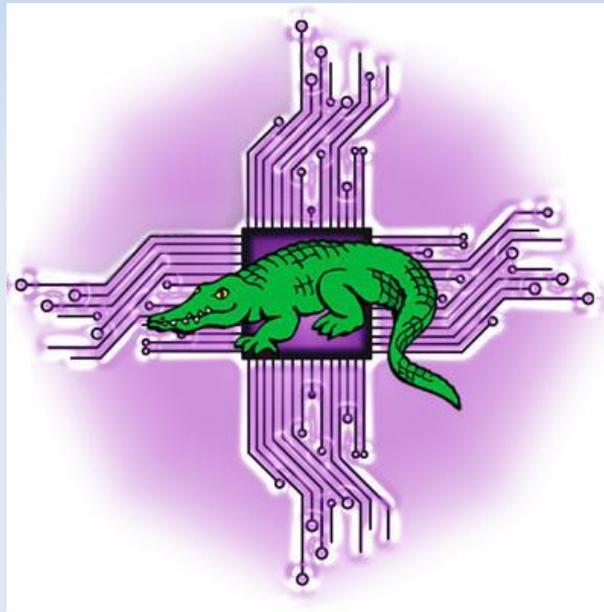


Opportunities and Issues in Free Software

Presentation to Software Freedom Day 2014



September 20, 2014

<http://levlafayette.com>

Opportunities and Issues in Free Software

1.0 What Is Freedom Anyway?

1.1 All politicians seem to claim that they support freedom - including those who wish to restrict even self-regarding rights. Freedom is used as a banner for wars (e.g., "Operation Enduring Freedom").

1.2 Freedom (from Old English) or liberty (from French and Latin) are defined as the absence of constraints. Eric Fromm (1941) in "The Fear of Freedom" drew the distinction between "freedom to" and "freedom from" which was followed up by Isaiah Berlin (1958) "Two Concepts of Liberty".

1.3 Freedom from or positive liberty represents enablement through resources and negative liberty representing absence of external restraint. The two may be in conflict.

Opportunities and Issues in Free Software

2.0 Freedom and Software

2.1 Software freedom is usually described as in "free as in speech", but also (indirectly) "free as in price". Free Software provides the opportunity to review, modify, and distribute - primarily a matter of liberty. Different licenses provide variable levels of "software freedom" (e.g., GPL, BSD, public domain, proprietary etc).

2.2 Software freedom is a particular challenge to software patents. Patents are a temporary monopoly to a technology that is meant to encourage investment for public disclosure. Patents are usually applied to the physical embodiment of processes. The main problem with software patents is that they restrict, in software, the implementation of a process, method, or idea; all of which (Church-Turing thesis) is a function on the natural numbers.

2.3 As a result software patents *reduce* freedom and impede computer technology. In addition to 'patent trolls' that misuse the legal system, there are examples of strategic patents claims that have weakened development in audio development, graphics, compression algorithms, register allocation, and memory checking. This is *bad* for scientific research, and economic wealth.

Opportunities and Issues in Free Software

3.0 Free Software Is Winning

3.1 Free software however is winning the majority of computer share. Android Linux has 78% of the market (IDC, 2013Q4) on mobile phones, Linux systems dominate Internet servers with 65% share (W3Techs Feb 2014), and supercomputers Linux has an overwhelming majority with 97% (Top500, June 2014). When Mark Shuttleworth launched the Ubuntu distribution in 2004 he also introduced "bug #1" for resolution: "Microsoft has a majority market share". In 2013 this bug was closed. However in the laptop and desktop market, MS-Windows has an overwhelming market share as of May 2014 (c91%, OS X 7%, Linux 2%) (Net Applications, c91%, OS X 7%, Linux 2%)

3.2 Free software is winning for a number of reasons: (a) It is derived and mostly compatible with Unix operating systems, dating back to 1972, itself based on Multics from 1964 - many years of development (b) it is technically superior on the metrics of scalability, stability, security, and efficiency (c) software depends on short-term inspiration and long-term success depends on a thorough and critical process of open peer review (d) as a result there is a powerful community of enthusiasts, professionals, and supportive businesses.

3.3 Computational devices are becoming more powerful (roughly doubling in processing and storage capacity every two years). Computing is becoming increasingly ubiquitous; "the Internet of things" 30 billion devices connected by 2020 (ABI Research).

Opportunities and Issues in Free Software

4.0 Issues Confronting Free Software

4.1 Reducing competitive threats as a path to profit reduces productivity as it requires monopolistic behaviour. Monopolistic companies can, and do, provide damaged goods which deliberately reduce overall utility and overall production. It is easier to "make money" through monopolistic behaviour as a form of 'rent-seeking'. Therefore some software companies, rather than providing a more competitive product, will seek to reduce competition.

4.2 Free software does suffer from a free rider problem. There is limited economic incentive to contribute to the development of free software, but significant economic incentive to draw from the common resource. This would not be a problem except software development is dynamic.