (the GPL and other free software licenses) for it even before I decide to use it. The approval process for proprietary software can take at least a year in the two school systems for which I've worked (I'm confident that is not atypical), which would kill any spontaneity in a technology education program. With free software, by contrast, we can experiment with and implement any software we wish as soon as we feel the need to do so. Being limited only by our imaginations, abilities, and willingness to work greatly expands the realm of the possible.

A bigger program in a new location

Just as the new CS program was taking off at Forestville, I changed jobs and moved to Yorktown High School in Arlington, Virginia. I was there two years before I was approached by the Instructional Technology Coordinator (ITC) asking me if I would be interested in teaching computer science the following year. There was one catch: the school didn't have the funds for a new computer lab, so I would have to teach in a lab made up of donated equipment. I'm sure my eyes must have sparkled when she told me this. I replied that I would be delighted to teach in a lab of donated computers, provided I could configure them as I wished. I explained to her that all of the software I would use would be free software, so the school would not have to worry about purchasing anything.

With the backing of the ITC, I set to work building a computer science lab with donated equipment and free software. Again, students were intimately involved from the beginning. Over the summer of 1997 two students and I setup the new lab. One of these students served as the first student system administrator. He recently sent me an email in which he had this to say about his experience, "I can tell you that the opportunities that lab provided gave me more real world experience than my 5 years of college..." There has been a new student system administrator each year since. Several of them have been hired into IT jobs right out of high school. Two of them won the Arlington Chamber of Commerce's Technology Young Mind award. In all cases they got invaluable, hands-on, real world experience. I've actually institutionalized the process so that the system administrator for the following year begins their term in the last quarter of the previous year. This means that the outgoing sys admin is still available for training and support.

The free software computer science lab at Yorktown is now in its eighth year. Each year the project has taken on new challenges and kept abreast of the newest happenings in the field. Some of the highlights include:

1997-1998

- First year of GNU/Linux CS lab at Yorktown —stand alone 386/486 computers, software installed from floppy disks.
- In Spring, a GNU/Linux server setup by student admins as a squid caching Web server providing Web access to school's shared computer lab.

1998-1999

- CS lab gets Internet connection.
- Two school Web servers: http://yhspatriot.net (http://yhspatriot.net) and http://linus.yhspatriot.net (http://linus.yhspatriot.net) go online.
- NIS/NFS used for shared directories in CS lab.
- E-Mail server setup for student accounts.

1999-2000

- First year using Python in intro CS class (see Using Python in a High School Computer Science Program (http://www.elkner.net/jeff/pyYHS/year01/pyYHS.html)).
- Yorktown High School Linux Users Group (YHS-LUG - http://yhslug.tux.org (http://yhslug.tux.org"))

formed by a student, a teacher, and a community member.
- Three students and teacher attend 8th International Python conference.

2000-2001
- Collaboration between Yorktown CS and Arlington Career Center T.V. Production on Introducing Python (http://www.ibiblio.org/obp/pyBiblio/pythonvideo.php) video.
- Guido van Rossum visits Yorktown.

2001-2002
- Two students and a professional mentor launch the pyKarel (later to become GvR (http://gvr.sf.net)) project.

2002-2003
- CS lab switches to Linux Terminal Server Project (LTSP) architecture (see LTSP: Moving into the Mainstream (http://www.elkner.net/jeff/articles/mainstream.html)).
- Red Hat Road Tour visits YHS-LUG.

2003-2004
- Dr. Vinton Cerf gives Interplanetary Network (IPN) presentation to YHS-LUG (November).
- Richard Stallman gives "The Free Software Movement and the GNU/Linux Operating System" presentation to YHS-LUG (March).
- YHS-LUG changes name to "Young Hackers and Scholars Libre Users Group".

Reaching out to the community

While a crusade to change the world can begin in your own back yard, it won't go very far if it stays there. Right from the start I have been interested in promoting the use of free software wherever and whenever I can, and in linking up with others interested in free software projects. Fortunately, this has not been difficult to do.

As anyone familiar with free software knows, once you begin using it seriously you almost immediately become part of a world wide, online community of users and supporters. Through the use of websites, email lists, and Internet relay chat (IRC), the GNU/Linux user community had won InfoWorld's Best Technical Support Award in 1997, the same year we were setting up our first lab at Yorktown. It was the support of this community that made it possible for a high school teacher and a hand full of students to setup a fancy, state of the art computing facility with little more going for them initially then the gumption to do it. Learning to "use the Web" as a resource for solving technical problems is now an essential part of our program. It is the students who can do this the best that become the big project programmers and system administrators while still at Yorktown and who land jobs immediately upon leaving.

Joining the Python community in 1999 provided the opportunity for us to become contributors to the Free Software movement. Python was still a relatively new language and the community around it was still fairly small. By becoming one of the first secondary school teachers to use Python as a language for learning programming in a computer science classroom, I was forced to play an active role in the Python community. The lack of teaching materials meant that I had to create my own. How to Think Like a Computer Scientist: Learning with Python (http://ibiblio.org/obp/thinkCS/python/english/) and the Open Book Project (http://ibiblio.org/obp) grew out of this need. This in turn got me thinking about the possibilities for collaboration opened up by the World Wide Web.

When a professional programmer wrote me an email about a pet project of his on which he would be willing to

mentor interested student developers, the next logical step in advancing the CS program at Yorktown was taken. pyKarel (now Guido van Robot (http://gvr.sf.net) or GvR) was a Python implementation of Karel the Robot, a teaching language developed for the purpose of teaching programming concepts to beginners. By working on this project two mutually beneficial goals would be achieved:

1. Three very talented high school students would be directly mentored by a professional programmer, giving them skills and experience not usually available to high school students.
2. The result of their effort would be a tool which would be used by and directly benefit their peers.

Each new school year when we begin our study of programming using GvR I tell my classes that they are using a program written by former students at their own school. I tell them that the source code for this program is readily available, and that if they are interested they could end up working on this or similar projects. I also tell them that by simply using the program and reporting bug or feature requests to the developers, they are actively contributing to the project.

The pyKarel project established a model that I have since used whenever the opportunity arises: join bright students with real world projects to which they can make effective contributions. I consider it part of my job to be on the lookout for such opportunities. The benefits to these kinds of experience are tremendous. Students learn best when what they are doing *matters*. Having them work with professional programming mentors gives them an apprenticeship opportunity not usually available in either high school or college. I've been told by several of my former students back visiting from college that the opportunities they had to work on projects like these made them better prepared than their peers for college study of Computer Science.

Currently, there are two projects in the works, DTN and CanDo. Motivated by Dr. Vinton Cerf's visit to Yorktown last year, three students are testing the Delay Tolerant Networking (http://www.dtnrg.org) (DTN) software for their Science Fair Projects. With help from a long time YHS-LUG supporter and guidance from an engineer at NASA, they are designing a relay system to control through software the timed breaking of an Ethernet connection, which will allow them to run data experiments on the bundle switching effectiveness of DTN.

Students learn best when what they are doing *matters*. Having them work with professional programming mentors gives them an apprenticeship opportunity not usually available in either high school or college.

The CanDo (http://cando.sf.net) project will have a direct impact in the classroom. Career and Technical Education (CTE) teachers in the state of Virginia are charged with tracking the completion of state based validated tasks in each of the courses they teach. Virginia's CTE Resource Center (http://www.cteresource.org) provides task lists and Student Competency Record (SCR) forms to aid in this, but the job cries out for a Web-based tracking system to aid students, teachers, and administrators in using validated tasks as a tool for increased student learning.

The goal of the CanDo project is to provide a free, Web-based tool to support competency tracking. Additionally, CanDo will permit the Web-based delivery of curriculum keyed to the validated tasks, as well as Web-based "student portfolios" which students can use to demonstrate mastery. The program is being developed by two high school students and a professional mentor, who is also serving as project manager. With CanDo I am trying something new: hiring the programmers and project manager for the project. Both the students and the project manager are being paid for their work. Instead of spending thousands of dollars on proprietary software, I am working with the school system and a community organization to apply these resources collaboratively to develop a free software solution to our mutual needs.

The community becomes the world

More than 10 years ago I became attracted by the philosophy of the free software movement and what I saw then as its promise to turn information technology into a tool which could give voice to the voiceless and build communications bridges between peoples, fostering the growth of new communities. Information technology is a double edged sword: it can save lives by giving a doctor the information she needs to cure a patient, or it can ruin lives by replacing workers and putting them out of work. It can serve humanity by providing the information needed to reverse the environmental degradation of the planet, or it can be used by the short sighted and the greedy to put the world in ever deeper peril. The future will at least in part be determined by who holds this sword and which edge of it they use to cut a path to tomorrow.

From the beginning I had hoped to be able to use my computer knowledge and skills for the good of the world. A major opportunity presented itself a a few years ago to do just that. Two years ago I was approached by Jeffrey Coupe and Aziz Kaddouri to be on the board of a new non-profit organization they were forming called the The Center for Innovative Communities (http://cicinternationale.org/) (CIC). The goal of the CIC is to empower communities through the development of training and education programs that use free software technology.

Since that first meeting I have been working ever more closely with the CIC on a number of exciting projects. The CIC has set up a community computing center in Temara, Morocco, using LTSP technology. We are in the process of training young people in Morocco to use and maintain the center. In Arlington, we are maintaining five computer centers for a non-profit, affordable housing corporation. We have switched three of these centers to GNU/Linux, one using LTSP. This project has provided employment for several Arlington high school students, giving them the chance to acquire valuable skills while at the same time earning money and serving their community. The CIC has also become the main sponsor of the CanDo project, and is working to extend CanDo to be used in community education.

The road ahead

While my first decade as a free software crusader has been a fulfilling and exciting one, I feel confident that the best is yet to come. Around the world from communities to countries people are looking to free software to solve their information technology needs. They are increasingly coming to understand that free software provides them with a way out of dependency and at the same time fosters the development of the local IT industry. Since free software belongs to all of us, developing countries can use it without restriction, including modifying it to suit local needs, thus becoming a contributor to its development. In Venezuela, for example, the entire public administration sector will be switching to free software over the next two years. There are similar stories taking place throughout the world.

I plan to be working closely with the CIC to be part of these world wide efforts, and in the process to help build innovative communities for a better tomorrow.

About the author

Jeff Elkner is a computer science teacher at Yorktown High School in Arlington, VA. When he is not out crusading for free software, he enjoys spending time with his wife and two boys. They especially like watching movies, traveling, and eating out together.