FOSS in Mepco

Date: 05-11-2009

Arun S.A.G

Introduction

There are thousands of engineering colleges in India. Each year, a few hundred thousands of graduates and diploma holders earn their degree. They pay thousands of rupees as fees to get the best facilities. A part of this goes to buying the softwares which they use either in college lab or for doing their assignments at home.

For every software which India's engineering students use; there is an alternate Free Software program available. Free Software(http://www.gnu.org/philosophy) offers freedom, accelerated possibilities and wider vistas to our students. But in a 'talent rich, resource poor' country like India, it is also an issue of costs. Huge amounts of resources will be saved, and productivity gained, while deploying Free Software. This will not only make engineering education more cost-effective, but also more productive.

Free/Libre and Open Source Software has many other advantages such as reliability, performance and security; building up of long-term capacity within the state and country itself; the Free (as in freedom) philosophy; encouraging innovations; offering alternatives to illegal copying; throwing up many possibilities in localisation; helping students vastly by allowing them to learn from the source code; getting access to literally thousands of tools; in addition, of course, to lower costs.

Issues related to Free/Libre and Open Source Software (FLOSS) in education are outlined very well at http://www.iosn.net/education/foss-education-primer/
Other than cost benefits, Free Software has other extremely pertinent advantages. We would urge you to serious consider the following:

* No-fee licensing
* Ease of license fee management
* Better large-scale programmability
* Easier integration
* Better performance
* Development convenience
* Better support

Niranjan Rajani in a study on FLOSS in the developing world, notes, "FLOSS has a complementary and reciprocal relationship to education. One needs an educated section of the population to fulfil the full potential of FLOSS, and at the same time FLOSS helps, enhances, and complements education by providing tools to promote education."

In the case of education in computer sciences, FLOSS provides opportunities which nothing else can, as the Finland-based researcher Rajani points out:

* Unrestricted access to the source code.
* An environment of unlimited experimentation and tinkering.
* Collaboration and interaction with a community of programmers, coders and users around the world.

Action Plans to Promote FOSS

1. Creation of separate club for FOSS:

The club can be named as Mepco FOSS club. Weekly events can be conducted by this club on technical topics. Yearly, a major event can be conducted with the experts from member s of Indian Linux user group -chennai (www.ilugc.in) .

2. Installation of GNU/Linux in all the lab machines:

GNU/Linux (ubuntu, fedora)and windows can be dual booted. Install every Lab machines with Linux. Encourage Students to use GNU/Linux whenever its possible. With GNU/Linux you don't have to worry about viruses.

3. Provide training to students and faculty on GNU/Linux:

Basics on how to use GNU/Linux should be taught to students and faculty.Basics include How to install software?, How to browse files with GUI? How to get help from IRC ?, How to browse the web? How to share files? How to Play videos/audios? etc.

4. Encourage participation in prestigious Google Summer of Code(GSOC) :

Google Summer of Code is a global program that offers student developers stipends to write code for various open source software projects. please vist http://code.google.com/soc/ for more info. FYI, a student from jaya engineering college and Aandal azhagar engineering college participated in GSOC.

5. Replace proprietary software's used in Lab's with Free and Open source software:

Replace,

- Photoshop with Gimp.
- Matlab with octave/scilab.

- Windows Operating systems with Fedora/ubuntu.
- Microsoft office with Openoffice.
- Oracle with mysql.
- RationalRose with umberrlo.
- ASP/ASP.NET technologies with PHP/Ruby on rails based websites/webapplications
- ActiveDirectory with NIS/LDAP based authentication mechanisms.

6. Use Version control system in academic labs:

Students should be taught to use version control systems like git in lab itself; This really increases the passion to code;

7. Encourage Sharing software and source code:

Students should stop reinventing the wheels when doing projects; Each student should be asked to work in a Opensource project and contribute code to it;

8. Start Open source projects:

Mepco should start its own open source projects and invite student and community participation.Which happens in foriegn universities;

9. Create a local mirror of software repository of popular distributions:

Ubuntu and Fedora repositories can be downloaded/borrowed and kept in a centralized server, so that students can install opensource software in their laptops and lab computers without having to download software from the internet. (helps saving bandwidth)

10. Replace college IT infrastructure with FOSS:

College websites/mail/intranet/library management applications can be replaced with FOSS based solutions. For example: website can be redesigned with-Drupal/PHP web apps ; Library -koha;

11. Create a Website for FOSS club using Open source technology:

Create a drupal based website for FOSS club; Create an IRC help channel for FOSS club; Create a mailing list/group for FOSS club where students get to ask their doubts; Create a wiki (media wiki) for FOSS club where students and faculties write HOWTO and FAQ on a techincal topic.

12. Students who participate/do FOSS projects can be encouraged by adding their name in wiki/website : (Don't waste money with certificates)

13. Ask the students/faculty to take part in external Linux user groups like ilugc.

14. Promote/use GNU/Coding standards and Mailing list guidelines.

15. Stop paying to Microsoft.

Events in FOSS Club:

1. Introduction to Freesoftware philosophy.
2. Introduction to ubuntu.
3. Introduction to Fedora .
4. Introduction to GNU/Linux command line.
5. Introduction to GIMP.
6. PHP - Workshop (2 week event)
7. Drupal/Joomla Event.
8. Introduction to Mysql.
9. Introduction to python.
10. Introduction to emacs.
11. Introduction Vi -editor
12. Introduction to Gtk+ and Qt GUI programming;
13. Ruby on rails.
14. Conducting installtion festivals.
15. Introduction to Version control systems like git and svn.
16. Introduction to bugzilla.
17. Fedora activity day.
18. Distributions release parties.
19. Introduction to IRC.
20. Introduction to linux kernel.
21. Introduction to virtualizantion with qemu and virtual box.
22. Introduction to various Desktop environments and Window mangers.
23. Introduction to openoffice.
24. Introduction to LaTex.
25. Introduction to Functional programming languages like haskell.

Links for students to work with the open source community:

http://fedoraproject.org/join-fedora -Fedora project

http://demo.ilugc.org.in/ - Indian Linux user group -chennai