preamble: free software

the meaning of open and free software and data formats

by Timo "Blazko" Boewing, mail@neveprise.de

Version: 0.01, 2002-03-06

This article discusses the importance of free software for our daily work and fun.

Table of contents

- 1.0 introduction
  - 1.1 definition: closed source and non-free software
  - 1.2 definition: open source and free software
- 2.0 evaluation
  - 2.1 non-free / closed source
  - 2.2 free / open source

1.0 introduction

> top of page

Most probably you are reading this with your www client, the browser, straight from the internet. You know the net: www, email, irc and much more. You have the choice of different tools to work with all these standards and are also able to create them on your own, for example a website using HTML. HTML - for example - is an open standard; it can be created with any text editor, the specifications are public and you have the choice between several clients (e.g. Mozilla, Konqueror, Galeon, Opera, Lynx or Internet Expl*rer).
> You also know that the www is very huge, nearly everyone has his pages up there. The web would not be what it is today if all would base on closed specs. These open standards have driven all, cos everyone can do it, without having to pay patent licenses and that. <p>

To anticipate a prominent prejudice: free software is not defined by its financial costs, free software can cost you money. The meaning of free software in short: free like in "freedom" and not like in "free beer".

1.1 definition: closed source and non-free software

> top of page

Non-free software is normally developed by a private enterprise or organization. There are employees that are payed for developing software or doing research. This software is solde to the public but only in binary form.
> As a consequence of not having the sources and specs, you cannot:

- modify the product to fit your purposes
- do bug fixes
- enhance security
- check it for security leaks or even troijan stuff

1.2 definition: open source and free software

> top of page

Open sources and free software is commonly developed by more than one company, it is often developed by a so-called community. However, the characteristics of free software can be described as follows:

- bases on public standards and systems
- can be received source and (almost always) additionally in binary form
- can be customized, enhanced in any way:
  - additional features
  - bug fixes
  - security improovements (kinda bugfix, uh?)
- it can be veryfied for security holes and other problems
- normally, non-free stuff cannot be derived from it
- often cost-free, but not a must

2.0 evaluation

> top of page

Following, a brief and kinds subjective argumentation on non-free vs free software.

2.1 non-free / closed source

> top of page

Supposed you have read 1.1, you know the basic characterists of non-free software. As a consequence, there are the following consequences:

- besides security affairs, closed sources are developed by a small amount of people; time is money, thus there happen few code reviews (if at all), so that performance and common code quality is often poor
- closed software and data formats can be used as a tool to gain monopolism; other developers have to pay license fees or may not use their formats at all; the public depends on the corporation developing the original
- privacy: you do not know what information exactly is exchanged to others, especially in networked environments

Cos you have few possibilities to be actively involved in development, you may post bug submissions or feature proposals and wait and pray for that features to be implemented.

Regarding the "monopolism" issue, you may throw into discussion that it is absolutely okay if a good product or standard is honored, cos many time and money flows into development and employees. But i think that especially standards are developed in public, so that a corporation should earn their berries by providing better software (the implementation part thus) and services.
> That’s like with HTML: the standard is public and worked out by several corperations and orgs but the implementations are done by the firms themselves. And it is no must to give away these products for nothing. <p>

2.2 free / open source

> top of page

Basically, the disadvantages of non-free software are the advantages of free software, so i will not mention them all.
> However, many people often say that free software is not as stable and feature-rich as their non-free opponents. I think this really applies for very young projects. But as history has shown, well-managed projects grow in quality and functionality. If professional developers are involved in the work, existing code is reviewed and bugs are fixed, and that not only by the core team developers but from people from all over the world who can get the sources and review these. <p>

Some good examples for free software / tools with no claim for completeness:

- the Mozilla project
- Linux
- GNU tools and systems
- GNOME
- The K Desktop Environment (KDE)
- NetBSD
- FreeBSD

For my very personal experience, there is no need to use non-free software. It is not about beeing stingy but it is about supporting the right ones. I do not want to support parties that think of their software as the Holy Grail and demand unfair licence fees and keep their stuff closed to establish tight customer bindings and use their products for politics instead of letting the highest quality stuff win.
> And like said, it is not about ruining the business, cos free software must not be for no cost. <p>

Commonly, closed tech also gets patented. Latest issues as seen on Slashdot revealed to what these politics can lead. On the one hand, there are some patents you can only laugh of (e.g. all-day processes that no one thinks of have been patented), on the other hand serious problems arise from patends: they hinder private developers or little firms to create own software making use of common but patented tech or data formats. As a consequence, only the bigger corporations can afford the development and consequencing financial success.
> As the past /www.w3.org/Archives/Public/www-patentpolicy-comment/" target="_blank">RAND discussion of the W3C revealed, the future of the www would be in danger if patended tech may get into important future standards. Well, just imagine you had to pay license feess for every viewed HTML page of your web site...

> top of page