SCALE 11x: Successful GNU-Linux labs in schools

Welcome and overview

This presentation is for a non-technical audience wanting to get the experiences of a simple end user who has been involved in setting up GNU-Linux labs in schools. The single most important point that I hope you take away from this talk is that for the Linux lab to succeed, there must be at least one teacher on staff at the school who is given at least 3 quiet hours per week ON THE CLOCK to address to the needs of the computer lab and who is passionate . Without this key ingredient, the lab will fail, whether it is Linux or Apple or Microsoft.

The second point that I would like to impress upon you is that this teacher cannot do it alone. In order for the computer lab to work, the teacher must have the support of at least one, but preferably two or three, highly technical people who can come into the school to trouble shoot the machines. It might be possible for some of these people to dial in to solve the problem, but for the most part, actual physical presence at the school is needed, or the lab will fail, whether it is Linux or Apple or Microsoft.

My background

First, I would like to give you a little background about me and Partimus. I am a sinple end user who started using GNU-Linux in 2000. I was introduced to Linux by Holden Aust, who was a systems administrator for a major asbestos litigation defense firm in San Francisco. At the time, I had my own small personal injury law firm.

I was concerned by reports that I was hearing of the growing proliferation of computer viruses. I happened to bump into Holden in the elevator one day, and from prior elevator conversations, I knew that he was a system administrator. I asked him what he felt was the best anti-virus software out there. He said “Linux.”

That conversation was the start of a long journey to migrate from Microsoft Windows to Linux. Since that time, I have been delighted by the stability of GNU-Linux, and have been amazed by the generous spirit of the free software community.

In fact, it is the social benefits of using free software that is the third most important point that I hope you take away from this talk. By helping others, you help yourself. Givers gain.

As I mentioned, I am a simple end user. I know about 20 script commands, which are the commands that you enter into the DOS-like blinking cursor interface that most simple end users never need to use, in either Linux, Apple, or Microsoft Windows. I sell insurance for a major insurer for a living. I do not make money from computers, and never have. And yet I am the Executive Director of an organization that installs computers in public schools.

I hope that fact proves to you that you do not have to be a technical person to use Linux. If you are a non-technical person, you too can use Linux easily. Linux is point and click simple. In fact, I find that Linux is easier to use than Microsoft Windows, which I am forced to use at my work.

In 2004, I wanted to share with others the benefits that I had found by moving to Linux. I felt this way because I was convinced that computers were the new cultural platform. Whoever controlled desktop computers would soon control the Internet. I knew from my training as a lawyer that it was permissible for shopping mall owners to limit public expression on their premises. I didn't want to see the Internet turned into a big shopping mall, where only Microsoft Windows users or Apple users got to post their views to the Internet. I saw Linux as a way to assure that the Internet would always be truly free and open to all.

So I wanted to see Linux spread as widely as possible, but I knew that most people were brand-sensitive, and were thus committed to either Microsoft Windows of Apple. I knew that Linux would never develop brand recognition until it was successful; but it would not be successful until it had achieved brand recognition. Linux adoption suffered from the classic chicken-and-egg problem.

It then occurred to me that I needed to find simple end users who would be willing to give Linux a try. I tried to give out Linux CDs on buses, trains, planes and public sidewalks, but I learned that retail-level Linux advocacy was fraught with the problem of end users needing more hand-holding support than I could offer. I needed to train someone who would train and support others, but here again I ran into the chicken-and-egg problem: no one would pay someone to train them to use Linux until they first accepted Linux, but they would not accept Linux until they first learned how to use it. So it occurred to me that I needed to train someone whose JOB it was to train others, someone like a …. TEACHER!

Fortunately for me, I live right across the street from a public charter school that was starting up, the KIPP San Francisco Bay Academy. I knew that they would likely be willing to listen to a pitch about Linux, because schools have been in such dire financial straights since 1978, when a law passed making it very difficult for California counties to raise taxes to fund their schools. I approached the principal and proposed that she allow me to meet with her to discuss a gratis operating system that her school would be able to use in perpetuity.

She was intrigued. We met, and she ended up agreeing to pay for a Linux lab. I installed a thin-client Linux lab in the school with a group of local Linux afficianados. The principal continued to use that lab until 2009, when the school merged with other similar schools, and the Microsoft systems administrator they hired decided to replace the Linux lab with a Microsoft Windows lab. At that time, we moved the lab to another public charter school about 3 blocks away, and continued to support individual teachers with computers small networks in their classrooms.

Partimus was founded in 2006 by co-founders Maile Urbancic and Cathy Malmrose. In 2008, the group of loosely-allied Linux volunteers who had put the computers into the KIPP San Francisco Bay Academy decided to band together under the Partimus umbrella. In 2009, the founders withdrew from active management of Partimus, and I was elected the Executive Director of the organization. We are now actively involved in 4 schools, where we either run Linux labs or smaller networks.

Who is involved with Partimus

Partimus currently is comprised of a board of five members, who tend to be the most active volunteer contributors to the Partimus projects, with the exception of our lead systems engineer, James Howard, who spearheads most of the work that is done in Partimus. There is value in understanding the composition of the Partimus board, because we are sort of a snapshot of the minimum that is needed to run a successful Linux lab.

James Howard has worked as an GNU/Linux professional for over 10 years and in the nonprofit space since 2003. His focus with Partimus has been on research and development, systems deployment, and providing technical guidance and support to Partimus clients. He is responsible for building and maintaining core technology infrastructure deployed in school and community center computer labs. His goal is to build highly stable labs that run with minimal overhead. James is a Red Hat Certified Architect but primarily works with Ubuntu Linux on Partimus projects. He also collaborates with Linux sysadmins in the open source community at large and is active in his neighborhood association.

Any successful Linux lab will need someone like James who is willing and able to address the deep technical questions that come up. James will often go over to a school by himself and address these deep issues.

Grant Bowman brings 24 years of professional Internet experience in roles such as founding Director of Silicon Valley Public Access Link (svpal.org), IT Manager, Sr. Systems Engineer and Consultant in large corporate environments like Bank of America, Adobe and SuSE Linux, Inc. as well as agile startup and open source project environments. Mr. Bowman is an elected leader of ubuntu-us-ca, a recognized Ubuntu Member, an olpcsf.org contributor, and a Fedora Ambassador. During 2011 he taught in Nairobi, Kenya with Dreamfish.

Any successful Linux will also need someone like Grant, because the primary systems administrator, like James Howard, cannot be everywhere at once. Grant is also a board member, and is highly engaged on issues of of non-profit governance. A successful Linux lab will need someone like that to attend to the keeping the organization running as an organization.

Elizabeth Krumbach, our Treasurer, has been involved with coordinating Linux community groups since 2003 and is currently a sitting member of the Ubuntu Community Council. Her work within Ubuntu primarily focuses on coursework development for in-classroom and internet-based teaching materials for learning to use Ubuntu, she is one of the leaders of the Ubuntu Women project which seeks to get more women involved with and using Ubuntu and is one of the leaders of Ubuntu California. She currently works as an Automation and Tools Engineer on the infrastructure team for the OpenStack project.

Elizabeth has technical skills equal to those of James Howard. Again, no one person can do it all, it takes a community.   Elizabeth also does much of the outreach to the world through her maintenance of the Patimus website, and through her connections on the Ubuntu community council, which is the entity which sets policy for the world's most widely used Linux distribution.

Beth Lynn Eicher, our Organizational Consultant, is also the treasurer and founder of the Ohio

LinuxFest Corporation. She brings over seven years of non-profit and volunteer management for Free Software causes. Ever since her research internship at Pittsburgh Supercomputing Center in 1999, she has been committed to the proliferation of the Linux desktop. In 2002, she deployed for the first Linux desktop classroom at Carnegie Mellon School of Computer Science. Beth Lynn's experience with making a large scale event happen such as the Ohio Linux Festival helps us understand how to run our small non-profit.

Co-founder Maile Urbancic is still involved in a very significant way by providing us with a funding resource through the sale of Ubuntu-themed necklaces and earrings manufactured by her Academia Boutique website.

Maile Urbancic, our co-founder, is a community organizer and freelance digital media professional with a passion for helping children achieve their intellectual potential. She has degrees in mathematics and educational psychology, and has taught computer literacy to low-income children for over 10 years. She is also the founder of Boutique Academia, a small business that specializes in math, science, and technology themed accessories for women. She is till involved in a very significant way by providing us with a funding resource through the sale of Ubuntu-themed necklaces and earrings manufactured by her Academia Boutique website.

Bethany Doolin, the newest addition to our board, brings five years of experience in sales and marketing to Partimus, as well as a varied background in education and technology. She enjoys being a catalyst for change. While a Web Developer at UC Berkeley, she took the lead for various technological projects in different departments. Her MSED in Educational Technology Leadership from California State University gives her the skills to promote organizational change through technology. We are looking forward to the greater understanding of the educational uses of technology that she she will bring to our board.

Jim Stockford, the founder of Systemateka, provides transportation and installation assistance. He volunteers his SUV to pick up donated hardware, and then provides assistance in setting up the machines once they have arrived on site.

My role has been making the initial contact at some of our school sites to let the schools know of the existence of our services; and then providing in-school hand-holding support to assist the teachers during class time to address basic typing and mousing issues raised by students. At one point, I was spending about 20 hours per week in schools assisting the teachers. I have also organized some of the equipment donations, and have struck up relationships with our upstream suppliers, as well as arranging for the responsible end of life recycling of the machines.

Speaking of equipment donation, where do you get the equipment?

Our equipment has come from four basic places: the Alameda County Computer Resource Center and its sister Marin County Resource Center; from Professional Computer Support, run by Dan Hernandez; from BNI-related donations; and from random sources. There is no Free Geek store in the San Francisco bay area, surprisingly, but Free Geek is in many areas of the country.

The point is the value of word of mouth networking. If you join a networking organization, such as Business Networking International, which I highly recommend, you will find that the machines will come to you. Put the word out, and the machines will come to you.

In fact, the issue is not how to get machines, but how to get quality machines. Make sure to let people know that you only want working, quality machines, unless you have a significant staff to be able to cannibalize machines and put together new machines. If you have ever been to a Free Geek store, you know that triaging and repairing machines is a huge, huge task unto itself. Don't take it on unless you have the staff and storage space to do it.

What are the benefits of using free open source software in public schools?

The benefit of using free software is that you will never again have to spend mountains of time fundraising for software. And you probably also will not have to spend time fundraising for machines to stay on the endless upgrade cycle, either, because free software runs better on older hardware than most proprietary software.

This is huge. Free open source software is not like anything else that came before it. Free software literally turns economic theory on its head. Free software is an anti-rivalrous good. Most things in the world are rivalrous goods: cars, buildings, roads, bridges, food, are all diminished by consumption. The more you have, the less there is for me.

Free software is the opposite. The more you consume free software, the more there is for me. Your consumption increases the supply. Simply by using the software, you promote it and test it. Developers want to work on popular free software projects; which improves the software; which makes more people use it; which makes more developers want to contribute to it;and so forth. Mozilla Firefox is a classic example. It grew and grew and grew in features and usage from the time it was developed until another free software browser was introduced, Google Chromium (also known as Chrome, which is the same thing as Chromium, but with a proprietary logo).

And you will get a community. The fact that you are using free software means that you will be able to find volunteers to support your school project. People like to give to givers. Givers gain. By using free software, you will be giving to others, which will incentivize people to give to you.

What are the best practices of setting up the computer labs?

Community, community, community. You have to be part of their school's community, or they won't contact you, and you will never find out what is going on until it is too late. Teachers are so busy that they often don't have time to even report problems in the lab. If you are able to spend some time in the school every week, you will much more likely find that the teachers are talking with you and communicating their victories and set-backs. They will develop a bond with you that will be essential when it comes time to decide how much time and money to allocate to your project.

By being involved with their life, you will also be able to understand their functional requirements. This is key. You need to sit down with them well in advance of the school year in which you are going to roll out the lab. Talk with them about what they want to do with the lab. Find out who the key teachers are who will be using the lab. Talk with those teachers, and find out what their expectations, experience, and needs are.

Software: make a list of functionality that they will need. Make a list of software that will support that functionality, and tell the teachers what you have to offer, and get their feedback.

Thin v. fat client: Our first labs were thin client machines. The advantage of thin clients is that there is less support. All the work lies only with one machine, more or less. The disadvantage is that thin clients are very demanding on the system, especially if you plan to edit video. You want to have as much data and software running on the individual clients as possible, especially if you are editing video.

Of course, you will want to have the status of the software determined by the server, so that you can easily push out updates to the clients from the server. It is prohibitive to have to install software on each of the clients. User authentication should also be determined by the server, so that each year, your chief sys admin can simply get a list of the names of the students who will be in each class, and then automatically populate the server with accounts for those students. It will also allow the sys admin to systematize the user passwords for each account.

One disadvantage to this system is that you will need to have assigned seating. You will want to have a unique machine for each student, and a unique account on that machine for each student. This will take a little bit of discipline, but the end result is worth it.

You will also need to establish a method for students to signal that they are having issues with the computers. Typically, students will speak up out of turn to let the teacher know that their computer is not working, and this creates a noisy disruption for the class, and takes the teacher’s attention away from the class as a whole. The ideal situation is to have someone like a computer technology assistance who can circulate through the class, answering basic mousing and browsing questions, and addressing log on issues and other basic userland questions. One of our schools required students to place colored blocks on the top of their machines: red if the question is urgent; yellow if less urgent; and green if the question is a style question.

Why is it so important to have single dedicated teacher take on responsibility for the lab?

As I mentioned at the beginning of my talk, the key to successful Linux labs lies in having at least one dedicated teacher who loves the concept of free software, who will be given 3 hours of time by the administration to work on the lab per week. Here is a list of things that teacher will need to do with those three hours:

1 Field questions from teachers using the lab;
2 Managing the scheduling of the lab, meaning which teachers use the lab when;
3 Checking loose cable connections;
4 Replacing bad mice and keyboards;
5 Establishing protocols for enforcing good behavior in the lab
6 Monitoring the maintenance of the lab, to make sure that each class cleans up the lab after use, so that the lab does not become a pigsty;
7 Investigating software and current trends in computer use, and forwarding requests for software to the systems administrator, so that the administrator can put those applications in the packages to be pushed out to the clients;the system;
8 Addressing issues with the parental control filters, such as the filtering of websites.

It is also important for one teacher to assume ownership of the lab, so that the teacher can encourage other teachers to stay on top of their students in keeping the lab in good working order. ownership of the lab. The importance of instilling a sense of ownership and pride in the lab cannot easily be exaggerated. If the students and teachers feel that the lab is of good quality, they will maintain it and use it. If the students and teachers maintain and use the lab, the sys admins and supporting volunteers will do a better job in keeping the lab in good working order.

Having a dedicated teacher who is committed to the lab will also mean that the sys admins will have an easier time getting access to the building and the lab to do their work. Also, the teacher will be help communicate to the sys admins how the staff and students view the lab and what their needs are. A dedicated teacher will also be able to spot any changes in policy that might happen from the school administration, such as proprietary software fan, who wants to make waves about having free software in the school. As I mentioned earlier, Partimus lost a computer lab in one school due to the fact that a partisan Microsoft systems administrator wanted the Linux lab out of the school. It would have been nice to be part of the process of choosing that systems administrator, so that we would have know that they school was hiring someone, so that we could have gotten a pro-Linux sys admin, rather than a pro-Microsoft sys admin.