

# On licensing and other conditions for contributing to widely used open source projects: an exploratory analysis

Jonas Gamalielsson  
University of Skövde  
Skövde, Sweden  
jonas.gamalielsson@his.se

Björn Lundell  
University of Skövde  
Skövde, Sweden  
bjorn.lundell@his.se

## ABSTRACT

Open source software (OSS) projects are provided under different open source licenses and some projects use other conditions (in addition to licensing terms) for contributors to adhere to. Licensing terms and conditions may affect community involvement and contributions, and are perceived differently by different stakeholders in different OSS projects. The study reports from an exploratory analysis of licensing terms and other conditions for 200 widely used OSS projects, and an investigation of the relationship between licensing terms and other conditions for contributing. We find that strong copyleft licenses are most common and are used in the majority of the projects. Further, a clear majority of the OSS projects use no specific other condition for contributing in addition to the license terms. However, a clear majority of the OSS projects supported by foundations use other conditions for contributing in addition to the license terms. Finally, use of no specific other conditions in addition to the license terms is more common for projects using strong copyleft licensing compared to projects using non-copyleft licensing.

## Author Keywords

open source; licensing; conditions; agreements

## ACM Classification Keywords

K.5 Legal Aspects of Computing  
K.6.3 Software Management

## 1. INTRODUCTION

Software is developed and provided under a range of different conditions, including a number of different licenses which are recognised by the Open Source Initiative (OSI) as open source licenses [29]. Open source is considered as an important driver for software development and innovation in several domains [4]. For example, complex products developed in the automotive sector rely on software which is provided under a range of conditions

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.

*OpenSym '17*, August 23–25, 2017, Galway, Ireland  
© 2017 Copyright is held by the owner/author(s).  
ACM ISBN 978-1-4503-5187-4/17/08.  
<https://doi.org/10.1145/3125433.3125456>

and different open source licenses [25].

When OSS projects make decisions concerning under which specific open source licenses and other conditions that developed software will be provided, it is evident that such decisions may have significant impact on its community and its future evolution. A specific choice of license made by a specific OSS project may be preferred by some individuals and companies, whereas others may dislike the decisions made and therefore decide not to engage with and contribute to software in the specific project. For example, it has been argued that companies may avoid engaging with GPL-licensed OSS projects [21], whereas other research shows that companies in the embedded systems area may prefer engaging in OSS projects which provide software under the GPL [22]. In fact, it has been shown that copyleft licenses can promote participation of small companies in OSS projects [22].

Apart from licensing terms, a number of OSS projects (and associated governing organisations) use a variety of different types of conditions for individuals and organisations wishing to contribute to a project. Conditions include contributor agreements, copyright assignments and disclaimers, developer certificates of origin, and conformance to committer requirements. There are different opinions and views concerning use of such conditions for different stakeholders in different OSS projects. Community members may perceive an increased bureaucracy that makes it harder and less motivated to contribute. For example, it has been reported that contributors to the OpenOffice.org project (before the LibreOffice fork) “over time perceived frustration and discontent due to various circumstances in the project” [9], which included concerns over use of copyright assignments. Increased bureaucracy may also result in that “motivation might decrease if it takes time for the contributions to be incorporated” [34]. It has also been argued that “Copyright assignment introduces asymmetry in the relationship between the copyright holder and outside contributors” and that “This inequality creates a barrier to involvement by other contributors.” [34]

Some OSS projects, especially those maintained by foundations and commercial organisations, may perceive use of different kinds of conditions as a necessary means for avoiding legal disputes and for their own legal protection. For example, it has been claimed that

“contributor agreements avoid, as far as possible, any future legal issues regarding the individual contributions such as disputes over origin, or ownership of rights over the code, or content of the product.” [14] Further, in the context of the OpenOffice.org project, it has been stated that: “In the case of Sun, most of its contribution is accounted for by OpenOffice, for which Sun holds the copyright. The entire codebase of OpenOffice is not, in fact, Sun’s sole creation, but contributors – individuals and other firms, small and big – sign an agreement assigning Sun joint copyright of their contributions, in order to simplify licensing and liability management” [12].

Licensing terms and other conditions for contributing to OSS projects are fundamental to individuals and organisations involved in those projects and their associated communities. Consequently, any organisational decision concerning involvement in OSS projects need to be based on awareness of current practice for use of licensing terms and other conditions, which significantly may impact on the potential for successful future evolution of OSS projects. Therefore, the overarching goal of this study is to characterise how widely used OSS projects are licensed and to how different types of other conditions are used for those projects. Specifically, the focus is on a larger number of widely used OSS projects ranked by user count on OpenHub. Through an analysis of publically available information on licensing and use of other conditions for contributing for 200 OSS projects, we investigate practices concerning licensing and other conditions for contributing.

We make three principal contributions. First, we establish an overall characterisation of licensing terms in widely used OSS projects. Second, we establish an overall characterisation of use of other conditions for contributing<sup>1</sup> to widely used OSS projects. Third, we report on observations concerning the relationship between licensing and use of other conditions for contributing to different widely used OSS projects.

The following research questions are formulated:

Q1: How are different licenses (and categories of licenses) used in different widely used OSS projects?

Q2: How are other conditions for contributing used in different widely used OSS projects?

Q3: What is the relationship between licensing and use of other conditions for contributing in different widely used OSS projects?

The rest of this paper is organised as follows. A background is presented on open source licensing, other conditions in OSS project contexts, and previous research. This is followed by the research approach, results, and analysis. Finally, the study presents discussion and conclusions.

---

<sup>1</sup> This may include contributions in terms of new source code, bug fixes, documentation, etc.

## BACKGROUND

### Open source licensing

The concept *copyleft* is central to several OSS licenses and constitutes a distinct contrast to other licenses which are often referred to as permissive (or “non-copyleft”) OSS licenses [30]. It has been claimed that “the central idea of copyleft is that we give everyone permission to run the program, copy the program, modify the program, and distribute modified versions--but not permission to add restrictions of their own. Thus, the crucial freedoms that define ‘free software’ are guaranteed to everyone who has a copy; they become inalienable rights.” [36] Essentially, copyleft “refers to licenses that allow derivative works but require them to use the same license as the original work.” [30] Hence, as “a concept, copyleft takes copyright (where the creator of a work is the owner of the copyright in code they develop and third parties may use it only with the owner’s consent) and turns it around.” [6] The effects of the GPL copyleft license has been elaborated as follows: “The GPL targets a specific audience: those most likely to withhold contributions to the code base. Software licensed under the GPL may thus be less attractive to commercial actors who would like to make proprietary derivative works. While it has been argued that this clause inhibits commercial adoption and innovation of GPL code, it can also be considered a clever way to prevent behavior that might threaten the sustainability of freely available code.” [28] Further, it should be noted that there are different views concerning the commercial attractiveness for GPL amongst practitioners. For example, it has been reported from the embedded systems domain that consultants “express clear preference for Open Source licensing, and in particular GPL in this domain” [22].

Different OSS licenses have different legal effects and categorisations of OSS licenses have been presented by practitioners, researchers, and representatives for the OSI. Central to most (if not all) categorisations is that there is a clear distinction between permissive OSS licenses and those that have a copyleft effect, and the scope for the reciprocity is often referred to as “strong” or “weak” copyleft. For example, open source licenses has been categorised into three major groups: permissive licenses (including MIT, 2- and 3-clause BSD, and Apache v2.0), weak copyleft licenses (including GNU Lesser General Public License versions 2.1 and 3, and Mozilla Public License 2.0), and strong copyleft (including GNU General Public License versions 2 and 3, and the Open Software License) [20].

Software provided under a permissive license “can be distributed as part of a larger product under almost any other license, while the only requirement is the attribution to the original authors.” [20] The difference between strong and weak copyleft licenses lies in the permission given for derivative work: “Any derivative work of strong copyleft-licensed software needs to be distributed with the same

license”, whereas for weak copyleft licensed software “the derivative work can be distributed under another license as long as it has not modified the weak copyleft-licensed software used” [20].

The importance of copyleft for users of open source software has been emphasised as follows: “The primary legal regime that applies to software is copyright law. Copyleft, which uses functional parts of copyright law to achieve an unusual result (legal protection for free sharing) forms the core legal principle of these licenses. It modifies, or ‘hacks’ copyright law, which is usually employed to strengthen the rights of authors or publishers, to strengthen instead the rights of users.” [27] Further, there are innovation benefits from use of copyleft licensed software resources, and it has also been claimed that “due to the fact that now many users can decide independently on the use of one and the same resource, the chances for its creative employment, for a follow-on invention, get multiplied.” [43] The importance of contributions in the context of copyleft has also been highlighted: “Copyleft licenses require that those who take material from the common pool give something back as well” [8].

A number of OSS projects and tools for license analysis of OSS projects have been provided, including the FOSSology project [13], the Binary Analysis Tool [15], and the Ninka tool [10]. These three tools, and a variety of other tools for license analysis, are compared in [20].

In addition to copyright, the complex issue of patents is also central to open source licenses. Some open source licenses are ‘silent’ on the issue of patents, whereas others address patents to varying extents and in different ways. Further, any organisation wishing to implement specific file formats and algorithms also need to consider the issue of patents and the risk for litigation. Open source licenses that do not consider patents include the licenses 3-Clause BSD and MIT, whereas other licenses (e.g. General Public License version 3 and Mozilla Public License 2.0) contain explicit patent clauses. For example, the “license grant in certain licenses (such as Apache 2.0, Eclipse, Mozilla, and GPL 3.0) runs with the license to the software. If a contributor to the code has a patent on the code, that contributor grants a patent license to all recipients of the code to enable them to exercise the open source license.” [24] Further, it should be noted that some “companies also choose to release code under a license with no patent grant—like BSD—and then grant a separate patent license.” [24] In addition, as elaborated by a former president of the OSI, it has been expressed that “many of the contributors to open source projects are patent-holding companies, this means you are the automatic recipient of patent licenses. When you innovate and contribute to the project, your innovations share the protection provided by the license. What must you do to be protected? First, make sure the software you use is under one of these modern licenses; older licenses like BSD and MIT don’t mention patents.” [33]

### **Other conditions for contributing to OSS projects**

Many OSS projects use other conditions for contributing in addition to licensing terms. One example of such conditions is contributor agreements<sup>2</sup> which “are contracts by which software developers transfer or license their work on behalf of an open source project. This is done for convenience and enforcement purposes, and usually takes the form of a formal contract.” [14] There are also copyright assignments<sup>3</sup> and copyright disclaimers that OSS projects may use for transferring or abandoning copyright in the context of contributing to a project. Further, the licensing terms that are to be used in combination with different types of agreements should be negotiated amongst stakeholders, as stated in the following: “The type of licence that will be appropriate for use in a CLA/CAA will depend on the specific interests of the parties to that CLA/CAA.” [7]

It is also important to recognise the inherently global nature of open source and that any analysis of licensing issues needs to consider many different jurisdictions. For example, previous research has identified significant differences between German and US law: “Irrespective of the licence type, breaches of standard CLA/CAA terms will usually not be copyright infringements under German law. In the US, by contrast, the violation of such terms may well amount to copyright infringement.” [7]. Overall, it has been claimed that “CLA/CAAs are a global phenomenon and their use is seldom limited to one jurisdiction, it is the author’s conviction that fully understanding CLA/CAAs requires an international perspective.” [7] and that the “international composition of FOSS projects requires organisations and companies conducting those projects to understand the basic private international law principles of the law of contracts and copyright and to shape their strategy for the governance of the projects on the basis of these principles.” [26]

The language used in different types of agreements is also important to consider, and it has been stated that a “CLA/CAA needs to use standard language for identifying such terms as conditions, as opposed to covenants. Such language could be ‘provided that’ or ‘conditioned upon’. In case of a breach of a CAA, however, copyright infringement will only occur if the breach causes the assignment to terminate. If this is the desired outcome, the CAA should contain language that clarifies this legal effect of breaches.” [7]

Use of different kinds of conditions (including agreements) for contributing to OSS projects may result in tension between different stakeholders (including volunteer contributors and corporate OSS project maintainers). Discussions on this topic in different fora can be useful, and it has been stated that the current debate around contributor

---

<sup>2</sup> Also referred to as contributor license agreement (CLA)

<sup>3</sup> Also referred to as copyright assignment agreement (CAA)

agreements and different drafting options is “as an opportunity to further develop best practices for legal strategies to reduce friction involved in moving rights between developers and free and open source software projects.” [23]

Copyright in the context of open source is complex and it is essential for any contributor to understand fundamental copyright principles. Previous research (e.g. [18]) stresses that such understanding may be lacking. For example, it has been argued that “when software is developed in an open source model, copyright issues abound, and many of these copyright issues are not well understood by software developers.” [18] and that when “a wide range of hands can touch the open source code, ownership and rights in the code can become blurred. Moreover, not all code contributions to an open source project will be protected by copyright.” [18]

### Previous research

There is previous research related to open source licensing that concerns modeling of software licenses & license architecture of systems (e.g. [1,2]), license evolution (e.g. [32,35]), and automated support for license analysis (e.g. [10,11,19,38]). Other research focuses on the impact of license choice and organisational sponsorship on success in open source development projects [37]. There are also studies which have a focus on license usage and license changes for all Java projects on GitHub [39,40,41] and motivation and impact of changed licensing for a selection of projects [42]. Further, there is research that concerns license inconsistencies in a large number of OSS projects [44,45].

Related research on other conditions for contributing to OSS projects includes a study involving a comparative analysis of copyright assignment and license formalities for open source contributor agreements [14]. A different study reports from an exploratory literature survey on copyright assignments and contribution license agreements in FOSS projects [34]. The adoption of copyright assignment agreements and copyright license agreements in different OSS projects has been studied through a qualitative approach [16]. Further, the internationalisation of contributory copyright assignments and licenses is addressed in [26]. Another study focused on how community contributions are managed through a case study on the Android and Linux software ecosystems [3].

To the best of our knowledge, no earlier study has presented results concerning licensing terms, other conditions for contributing, and the relationship between licensing terms and other conditions for widely used OSS projects. Hence, our study contributes to filling that knowledge gap.

### RESEARCH APPROACH

Through an exploratory analysis of widely used OSS projects, we establish a characterisation of licensing terms

and use of other conditions for contributing to those projects. Specifically, names (and associated meta-data) for the top 200 OSS projects on OpenHub<sup>4</sup> with respect to user count were collected during March 2017 using custom made Perl scripts utilising the OpenHub (also known as Ohloh) API<sup>5</sup>. We consider use of highly ranked OSS projects with respect to OpenHub user count to constitute a representative set of widely used open source projects. Use of the top 200 OSS projects according to user count is a purposeful choice guided by visual inspection of the decline in user count as a function of sorted project index.

Using a manual web search<sup>6</sup> applied to the domain names (if available) for the selected OSS projects and specific key words (including “license”, “contribute”, “agreement”, and “conditions”), information was collected concerning licenses used and other conditions for contributing for each of the top 200<sup>7</sup> OSS projects. The collected data was analysed and summarised using custom made Perl scripts. It should be noted that conditions encountered during data collection were sorted into seven emerging categories (No specific condition, Individual contributor agreement, Corporate contributor agreement, Copyright assignment, Copyright disclaimer, Developer certificate of origin, and Conformance to committer requirements). It should also be emphasised that *currently used* licensing terms and other conditions were collected (i.e. historical licensing terms and other conditions were not collected).

A number of software projects were encountered during data collection that are not recognised by the OSI as open source software (provided under conditions including: BitTorrent Open Source License, ImageMagick License, Public Domain, Ruby License, OpenLDAP Public License v2.8, OpenSSL license, and Vim License), and these projects were therefore not included in the set of top 200 OSS projects. Other “projects” that were excluded are different Linux distributions and environments (including Android, Arch Linux, CentOS, Debian, Gentoo Linux, KDE, and Ubuntu) that each contain many software projects provided under different licensing terms and conditions.

## RESULTS

### Characterisation of licensing in OSS projects

Table 1 shows the number (and proportion) of OSS projects that use a specific type of license. Some projects may use several licenses. We have chosen to omit “or later” for licenses in order to simplify the table. For example, when referring to “GNU General Public License version 2” the table includes OSS projects which provide software under

---

<sup>4</sup> <https://www.openhub.net/>

<sup>5</sup> [https://github.com/blackducksoftware/ohloh\\_api/](https://github.com/blackducksoftware/ohloh_api/)

<sup>6</sup> Through use of the Google search engine.

<sup>7</sup> The top 200 projects are listed in Appendix A.

both “GNU General Public License version 2 only” and under “GNU General Public License version 2 or later”. Different kinds of exceptions in combination with a license have also been omitted. For example, “GNU General Public License version 2 + classpath exception” has been categorised as “GNU General Public License version 2”. It can be observed that there are 25 different open source licenses<sup>8</sup> used in the 200 selected OSS projects. Further, we note that the strong copyleft licenses “GNU General Public License” versions 2 and 3 are the two most frequently used licenses among the selected OSS projects appearing in 32% and 17% of the projects, respectively. There is also a variety of less common licenses of which 11 licenses (41% of all licenses used) only are used in a single project.

License	#	%
GNU General Public License version 2	64	32
GNU General Public License version 3	33	17
Apache License, Version 2.0	24	12
GNU Lesser General Public License version 2.1	19	10
MIT License	17	9
3-Clause BSD License	14	7
Eclipse Public License 1.0	8	4
GNU Lesser General Public License version 3	6	3
Mozilla Public License 2.0	5	3
2-Clause BSD License	3	2
BSD-like License	3	2
GNU Affero General Public License version 3	3	2
GNU Library General Public License version 2	3	2
PHP License 3.0	3	2
GNU General Public License version 1	2	1
MIT-like License	2	1
Academic Free License 2.1	1	<1
Artistic License 2.0	1	<1
Boost Software License 1.0	1	<1
Common Development and Distribution License 1.0	1	<1
IBM Public License Version 1.0	1	<1
ISC License	1	<1
LaTeX Project Public License, Version 1.3c	1	<1
zlib/libpng License	1	<1
Open Group Test Suite License	1	<1
PostgreSQL License	1	<1
Python Software Foundation License Version 2	1	<1

**Table 1. Number and proportion of OSS projects using a specific license.**

Table 2 is a variant of Table 1 where licenses have been aggregated into four (of the total five) license categories related to copyleft proposed by the Institute for Legal Questions on Free and Open Source Software (<http://www.ifross.org/en/license-center>)<sup>9</sup>. It is evident that

<sup>8</sup> “BSD-like License” and “MIT-like License” are considered variants of the BSD- and MIT Licenses and are therefore not counted as being part of the 25 licenses.

<sup>9</sup> The concepts “Without copyleft” and “Restricted copyleft” are equivalent with the terms “Permissive” and “Weak copyleft” as described in [20].

licenses with strong copyleft are most widely used in the selected OSS projects and the majority of OSS projects (55%) use such licenses. We also note that licenses without copyleft are more frequently occurring than restricted (weak) copyleft licenses. Further, it can be observed that restricted choice licenses<sup>10</sup> (with restrictions in how modifications of the software can be distributed) are only used in two OSS projects.

License category	#	%
Without copyleft	70	35
Strong copyleft	109	55
Restricted copyleft	33	17
Restricted choice	2	1

**Table 2. Number and proportion of OSS projects for different license categories.**

### Characterisation of other conditions for contributing to OSS projects

Table 3 illustrates the number (and proportion) of OSS projects using different specific conditions for contributing. It can be observed that a clear majority (67%) of the selected OSS projects use no specific condition for contributing (in addition to the license terms). We also note that individual and corporate contributor agreements are not uncommon (used in 19% and 15% of the projects, respectively). Examples of projects using such agreements are OSS projects governed by foundations and other types of organisations, including the Apache Foundation, the Eclipse Foundation, and Oracle Corporation. In addition, contributor agreements are used in OSS projects including Chromium, NetBeans IDE, OpenSSL, Python, and Qt. The use of copyright assignments and copyright disclaimers are only identified in different GNU projects (e.g. Bash, GNU Compiler Collection, GNU make, Wget, and GNU Emacs) governed by the Free Software Foundation (FSF). Developer certificate of origin is used in projects including Samba and Eclipse governed projects. Conformance to committer requirements is for example explicated for committers to Mozilla repositories (including the OSS projects Firefox and Thunderbird).

Condition	#	%
No specific condition (nsc)	133	67
Individual contributor agreement (ica)	38	19
Corporate contributor agreement (cca)	30	15
Copyright assignment (ca)	24	12
Copyright disclaimer (cd)	24	12
Developer certificate of origin (dco)	6	3
Conformance to committer requirements (ccr)	5	3

**Table 3. Number and proportion of OSS projects using a specific condition for contributions.**

<sup>10</sup> In Table 1, “Artistic License” and “LaTeX Project Public License 1.3c” belong to the “Restricted choice” category (<http://www.ifross.org/en/license-center>).

**On the relationship between licensing and other conditions for contributing to OSS projects**

Table 4 illustrates the relationship between licensing and different conditions for contributing by showing the number (and proportion) of OSS projects that use specific combinations of license and condition for contributing. It can for example be noted that a number of projects (governed under the Free Software Foundation) which are licensed under GNU General Public License versions 2 and 3 use copyright assignments and copyright disclaimers. We also observe that the majority of OSS projects under “GNU General Public License version 2” do not use any specific conditions for contributing. Further, the majority of OSS projects using “Apache License, Version 2.0” (including all projects governed by the Apache Foundation) use individual and corporate contributor agreements. It can also be observed that for a number of other licenses without copyleft (including MIT License, MIT-like License, 3-Clause BSD License, 2-Clause BSD License, and BSD-like License) a clear majority of the projects do not use any

specific condition for contributing (in addition to the license terms).

Table 5 is a variant of Table 4 where licenses have been aggregated into the four earlier introduced license categories related to copyleft. It can be observed that use of no specific conditions (in addition to licensing terms) is more common for projects using strong copyleft licenses compared to projects without copyleft licensing. Further, it can be noted that contributor agreements are used to a greater extent for projects without copyleft licensing compared to projects using strong or restricted copyleft licenses. It can also be observed that copyright assignments and disclaimers are used for a larger number of OSS projects using strong copyleft licenses (and governed by the Free Software Foundation). In addition, it is evident that no copyright assignments and disclaimers are used for projects without copyleft licensing.

License	nsc		ica		cca		ca		cd		dco		ccr	
	#	%	#	%	#	%	#	%	#	%	#	%	#	%
GNU General Public License version 2	52	26	6	3	5	3	3	2	3	2			3	2
GNU General Public License version 3	9	5	4	2	1	<1	20	10	20	10	1	<1		
Apache License, Version 2.0	6	3	18	9	17	9					1	<1		
GNU Lesser General Public License version 2.1	17	9	1	<1			1	<1	1	<1				
MIT License	14	7	3	2	1	<1								
3-Clause BSD License	12	6	2	1	2	1								
Eclipse Public License 1.0	3	2	5	3	5	3					5	3		
GNU Lesser General Public License version 3	4		2		1	<1					1	<1		
Mozilla Public License 2.0	3	2											2	1
2-Clause BSD License	3	2												
BSD-like License	3	2												
GNU Affero General Public License version 3	2	1	1	<1										
GNU Library General Public License version 2	3	2												
PHP License 3.0	3	2												
GNU General Public License version 1	2	1												
MIT-like License	2	1												
Academic Free License 2.1	1	<1												
Artistic License 2.0	1	<1												
Boost Software License 1.0	1	<1												
Common Development and Distribution License 1.0			1	<1	1	<1								
IBM Public License Version 1.0	1	<1												
ISC License	1	<1												
LaTeX Project Public License, Version 1.3c	1	<1												
zlib/libpng License	1	<1												
Open Group Test Suite License	1	<1												
PostgreSQL License	1	<1												
Python Software Foundation License Version 2			1	<1	1	<1								

**Table 4. Number and proportion of OSS projects using a specific license in combination with a specific condition for contributing.**

License category		nsc	ica	cca	ca	cd	dco	ccr
Without copyleft	#	46	24	21			1	
	%	23	12	11			<1	
Strong copyleft	#	68	15	10	23	23	6	3
	%	34	8	5	12	12	3	2
Restricted copyleft	#	26	4	2	1	1	1	2
	%	13	2	1	<1	<1	<1	1
Restricted choice	#	2						
	%	1						

**Table 5. Number and proportion of OSS projects for license categories in combination with other conditions used.**

## ANALYSIS

Concerning licensing practices in widely used OSS projects, a major finding is that strong copyleft licenses are most common among the selected projects and are used in the majority (55%) of the projects. Further, the vast majority of the 200 investigated OSS projects are provided under a clear minority of all open source licenses recognised by the OSI. In fact, the 25 observed licenses is less than a third of the 80 approved OSI licenses, and there are only 9 different licenses that are used by 5 or more of the top 200 projects (where several of these are different versions of a license).

When applying a more coarse grained aggregation of licenses, we find that 60% of all OSS projects (120 projects) are provided under the GPL-family of licenses (including different versions of AGPL, GPL, and LGPL), 12% (24 projects) under Apache 2.0, 10% (19 projects) under the MIT (or a MIT-like) license, 10% (20 projects) under a BSD (or BSD-like) license, 4% (8 projects) under EPL 1.0, 3% (5 projects) under MPL 2.0, 2% (3 projects) under PHP License 3.0, and only a single OSS project (representing less than 1%) provided under any of the other open source licenses. It was also found that in total 5 OSS projects use different types of GPL-family licenses with different kinds of restrictions. For example, the GNU Compiler Collection uses “GNU General Public License v3.0 + runtime library exception”.

We note that 5 projects in Table 1 provide software under a “BSD-like” license or an “MIT-like” license, which are considered variants of the OSI recognised BSD- and MIT licenses. We acknowledge that the terms under which these 5 projects are provided may deviate from the terms in the licenses recognised by the OSI. For example, the MIT-like license for the curl project (with rank 65 in the list of top 200 projects) differs from the OSI recognised license text in that the copyright statement is more elaborated (“Copyright (c) 1996 - 2017, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.”) compared to the OSI recognised license text, the paragraph explicating the rights is somewhat simplified (“Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.”) compared to the OSI recognised license text, and there is an additional paragraph to prevent misuse of the

copyright holders’ names (“Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.”). Another example is the BSD-like license for the bzip project (with rank 48 in the list of top 200 projects) which has four clauses. The first clause is identical to the first clause of both 2-Clause BSD and 3-Clause BSD. The fourth clause is similar to the third clause in 3-Clause BSD. However, the second clause (“The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.”) and the third clause (“Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.”) are not similar to any clause in neither the 2-Clause BSD nor the 3-Clause BSD license.

Concerning other conditions for contributing, a major finding is that a clear majority (67%) of the selected OSS projects use no specific other condition for contributing (in addition to the license terms) and that use of individual- and corporate contributor agreements, and copyright assignments and disclaimers are used in at least 12% of the projects. We acknowledge that categories for conditions may to some extent be overlapping. For example, both the individual and corporate contributor license agreement for Apache Software Foundation includes text related to copyright, and thereby overlaps the category “Copyright assignment”.

Concerning the relationship between licensing and other conditions for contributing a major finding is that use of no specific conditions (in addition to the license terms) is more common for projects using strong copyleft licensing (69% of projects) compared to projects using non-copyleft licensing (47%). From further analysis of results presented in Table 5 through an abstraction into four quadrants along the dimensions “copyleft/non-copyleft” and “other conditions/no other conditions” (see Table 6), it is evident that 22% of all OSS projects (44 projects) use copyleft licensing and other conditions, 45% (89 projects) use copyleft licensing and no other conditions, 12% (24 projects) use non-copyleft licensing and use other conditions, and 25% (50 projects) use non-copyleft licensing and use no other conditions. Hence, at this level of abstraction it is clear that it is considerably more common that both copyleft licensed projects and non-copyleft licensed projects use no other conditions in addition to licensing terms. When only considering GPL licensing (including AGPL) instead of all copyleft licenses, the finding is similar: 18% of all OSS projects (36 projects) use GPL licensing and other conditions, 32% (64 projects) use GPL licensing and no other conditions, 18% (35 projects) use non-GPL licensing and use other conditions, and 40% (79 projects) use non-GPL licensing and use no other conditions.

	other conditions		no other conditions	
	#	%	#	%
copyleft	44	22	89	45
non-copyleft	24	12	50	25

**Table 6. Number and proportion of OSS projects for combinations related to copyleft and other conditions.**

In order to investigate whether there is a difference in used licensing terms and other conditions within two subsets based on the ranking of the selected 200 OSS projects, we divided the projects into the top 100 and bottom 100 projects (based on user count) and compared. Concerning use of licensing terms (see Table 7), it was found that strong copyleft licensing is used in 60% of the top 100 projects and in 49% of the bottom 100 projects. Further, restricted (weak) copyleft licensing is used in 19% of the top 100 projects and in 14% of the bottom 100 projects. It was also found that licensing without copyleft (permissive licensing) is used in 26% of the top 100 projects and in 44% of the bottom 100 projects. Hence, copyleft licensing is more dominating amongst the top 100 projects (compared to the bottom 100), and licensing without copyleft is relatively more common in the bottom 100 projects (compared to the top 100).

License category	top 100		bottom 100	
	#	%	#	%
Without copyleft	26	26	44	44
Strong copyleft	60	60	49	49
Restricted copyleft	19	19	14	14
Restricted choice	2	2	0	0

**Table 7. Number and proportion of (the top 100 and bottom 100) OSS projects for different license categories.**

Concerning use of other conditions (see Table 8), it was found that no other conditions are used in 56% of the top 100 projects and in 77% of the bottom 100 projects. Further, individual and corporate contributor agreements are used to similar extents for the top 100 projects (21% and 16%) and bottom 100 projects (17% and 14%). Copyright assignments and disclaimers are used more extensively in the top 100 projects (20% and 20%) compared to the bottom 100 projects (4% and 4%). The dominance of strong copyleft licensing and more extensive use of copyright assignments and disclaimers among the top 100 projects can be explained by the fact that there are considerably more GNU projects governed by the Free Software Foundation among the top 100 projects. This abundance of GNU projects is also reflected in a much higher proportion of copyright assignments and disclaimers for projects using strong copyleft licensing.

A substantial proportion of the 200 OSS projects are supported (or governed) by foundations (77 OSS projects, i.e. 39% of the 200 projects). For this reason we analyse whether such OSS projects tend to be licensed differently or use conditions for contributions differently compared to the full set of OSS projects. The number (and proportion) of all 200 OSS projects that are supported (or governed) by

foundations is presented in Table 9. It can be observed that 77 OSS projects are supported by in total 20 different foundations, where the Free Software Foundation supports the largest number of OSS projects.

Condition	top 100		bottom 100	
	#	%	#	%
No specific condition	56	56	77	77
Individual contributor agreement	21	21	17	17
Corporate contributor agreement	16	16	14	14
Copyright assignment	20	20	4	4
Copyright disclaimer	20	20	4	4
Developer certificate of origin	3	3	3	3
Conformance to committer requirements	3	3	2	2

**Table 8. Number and proportion of (the top 100 and bottom 100) OSS projects using a specific condition for contributions.**

Foundation	#	%
Free Software Foundation	24	12
Apache Software Foundation	15	8
GNOME Foundation	10	5
Eclipse Foundation	5	3
Linux Foundation	3	2
Mozilla Foundation	3	2
Xiph.Org Foundation	3	2
JS Foundation	2	1
Blender Foundation	1	<1
Cake Software Foundation	1	<1
Django Software Foundation	1	<1
FreeBSD Foundation	1	<1
Node.js Foundation	1	<1
Python Software Foundation	1	<1
The Document Foundation	1	<1
The Perl Foundation	1	<1
Wikimedia Foundation	1	<1
Wireshark Foundation	1	<1
WordPress Foundation	1	<1
X.Org Foundation	1	<1

**Table 9. Number and proportion of OSS projects supported by different foundations.**

Table 10 is similar to Table 2 and shows license categories for the OSS projects supported by foundations. The proportion in Table 10 is proportion of all OSS projects supported by foundations. Compared to Table 2, it can be observed that proportions for different license categories are similar, with a somewhat higher proportion of strong copyleft licensing (60% vs. 55%) and a somewhat lower proportion of restricted copyleft licensing (13% vs. 17%) among OSS projects supported by foundations.

License category	#	%
Without copyleft	27	35
Strong copyleft	46	60
Restricted copyleft	10	13
Restricted choice	1	1

**Table 10. Number and proportion of OSS projects supported by foundations for different license categories.**

Table 11 is similar to Table 3 and presents use of other conditions for the OSS projects supported by foundations. The proportion in Table 11 is proportion of the OSS projects supported by foundations. Compared to Table 3, it can be observed that the proportion of OSS projects supported by foundations that use no specific condition for contributions is considerably smaller than for the full set of 200 OSS projects (31% vs. 67%). Consequently, a considerably larger proportion of OSS projects supported by foundations use different other conditions for contributions.

Condition	#	%
No specific condition	24	31
Individual contributor agreement	24	31
Corporate contributor agreement	22	29
Copyright assignment	24	31
Copyright disclaimer	24	31
Developer certificate of origin	5	6
Conformance to committer requirements	5	6

**Table 11. Number and proportion of OSS projects supported by foundations using a specific condition for contributions.**

## DISCUSSION AND CONCLUSIONS

### Discussion

We find both differences and similarities between licensing terms for the 200 widely used OSS projects analysed in this study and licensing terms for the more than two million OSS projects in the Black Duck knowledge base [5]. For example, we note that use of GPL licensing is much more common for the analysed 200 projects in this study compared to the use of GPL licensed projects in the Black Duck database. This may indicate that GPL licensing is more common in widely used OSS projects. It is also evident that the MIT license is considerably less common for the projects analysed in this study compared to the use of the MIT license in projects in the Black Duck database. Interestingly, we note that the proportion of Apache and 3-Clause BSD licensed OSS projects among the 200 selected projects in this study is similar to the proportion of Apache and 3-Clause BSD licensed projects according to the Black Duck database.

The OSI has long had the ambition to try to counter the issue of open source license proliferation for several reasons: too many licenses makes it hard for licensors to choose a license, some licenses do not interoperate well with other licenses, and if there are too many licenses in a multi-license distribution of software it is difficult for the licensor to understand what is agreed upon [31]. In light of the effort undertaken by the OSI to counter license proliferation, we find that most of the investigated OSS projects are provided under a small subset of all open source licenses recognised by the OSI.

Amongst all software projects which claim to use some form of a BSD license, there are those licenses which are open source licenses (2-Clause BSD and 3-Clause BSD) and there is a license which clearly is not an open source

license (4-Clause BSD). Further, there are projects which provide software under modified versions of the 3-Clause BSD and 2-Clause BSD licenses. OSS projects identified in this study that are provided under a “BSD-like” license or an “MIT-like” license are considered variants of the BSD- and MIT licenses that are recognised by the OSI. We acknowledge that the terms under which such projects are provided may deviate from the terms in the licenses recognised by the OSI to an extent which goes beyond what the outcome from a comprehensive legal analysis would assess to be OSI recognised licenses. However, such a legal analysis is beyond the scope of this paper.

Experiences obtained from the data collection and analysis performed during conduct of this study show that it can for some projects be difficult to determine how a specific OSS project is licensed. Information about licensing terms can be provided at a variety of different locations (e.g. somewhere at the project web site, in different text files in repositories, in the header of source code files, etc.). The licensing terms can also be presented in a variety of different ways. For example, sometimes only the license name is stated (with or without specifying the specific version of the license), and in other cases the license text is provided without stating what license the license text represents. In light of these experiences, efforts aimed at formalising and structuring licensing information are considered useful in order to facilitate interpretation and license compliance for OSS projects. One such effort is the Software Package Data Exchange (SPDX) project which provides a set of standards for communication of components, licenses, and copyrights that are associated with software (see <https://spdx.org/>). In addition, the use of tools for license analysis of OSS projects (e.g. the FOSSology project [13]) provides further means for determining how a specific OSS project is licensed.

For an OSS project that uses a contributor agreement or a copyright assignment, a potential contributor may choose to (or choose not to) contribute to that project depending on what legal entity (e.g. organisation) rights are transferred to when signing the contributor agreement or copyright assignment. This, in turn, can impact on the extent to which an OSS project can manage to attract external contributions. Any decision concerning whether or not a potential contributor trusts the legal entity to which rights needs to be transferred (as a requirement for providing a contribution) may involve significant considerations. In fact, the importance of the holder of rights is manifested in a quality assessment framework for OSS projects (QSOS, see <http://www.qsos.org/>), which uses the copyright holder in a project as an assessment criterion where points are awarded differently depending on who has the copyright.

GPL licensing can promote competition and protect the openness of software, as stated by an internationally recognised IT law attorney: “GPL is popular not just among developers but also among companies because it helps

secure a proper competition with regard to a particular software product and prevents unfair withholding of improvements of the software released in the Free Software world” [17]. The use of contributor agreements in OSS projects and copyright assignments in GPL licensed OSS projects can potentially facilitate the protection of openness in case there are license violations and a specific OSS project needs to be defended [14]. Specifically, it has been claimed that “contributor agreements avoid, as far as possible, any future legal issues regarding the individual contributions such as disputes over origin, or ownership of rights over the code, or content of the product.” [14] and that “by having copyright assigned to a project’s director or administrative institution, the contributor can be assured that they will be able to enforce copyright in case of license breach or copyright infringement.” [14]. Further, the `gpl-violations.org` project (see <http://gpl-violations.org/>) is one initiative that aims to raise public awareness about infringements of GPL licensed software.

Analysis of the specific content of, and differences between, the conditions encountered for the selected projects is beyond the scope of this study. However, for future work it would be interesting to study this more in-depth and also consider other characteristics of OSS projects in the analysis (e.g. type of OSS project, corporate involvement, community activity, etc.).

For future work it would also be desirable to collect data for a larger number of projects to get a more extensive material for statistics and potentially automate the process to a greater extent. We initially planned to use the licensing information for OSS projects as provided in the OpenHub database. However, it was found that OpenHub licensing information for projects often was outdated or inaccurate. Hence, manual search was considered necessary in order to obtain more reliable and up-to-date licensing information.

A different focus for future work is on use of license terms which deviate from recognised OSI licenses. Such deviations may have limited or no implications in some practical situations, whereas they may have fundamental implications in other situations. One such circumstance may be possible implications from directives or policy. We note that in some circumstances (e.g. in a policy context) it may be essential to clarify if a specific software project is provided under terms which are recognised as being provided under one (or several) specific open source licenses as recognised by the OSI. For example, in situations when a specific EU or national policy or framework agreement used for public procurement includes references to open source software which impact on what can (and what cannot) be done under the terms of such contracts.

We acknowledge that there may be a tendency that many of the widely used projects (by user count) on OpenHub are community governed projects and that such projects may have a tendency to use copyleft licenses. It is also likely

that OSS projects that have been developed over a longer time period have accumulated a larger number of users and therefore have a higher ranking by user count on OpenHub.

## Conclusions

The study reports from an exploratory analysis of 200 widely used open source projects in which licensing terms, other conditions for contributing, and the relationship between licensing terms and other conditions for contributing, have been analysed.

We find that strong copyleft licenses are most common amongst the investigated widely used open source projects, and are used in the majority of the projects. Further, a clear majority of the selected open source projects use no specific other condition for contributing in addition to the license terms. However, a clear majority of the open source projects supported by foundations use other conditions for contributing in addition to the license terms. Finally, use of no specific other conditions in addition to the license terms is more common for projects using strong copyleft licensing compared to projects using non-copyleft licensing.

Findings from the analysis of use of licenses and other conditions for contributing for widely used open source projects constitute an important contribution towards deeper insights concerning challenges related to terms under which open source projects are provided.

## ACKNOWLEDGEMENTS

This research has been financially supported by the Swedish Knowledge Foundation (KK-stiftelsen) and participating partner organisations in the LIM-IT project. The authors are grateful for the stimulating collaboration and support from colleagues and partner organisations.

## REFERENCES

1. Thomas A. Alspaugh, Hazeline U. Asuncion, and Walt Scacchi. 2009. Intellectual Property Rights Requirements for Heterogeneously-Licensed Systems. In *Proceedings of the 2009 17th IEEE International Requirements Engineering Conference*, RE (RE '09). IEEE Computer Society, Washington, DC, USA, 24-33. DOI=<http://dx.doi.org/10.1109/RE.2009.22>
2. Thomas A. Alspaugh, Walt Scacchi, and Hazeline U. Asuncion. 2010. Software Licenses in Context: The Challenge of Heterogeneously-Licensed Systems. *Journal of the Association for Information Systems* 11, 11: Article 2.
3. Nicolas Bettenburg, Ahmed E. Hassan, Bram Adams, and Daniel M. German. 2015. Management of community contributions - A case study on the Android and Linux software ecosystems. *Empir Software Eng* 20:252-289.

4. Swapnil Bhartiya. 2016. Open source speeds innovation, plays major role in NASA's mission. Retrieved April 3, 2017 from <http://www.cio.com/article/3094108/linux/open-source-speeds-innovation-plays-major-role-in-nasas-mission.html>
5. BlackDuckSoftware. 2017. Top Open Source Licenses. <https://www.blackducksoftware.com/top-open-source-licenses>
6. Amanda Brock. 2013. Understanding Commercial Agreements With Open Source Companies. In *Thoughts on Open Innovation*. Shane Coughlan (ed), OpenForum Europe. ISBN 978-1-304-01551-8. pp. 119-139.
7. Tim Engelhardt. 2013. Drafting Options for Contributor Agreements for FOSS: Assignment, (Non)Exclusive Licence and Legal Consequences. A Comparative Analysis of German and US Law. *SCRIPTed* 10,2:149–176.
8. Richard Fontana, Bradley M. Kuhn, Eben Moglen, Matthew Norwood, Daniel B. Ravicher, Karen Sandler, James Vasile, and Aaron Williamson. 2008. *A Legal Issues Primer for Open Source and Free Software Projects*. Software Freedom Law Center, New York.
9. Jonas Gamalielsson and Björn Lundell. 2014. Sustainability of Open Source software communities beyond a fork: How and why has the LibreOffice project evolved?, *Journal of Systems and Software* 89,1:128-145.
10. Daniel M. German, Yuki Manabe, and Katsuro Inoue. 2010. A sentence-matching method for automatic license identification of source code files. In *Proceedings of the IEEE/ACM international conference on Automated software engineering (ASE '10)*. ACM, New York, NY, USA, 437-446. DOI=10.1145/1858996.1859088 <http://doi.acm.org/10.1145/1858996.1859088>
11. Daniel M. German and Massimiliano Di Penta. 2012. A Method for Open Source License Compliance of Java Applications. *IEEE Software* 29,3: 58-63.
12. Rishab A. Ghosh. 2006. Study on the: Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU. Final report. Merit. 20 November.
13. Robert Gobeille. 2008. The FOSSology project. In *Proceedings of the 2008 international working conference on Mining software repositories (MSR '08)*. ACM, New York, NY, USA, 47-50. DOI=<http://dx.doi.org/10.1145/1370750.1370763>
14. Andrés Guadamuz and Andrew Rens. 2013. Comparative Analysis of Copyright Assignment and Licence Formalities for Open Source Contributor Agreements. *SCRIPTed* 10,2:207-230.
15. Armijn Hemel, Karl Trygve Kalleberg, Rob Vermaas, and Eelco Dolstra. 2011. Finding software license violations through binary code clone detection. In *Proceedings of the 8th Working Conference on Mining Software Repositories (MSR '11)*. ACM, New York, NY, USA, 63-72. DOI=<http://dx.doi.org/10.1145/1985441.1985453>
16. Sylvia F. Jakob. 2014. A Qualitative Study on the Adoption of Copyright Assignment Agreements (CAA) and Copyright License Agreements (CLA) within Selected FOSS Projects. *JIPITEC* 5,2:105-115.
17. Till Jaeger. 2010. Enforcement of the GNU GPL in Germany and Europe. *JIPITEC* 1,1:34-39.
18. Omar Johnny, Marc Miller, and Mark Webbink. 2010. Copyright in Open Source Software -Understanding the Boundaries. *International Free and Open Source Software Law Review* 2,1: 13-38.
19. Georgia M. Kapitsaki, Frederik Kramer, and Nikolaos D. Tselikas. 2016. Automating the license compatibility process in open source software with SPDX. *Journal of Systems and Software*. <http://dx.doi.org/10.1016/j.jss.2016.06.064>
20. Georgia M. Kapitsaki, Nikolaos D. Tselikas, and Ioannis E. Foukarakis. 2015. An insight into license tools for open source software systems. *Journal of Systems and Software* 102:72-87.
21. Sandeep Krishnamurthy. 2005. An Analysis of Open Source Business Models. In *Perspectives on Free and Open Source Software*. Joseph Feller, Brian Fitzgerald, Scott A. Hissam, and Karim R. Lakhani (eds). MIT Press, Cambridge, Massachusetts, 279-295.
22. Björn Lundell, Brian Lings, and Anna Syberfeldt. 2011. Practitioner perceptions of Open Source software in the embedded systems area. *Journal of Systems and Software* 84:1540-1549.
23. Catharina Maracke. 2013. Copyright Management for Open Collaborative Projects - Inbound Licensing Models for Open Innovation. *SCRIPTed* 10,2:140-148
24. Heather Meeker. 2017. *Open (Source) for Business: A Practical Guide to Open Source Software Licensing*. Second edition, CreateSpace Independent Publishing Platform, North Charleston, SC.
25. Mercedes-Benz. 2017. License Agreement Supplement. Retrieved April 3, 2017 from [http://moba.i.daimler.com/bai-cars/ba/foss/content/en/assets/FOSS\\_licences.pdf](http://moba.i.daimler.com/bai-cars/ba/foss/content/en/assets/FOSS_licences.pdf)
26. Axel Metzger. 2013. Internationalisation of FOSS Contributory Copyright Assignments and Licences: Jurisdiction-Specific or Unported. *SCRIPTed* 10,2:177-206.

27. Eben Moglen and Mishy Choudhary. 2014. Software Freedom Law Center Guide to GPL Compliance. Retrieved April 4, 2017 from [https://www.softwarefreedom.org/resources/2014/SFLC-Guide\\_to\\_GPL\\_Compliance\\_2d\\_ed.pdf](https://www.softwarefreedom.org/resources/2014/SFLC-Guide_to_GPL_Compliance_2d_ed.pdf)
28. Sioban O'Mahony. 2003. Guarding the commons: how community managed software projects protect their work. *Research Policy* 32,7:1179–1198.
29. OSI: Open Source Initiative. 2017a. <https://opensource.org/>
30. OSI: Open Source Initiative. 2017b. <https://opensource.org/faq#copyleft>
31. OSI: Open Source Initiative. 2017c. <https://opensource.org/proliferation-report>
32. Massimiliano Di Penta, Daniel M. German, Yann-Gaël Guéhéneuc, and Giuliano Antoniol. 2010. An exploratory study of the evolution of software licensing. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 1 (ICSE '10)*, Vol. 1. ACM, New York, NY, USA, 145-154. DOI=<http://dx.doi.org/10.1145/1806799.1806824>
33. Simon Phipps. 2013. 4 ways open source protects you against software patents. *Infoworld*. 8 November 2013, <http://www.infoworld.com/article/2609614/open-source-software/4-ways-open-source-protects-you-against-software-patents.html>.
34. German Poo-Caamaño and Daniel M. German. 2015. The Right to a Contribution: An Exploratory Survey on How Organizations Address It. In *Open Source Systems: Adoption and Impact*. Ernesto Damiani, Fulvio Frati, Dirk Riehle, and Anthony Wasserman (eds). OSS 2015. IFIP Advances in Information and Communication Technology, vol 451. Springer, Cham, 157-167.
35. Walt Scacchi and Thomas A. Alspaugh. 2012. Understanding the role of licenses and evolution in open architecture software ecosystems. *Journal of Systems and Software* 85:1479-1494.
36. Richard Stallman. 1999. The GNU operating system and the free software movement. In *Open Sources*. Chris DiBona, Sam Ockman, and Mark Stone (eds). O'Reilly, Sebastopol, CA, 31-38.
37. Katherine J. Stewart, Anthony P. Ammeter, and Likoebe M. Maruping. 2006. Impacts of License Choice and Organizational Sponsorship on User Interest and Development Activity in Open Source Software Projects. *Info. Sys. Research* 17,2:126-144. DOI=<http://dx.doi.org/10.1287/isre.1060.0082>
38. Timo Tuunanen, Jussi Koskinen, and Tommi Kärkkäinen. 2009. Automated software license analysis. *Autom Softw Eng* 16,3:455-490.
39. Christopher Vendome, Gabriele Bavota, Massimiliano Di Penta, Mario Linares-Vásquez, Daniel German, and Denys Poshyvanyk. 2016. License usage and changes: a large-scale study on gitHub. *Empir Software Eng*. doi:10.1007/s10664-016-9438-4
40. Christopher Vendome, Mario Linares-Vásquez, Gabriele Bavota, Massimiliano Di Penta, Daniel German, and Denys Poshyvanyk. 2015a. When and why developers adopt and change software licenses. In *2015 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, Bremen, 2015, 31-40. doi: 10.1109/ICSM.2015.7332449
41. Christopher Vendome, Mario Linares-Vásquez, Gabriele Bavota, Massimiliano Di Penta, Daniel German, and Denys Poshyvanyk. 2015b. License usage and changes: a large-scale study of Java projects on GitHub. In *Proceedings of the 2015 IEEE 23rd International Conference on Program Comprehension (ICPC '15)*. IEEE Press, Piscataway, NJ, USA, 218-228.
42. Robert Viseur and Gregorio Robles. 2015. First Results About Motivation and Impact of License Changes in Open Source Projects. In: *Open Source Systems: Adoption and Impact*. Ernesto Damiani, Fulvio Frati, Dirk Riehle, and Anthony Wasserman (eds). OSS 2015. IFIP Advances in Information and Communication Technology, vol 451. Springer, Cham, 137-145.
43. Dan Wielsch. 2010. Governance of Massive Multiauthor Collaboration – Linux, Wikipedia, and Other Networks: Governed by Bilateral Contracts, Partnerships, or Something in Between?. *JIPITEC* 1,2:96-108.
44. Yuhao Wu, Yuki Manabe, Tetsuya Kanda, Daniel M. German, and Katsuro Inoue. 2015. A method to detect license inconsistencies in large-scale open source projects. In *Proceedings of the 12th Working Conference on Mining Software Repositories (MSR '15)*. IEEE Press, Piscataway, NJ, USA, 324-333.
45. Yuhao Wu, Yuki Manabe, Tetsuya Kanda, Daniel M. German, and Katsuro Inoue. 2016. Analysis of license inconsistency in large collections of open source projects. *Empir Software Eng*. doi:10.1007/s10664-016-9487-8

## APPENDIX A

This appendix contains a list of the OSS projects included in this study, sorted on OpenHub user count (shown in brackets for each project).

1. Mozilla Firefox (13071)
2. Apache HTTP Server (9409)
3. MySQL (9130)
4. Apache Subversion (8488)
5. PHP (7746)
6. Linux Kernel (7197)
7. Bash (6381)
8. Firebug (5915)
9. Git (4950)
10. GIMP (4411)
11. Apache OpenOffice (4360)
12. GNU Compiler Collection (4210)
13. PuTTY (4207)
14. phpMyAdmin (4114)
15. Python programming language (3865)
16. GNU grep (3686)
17. VLC media player (3551)
18. TortoiseSVN (3525)
19. sudo (3380)
20. Thunderbird (3262)
21. OpenSSH (3073)
22. jQuery (3040)
23. X.Org (3025)
24. GNU tar (2952)
25. Eclipse IDE for Java (2771)
26. GNU Make (2627)
27. 7-Zip (2477)
28. GNU Core Utilities (2459)
29. Wget (2410)
30. GNOME (2388)
31. Chromium (Google Chrome) (2165)
32. GNU GRUB (2104)
33. Pidgin IM (ex-Gaim) (2057)
34. PostgreSQL Database Server (2034)
35. FileZilla (1944)
36. CakePHP (1770)
37. rsync (1765)
38. GNU Screen (1712)
39. Notepad++ (1660)
40. Apache Tomcat (1658)
41. man (1604)
42. WordPress (1558)
43. Inkscape (1544)
44. JUnit (1522)
45. Trac (1460)
46. GNU findutils (1454)
47. Perl (1393)
48. bzip2 (1380)
49. Subclipse (1370)
50. MPlayer (1314)
51. GDB (1264)
52. Wireshark (1253)
53. Apache Ant (1242)
54. GnuPG (1214)
55. Samba (1183)
56. GNU sed (1183)
57. Apache Maven 2 (1159)
58. Web Developer (Browser Add-on) (1145)
59. GNU Emacs (1129)
60. Hibernate ORM (1127)
61. LaTeX (1096)
62. Spring Framework (1094)
63. Django (1093)
64. log4j (1088)
65. cURL (1078)
66. Audacity (1066)
67. GNU Diff Utilities (1066)
68. Common Unix Printing System (CUPS) (1052)
69. Wine (1034)
70. Ruby on Rails (994)
71. Oracle VM VirtualBox (988)
72. Mercurial (983)
73. Scripting Layer for Android (969)
74. NetBeans IDE (941)
75. Nmap Security Scanner (933)
76. FFmpeg (928)
77. Funambol Client for Mozilla Thunderbird (918)
78. WinSCP (915)
79. GNU C Library (914)
80. Adblock Plus (903)
81. GTK+ (866)
82. phpBB Forum Software (848)
83. nginx (834)
84. GNU binutils (808)
85. Drupal (core) (790)
86. LibreOffice (778)
87. GNU Autoconf (777)
88. MediaWiki (776)
89. Postfix (759)
90. Cygwin (747)
91. Facebook Plugin for Pidgin (739)
92. AmaroK (719)
93. GNU Automake (719)
94. LAME (Lame Ain't an MP3 Encoder) (675)
95. gzip (656)
96. RubyGems (655)
97. NHibernate (653)
98. CVS: Concurrent Versions System (650)
99. Eclipse Web Tools Platform (WTP) (649)
100. Qt 4 (640)
101. Valgrind (637)
102. Joomla! (618)
103. zsh (614)
104. Doxygen (611)
105. NUnit .Net unit testing framework (610)
106. Apache Xerces2 J (598)
107. BIND (Berkeley Internet Name Domain) (594)
108. jQuery UI (592)
109. Evince (591)

110. Blender 3D (580)
111. tcpdump (571)
112. SpamAssassin (538)
113. Rake (532)
114. Jetty: Java based HTTP, Servlet, SPDY, WebSocket Server (523)
115. WinMerge (521)
116. irssi (521)
117. gedit (520)
118. FLAC - Free Lossless Audio Codec (515)
119. K3b (506)
120. CMake (504)
121. Boost C++ Libraries (490)
122. Prototype Javascript Framework (487)
123. Vaadin (477)
124. OpenVPN (472)
125. lighttpd (469)
126. mutt (468)
127. IPython (440)
128. GLib (438)
129. Zend Framework (438)
130. Clam AntiVirus (431)
131. PEAR (430)
132. Adium (428)
133. FreeBSD (422)
134. FUSE (420)
135. udev (419)
136. GNU Libtool (418)
137. GraphViz (416)
138. dpkg (415)
139. memtest86+ (396)
140. gettext (391)
141. AnkhSVN (390)
142. script.aculo.us (387)
143. MongoDB (386)
144. Konqueror (383)
145. Nautilus (383)
146. GNU Parted (376)
147. FindBugs (376)
148. Transmission (374)
149. Java (374)
150. XAMPP (371)
151. Apache log4net (367)
152. Greasemonkey (367)
153. JBoss Application Server (366)
154. Eclipse PHP Development Tools (PDT) (365)
155. X-Chat (364)
156. Dia (363)
157. Hudson (359)
158. GNU netcat (359)
159. Ghostscript (358)
160. Kate (KDE) (357)
161. NetworkManager (356)
162. ZXing (354)
163. ASP.NET MVC (350)
164. GStreamer (350)
165. GWT (formerly Google Web Toolkit) (349)
166. Gawk (344)
167. libpng (342)
168. libvorbis (342)
169. Vuze (formerly Azureus) (340)
170. Apache Commons Collections (338)
171. Smarty (335)
172. Xdebug (333)
173. iptables (332)
174. S.M.A.R.T. Monitoring Tools (331)
175. ibus-cloud-pinyin (330)
176. Jenkins (327)
177. Checkstyle (326)
178. xterm (324)
179. RSpec (321)
180. YSlow (321)
181. Totem (320)
182. Bazaar (317)
183. Symphony (317)
184. Pitivi (316)
185. MantisBT (316)
186. Apache Commons Logging (304)
187. Node.js (302)
188. WebKit (300)
189. Bugzilla (298)
190. Lucene (293)
191. D-Bus (293)
192. NumPy (293)
193. FreeMind (292)
194. ibus (292)
195. GNU Midnight Commander (291)
196. PHPUnit (291)
197. GNOME Terminal (287)
198. Groovy (286)
199. Redmine (285)
200. libogg (283)