FREE YOUR COMPUTER

A Brief Overview of What Free and Open Source Software Is

Anant Narayanan
Malaviya National Institute of Technology

Blitzschlag 2008
February 3

Overview

* Philosophy
  Reasoning behind the free software movement

* Evolution
  How it became what it is today

* Why you should be bothered

* How you can get involved

In the Beginning

* Software was free
* Was not considered as a commodity that could be sold
* Most hardware vendors gave away software for free with their hardware
* Developers, Hobbyists and “Hackers” freely exchanged code and continuously made community based improvements

And then…

* Bill Gates wrote a (now famous) letter to computer hobbyists accusing them of "stealing software"

* Wanted to make software a commodity in the market, one that could be sold for profit

* The rest is history

* The question is, does software really work like other commodities?

Ask Yourself

* Physical items always cost something, because giving the item to someone else means you don't have it anymore

* What about ideas, knowledge, and by extrapolation - software?

* Do you pay your college for sharing knowledge or for physical infrastructure and amenities?

* Do you think ideas should be patentable? Do you think software patents are fair?

Round-a-about

* Universities continued to publish source code of all software created - the Berkeley Standard Distribution (BSD family) is a good example
* The commercial market, however took a different path - the one that most of you are aware of
* But now, free software is making a comeback and posing a legitimate threat to the existing software business model

Free as in Freedom

* The word free in the context of free software emphasizes freedom, not price
* The free software movement was started by Richard Stallman when he was working at the AI Laboratory at MIT
* The goal was to develop a completely free operating system for a general purpose computing device

The Two Pillars

* GNU - GNU's Not Unix
  Though it originally aimed at a completely free operating system, they completed only the user-land portion in time

* Linux
  Thankfully, Linux came along - the brainchild of Linus Torvalds. Linus decided to release Linux as free software

* And not before long, people began combining GNU with Linux to create a completely free operating system

GNU

Linux

Free Software Means...

* Freedom to run the program for any purpose
* Freedom to study how the program works and adapt it to your needs
* Freedom to redistribute copies
* Freedom to improve the program and release your improvements to the public

Open Source Software Means…

* A quicker, more efficient way of writing high-quality software
* Hundreds of people from all over the world collaborating on a single project
* The result: technically superior software for the price we all love: zero

Now it's grown

* We are fortunate to have access to free and open source software that can do everything equivalent proprietary software
* As a bonus, the free software equivalent often outperforms the proprietary equivalent
* The "Linux" operating system that you may be familiar with is actually a huge pile of several free software components that are integrated today

Examples

* Apache
* OpenOffice
* Gnome / KDE
* Pidgin
* Mozilla Firefox / Thunderbird
* and of course GNU + Linux, and many many more…
* <put your project here>

Alright, so what?

* As computer science students, FOSS brings you the unique opportunity to learn how software works in the "real world" beyond laboratory assignments
* You not only learn to program in C, you learn how a C compiler works
* You not only learn how to use an operating system, you learn how an OS is actually written
* And so on…

How FOSS is made

* Part time contributors
* Full time (paid) contributors via company sponsorship
* Guess who can become the former?

That's right, YOU!

Anyone can contribute

* Completely meritorious system
* You are respected as much as you contribute, and since anyone is free to contribute, the project is often lead by the person who makes the maximum contributions
* It doesn’t take too much effort to begin contributing, let’s take a look…

Getting in touch

* The primary method of communication amongst most developers in an open source project is the mailing list
* Subscribe, lurk, post, contribute
* Real time interaction is most often done on IRC
* Remember, these are guidelines, check out the actual project to see what rules they follow

Things you'll need

* Preferably running GNU/Linux (but BSDs, OS X, and even Windows may work!)
* Version Control Systems
* Patch creation and review tools
* Email and IRC Client
* Some time
* Coding is not the only way you can contribute to a project

Going Open Source

* Let's hope I've convinced you to open source every project you create from now…

* If not, let me convince you!

Licensing

* Fundamentally two types of licenses you can choose from:
  * Liberal (BSD-like)
  * Conservative (GPL-like)
* Let's take a look at a bunch of licenses and help you choose one for your project

QUESTIONS?

http://www.kix.in/talks/

THANK YOU!