Faculty of Informatics
Masaryk University

IV064

Information Society

Open access to source code, and results, open systems

Adrián Tóth (491322)

November 6, 2019

Contents

1 The Open Access **2**

2 Success of Open Source Projects **3**

2.1 The Open Source Community 3

2.2 Source Code Availability and Hosting 3

2.3 Collaboration and Contribution 4

2.4 Feedback and Support 4

3 Open Source Consequences **4**

3.1 Influencing Impact of Open Source 4

3.2 Threats, Weaknesses and Vulnerabilities 4

3.3 Pros and Cons 4

4 The Open Systems **4**

5 My Notes **4**

6 References **5**

1 The Open Access

Nowadays, the open access term is used in many ways. Firstly, let us declare the meaning of this expression within the context of IT to better understand other things that will be mentioned later. You may heard phrases like '*Open access to source code*' as well as '*Open source software*'. Accordingly, this phrases may also appear under the '*free*' acronym, which is not the proper notion. There is an article [12] about the contrast of these two terms, which describes and explains the differences between them. All of these names indicates one attribute (model, philosophy or methodology) of the software development - the right of free access to the source code of a software to anyone known as the open source philosophy. This right stands for the free way of software source code free usage, inspection, modification and distribution that may be restricted by further open source licenses such as GPL – GNU General Public License, which will be described later. To clarify the veritable meaning of the open source let us specify the correct definition itself - the open source definition.

Definition 1.1. [2, 10, 13]

> Open source does not just mean access to the source code. The distribution terms of an open source software must comply with the following criteria:
>
> - Free Redistribution
> - Inclusion of Source Code
> - Inclusion of Derived Works
> - Integrity of The Author's Source Code
> - No Discrimination Against Persons or Groups
> - No Discrimination Against Fields of Endeavor
> - Distribution of License
> - License Must Not Be Specific to a Product
> - License Must Not Restrict Other Software
> - License Must Be Technology–Neutral

As has been noted above, the definition is denoting a specific software type which is made freely available with respect to modification and distribution to anyone including license restrictions and non-discriminatory rules. At the present time, this influencing fact has affected not just the software itself but the development as well. Open source projects, products, or initiatives embrace and celebrate principles of open exchange, collaborative participation, rapid prototyping, transparency, meritocracy, and community-oriented development [11]. The strongest side of the open source is its community together with the collaboration power. The open source community is diverse and highly motivated [9].

Related to open access, you might probably notice abbreviation such as *FOSS* or *FLOSS*. *FOSS* stands for *Free and open–source software* while *FLOSS* means *Free/Libre and Open Source Software*. These software projects are a form of commons where individuals work collectively to produce software that is a public, rather than a private, good [3].

At the present time, there is a countless amount of open source projects. Many of them become popular quite quickly and has drawn the interest of academia and industry. For students open source software is an arena for learning, and the industry needs software engineers acquainted with the theoretical and practical aspects of open source software development [6]. Several projects are running for decades, some of them are just at the beginning of their dawn. As an illustration, the Table 1 below contains some of the well known and still running projects under the terms of open source.

Project	Year	Author(s)
Unix OS	1969/1970	Ken Thompson, Dennis Ritchie & others
GNU Project	1983	Richard Stallman
Linux Kernel	1991	Linus Torvalds

Table 1: The oldest and still existing open source projects [5, 4, 14].

In contrast to the open source, its antonym is known as proprietary sometimes referred as non-free. In conclusion, it indicates copyright restrictions that prevent unrestricted distribution or reuse of the software. The pros and cons of these two types of software philosophy will be explained later in the following sections.

2 Success of Open Source Projects

In the face of the fact, that nowadays it is very hard to be successful without any reward for the given effort, open source projects have their characteristic attribute that they are invincible in this field. The roots of success lie in the kindness and helpfulness of people who are developing software for the others as a hobby and for free. These developers, representing and formulating the community, are the crucial heroes for the success. As open source continues to prosper, the topic '*why is open source successful*' becomes the subject of more and more academic research [1].

2.1 The Open Source Community

Who and what is the open source community? Sometimes the community is referred as anybody, sometimes it is restricted to the group of developers only. It depends, in some cases from the license itself too. We may consider, that the community is a group of people who are interested in the open source software, not just the developers of that software. Thereby, anybody related or connected to the usage, development, improvement or enhancement of the open source software can be considered as a member of the open source community. Because of the increasing interest for a software by absolutely anyone, the community has uncommonly fast grow, which has a positive affect on the software itself.

The community is strong, truly powerful. Together, collaboratively the community members are co-creating a single masterpiece of work, while they share their own intellectual properties as a subparts of the overall outcome. Many people are not just enhancing the pure software, but also their own skills. Learning new, yet unknown, things from the others can be considered as a benefit of being an open source community member. Do not forget to note that the community is worldwide in nature, so projects with larger scale are developed internationally such as projects listed in Table 1.

2.2 Source Code Availability and Hosting

To allow collaboration and contribution it is necessary to share the source code in some way as it is stated in the open source definition. In most cases these open source projects are using version control repositories or they may use something else, which are publicly hosted to make it available. Available version control repository hostings such as GitHub[1], GitLab[2] or Bitbucket[3] are the most common ones. Their advantage is not only the management of change in the source code but also saving metadata such as related author, date of change and so on. Metadata might include information such as source code location, contributors, license, references and how to cite the software [8]. Using a properly

[1] github.com

[2] gitlab.com

[3] bitbucket.org

configured version control repository with open access allows to save the derived works, authors of each modification and further requirements easily.

2.3 Collaboration and Contribution

We have already described the open source community and the availability of the source code and its derivations in the sections above where we also slightly mentioned the community contribution. Now, let us take a closer look at the pure contribution and how it is performed by the members of the community.

To increase the intelligibility, many of open source project are using common conventions developed by the time. Many open source best practices fly in the face of traditional software development methods [7]. There are many books, articles and tutorials about the best practices on how to make better and easier collaboration and reduce mistakes by preventing common faults and errors. The strategy of open source development is still evolving by the time. Nowadays, due to the increase of the contribution count, software releases are faster and faster. The proprietary software projects will never reach the same speed of release announcement as the open source projects have.

2.4 Feedback and Support

3 Open Source Consequences

3.1 Influencing Impact of Open Source

3.2 Threats, Weaknesses and Vulnerabilities

3.3 Pros and Cons

4 The Open Systems

TODO

5 My Notes

TODOREMOVE!

- How Successful Open Source Projects Work, and How and Why to Introduce Students to the Open Source World - https://ieeexplore.ieee.org/document/4556977
- Free and Open Source Software - https://ieeexplore.ieee.org/document/5662568
- Licenses of Open Source Software and their Economic Values - https://ieeexplore.ieee.org/document/1620054
- 1 - Open Source Software: Definitions and History - https://www.sciencedirect.com/science/article/pii/B9781555583200500027
- B - The Open Source Definition - https://www.sciencedirect.com/science/article/pii/B9781555583200500167

6 References

[1] DeKoenigsberg, G.: *How Successful Open Source Projects Work, and How and Why to Introduce Students to the Open Source World*. 2008. pp. 274–276. doi:10.1109/CSEET.2008.42. [Online; Accessed: 2019-10-28].
Retrieved from: https://ieeexplore.ieee.org/document/4556977

[2] DiBona, C.; Ockman, S.; Stone, M.; et al.: *Open Sources: Voices from the Open Source Revolution*. 1999. [Online; Accessed: 2019-10-27].
Retrieved from: https://www.oreilly.com/openbook/opensources/book/

[3] English, R.; Schweik, C. M.: *Identifying Success and Tragedy of FLOSS Commons: A Preliminary Classification of Sourceforge.net Projects*. In *First International Workshop on Emerging Trends in FLOSS Research and Development (FLOSS'07: ICSE Workshops 2007)*. May 2007. pp. 11–11. doi:10.1109/FLOSS.2007.9. [Online; Accessed: 2019-11-06].
Retrieved from: https://ieeexplore.ieee.org/document/4273082

[4] Free Software Foundation, Inc.: *Initial Announcement*. 2014-04-12. [Online; Accessed: 2019-10-28].
Retrieved from: https://www.gnu.org/gnu/initial-announcement.en.html

[5] Israeli, A.; Feitelson, D. G.: *The Linux kernel as a case study in software evolution*. vol. 83, no. 3. 2010: pp. 485 – 501. ISSN 0164-1212. [Online; Accessed: 2019-10-28].
Retrieved from: http://www.sciencedirect.com/science/article/pii/S0164121209002519

[6] Jaccheri, L.; Osterlie, T.: *Open Source Software: A Source of Possibilities for Software Engineering Education and Empirical Software Engineering*. In *First International Workshop on Emerging Trends in FLOSS Research and Development (FLOSS'07: ICSE Workshops 2007)*. May 2007. pp. 5–5. doi:10.1109/FLOSS.2007.12. [Online; Accessed: 2019-11-06].
Retrieved from: https://ieeexplore.ieee.org/document/4273076

[7] Katsamakas, E.; Xin, M.: *Open source adoption strategy. Electronic Commerce Research and Applications*. vol. 36. 2019: page 100872. ISSN 1567-4223. doi:10.1016/j.elerap.2019.100872. [Online; Accessed: 2019-11-04].
Retrieved from: https://www.sciencedirect.com/science/article/pii/S1567422319300493

[8] Kuzak, M.; Harrow, J.; Jimenez, R. C.; et al.: *Lesson Development for Open Source Software Best Practices Adoption*. In *2018 IEEE 14th International Conference on e-Science (e-Science)*. Oct 2018. pp. 19–20. doi:10.1109/eScience.2018.00011. [Online; Accessed: 2019-11-04].
Retrieved from: https://ieeexplore.ieee.org/document/8588630

[9] Opensource.org: *Community and Collaboration*. [Online; Accessed: 2019-10-28].
Retrieved from: https://opensource.org/community

[10] Opensource.org: *The Open Source Definition*. 2007-03-22. [Online; Accessed: 2019-10-27].
Retrieved from: https://opensource.org/docs/osd

[11] Red Hat, Inc.: *What is open source?* 2019. [Online; Accessed: 2019-10-27].
Retrieved from: https://opensource.com/resources/what-open-source

[12] Richard Stallman: *Why Open Source misses the point of Free Software*. 2019-04-28. [Online; Accessed: 2019-10-28].
Retrieved from: https://www.gnu.org/philosophy/open-source-misses-the-point.en.html

[13] Software in the Public Interest, Inc. and others: *Debian Social Contract*. 2019-07-16. [Online; Accessed: 2019-10-27].
Retrieved from: https://www.debian.org/social_contract.en.html#guidelines

[14] Spinellis, D.: *Half-century of Unix: History, Preservation, and Lessons Learned*. MSR '17. IEEE Press. 2017. ISBN 978-1-5386-1544-7. pp. 1–1. doi:10.1109/MSR.2017.1. [Online; Accessed: 2019-10-28].
Retrieved from: https://doi.org/10.1109/MSR.2017.1