Differences between open source and free software

Akira Taguchi
Department of Computer Science
University of Helsinki
Finland
akirahattivatti@gmail.com ORCID: 0009-0007-0343-4540

Abstract—Open-source software development is praised by an increasing amount of big technology companies. The outcome of open-source development is however problematic to the developers themselves. Free software attempts to solve these problems. This paper examines the differences between open source and free software definitions and their usage. The results show some examples of the differences. The paper then attempts to go through some of the phenoma behind the unclear differences. Both open source and free software share some of the same development methods. Free software however doesn't share all of the same sustainability issues open source does.

Index Terms—open-source software, open source, free software, software freedom, libre software, proprietary software, closed source software, copyleft

I. Introduction

Open-source software development has seen an increasing interest in the recent years [1]. Open source gives all kinds of different benefits ranging from enhanced security to extra labor force from outside the company. The benefits also include faster feedback loop between end-users and the developers, increase in agility and decrease in costs because every developer are able to make bug fixes, feature requests and many other types of contributions regardless of their employment relation to the company. Enhanced security is usually justified by Linus' law, formulated by Eric Raymond: "given enough eyeballs, all bugs are shallow" [2]. Open-source also enables companies to maintain a closed-source, commercialized version of that same open-source software and sell it with some extra features to the customers, allowing the company to profit even more from the open-source approach. The list of benefits is long and growing on-par with the technology advancements. It is no wonder the interest in open-source software development has grown with the recent years.

It's most known opposer, closed-source software, comes with it's own benefits like increased revenue because a competing company cannot create a competitive copy with the same source code. Closed-source, or proprietary software is argued, against Linus' law, to have a higher rate of bug uncoverage due to an in-depth review by fewer people as opposed to in open-source approach [3]. This is often referred to as a higher level of security in proprietary software.

Free software is the least known opposer to both open source and proprietary software. It was first defined by the Free Software Foundation, or the FSF in 1986: "The word "free" in our name does not refer to price; it refers to freedom. First, the freedom to copy a program and redistribute it to your neighbors, so that they can use it as well as you. Second, the freedom to change a program, so that you can control it instead of it controlling you; for this, the source code must be made available to you." [4]. The difference between proprietary software and free software might be more obvious than the difference between open source and free software. According to the second freedom in the definition of free software "the source code must be made available to you" open source doesn't always go down this path. Because open-source software can be made closed-source for commercial or some other purposes, the FSF does not see open source as a form of free software [5].

Free software has seen declined interest in the last four decades [6]. Free software prohibits closed-source derivatives from the free software's source code. Because most companies profit from their commercial, closed-source versions, companies like Microsoft have taken actions to demonize free software as a whole throughout the years [7]. Free Software Foundation's head figure, Richard Matthew Stallman is also being demonized with hate speech [8]. Aside from systematic demonization of free software it has also been criticized as failing on a practical, -ideological, economical and political level [9].

The objective of this paper is to find out how the differences between open source and free software are perceived in the academic field. This consists of mostly academic papers in academic journals like IEEE, ACM or other journals found from Google Scholar. The objective of the paper also includes the possible reasoning behind the perception of the two aforementioned approaches to software development. Legal-heavy parts of the open source and free software definitions are outside the scope of this paper.

Next the paper describes the methods used to find out the perceptions in the difference of open source and free software in the academic field. After describing the methods we take a look at the results. Finally the paper discusses the phenomena around the research scope. This includes nexts steps in concrete action, the new term "openwashing" and further research.

II. Methods

The papers are gathered from IEEE Xplore, ACM Digital Library and Google Scholar. The papers are picked based on the two factors: numbers cited and keyword accuracy. The first one is self explanatory but the latter needs explanation. The keyword accuracy is higher if the title or other contents of the paper contain the words "free", "libre" or "FOSS" for example. From the author's previous experience the keyword "open source" appears always with the a paper discussing free software. The paper will present discoveries of other synonyms for free software.

A. Papers from IEEE

TABLE I
Papers from IEEE

Title	Number of citations*	Keywords	Publishe
Free/Libre open-source software development: What we know and what we do not know	269	free, libre, open-source	2008

*Number of citation checked November 11th 2023.

IEEE, ACM and Google Scholar
Difference (keltytwobits)
Open Source Definition
Free software definition

III. Results

Academics do not know the difference. Why? Demonization of free software

IV. Discussion

Openwashing

GPL is not enough (sfc2023 gpl-rhel-analysis, ferguson 2006 gpl debug)

So what: better license, copyleft companies, increase active discussion, cc by

Further research

References

[1] S. Raghunathan, A. Prasad, B. Mishra, and H. Chang, "Open source versus closed source: software quality in monopoly and competitive markets," IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 35, no. 6, 2005.

[2] E. Raymond, "The cathedral and the bazaar," Knowledge, Technology & Policy, vol. 12, no. 3, pp. 23–49, 1999.

[3] D. LeBlanc and M. Howard, Writing secure code. Pearson Education, 2002.

[4] Free Software Foundation, "GNU's bulletin," https://www.gnu.org/bulletins/bull1.txt, 1986, [Online; accessed 24-October-2023].

[5] R. Stallman, Free software, free society: Selected essays of Richard M. Stallman. lulu.com, 2002.

[6] Wikipedia contributors, "History of free and open-source software," https://en.wikipedia.org/w/index.php?title=History_of_free_and_open-source_software&oldid=1162363354, 2023, [Online; accessed 16-October-2023].

[7] C. Mundie, "Speech transcript - craig mundie, the new york university stern school of business," https://web.archive.org/web/20231018013917/https://news.microsoft.com/speeches/speech-transcript-craig-mundie-the-new-york-university-stern-school-of-business/, 2001.

[8] "Debunking false accusations against richard stallman," https://web.archive.org/web/20230624091156/https://stallmansupport.org/debunking-false-accusations-against-richard-stallman.html, 2023.

[9] "A critique of free software," https://web.archive.org/web/20230325192431/https://www.paritybit.ca/a-critique-of-free-software, 2021.