An Introduction to FOSS

COMP8440: FOSSD
Lecture 1

What is FOSS?

- Free and Open Source Software
  - What does 'free' mean?
  - Does providing source make it Open Source?
  - Who makes FOSS? Why do they do it?
  - Is FOSS just for fanatics and hobbyists?
  - Can you make money with FOSS?
  - Does FOSS make for better software?

The many facets of FOSS

- FOSS is different things to different people
  - Some see it as a resource
  - Some see it as an ethical choice
  - Some see it as a technical choice
  - Some see it as the enemy
  - Some see it as just another type of software

What is FOSS to you?

Why give away software?

- People have many different reasons
  - Ethical choice
  - Participate in community and gain recognition
  - Reduce future maintenance burden
  - Easier/cheaper than commercial release
  - Can you think of other reasons?
- What do you potentially get back?
  - Thanks from users – sense of significance
  - Recognition of your coding skills
  - Help developing your software from other programmers
  - Job offers
  - Use of a large body of source code, much of it high quality

Free as in Freedom

"Free software" is a matter of liberty, not price. To understand the concept, you should think of "free" as in "free speech," not as in "free beer"

(Richard Stallman, The Free Software Definition)

The Four Freedoms

* The freedom to run the program, for any purpose (freedom 0).
* The freedom to study how the program works, and adapt it to your needs (freedom 1). Access to the source code is a precondition for this.
* The freedom to redistribute copies so you can help your neighbor (freedom 2).
* The freedom to improve the program, and release your improvements (and modified versions in general) to the public, so that the whole community benefits (freedom 3). Access to the source code is a precondition for this.

(Richard Stallman, The Free Software Definition)

Open Source

- Open Source definition
  - Open Source Initiative holds Open Source trademark
  - Stewards of the Open Source Definition
  - Similar in spirit to the four freedoms, but different emphasis

Let's have a look at the Open Source Definition:
http://www.opensource.org/

How big is FOSS?

- How many projects?
  - Certainly in 100s of thousands
  - Over 400,000 projects registered on one site alone (sourceforge.net)
  - Github.com claims “millions of open-source projects”
  - Similar number of developers?
- How many users?
  - Depends on what you call a 'user' !
  - Certainly many millions
- What types of products?
  - Not just for servers and desktops
  - Huge presence in embedded market


the BIZARRE CATHEDRAL
BY
MERC + CRIMPERMAN
SEE? THIS IS WHAT I LOVE ABOUT WINDOWS - THINGS JUST WORK
debian
I PLUGGED IN MY ROUTER AND BINGO - I HAVE A WEB CONNECTION! THIS IS WHAT PEOPLE WANT!!
debian
DOES HE KNOW IT'S THAT EASY ON MAC TOO?
SHALL I TELL HIM THE ROUTER RUNS ON LINUX?
debian
http://www.freesoftwaremagazine.com
(c) Copyright 2008 Ryan Cartwright CC: By-NC-SA

FOSS Community

- Developer community
  - Each FOSS project develops its own sub-culture
  - Most have common elements
- Some common factors
  - Strong leaning towards meritocracy
  - Right to fork, but reluctance to fork
  - Technical focus
  - Openness in communication

FOSS Culture

- Originally developed from “hacker” culture
- Not uniform – wide variance
  - “Free software” vs. “Open Source”
  - Degree of hostility to proprietary software
- Strong emphasis on giving credit where due
  - Names in copyright notices
  - Author and committer names in SCM
  - Credit on web sites
- Self deprecation – modesty encouraged
  - Unusually honest documentation
  - Projects getting stuck at version 0.9.x
- Values technical reality over image and appearance

Forking

- Right to fork
  - Almost a definition of FOSS
  - Necessary to guarantee independence
- Reluctance to fork
  - If a fork is needed, then something is wrong
  - A fork is usually an indication of dissatisfaction with project leadership
- Branches – temporary forks
  - A branch is a fork where the intention is to merge
  - Some branches are long lived, most are short lived
  - Unlike forks, branches are encouraged
- Is a new OS distribution a fork?
  - Depends on intention to share code

Credit

- Strong emphasis on credit
  - Taking code without credit is frowned upon
  - credit system is built into most FOSS licenses
  - big contrast with proprietary projects
- Giving credit
  - Copyright headers
  - CREDITS file or ChangeLog
  - Website credit (some automated)
  - Credit in commit messages

First Lab - TuxPaint

- Download, compile, install
  - Look carefully at the project while working with it
  - Who wrote it? Who supports it?
  - How do the docs differ from a proprietary app?
  - How would you submit a bug report?
  - How would you get help?
  - How many people use TuxPaint?
  - How active is development?
- Lab docs
  - make sure you read the lab work docs!
  - especially look at the 'build tips' page
  - Remember to join the lab mailing list
  - Ask if you get stuck!