


RESEARCH

Open Access



Support mechanisms provided by FLOSS foundations and other entities

Juho Lindman^{1*}  and Imed Hammouda^{2,3}

Abstract

Foundations function as a vital institutional support infrastructure for many of the most successful open-source projects, but the different roles played by these support entities are understudied in Free/Libre and Open Source Software (FLOSS) research. Drawing on Open Hub (formerly known as Ohloh) data, this paper empirically investigates how these entities support projects and interact with other projects. This study was conducted using the Theoretical Saturation Grounded Theory approach given the large volume of data on hand. The findings are synthesized as a taxonomy of support entities, a categorization of support mechanisms and a set of dynamics of the interactions between different FLOSS support entities.

Keywords: FLOSS, Open source, Open source foundations

1 Introduction

Traditionally, economic activities have been organized into organizational hierarchies or markets that follow price signals [4]. The third mode of production is called commons-based peer production, which brings about dramatic changes in economic organizing [3]. Relations between FLOSS developers, projects and their support entities, such as foundations, are of key importance in understanding this emerging landscape. The continued success of Free/Libre Open Source Software (FLOSS) is thus attributable to the evolution of its projects and contributors [2, 9, 13], but more research is needed concerning the entities¹ that support FLOSS, such as foundations [15]. These entities support individual FLOSS projects in different ways, but their dynamics remain an understudied phenomenon. In addition, interactions between these entities and developers pose several questions for further study.

We address these gaps in our empirical investigation of how these entities support and interact based on Open Hub² data. In particular, our research ques-

tions and related contributions can be formulated as follows:

RQ1. *What defines a FLOSS support entity?* As a starting point, we study the initial characteristics of the support entities as a step toward formulating theories in the area. We present such characteristics as a taxonomy of different attributes.

RQ2. *How do FLOSS support entities support FLOSS projects?* We focus on identifying the support mechanisms provided by support entities in order to maintain the sustainability and viability of FLOSS projects in the ecosystem. The identified support mechanisms help us reflect on the role of the support entities in the ecosystem.

RQ3. *How do FLOSS support entities interact between each other?* We investigate the different ways FLOSS support entities interact and collaborate in order to understand the dynamics of the FLOSS ecosystem. Our findings reveal traces of a complex interplay in the FLOSS ecosystem. We argue that the study results are of relevance to both researchers and practitioners.

* Correspondence: juho.lindman@ait.gu.se

¹Applied IT, University of Gothenburg, Gothenburg, Sweden
Full list of author information is available at the end of the article

The paper is organized as follows: Section 2 provides a background on the related research. Section 3 presents the methodological details of data collection and analysis. The findings are reported and discussed in Section 4. Finally, the limitations and validity threats are also discussed at the end of Section 4.

2 Literature

FLOSS foundations support their community projects in many different ways. We explore FLOSS support entities (“Foundations,” “Organizations”) and their relationships to the projects with which they are affiliated. Generally, these entities are an association of developers from the different companies and organizations supporting the development of specific FLOSS projects.

2.1 Support entities and FLOSS developers

It is noteworthy that the role of these support entities is not like that of a manager; rather, it is like that of a steward supporting community-driven projects. For example, Riehle [15] demonstrates how FLOSS support entities manage and ensure the long-term survival of their projects. Examples of such support entities include ASF, Linux Foundation and Eclipse Foundation.

Understanding the role of these support entities helps us better understand the institutional arrangements that complement individual development efforts and provide infrastructural stability to the communities in charge of the actual software development. These entities are linked to FLOSS projects and facilitate their implementation, providing various kinds of financial support and legal protection [15]. This makes the projects slightly less dependent on volunteer development efforts and enhances their credibility. Tasks carried out in the support entities can include infrastructural and back office work such as maintenance of member databases and financial reporting [14].

Riehle and Berschneider [14] investigate the different options faced by projects when they mature: whether to try to join an existing support entity or to start a novel support entity. There is also the option of not organizing activities through a support entity. Finding a match that suits the community goals is critical to securing long-term project viability [14]. This founding of support entities can also be characterized as one stage in the development of the life cycle of a FLOSS community [10, 11]. For example, de Laat [10] discusses the development of governance in terms of three stages: 1) spontaneous, 2) internal

and 3) external. In this kind of situation—i.e., when a support entity is seen as a “shell” that protects the development community [8] or is legally “guarding the commons” [11]—the boundary between the FLOSS community and FLOSS support entity becomes analytically slightly more difficult to demarcate separately. In this study, we try to discuss community governance and the entities’ activities separately, especially focusing on the activities carried out by the support entities.

2.2 Support entities and FLOSS companies

Outside FLOSS research, it is still surprisingly common to encounter a misunderstanding that posits that FLOSS foundations and projects are somehow automatically and essentially antibusiness. Research shows that many different kinds of company–community relationships exist, ranging from firm-driven development to more decentralized approaches [5, 12]. On the FLOSS company side, the classification of FLOSS businesses as hybrid companies, pure-play FLOSS companies and FLOSS process-oriented companies is widely used [7].

Schaarschmidt et al. [17] provide a typology that divides projects into firm vs. community-initiated and single-vendor vs. multivendor to discuss different governance mechanisms used by the companies. It is worth noting that even if developers were employed by software companies, FLOSS communities would rely on indirect control, voluntary task assignments and lateral authority [6]. One example is the Linux kernel community, which is driven by a group of core developers who earn their salaries in commercial software companies [1].

In contrast to development-driven communities, FLOSS support entities have bylaws, membership roles and formal positions often held by entrusted representatives of companies that support particular projects, although there exist large variations on how exactly support entities are organized. While these support entities are stewards and do not decide the road maps of development, providing these formal positions also seems to be attractive to the supporting companies; for example, providing access to a project and showing belonging to the wider community of FLOSS proponents.

2.3 Support entities and FLOSS projects

There is considerable variation in the services the support entities provide. FLOSS support entities may have other responsibilities related to the hosting and management of FLOSS projects. Responsibilities include: (i)

organizing community projects (ii) marketing, (iii) managing intellectual property (IP) rights and (iv) setting strategic directions. Support entities may serve as legal representatives and provide a means to enforce the protection of community-generated content using IP legislation [13]. Some projects require contributor agreements that are managed by support entities.

Entities may also serve as a platform for commercial companies that, for legal and institutional reasons, tend to prefer working with organizations rather than individual developers.

Occasionally, for commercial organizations, support entities build consensus by offering to impact the development road maps or voice their concerns about the direction of the development or legal issues surrounding specific community decisions. Foundations may also accept monetary donations and use these funds to support development efforts or maintain the necessary infrastructure. Support entities may also use funds to not only support development efforts and infrastructure directly but also for community building (e.g., funding conferences, recruitment of new members etc.).

3 Methodology

In this paper, we empirically investigate the FLOSS environment, the role of support entities and the relationships between support entities. This study was conducted using the Theoretical Saturation Grounded Theory approach, which is a form of a qualitative data collection and data-analysis methodology [8]. The main purpose of this research approach is to build toward a classification framework through a continuous comparative analysis of qualitative data collected by the sampling process. This approach supports data collection along with the data-analysis process. This research approach is also used to assess any sort of patterns (or) variations in a research area.

The Open Hub data repository (formerly known as Ohloh) was used as a primary data source for this study. This source holds key information about the support organizations concerning their sectors, development focuses, licensing policies, membership types and structure. The data repository also holds other information, such as projects and committers, which can be used to determine the relationships between support entities and projects. Open Hub can be accessed using the API.³ We used this repository to identify the relationship a support entity may have with another entity as well as a support entity's portfolio projects. We used Open Hub data from all FLOSS support entities that host at least one project. We have included an example from Mozilla Foundation in Appendix 1.

Support entity websites are another main source of data. These websites contain key information about support and services, incubation processes, project governance, maintenance, project-development practices, IP management, license-agreement policies, hosting services and so on. This information was used to map how the entities provide support for projects.

We used a Java program to parse the API data from the XML data format to plain text and then stored them in a database. We collected data from 88 FLOSS support entities. We have included an example of XML data in Appendix 2. Saturation point was considered to be reached when 20 sampling cases were investigated without obtaining new findings from the data.

4 Results and discussion

In this section, we first present our findings related to the three research questions and then discuss their implications. In section 4.3, we discuss the implications and limitations.

4.1 Results

This section on the results is organized based on the three research questions of the paper. First we discuss the definitions of a support entity, discuss how these entities support FLOSS projects and then finally discuss how support entities interact with each other.

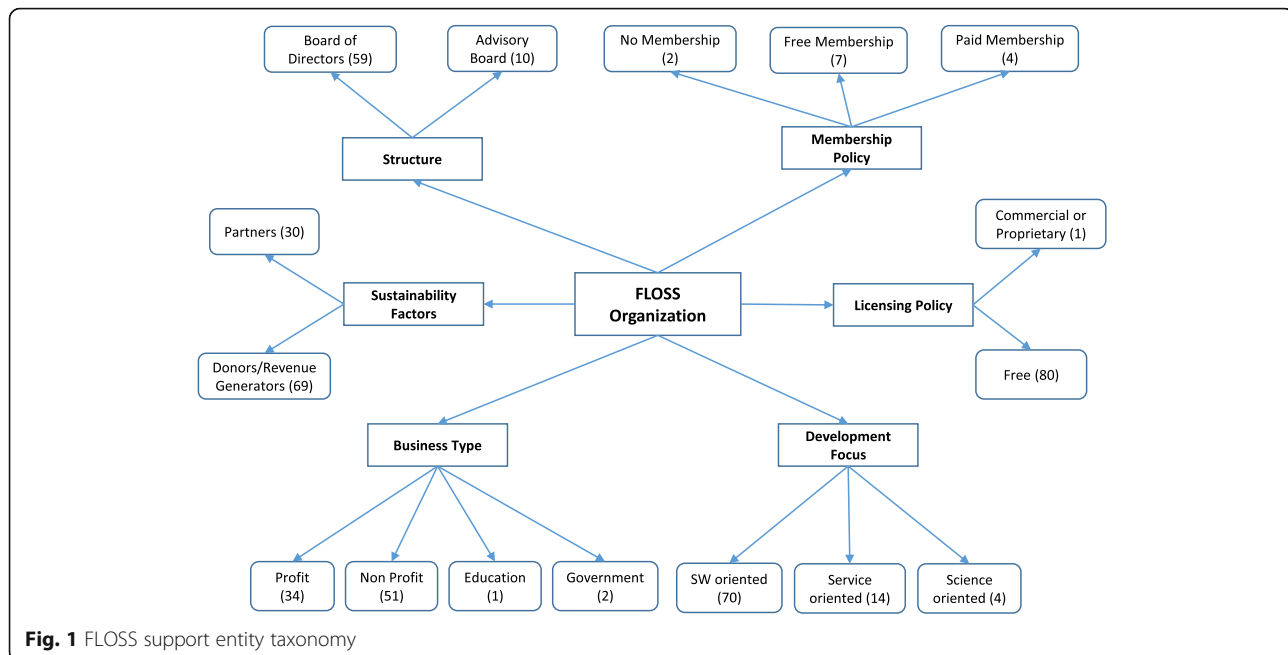
4.1.1 RQ1. What defines a FLOSS support entity?

To answer our research question RQ1, we report the characteristics of the different support entities by grouping these entities according to a taxonomic classification. The taxonomy outlines support entities' activities and highlights some issues related to legal, financial and governance topics. This is depicted in Fig. 1. The numbers in Fig. 1 denote the number of cases identified for each characteristic.

As mentioned earlier, we collected data from 88 FLOSS support entities. However, some entities did not have explicit policies on certain topics. For instance, we could identify a clear organization structure for 60 entities only. In contrast, we were able to find out about the organization business type of all 88 entities.

In order to explain how we derived the taxonomy and variations on the data, we next provide two examples of the comparison in the theoretical sampling process and cases.

Example 1: *Case 1: Apache Software Foundation (ASF) is a nonprofit organization that is primarily sustained by donors (for example, volunteers and*



companies). ASF is governed by the board of directors, mostly deals with software-related projects, hosts free software license projects only and maintains free membership policy. Case 2: Wikimedia Foundation is also a nonprofit organization that is sustained by both donors and partners, unlike ASF, which is sustained only by donors. Wikimedia is governed by an advisory board instead of a board of directors. Wikimedia hosts only free software license projects, similar to ASF.

By comparing Case 1 and Case 2, we thus analyze that Case 1 and Case 2 have a similar organization business type and have slight variations in the form of the governance structure and sustainability factors.

Example 2: Case 7: Twitter is a profit organization that primarily focuses its development on service-related projects. Case 8: Los Alamos National Lab is a government organization that primarily focuses its development on science-related projects.

By comparing Case 7 and Case 8, we can notice that Case 7 and Case 8 have a different business type and organization development focus.

Following our criterion for saturation point, we did not find any new emerging attributes from Case 56 to Case 75. We thus decided to end our theoretical sampling process at that point. The resulting figure is thus Fig. 1.

A **Profit (or) commercial FLOSS** support entity generates revenue via sales of products, services and solutions. Such entities collaborate with different corporations and technical partners. In contrast,

nonprofit foundations are primarily sustained through volunteer donations. Such foundations collaborate with external companies, educational institutions and other stakeholders to obtain funds to support projects. Most of these organizations are also primarily governed by a board of directors (BOD). **Government FLOSS** mostly consists of science-related projects. The funding for such projects mainly comes from public sources. **Education FLOSS** primarily comprises educational institutions. These support entities focus, for the most part, on providing education to the general public and are sustained through donations from public sources and student fees.

Organizations list different development focuses. Options include **S/W oriented**, **service oriented** and **science oriented**. Most service-oriented support entities are of the for-profit type, while science-oriented support entities are often government based and educational.

FLOSS support entities may support projects that use either **free software license** projects or **commercial or proprietary software license** projects. A free software license grants the user a piece of the software extensive rights to modify and redistribute that software. A commercial or proprietary software license is produced for sale or to serve commercial purposes.

FLOSS support entities evolve through different kinds of **donors and revenue generators** and **partners**, such as volunteers, corporations, open-source

organizations, software products, government agencies, educational institutions and investors.

FLOSS support entities are governed by two different governance modes: a BOD and an **advisory Board** (AB). A BOD has decision-making authority and is responsible for governing the support entity. BOD committee roles may include founder, investor and director. In contrast, an AB does not have decision-making authority, and it is only responsible for assisting or giving advice within an organization. AB committee members can have roles such as senior manager, executive, volunteer and so on.

FLOSS support entities have different types of membership schemes. The **no membership** (NM) type has no members within the support entity. The **free membership** (FM) type allows any member to join without a membership fee. The **paid membership** (PM) type allows only paid members to take part.

4.1.2 RQ2. How do support entities support FLOSS projects?

To answer our research question RQ2, we explored how entities support FLOSS projects. We grouped our findings as (described in detail in Table 1 below): services, incubation process, project governance, project maintenance, IP, project acceptance and hosting services. Table 1 summarizes the key support mechanisms.

We used the same saturation point criterion to identify the support mechanisms. The following set of cases explains our data analysis process.

Case 1: Apache Software Foundation (ASF) provides various support and services to its foundation projects. New projects can be created only when they go through an incubation process. ASF is one of the few organizations that assigns a single project management committee (PMC) to govern its foundation projects.

Table 1 How support entities support FLOSS projects

Support	Explanation
Services	FLOSS support entities can provide legal, financial and consulting services to their projects. Support entities can provide tools and offer advice on how to raise funds. Support entities can also provide essential support on how to protect projects' IP and financial contributions, and they can limit the legal exposure of an individual contributor to portfolio projects; examples include ASF and Gentoo.
Incubation Process	Support entities have different guidelines regarding how a portfolio project can be created. Many support entities require an incubation process. Created projects enter the incubation process. Some processes are mandatory quality-control mechanisms. In some FLOSS support entities, incubation processes are used to create new versions of the existing projects and not for creating new projects. Some FLOSS projects start with a preexisting code before going through the incubation process. These incubation processes are useful for new projects with respect to learning community norms and processes. Projects in incubation are monitored by designated mentors. There are some variations: <ul style="list-style-type: none"> • The incubation process is only used to create the new versions of an existing project and not for creating entirely new projects; for example, the Wikimedia Foundation. • Individuals are responsible for the creation of projects. However, in the case of the Eclipse Foundation, a project can be started/created with a preexisting code. • A project can be started/created by anyone with the necessary skills.
Project Governance	Support entities may assign a project management committee (PMC) consisting of people to govern or manage projects and subprojects. Support entity mentors usually work with the PMC to facilitate a project's evolution; examples include ASF and Tryton.
Project Maintenance	Project data are maintained by either a PMC or by projects themselves (e.g., ASF).
Intellectual Property (IP)	FLOSS support entities' IP management enables software developers from different organizations to participate in software development. Tried-and-true practices exist to support software IP management and to foster a growing community. FLOSS support entities protect a developer's contribution to portfolio projects when the developer signs a contributor license agreement (CLA). A CLA is specifically designed to protect a developer's contribution. Organizations do not usually protect the hosted projects managed by third parties with a CLA; for example, Outercurve Foundation, Eclipse and Gentoo. <ul style="list-style-type: none"> • A project might receive an organization's IP clearance for contributions and third-party libraries. • IP management enables and encourages the participation of organizations' software developers to develop software collaboratively in a FLOSS community. • When a CLA is signed by developers, the entity protects the contributions of its portfolio projects; for example, Twitter and 52 NIFGOSS. • However, third parties managing hosted projects within the entity are not protected by a CLA.
Project Acceptance	Projects need to be championed by a sponsor (e.g., if the sponsor is the foundation board); for example, Outercurve Foundation.
Project Hosting	Organizations provide a project-hosting infrastructure and tools to promote FLOSS development; for example, OSGeo and Genivi Alliance. <ul style="list-style-type: none"> • The support entity hosts projects and a wide variety of other mailing lists for projects, committees and special-interest groups.

Case 2: Within the Wikimedia Foundation, we have identified that developers cannot entirely create a new project by going through the incubation process. They can only start a new language version of an existing project by going through the incubation process.

By comparing Case 1 and Case 2, we determined that the purpose of the incubation process used within ASF and Wikimedia Foundation is different.

Case 7: We identified that Twitter requires developers from companies to accept and submit a contributor license agreement (CLA) so that their contributions are protected by Twitter.

Case 13: NIFGOSS can host open-source projects managed by third parties. However, it does not protect the contributions made by the third-party developers since the contributions are not covered by CLA.

By comparing Case 7 and 13, we found that different support entities have different strategies and mechanisms related to handling developer contributions.

Case 25: We identified that Genivi Alliance provides hosting services to its foundation projects.

Case 38: We identified that the MirOS project can be created/started by anyone who has the necessary skills.

Case 50: Tryton Foundation projects are divided into subprojects. We identified that each subproject is also assigned a project leader.

According to our initially set criteria, Case 51 to 70 did not provide us with any new emerging data; thus, we decided to end our theoretical sampling process.

Table 1 reveals that FLOSS support entities such as ASF, Gentoo and SpringSource provide various forms of support and services to portfolio projects. Organization incubation processes are used in ASF, Wikimedia Foundation, Eclipse Foundation and the MirOS project. Foundations such as ASF and Tryton assign a PMC to govern their projects. Some foundations, such as KDE, have limited hierarchical structures. Some support entities (e.g., the Outercurve Foundation, Eclipse and Gentoo) own the IP rights to protect their portfolio projects.

4.1.3 RQ3. How do FLOSS support entities interact with each other?

According to the Open Hub API definition, a portfolio project is claimed (or) hosted under a specific FLOSS support entity. FLOSS support entities and FLOSS projects are specified using unique ID numbers. From the collected API data, we explored the relationships between two FLOSS support entities by determining whether there are any overlapping projects between them, i.e., any project with the same project ID hosted under or claimed by more than one entity as a portfolio project. The data also contained

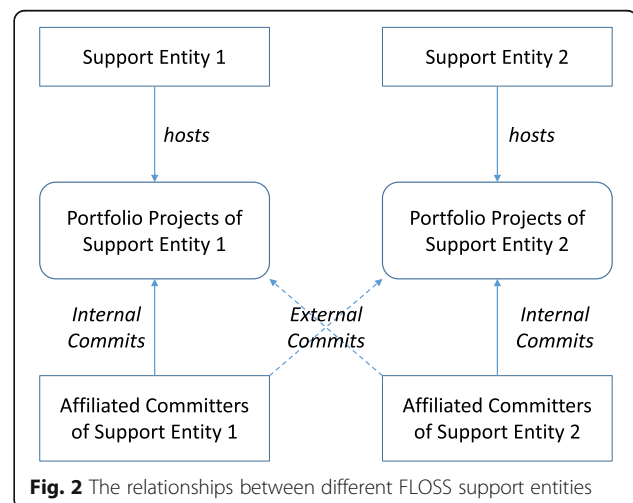
information about which developers are affiliated with which projects.

In the generated database, we identified whether entities with unique IDs have (1) connections with projects affiliated with other entities and (2) whether an entity's affiliated developers contribute to the projects of other entities. These different scenarios are described in Fig. 2 below. As depicted in the figure, a portfolio project of support entity A is a project that is hosted by entity A. A FLOSS project has affiliated committers who may contribute to projects hosted by other support entities through external commits.

This investigation is relevant to our research question from at least two perspectives. First, it would be interesting to determine whether support entities provide support mechanisms to portfolio projects only or to other non-portfolio projects as well. Second, in the latter case, it would be interesting to investigate whether the external commits of affiliated committers play a role in supporting non-portfolio projects.

As an example, Mozilla Foundation has a number of outside non-portfolio projects to which Mozilla's affiliated developers are contributing to. On the other hand, support entity Homebrew has only one project with three affiliated committers contributing to that project only. We did not find any projects hosted under or claimed by multiple organizations as a portfolio project.

We then used a manual approach to search for the appropriate information through relevant online sources (e.g., foundations' websites and forums) to describe the identified relationships between the FLOSS support entities. In order to identify the key reasons behind these relationships, through our set criteria for the Theoretical Saturation Grounded Theory approach, we have



considered each and every relationship between any two entities within the FLOSS relationship network as a sampling case. We then qualitatively investigated and analyzed the identified details of the relationships and grouped them.

Based on our qualitative analyses, we list the identified reasons explaining support entities' interactions. Two FLOSS support entities may have a relationship due to the following key reasons: plug-ins, sponsorships, tie-ups, packages, reliance, key persons and hosting (see Table 2 for detailed descriptions). We have listed the identified links in Appendix 3.

4.2 Discussion

In this section we discuss our findings with respect to the implications for research and the industry.

4.2.1 Implications and future avenues for research

Our findings show the fertility of the overall research area of FLOSS support organizations and open several new avenues for further research. There are interesting research opportunities related to verifying and measuring the impacts of developer contribution and entities. Focusing on the relationship between individual projects and entities has potential for future contributions, e.g., as it pertains to governance, tensions or developer decision-making. Furthermore, support entities might enrich our understanding of traditional FLOSS research questions related to developer motivation, group collaboration or generation of a culture of

contributions. Support entities also seem to be one of the main enforcers of IP as well as the area in which IP-related discussions take place. One particularly interesting question might stem from looking into how support entity projects differ from projects that lack support entities. Anecdotal evidence suggests that larger, more established projects move toward the founding of support entities, but there are, of course, other ways to procure the same services.

Our methodology focuses on certain parts of the interplay between support entities, so we expect future studies will shed further light on the important and understudied role of these entities in supporting and governing FLOSS. Specifically, more quantitative methods may yield a better understanding of these entities in the future.

The collected data showed that different FLOSS support entities have relationships when affiliated developers from one FLOSS support entity contribute to other FLOSS support entities' portfolio projects. As this study mainly focused on support entities, we did not engage in an in-depth consideration of individual project information, although this information may offer more insight regarding specific projects and their committers.

Further, in this work we have not examined the connection between the key support mechanisms presented in Table 1 or the relationships' information presented in Table 2 and the characteristics taxonomy presented in Fig. 1. For example, the kinds of services provided by a support entity (e.g., legal, financial,

Table 2 The relationship between two support entities

Plug-ins/Add-ons	A FLOSS support entity may provide or produce plug-ins/add-ons to other FLOSS support entity projects and their products; for example, the Xfce desktop provides an add-on to Mozilla's Thunderbird application.
Sponsorship	A FLOSS support entity may provide funding or sponsorship to contributors to other FLOSS support entities and portfolio projects; for example, Twitter provides financial funding and contributes to the Apache Software Foundation, and Yahoo also provides financial funding to the OpenStack Foundation.
Tie-ups	FLOSS project software might have a tie-up with other FLOSS support entity software. For example, Xfce and KDE desktops have tie-ups with Debian operating system.
Packages	A FLOSS support entity may provide packages for other FLOSS products and services. For example, Homebrew provides packages for KDE desktop applications to install on OS X. Homebrew also provides packages to Mozilla's add-ons on OS X.
Reliance	A FLOSS support entity may use another FLOSS support entity software, service, infrastructure, tool or product for its own business operations and services; for example, Sony Mobile and Yahoo use the OpenStack platform infrastructure for their business purposes.
Key persons	A key person—such as the founder, lead developer, maintainer or manager—from one FLOSS support entity might be employed by another FLOSS foundation. Both FLOSS support entities might have a single person as a common manager responsible for FLOSS projects; that is, a single person may act as a manager for both organizations' projects. For example, Tarent Solutions GmbH and the MirOS project have a single person managing their projects; the same person who founded the MirOS project is employed by Tarent Solutions GmbH.
Hosting	A FLOSS support organization might host and distribute other FLOSS support entity products and services; for example, BlackBerry hosts and distributes Adobe apps on BlackBerry World to BlackBerry mobiles. Moreover, a FLOSS support entity may provide generic modules and functions to work with other FLOSS support entity software implementations; for example, SaltStack provides generic modules and functions to work with the Apache Software Foundation implementation.

consulting etc.) (Table 1) may depend on whether the entity is nonprofit or for-profit. Similarly, the package-driven relationship (Table 2) may be motivated by the development focus of the organization in the sense that one expects software-oriented organizations to develop more packages than service-oriented ones.

4.2.2 Implications for industry

The findings have several intriguing implications for practitioners and the industry in general. These entities are key stakeholders for companies that rely on FLOSS projects in their operations or as a part of their offering.

For individual developers, these entities not only provide needed respite from IP issues but also provide an interesting avenue for how IP policies develop. Additionally, they provide the infrastructure of contribution and governance, helping relieve developers' burden and in turn allowing them to focus on software development-related issues.

For support entities, benchmarking the relevant projects leads to valuable opportunities for the learning and diffusion of best practices. The links between projects also offer new areas for the potential cooperation or mitigation of possible tensions. Obviously, these support entities also have a considerable aggregate voice in promoting FLOSS toward external stakeholders.

4.3 Limitations

We discuss a number of limitations/threats that may affect the validity of the research results along the guidelines of Runeson&Höst [16].

Construct validity threat is the extent to which the studied operational measures reflect what the researcher intended to study according to the research goals. In this research study, the main construct validity relates to our assumption that the terms "support entity," "organization" and "foundation" are used interchangeably. Indeed, different data sources use different terms when describing the FLOSS ecosystem. Our approach has been rather inclusive with respect to the used terms.

Internal validity threat is the prospect where external factors may affect the study results. The researcher might be aware of these factors, but others might not be aware of them [16]. Since we are conducting an exploratory study, there is a lack of existing evidence available on FLOSS support entities. We found that increasing growth of new entities and projects within the Open Hub repository may also affect our study results. We considered only 88 FLOSS support entities as the sample size, but the total number

of FLOSS support entities within the Open Hub is much higher. However, we believe that the sample we are using is relevant for analyzing patterns and trends in the evolution of FLOSS support entities. During our data collection process, some of the FLOSS support entities and individual projects within the Open Hub repository did not provide us with relevant data that are essential for conducting this study. It will be impossible for us to avoid or minimize this threat because it is not feasible to inquire about the missing information through the Open Hub repository. Further, the process of identifying relationships among different FLOSS support entities is a continuous process and might change over time. In order to minimize these validity threats, we argue that our study results are specific to the study period.

External validity threats relate to what degree the results of the study can be generalizable [16]. We argue that generalizability can be considered as a threat to validity in this study since we have used the Open Hub data repository as the main source for collecting data. To minimize this validity threat, we have considered the FLOSS support entity website data and other external web links to improve our study results.

Reliability is the prospect that is concerned with how the study data and data analysis are reliant on the researcher. This outlines how the results will be if the same study is conducted by other researchers [16]. Exceeding the saturation point by other researchers might reveal other attributes, new organization's support mechanisms and the relationships between different FLOSS support entities, which could affect our study results.

5 Endnotes

¹In this work, we use the term "support entities" yet recognize that, in many cases, "foundation" would also be applicable. However, we note that (1) not all of these entities are foundations and (2), even if they are, there are subtle differences regarding their legal and tax statuses in different jurisdictions. Thus, we limit these legal considerations, which fall outside the scope of this study, and employ the term "support entities."

²Open Hub (or Black Duck Open Hub) is a website that offers an open-source directory, community platform and set of analytics for open-source projects. The site contains in total 32,616,131,121 lines of code and indexes 472,078 projects that have in total 4,134,848 contributors. The site tracks 796,636 source control repositories. [27 January, 2018.] Open Hub website at <https://www.openhub.net>.

³https://github.com/blackducksoftware/ohloh_api

6 Appendix 1

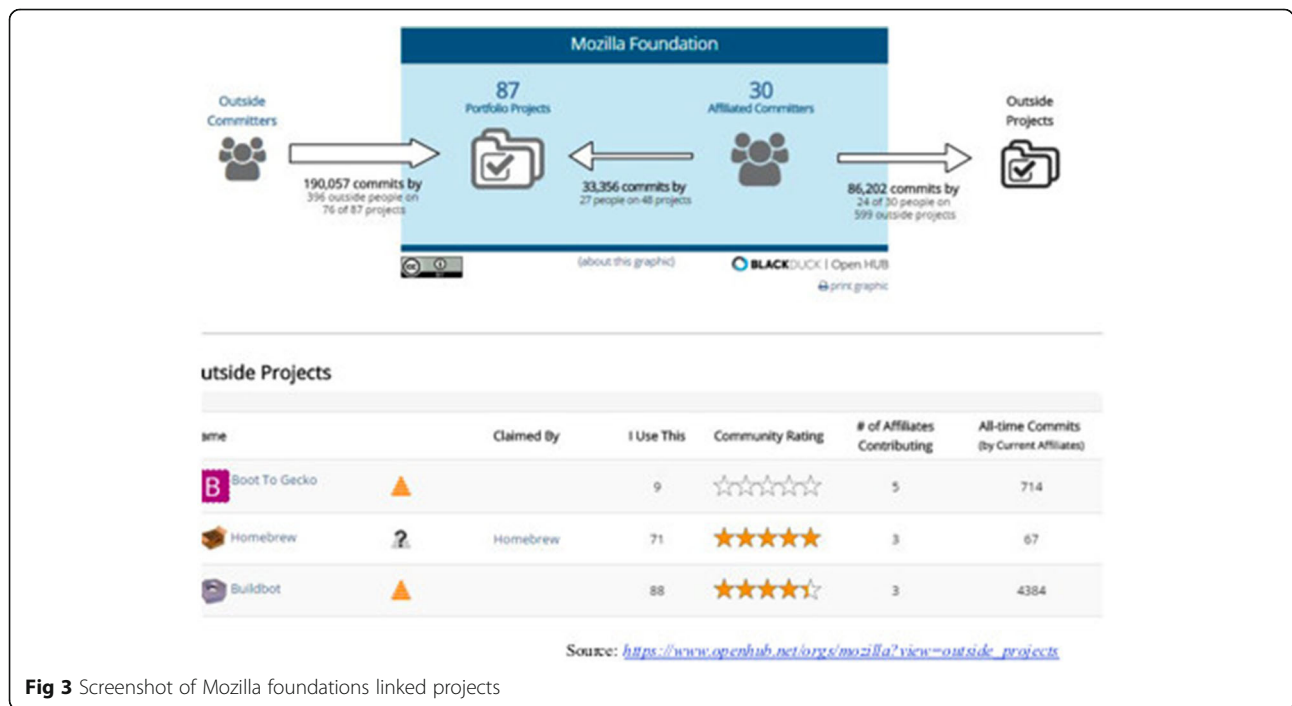


Fig 3 Screenshot of Mozilla foundations linked projects

7 Appendix 2

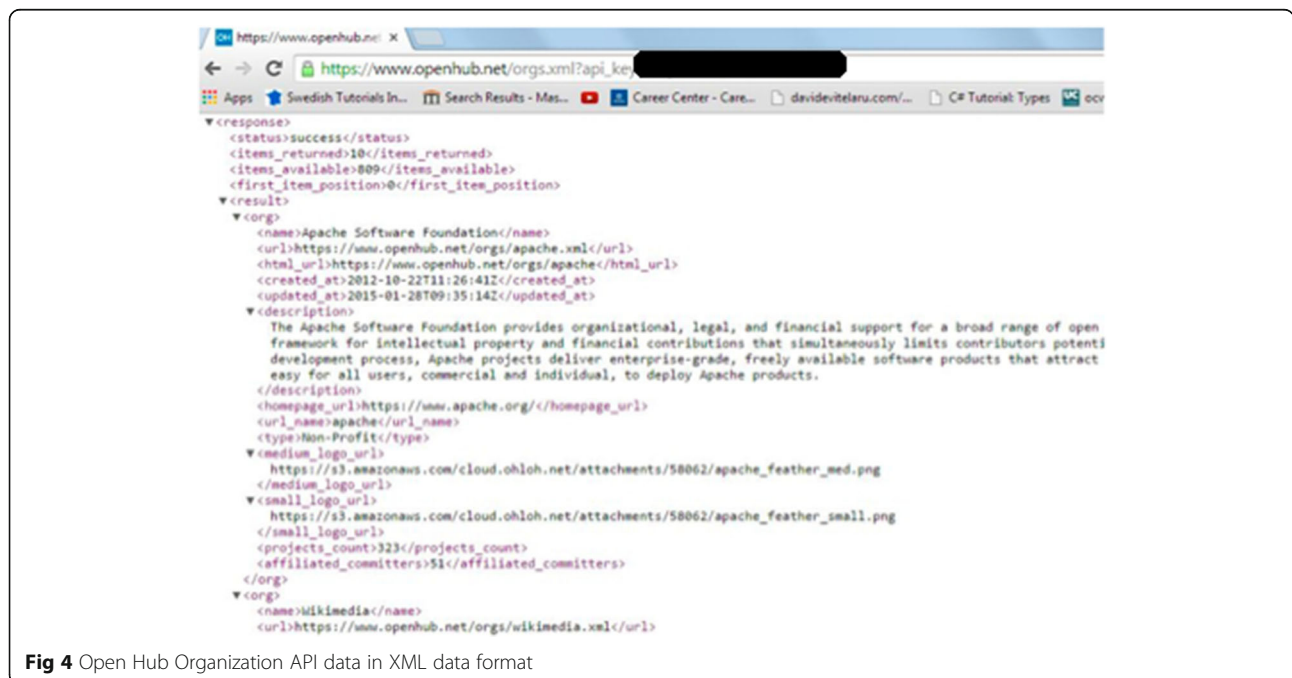


Fig 4 Open Hub Organization API data in XML data format

8 Appendix 3

Table 3 External linkages between supporting entities

Supporting entities	Identified linkage
1 ASF - Eclipse	<ul style="list-style-type: none"> Eclipse is providing plugins to Apache projects. [http://www.eclipse.org/subversive/] [https://code.google.com/a/eclipseorg/p/eclipsepluginapache/]
2 ASF- Mozilla	Linkage not identified
3 ASF - Twitter	<ul style="list-style-type: none"> Twitter is sponsoring ASF financially. [http://www.apache.org/foundation/thanks.html] Twitter is also contributing in ASF projects through their committers. [https://engineering.twitter.com/opensource/community]
4 ASF-JBOSS	<ul style="list-style-type: none"> JBoss web server collaborated / integrated with Apache http server. [http://www.jboss.org/products/webserver/overview/]
5 ASF - Debian	<ul style="list-style-type: none"> The Debian is providing Apache's packages to install Apache software under Debian operating system. [https://wiki.debian.org/Apache]
6 Wikimedia -Debian	<ul style="list-style-type: none"> The Debian provides packages for MediaWiki [https://wiki.debian.org/MediaWiki]
7 Wikimedia - SMC	Linkage not identified
8 Wikimedia -Openstack	<ul style="list-style-type: none"> Wikimedia is using Openstack to build test and development infrastructure. [http://readwrite.com/2011/01/03/howthewikimediafoundation/]
9 Wikimedia -Homebrew	Linkage not identified
10 Wikimedia - KDE	Linkage not identified
11 KDE - Debian	<ul style="list-style-type: none"> KDE desktop environment softwares are tieup with Debian. There is also KDEDebian maintenance team on KDE software's successful installation on Debian environment. [https://wiki.debian.org/KDE] [http://pkgkde.alioth.debian.org/]
12 KDE - VideoLan	Linkage not identified
13 KDE - XBMC	Linkage not identified
14 KDE - Gentoo	<ul style="list-style-type: none"> Gentoo is providing Packages for KDE desktop applications. [https://wiki.gentoo.org/wiki/KDE]
15 Mozilla -OWASP	<ul style="list-style-type: none"> Mozilla is using OWASP security tools and addons for its firefox browser. [https://addons.mozilla.org/Enus/firefox/collections/dennis_groves/owa/] [https://blog.mozilla.org/security/2012/09/13/owasp-zap-the-firefox-of-web-security-tools/]
16 The Xfce Desktop - Mozilla	<ul style="list-style-type: none"> The Xfce desktop provides addons for Mozilla Thunderbird application. [https://addons.mozilla.org/en-US/thunderbird/addon/xfce-adwaita/]
17 Mozilla - Debian	<ul style="list-style-type: none"> Mozilla Debian team provides various versions of Mozilla related packages on Debian OS based systems. [http://mozilla.debian.net/]
18 Saltstack - Mozilla	Linkage not identified
19 Mozilla - Adobe	<ul style="list-style-type: none"> Adobe provides plugins to Mozilla firefox browser [https://support.mozilla.org/en-US/kb/keep-flash-up-to-date-and-troubleshoot-problems]
20 Mozilla-OpenStack	<ul style="list-style-type: none"> Mozilla has experimented the Openstack cloud infrastructure and services. [https://bugzilla.mozilla.org/show_bug.cgi?id=963165]
21 HomeBrew-Mozilla	<ul style="list-style-type: none"> For installation of Mozilla addons on the OS X, the homebrew provides packages. [https://developer.mozilla.org/en-US/docs/Mozilla/Developer_guide/Build_Instructions/Mac_OS_X_Prerequisites]
22 Mozilla-LLVM project	<ul style="list-style-type: none"> Mozilla uses the LLVM's Clang compiler to compile the Firefox browser. [https://developer.mozilla.org/en-US/docs/Mozilla/Developer_guide/Build_Instructions/Building_Firefox_on_Windows_with_clang-cl]
23 Twitter-Mozilla	<ul style="list-style-type: none"> Twitter provides addons for mozilla firefox browser [https://addons.mozilla.org/Enus/firefox/addon/twitterapp/]
24 Twitter-HomeBrew	Linkage not identified
25 Los Alamos – Debian	<ul style="list-style-type: none"> The Los Alamos computer systems are preinstalled with Debian operating system. [http://lalinux.com/en/About]
26 Los Alamos - VideoLAN	Linkage not identified
27 Los Alamos - Gentoo	Linkage not identified

Table 3 External linkages between supporting entities (Continued)

Supporting entities	Identified linkage
28 Tarent Solutions – Debian	<ul style="list-style-type: none"> • The employee of the Tarent is a maintainer/contributor of Debian.
29 Tarent Solutions – Gentoo	Linkage not identified
30 Tarent Solutions – Miros	<ul style="list-style-type: none"> • The founder of the Miros Project organization is also an employee at Tarent Solutions GmbH. [https://www.mirbsd.org/wlog.htm] • He is the lead developer and manager of the FLOSS projects at these two organizations. [https://www.openhub.net/orgs/MirOS] [https://www.openhub.net/orgs/tarent] • Some of the commits to the Miros Project were sponsored by Tarent.
31 The Xfce Desktop - Debian	<ul style="list-style-type: none"> • The Xfce is one of the desktop options in the Debian desktop. [https://wiki.debian.org/Xfce]
32 The Xfce Desktop - OS Geo	Linkage not identified
33 Debian OS Geo	<ul style="list-style-type: none"> • Debian provides packages to OS Geo applications. [http://blends.debian.org/gis/tasks/osgeoewebsver]
34 The Miros Project - Debian	<ul style="list-style-type: none"> • Miros Project founder and manager is the maintainer at Debian project. [https://wiki.debian.org/ThorstenGlaser]
35 Adobe - Black Berry	<ul style="list-style-type: none"> • Black Berry is host and distributing Adobe applications for its mobiles [https://appworld.blackberry.com/webstore/search/adobe/?lang=en &countrycode = SE]
36 German Neuroinformatics - Mozilla	Linkage not identified
37 Yahoo - Openstack	<ul style="list-style-type: none"> • Yahoo has gold membership at Openstack foundation. Yahoo provides funding to Openstack. [http://www.openstack.org/foundation/companies/] • Yahoo is also using Openstack cloud computing platform for its operations and services. [https://www.infoworld.com/article/2623371/iaas/yahoo-s-open-source-iaas-could-up-the-ante-in-cloud-services.html]
38 Genivi – Debian	<ul style="list-style-type: none"> • The founder of Debian automotive is a key person at Genivi alliance. [https://archive.fosdem.org/2013/interviews/2013-jeremiah-foster/]
39 Saltstack – ASF	<ul style="list-style-type: none"> • The Saltstack provides generic modules and functions to work with all implementations of Apache. [https://github.com/saltstack-formulas/apache-formula]
40 Adobe – ASF	Linkage not identified
41 Adobe – Homebrew	Linkage not identified
42 Gentoo – Miros	Linkage not identified
43 Softlayer - Openstack	<ul style="list-style-type: none"> • Softlayer uses Openstack infrastructure for its business services. [https://www.computerworld.com/article/2923947/app-development/ibm-brings-openstack-to-its-softlayer-cloud.html]
44 Home Brew - KDE	<ul style="list-style-type: none"> • Home Brew is a package management software that easy installation of KDE's application on OSX. [https://github.com/KDE-mac/homebrew-kde]
45 The miros project - Home Brew	Linkage not identified
46 Plaimi – Gentoo	Linkage not identified
47 Sony Mobile - Openstack	<ul style="list-style-type: none"> • The Sony Mobile uses Openstack as its backend platform, which hosts software for connecting online gamers. [https://www.networkworld.com/article/2186653/cloud-computing/sony-division-moves-some-services-from-aws-to-openstack.html]
48 Sony Mobile - Eclipse	Linkage not identified
49 Sony Mobile - Logilab	Linkage not identified
50 Argonne National Labs - HomeBrew	Linkage not identified
51 Argonne National Labs - The LLVM	Linkage not identified
52 CZ.NIC - Debian	Linkage not identified
53 CZ.NIC - Homebrew	Linkage not identified
54 The Grid - GNOME	Linkage not identified

Acknowledgements

This paper is an extended version of the article presented at OSS 2017, under the title of Investigating Relationships between FLOSS Foundations and FLOSS projects. The authors would like to thank Bharat Kumar Mendu and Joshua Smith Soundararajan who collected the original data and provided parts of the early analyses.

Funding

No external funds were used to carry out this research.

Availability of data and materials

The API used in data collection can be found at: https://github.com/blackducksoftware/ohloh_api

Authors' contributions

IH provided guidance with the research design, data collection and data analysis. JL drafted the manuscript. Both authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Applied IT, University of Gothenburg, Gothenburg, Sweden. ²Mediterranean Institute of Technology, South Mediterranean University, Tunis, Tunisia. ³Department of Computer Science and Engineering, Chalmers and University of Gothenburg, Gothenburg, Sweden.

Received: 16 October 2017 Accepted: 9 February 2018

Published online: 22 February 2018

References

- Aaltonen T, Jokinen, J (2007) Influence in the Linux kernel community. In: Feller J, Scacchi B.F.W, Sillitti A (eds.) Open Source Development, Adoption and innovation. IFIP 234: 203–208. Springer, Boston.
- Aksulu A, Wade M. A comprehensive review and synthesis of open source research. *J Assoc Inf Syst.* 2010;11(1):576–656.
- Benkler Y. Coase's penguin, or Linux and the nature of the firm. *Yale Law J.* 2002;112(3):369–446.
- Coase RH. The nature of the firm. *Economica.* 1937;4(16):386–405.
- Dahlander L, Magnusson M. How do firms make use of open source communities? *Long Range Plan.* 2008;41:629–49.
- Dahlander L, O'Mahony S. Progressing to the center: coordinating project work. *Organ Sci.* 2011;22(4):961–79.
- Feller J, Fitzgerald B. Understanding open source software development. Boston: Addison-Wesley Longman Publishing Co Inc; 2002.
- Glaser B, Strauss AL. The discovery of grounded theory: strategies for qualitative research. New Brunswick: Transaction Publishers; 2009.
- Godfrey M., Tu M Growth, evolution, and structural change in open source software. In proceedings of the 4th international workshop on principles of software evolution. ACM press: 2001 103-106.
- de Laat PB. Governance of open source software: state of the art. *J Manag Gov.* 2007;11:165–77.
- O'Mahony S. Guarding the commons: how community managed software projects protect their work. *Res Policy.* 2003;32:1179–98.
- O'Mahony S. Non-profit foundations and their role in community-firm software collaboration. In: Software FJ, Fitzgerald B, Hissam S, Lakhani K, editors. Perspectives on free and open source. Cambridge: MIT Press; 2005. p. 393–414.
- Robles G, Amor JJ, Gonzalez Barahona JM, Herraiz I Evolution and growth in large libre software projects. In proceedings of the eighth international workshop on principles of software evolution (IWPSSE 2005). IEEE Computer Society 2005 165-174.
- Riehle D, Berschneider, S A model of open source developer foundations. Hammouda I, Lundell B, Mikkonen T, ScacchiW. 8th international conference on open source systems (OSS), Sep 2012, Hammamet, Tunisia. Springer, IFIP

- advances in information and communication technology, AICT-378:15-28, 2012, 2012 Open Source Systems: Long-Term Sustainability.
- Riehle D. The economic case for open source foundations. *Computer.* 2012;43(1):86–90.
- Runeson P, Höst M. Guidelines for conducting and reporting case study research in software engineering. *Empir Softw Eng.* 2009;14(2):131–64.
- Schaarschmidt M, Walsh G, von Kortzfleisch HFO. How do firms influence open source software communities? A framework and empirical analysis of different governance modes. *Inf Organ.* 2015;25:99–114.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com