Sustainability and Diversity of Open Source Software Communities:

Analysis of the Android Open Source Project

Remo Eckert, Andreas Mueller
University of Bern
Institute of Information Systems
Bern, Switzerland
e-mail: remo.eckert@iwi.unibe.ch, andreas.mueller@students.unibe.ch

***Abstract* — Open Source Software (OSS) projects rely on the efforts of thousands of software developers. The sustainability and diversity of these communities are two important factors for the long-term viability of OSS communities. This paper reviews the research on OSS sustainability and diversity. Drawing on the findings of measures on sustainability and diversity, we applied a number of the established metrics to the Android Open Source Project and found that over a third of all contributions originate from Google. Surprisingly, in 2015 we saw a decrease in the total number of contributors within the Android project. Findings from our analysis highlight the importance of sustainability and diversity for the development of OSS.**

Keywords-Open Source Software; Sustainability; Diversity; Android.

I. INTRODUCTION

Companies need to preserve their systems and digital assets for a considerable time. In such scenarios, problems would arise if the commercial vendor of adopted proprietary software were to leave the market or stop its development [1]. Popular OSS projects attract thousands of individuals and firms who collectively contribute to the software. OSS has seen a considerable increase in attention over the last few years. The success of various OSS projects such as Linux and Apache is now widely recognized. OSS has become a strategic asset for various firms who even choose to dedicate development resources to OSS projects [2]. Similarly, in the context of OSS development, if a dominant player leaves or stops its contributions to the project, this may influence the success of the OSS. Therefore, both for companies and for the individuals contributing to OSS projects, the stability and long-term success of the projects are key.

In many ICT sectors, OSS has been found to be relevant when it comes to sustainability [1]. According to Gamalielsson and Lundell [3], a primary factor for the success of any OSS project lies in the sustainability of the community. There are many aspects of OSS projects that can affect community sustainability, such as project management, incentives for contributors or the license of the project [4].

Building on diversity literature, Daniel et al. [5] show that diversity influences two critical outcomes of OSS projects - the community engagement and market success. Another recent study on social diversity by Aué et al. [6] investigated the relationship between project growth and the social diversity of OSS projects on GitHub. They found a statistically significant link between project rating and gender and geographical diversity. Drawing on measurements found in the literature on sustainability and diversity, this paper analyzes the sustainability and diversity of the Android Open Source Project (AOSP).

Section II presents our research questions. In Section III, previous research on OSS sustainability and diversity is shown. An overview of the AOSP and the OSS sustainability and diversity measures applied to the AOSP are outlined in Section IV. The discussion follows in Section V.

II. RESEARCH QUESTIONS

According to Chengalur-Smith et al. [7], sustainability is defined as “*the ability of an organism or an ecosystem to maintain its activity and productivity over time.*” However, this definition of sustainability is a vague concept. Our aim is to find measurements on OSS community sustainability. RQ1 deals with the sustainability of OSS communities and is as following:

- **RQ1: How can sustainability of OSS communities be measured?**

In their study, Daniel et al. [5] investigated whether different types of diversity influence the success of OSS. They found that projects in later stages benefit more from diversity than projects in earlier stages. RQ2 deals with the diversity of OSS communities and is as follows:

- **RQ2: How can diversity of OSS communities be measured?**

The goal of this paper is to show measurements of OSS community sustainability and diversity found in the literature, then apply and discuss them in the context of a well-known OSS project. This is why we chose the AOSP community.

 ISBN: 978-1-61208-590-6

III. LITERATURE REVIEW

This section analyzes existing work in the context of OSS sustainability and OSS diversity.

A. OSS Sustainabiliy

Gamalielsson and Lundell [3] underline how the sustainability of communities is one of the most important factors for the long-term sustainability of OSS itself. They highlight the significance of governance for sustainability, which is also identified by O'Mahony and Becky [8].

Ghapanchi [9] provides an overview of prior research on OSS development sustainability and summarizes the various drivers of OSS projects, such as: developer and user attraction, development base, project age, having developers with higher levels of different skills, project status & activity, having a nonmarket sponsor and having a copyleft license. He investigated the impact of a project's capabilities on its development sustainability and found that OSS projects are more likely to succeed if they are able to: process a higher percentage of suggested features, quickly remove identified defects and release the software them at a faster rate.

Chengalur-Smith et al. [7] tested software projects empirically in terms of their activity and contribution patterns. They used a model of project sustainability based on organizational ecology, termed Structural Equation Modelling. As a contribution to research, they provide a table of measurements, comprising indicators and descriptions. They derive the following statement as a conclusion: *"Sustainability requires certain levels of activity to be maintained over a long period of time"* [7].

Farmer and Norman [10] made a case study review on OSS sustainability in which they describe and analyze seven successful OSS projects. They define sustainability of OSS at two levels: Sustainability I with a more innovative early product stage and Sustainability II which is product and service oriented.

It is Wilson [11] and Gonzalez-Barahona [12] who provide a framework for how to measure OSS sustainability. Both have published their approaches on OSS sustainability measures on the Internet; neither has been published in a journal to date. Wilson [11] provides five key indicators as informal criteria for evaluating the sustainability of an OSS community. The key indicators are code contribution activity, release history, user community, longevity and ecosystems. Beginning with code contribution activity, contributions can be tracked and the community activity visualized through tables and charts. In combination with a release history, interesting insights about governance issues can be found. The user community is the core of the software project. Wilson [11] brings out its essence *"Software isn't sustainable without users"*. When it comes to longevity, projects pass through different phases: From creation through intense activity into a stable productive stage and then, finally, dying or becoming forked or replaced by a new project. The last key indicators shift the focus of the ecosystem onto developers and users of a project. Companies will engage or initialize an OSS project and provide their own software engineers, financial resources and other services to the project [11].Furthermore, Kilamo et al. [13] show the increasing trend for companies to release their proprietary software as OSS.

Finally, Gonzalez-Barahona [12] provides a group classification on OSS starting with the fundamental question for empirical research on OSS communities - which metrics should be used. His article classifies five metrics for sustainable OSS: activity, size, performance, demographics and diversity.

B. OSS Diversity

In their study, Daniel et al. [5] investigated whether different types of diversity influence OSS success in terms of community engagement and market success. They understand separation diversity to be the differences in position or opinion within the community. Variety diversity captures the range of information that members bring into the community. Disparity diversity specifies the power and resource differences within the communities. To measure community engagement, they calculate the contributions to the project, and to determine market success they evaluate the attention the project receives from users. In general, their empirical results show that diversity has both positive and negative effects on OSS project success. They found that projects in later stages benefit more from disparity and separation diversity than projects in earlier stages. If a project reaches later stages of development, it increases in size and complexity and attracts more users with more varied needs. If the project is dealing with more external stakeholders, it benefits from having more disparity and separation diversity within its own developers. Other findings are the positive effect of cultural separation diversity on market success where cultural separation has a negative effect on community engagement.

Vasilescu et al. [14] applied a regression analysis to GitHub data to study how gender and tenure diversity relates to team productivity and turnover. They explain why diversity attributes may be different in online groups (e.g., OSS communities) than for offline groups and identify four factors. Firstly, geographic and cultural dispersion is common in OSS and contributors rarely meet face-to-face. Secondly, OSS teams are fluid and rather task-focused. Thirdly, OSS teams are comprised mainly of volunteers and have a high turnover. Fourthly, often a small group of developers in OSS develop the majority of the software. They found that gender and tenure diversity are significant and positive predictors of productivity.

Vasilescu et al. [15] performed a user survey of software teams working on GitHub. They analyzed how teamwork and individual attributes were perceived by developers. One of their findings is that developers have embraced the inherent diversity from GitHub teams and, for the most part, benefit from it. Another positive effect is at the team level, where diversity can provide new ideas, perspectives, skills, and approaches to problem-solving.

In another study, Vasilescu et al. [16] gathered information on alias resolution, location data and gender inference techniques from a large dataset of GitHub projects. For the gender inference techniques, they applied an approach including heuristics (e.g., Russian surnames ending in -ova are female) and female/male frequency name lists for different countries.

Another recent study on social diversity from Aué et al. [6] investigated the relationship between project growth and social diversity of OSS projects on GitHub. They found a statistically significant correlation between project rating and gender and geographical diversity.

Alfaro [17] focuses on nationality diversity in global software development. Despite not including OSS, his research involves aspects relevant to OSS diversity. He states that global teams are diverse by nature since individuals come from different countries and cultural backgrounds, which is a general characteristic of many OSS projects. One of his findings is the positive effect between nationality diversity on team performance. In addition, he explains how teams with low temporal dispersion have performed better compared to teams with high temporal dispersion, independent of their degree of nationality diversity.

Diamant and Daniel [18] investigated developer's learning and the culture context in OSS projects. Their results show that diversity exposes developers to different work styles, problem-solving approaches and development techniques which offer opportunities for learning.

IV. THE CASE OF THE ANDROID OPEN SOURCE PROJECT

Android dominates the market for Mobile Operating Systems [19]. Google, under the holding Alphabet Inc. enjoys increasing economic supremacy and influence on Android. However, there are numerous other companies, non-profit organizations (NPO) and individuals who contribute to the project. The AOSP consists of an ecosystem of sub-projects and activities with numerous protagonists. There is a whole industry behind it: NPOs like, for instance, the Linux Foundation; several original equipment manufacturers such as Garmin and Huawei; as well as service companies such as eBay and Accenture [20].

Data gathering: To answer the two research questions, we cloned the AOSP repositories and extracted the commit history from the AOSP as described by Shihab et al. [21]. The commit history was cloned and converted to 1,144 XML files and consists of 14,150,546 data entries. The 1,144 XML files were merged into one large dataset which comprises the following information: author date, author e-mail, author name, committer e-mail, committer name, committer date, project, subject and commit hash codes. The author is the person who originally wrote the patch, whereas the committer is the person who applied the patch.

Data cleansing: Some entries were invalid due to data errors, missing content or unreasonable dates: for example, commits with a timestamp before the launch of the Android project, such as 1st January 1970 or, alternatively, dating from 25th April 2037. By choosing a start date of 2005-01.01, 132,527 (0.94%) observations were excluded from the dataset. The Android repositories were cloned on 2016-12-21. As the data for the full month December 2016 was not complete, choosing 2016-11-30 as the end date for the dataset meant that 9,639 (0.07%) observations were not included in the dataset. Finally, seven repositories are outside the chosen period. The final dataset consists of 14,008,380 observations from 1,135 repositories.

Single Commit Hash: An analysis of the dataset indicated how several commit entries contained exactly the same XML tags, such as alias, e-mail, date for authors and committers, subject and hash codes. The only difference was in the sub-project name. An investigation of the commit hash code's purpose and how it is generated, confirms that it is a unique identifier. The commit hash code is generated by an algorithm based on the commits content. If a change is part of several sub-projects, relevant titles and their commits will be cloned several times. In conclusion, it is questionable whether the overall number of commits for the whole AOSP is an appropriate community activity measure when an indefinite number of commits are cloned several times.

If the whole dataset of 14,008,380 observations is adjusted for the single commit hashes, the adjusted dataset retains 3,085,901 observations. This leads to the question: Which dataset is the correct one? This depends on what one wants to measure. If the overall activity of the AOSP is to be measured, then the 3,085,901 observations are more appropriate because work conducted on the code base is not overestimated through cloning the commit several times. On the contrary: if questions relating to individual committers' involvement in different sub-projects and activity in sub-projects is to be answered, then the 14,008,380 observations are appropriate.

A. *Android Community Sustainability*

To answer the first research question, this paper applies approaches and metrics from literature relating to OSS community sustainability.

Firstly, Gonzalez-Barahona [12] used five metrics to track an OSS community: activity, size, performance, demographics and diversity. Secondly, Wilson [11] suggested key indicators such as code activity, releases, user community, longevity and ecosystem to evaluate OSS community sustainability. To merge both approaches into a common basis and to start the investigation on the AOSP community development, this paper uses elements of community activity, size and demographics to answer RQ1.

Activity: Starting with activity, Gonzalez-Barahona [12] proposes this measure as a first view of how active a community is and recommends tracking the number of commits over time. As Wilson [11] describes, for a project to be sustainable, it must have contributors, and its codebase needs to be evolving.

Similar to Wilson [11], Figure 1 shows the time stamp of each commit over all Android sub-projects clustered into quarterly periods. As discussed, to relate the issue to cloned commits, the data in Figure 1 is adjusted to contain only one single commit hash.

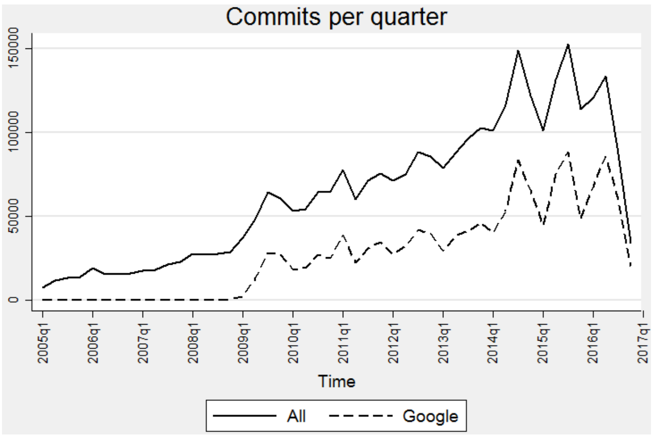

Figure 1. Commits per quarter (author perspective).

In general, Figure 1 indicates an increasing pattern of commits over time. To identify Google's share in overall commit activity, a second time series for commits originating from Google is included, using the e-mail domain part. As Figure 1 shows, Google's share of the AOSP is significant. A surprising finding is the commencement of Google's activity in early 2009, despite Google having already acquired Android Inc. in 2005. One possible explanation for this is that Android Inc. core developers may have switched to Google e-mail domains in early 2009.

Disregarding the drop in activity from summer 2016, the data curve for the AOSP indicates steadily increasing activity. This is the intuitive expectation of a successful project since, according to Wilson [11], it is "…*a good sign as it indicates that the project is picking up developers…*". A possible explanation for the decrease after summer 2016 could be that there is a time delay from the author code commit Git push until it is applied in the master branch. The following two explanations are proposed: A first possible reason for the delay could lie in the Android development process. The Gerrit process flow chart for the AOSP illustrates the different steps necessary for a change to be applied, a process that could result in a delay. It takes a certain amount of time until a proposed code change from an author, via his commit, is processed, tested, reviewed, potentially modified and finally submitted to the public depot for future synchronizations [22]. Secondly, the most compelling evidence comes from the AOSP web page which suggests that a reason for the delay could come from the next generation of Android, which would first be developed privately and then released to the public domain at a later stage [23].

Size: According to Gonzalez-Barahona [12], project size includes aspects of the number of people participating and the number of contributors in an OSS project. He specifically highlights the importance of active contributors because they lead the community and often deliver a major portion of the source code. Table I shows the 10 most active Android sub-projects, including their number of commits, authors and committers. As Table I shows, the AOSP attracts thousands of different authors.

TABLE I. 10 MOST ACTIVE ANDROID SUB-PROJECTS.

#	Sub-Project	#Commits	#Authors	#Committers
1	platform_external_linux-kselftest	616,038	15,731	581
2	kernel_hikey-linaro	567,369	15,001	576
3	platform_hardware_bsp_kernel_common_v4.4	564,251	14,814	546
4	kernel_msm	563,484	14,777	571
5	platform_hardware_bsp_kernel_common_v4.1	521,866	13,910	524
6	platform_hardware_bsp_kernel_imagination_v4.1	521,401	13,858	551
7	kernel_common	483,913	13,026	541
8	kernel_goldfish	483,913	13,026	541
9	kernel_mediatek	483,913	13,026	541
10	platform_hardware_bsp_kernel_freescale_picoimx	440,446	12,196	527

To understand the calculated figures and to compare them with the size of other OSS projects, we referred to the highest ranking projects on OpenHub.net which provides descriptive information about OSS projects, including the number of commits and different authors. Only Chromium with 578,455 commits and 5,406 authors almost reaches this level, but is in fact a Google-driven project too. Of the non-Google OSS, Mozilla Firefox with 343,841 commits and 4,813 authors reaches a comparable scale in the top-ranked AOSP sub-projects.

Demographics: Gamalielsson and Lundell [3] accentuated how important it is for an OSS project with long life-cycles to recruit new and retain current contributors to its community. An instrument to visualize this attribute on an OSS project is the aging chart proposed by Gonzalez-Barahona [12].

The main components of the aging chart are two graphical bar categories, which indicate the attracted and retained developers over time. Figure 2 shows the aging chart for the AOSP community. The attracted bar (red color) summarizes the new developers who contributed for the first time in that particular year. Unlike the aging chart proposed by Gonzalez-Barahona, a third bar is included to summarize the lost developers (blue color). This additional bar makes the representation more intuitive, as the retained developers (green color) can be estimated visually. In the Aging Chart shown in Figure 2, a developer is still considered as active

 ISBN: 978-1-61208-590-6

even if the developer made a break in some following years after the first contribution and then contributed again.

Figure 2. Aging chart of the AOSP.

The AOSP from 2005 onwards indicates a rapidly increasing trend in attracting new developers, which peaks in 2014, which is in general a good sign for the AOSP. Nevertheless, the number of developers leaving the project is also increasing and exceeds the number of new attracted developers in 2015. As indicated through the retained bars, the absolute peak of the AOSP developers is reached in 2014 and then starts to decline. 2016 is not representative, since the data set does not include the entire year. Furthermore, another fuzziness which influences 2016 data is the delay on reported commits from announcement until release on the Android repositories, as discussed. It seems that the AOSP reached its peak in 2014. It would be interesting to investigate for the coming years whether the AOSP manages to retain a stable amount of developers.

B. *Android Community Diversity*

To answer the second research question, this paper applies approaches and metrics from literature relating to OSS community diversity.

Wilson [11] described the importance of the ecosystem, taking into consideration the diversity between the companies that engage with a project. Likewise, Gonzalez-Barahona [12] questions what will happen if a major contributor leaves the project. Considering the composition of the AOSP in which a major firm such as Google takes on a significant role in the dispersion and enhancement of OSS, major contributors shall be identified.

To determine whether a contributor had a commercial, independent or educational background, an approach comparable to Heppler et al. [24] working with the domain-part of the e-mail address was used. The entries were manually selected and classified based on the number of contributions. If the domain-part included ".edu", for example, we labelled the entity as Education; ".org" specified an organization. Companies were manually identified through their company domain, such as "google.", "Samsung." or "ibm.". Overall, 197 entities were manually selected.

A total of 300,142 (9.8%) observations remain unidentified and will not be further disentangled, because the effort increases disproportionately for every fraction additionally identified. The search field controls the accuracy of the identified entities. More specific terms increase the accuracy but lower the possibility of identified entities. For instance, by choosing "intel" instead of "intel." more entries can be identified. It was decided on a case to case basis whether more specific or open terms were used to identify an entity.

Tables II and III list the top 15 entities and the number of commits from an author and committer perspective.

TABLE II. TOP 15 IDENTIFIED AUTHORS.

#	Entity	Branch	#Commits	%
1	Google	Firm	1,189,998	38.6
2	Individual Gmail	Individual	200,155	6.49
3	Jet Brains	Firm	145,140	4.7
4	Android	Firm	122,551	3.97
5	Intel	Firm	94,278	3.06
6	Linux Foundation	Organization	87,274	2.83
7	RedHat	Firm	74,110	2.4
8	Apple	Firm	73,855	2.39
9	Chromium	Organization	52,933	1.72
10	not provided	Individual	31,804	1.03
11	Gentoo	Organization	29,079	0.94
12	Suse	Firm	24,930	0.81
13	Samsung	Firm	24,621	0.8
14	IBM	Firm	23,415	0.76
15	Linaro	Organization	22,279	0.72

TABLE III. TOP 15 IDENTIFIED COMMITTERS.

#	Entity	Branch	#Commits	%
1	Google	Firm	833,857	27.02
2	Android	Firm	546,935	17.72
3	Linux Foundation	Organization	267,127	8.66
4	Jet Brains	Firm	145,381	4.71
5	Individual Gmail	Individual	140,068	4.54
6	RedHat	Firm	81,361	2.64
7	Intel	Firm	78,278	2.54
8	Apple	Firm	70,496	2.28
9	Suse	Firm	49,317	1.6
10	Chromium	Organization	42,110	1.36
11	not provided	Individual	32,499	1.05
12	Gentoo	Organization	28,792	0.93
13	Linaro	Organization	25,718	0.83
14	Go Lang	Individual	25,539	0.83
15	Kitware	Firm	23,706	0.77

The results demonstrate Google's dominant position with 38.6% of all identified commits. Other major contributors came from companies such as Intel, Apple, Samsung and IBM. Top ranked firms such as Suse and RedHat sell OSS services and distribute their own Linux distribution. Other top contributors are Jet Brains, Gentoo and Linaro. An interesting observation was that the Linux Foundation made a contribution to the AOSP that, at 8.66% was higher on the committer side than on the author side (2.83%). In other words, the Linux Foundation was more focused on reviewing and applying changes than in bringing in new ones.

 ISBN: 978-1-61208-590-6

Gini coefficient: The Gini coefficient is a measure to determine inequality. The Gini coefficient is often used to express a status criterion because it describes heterogeneity and is the most frequently used measure of inequality [25]. In our case, the Gini coefficient is used to portray the inequality in the distribution of contributions by authors. The Gini coefficient indicates how unequal contributions are distributed among the authors. A Gini coefficient of zero expresses equality, whereas a coefficient of one expresses inequality. Using the Gini coefficient for the AOSP allows comparison to other OSS or, in an extended perspective, to other applications in general. Figure 3 shows the Lorenz curve of the AOSP.

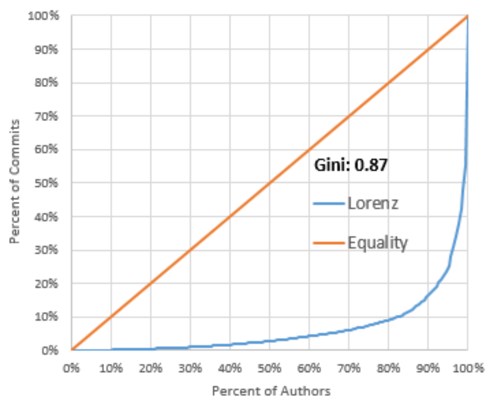

Figure 3. Lorenz curve for the AOSP.

The Gini coefficient of 0.87 for the AOSP is close to one, which describes a strongly unequal distribution. The main reason for this is Google's relative large share (38.6%) in the overall number of commits. However, an unequal distribution of commit activity in OSS projects is not uncommon [26].

V. DISCUSSION

This paper contributes to the literature on OSS community sustainability and diversity in two ways: First, it provides a broad literature review on OSS measurements of diversity and sustainability; and, second, it applies a number of these measures to the AOSP. Both the literature on OSS community sustainability and on diversity highlight the importance of the OSS ecosystem for the development of OSS.

According to Gamalielsson and Lundell [3], a primary factor for the success of any OSS projects is the sustainability of their community. Sustainability of OSS communities can be measured in various different ways, as our literature review shows. When it comes to diversity, there are effects on the long-term viability on OSS, since diversity influences the OSS success in terms of community engagement and market success, as Daniel et al. [5] show.

In our paper, we examine these two concepts with special regard to the AOSP project. As we have demonstrated, the AOSP ecosystem has grown steadily since 2005. The AOSP has been able to maintain a high level of activity over a prolonged period of time. As the aging chart indicates, the AOSP attracted new developers and kept them within the project between 2005 and 2014. However, in 2015, there is a surprising result: the number of lost developers is higher than the number of new developers attracted. Because the dataset for 2016 is incomplete, we are unable to determine whether this is the same for 2016. If Google is not able to attract and retain developers for the AOSP, the sustainability of the project may be decreasing. A project's ability to attract developers and active user resources was found to have a positive effect on project sustainability [7].
Google's dominant position within the AOSP cannot be overlooked. The overall size of the AOSP is vast and is separated into several sub-projects, each of which is comparable in size and activity to other complete OSS projects, such as Mozilla Firefox.

Although hundreds of different firms, organizations and individuals were identified, at 38.6% of all commits, Google's share is tremendous. Moreover, the Gini coefficient of 0.87 for the AOSP shows that the distribution is highly unequal. Google's dominant role poses risks for the AOSP if Google were to leave the project or decrease its investments in it.

Communities with a small number of major contributors are more dependent on those contributors than are communities with several major contributors. Diversity is therefore important for the resilience of an OSS community. The more diverse a community in terms of the different individuals and organizations contributing to the project, the less dependent it is on a single contributor and the more resilient it will be. However, having a major contributor for an OSS project can boost its development.

It would be beneficial for future research to analyze the implications on the project success when core contributors leave the project. The relationship between core contributors and project success could bring some interesting insights on the sustainability and diversity of OSS communities.

ACKNOLEDGEMENTS

The authors thank Michael Single for his constructive suggestions and comments on this research.

REFERENCES

[1] B. Lundell, B. Lings, and A. Syberfeldt, "Practitioner perceptions of Open Source software in the embedded systems area," *Journal of Systems and Software*, vol. 84, no. 9, pp. 1540–1549, Sep. 2011.

[2] G. von Krogh and S. Spaeth, "The open source software phenomenon: Characteristics that promote research," *The Journal of Strategic Information Systems*, vol. 16, no. 3, pp. 236–253, 2007.

[3] J. Gamalielsson and B. Lundell, "Sustainability of Open Source software communities beyond a fork: How and why has the LibreOffice project evolved?," *Journal of Systems and Software*, vol. 89, pp. 128–145, 2014.

[4] A. Bonaccorsi and C. Rossi, “Comparing motivations of individual programmers and firms to take part in the open source movement: From community to business,” *Knowledge, Technology & Policy*, vol. 18, no. 4, pp. 40–64, 2006.
[5] S. Daniel, R. Agarwal, and K. J. Stewart, “The Effects of Diversity in Global, Distributed Collectives: A Study of Open Source Project Success,” *Information Systems Research*, vol. 24, no. 2, pp. 312–333, Jun. 2013.
[6] J. Aué, M. Haisma, K. F. Tómasdóttir, and A. Bacchelli, “Social Diversity and Growth Levels of Open Source Software Projects on GitHub,” in *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 2016, p. 41.
[7] I. Chengalur-Smith, A. Sidorova, and S. Daniel, “Sustainability of free/libre open source projects: A longitudinal study,” *Journal of the Association for Information Systems*, vol. 11, no. 11, p. 657, 2010.
[8] S. O’Mahony and B. A. Bechky, “Boundary organizations: Enabling collaboration among unexpected allies,” *Administrative Science Quarterly*, vol. 53, no. 3, pp. 422–459, 2008.
[9] A. H. Ghapanchi, “Predicting software future sustainability: A longitudinal perspective,” *Information Systems*, vol. 49, pp. 40–51, 2015.
[10] J. Farmer and J. Norman, “A case study review of open source sustainability models.” JISC University of Oxford, 2007.
[11] S. Wilson, “How to evaluate the sustainability of an open source project,” 11-Dec-2013. [Online]. Available: http://oss-watch.ac.uk/resources/evaluatingsustainability. [Retrieved: August 2017].
[12] G.-B. Jesus M., “Top 5 open source community metrics to track,” 2015. [Online]. Available: https://opensource.com/business/15/12/top-5-open-source-community-metrics-track. [Retrieved: August 2017].
[13] T. Kilamo, I. Hammouda, T. Mikkonen, and T. Aaltonen, “From proprietary to open source—Growing an open source ecosystem,” *Journal of Systems and Software*, vol. 85, no. 7, pp. 1467–1478, 2012.
[14] B. Vasilescu *et al.*, “Gender and Tenure Diversity in GitHub Teams,” in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 2015, pp. 3789–3798.
[15] B. Vasilescu, V. Filkov, and A. Serebrenik, “Perceptions of Diversity on Git Hub: A User Survey,” in *Cooperative and Human Aspects of Software Engineering (CHASE)*, 2015, pp. 50–56.
[16] B. Vasilescu, A. Serebrenik, and V. Filkov, “A data set for social diversity studies of GitHub teams,” in *Proceedings of the 12th Working Conference on Mining Software Repositories*, 2015, pp. 514–517.
[17] I. Alfaro, “Nationality diversity and performance in global software development teams: The role of temporal dispersion and leadership,” in *International Conference on Information Systems (ICIS)*, 2010, pp. 116–134.
[18] E. I. Diamant and S. L. Daniel, “Learning in Open-Source Software (OSS) Development: How Organizational and National Culture Impact Developers’ Learning.,” in *International Conference on Information Systems (ICIS)*, 2010, pp. 66–76.
[19] International Data Corporation (IDC), “Smartphone OS Market Share 2016 Q3,” 2016. [Online]. Available: http://www.idc.com/promo/smartphone-market-share/os. [Retrieved: August 2017].
[20] Open Handset Alliance, 2017. [Online]. Available: http://www.openhandsetalliance.com. [Retrieved: August 2017].
[21] E. Shihab, Y. Kamei, and P. Bhattacharya, “Mining challenge 2012: The android platform,” in *Proceedings of the 9th IEEE Working Conference on Mining Software Repositories*, 2012, pp. 112–115.
[22] A. Hoog, *Android Forensics: Investigation, Analysis, and Mobile Security for Google Android*. Elsevier, 2011.
[23] Android Source, “Codelines, Branches, and Releases | Android Open Source Project,” 2017. [Online]. Available: https://source.android.com/source/code-lines.html. [Retrieved: August 2017].
[24] L. Heppler, R. Eckert, and M. Stuermer, “Who cares about my feature request?,” in *IFIP International Conference on Open Source Systems*, 2016, pp. 85–96.
[25] P. M. Blau, “A macrosociological theory of social structure,” *American journal of sociology*, vol. 83, no. 1, pp. 26–54, 1977.
[26] M. Goeminne and T. Mens, “Evidence for the pareto principle in open source software activity,” in *the Joint Porceedings of the 1st International workshop on Model Driven Software Maintenance and 5th International Workshop on Software Quality and Maintainability*, 2011, pp. 74–82.

 ISBN: 978-1-61208-590-6