A Two Phase Case Study on Implementation of Open Source Development Practices within a Company Setting

Alma Oručević-Alagić, Martin Höst
Department of Computer Science, Lund University, Sweden
(alma.orucevic-alagic, martin.host)@cs.lth.se

Abstract

Implementation of open source development practices within commercial settings can bring benefits such as improved source code quality, lower maintenance costs, and increased innovation. However, a widespread in-house implementation of the practices has not been observed. The goal of this research is to understand factors which hinder the implementation. For the purpose, development practices of a large, global software and hardware organization that bases its products on open source software, and has over a decade long experience of contributing to various open source projects were studied. The results were validated through a set of structured interviews and a focus group meeting. It is found that the initial implementation of the process has not been carried out in a planned and systematic way within the company. The results of the follow-up focus group meeting show that while the company's practices acquired a higher degree of alignment over a two-year period, the change was necessitated by a need to have a more efficient development effort across new, globally distributed, development sites.

1. Introduction

Open source software (OSS) has influenced the way software is produced and distributed. Some companies use open source as another type of off-the-shelf software, while others integrate it into their software products and actively participate and contribute code to open source communities, see for example Höst and Oručević-Alagić [6]. By participating in OSS development processes, companies gain experience in how online, distributed collaboration effort is organized, and how information and knowledge are managed in these projects. While the software built by OSS communities spans a wide range of domains, sizes, and maturity levels, an increased industry interest and involvement with OSS came with the emergence of a large, complex, and industry-grade software products, such as Linux, Android, and Hadoop [19].

Most common profile of the developers contributing to OSS projects is that of unpaid, geographically distributed, and well integrated into highly organized and structured fabric of OSS projects [13]. However, with increased usage and participation of the industry in OSS projects, a greater participation of paid developers can be noted, and as OSS gains mainstream acceptance, business models and industry involvement strategies mature as presented e.g. by Fitzgerald [2]. However, understanding how OSS development practices could be applied in-house to enhance proprietary software development process requires further investigation for several reasons. Scacchi [16] argues that OSS development is an interesting alternative approach to development of large systems and suggests that further research, especially using empirical examination, is conducted in order to better understand OSS development practices (OSDP). A description on adopting open source development practices within the organizations, also known as inner source, was proposed by Stol and Fitzgerald [18]. They show three important aspects of software projects that are run as inner source: types of projects, practices and tools, and people and management. While there exist a number of case studies showing successful adoption of OSS in-house, HP [11], Lucent [4], and Nokia [10], more evidence is still needed to better understand how companies can apply OSDP.

In this case study we present alignment of software development practices and OSDP in a large, international, software and hardware company, referred to as the Case Company, that bases its products on open source software. The company has a long experience of working with mature communities, and has recognized the value of the OSDP. This study is a second part of the two phase study [12] which tracks the adoption of OSDP in commercial setting over a two year period.

The outline of this paper is as follows. In Section 2, the background information on OSDP is presented. In Section 3, the research approach is further defined. Section 4

DOI reference number: 10.18293/SEKE2016-088

presents the obtained results, while Section 5 discusses and analyses the obtained results in some more detail. Finally, conclusions are drawn in Section 6.

2. Background and related work

Examples of studies that demonstrate successful application of OSDP within commercial settings are the ones conducted in HP [11], Lucent [4], and Nokia [10]. The studies analyzed, for example, development of a complex software product across departments [4] using OSDP, and transitioning of an entire development to adopt OSDP [11]. The software produced in such manner is called "inner source", "progressive open source", or "closed open source".

A case study conducted by Stol et al. [17] focuses on the challenges of building and integrating software products developed as a shared asset. In this case study the focus is on challenges of developing and integrating software developed as a shared asset within the company setting, and comparing these challenges with the challenges of integrating an open source software product developed outside of the company. The Stol et al. research [17] concludes that organizations can benefit in adoption of OSDP, but that more research in the area is needed to further identify and address the challenges of OSDP in company settings.

Melian and Mähring conducted a study in HP [11] observing the process of progressively transitioning HP's development team to work under OSDP. The motivation for introduction of POS in HP is the business need to increase the development cost efficiency and shorten time to market by making software highly modular and reusable asset. The research has produced a comparative listing of open source and "progressive open source" development practices. Some of the biggest differences between the two practices lied in the aspects of organizational structure, time and budget to deliver, abundance of available human resources and reward system. They conclude that implementation of OSDP within a corporate setting can bring long lasting benefits in terms of development efficiency and code quality, but they also state that more research is needed to address differences in reward system, and control and monitoring of individual participants.

A case study conducted by Gurbani and Gavert in Lucent [4] provides another relevant insight into what happens when a software product is developed within a company as a shared asset, and employees from other departments are involved in its development through development process compliant with OSDP. The lessons learned form that case study are that source code ownership and the "many eyeballs" contributing to a transparent development process facilitate efficient software development especially if the software product is shared and highly utilized across different departments as it was the case in that case study [4].

All the studies referenced above ([11], [10], [17], [4]) identified the importance of having a common set of standard development tools, a single version control system, and a standardized change management system.

For the purpose of this study a set of Open Source development practices was identified based on the work by Fogel [3] and outlined in Table 1. The work details the most important aspects and characteristics of free software development, based on experiences gained with the Apache Subversion project [1], the OSS source code version control system with widespread use in open source and corporate setting. It provides a valuable insight on how the Subversion [1] community has been built and sustained over a period of twelve years. Besides analyzing the infrastructure needed to support the project in an online environment, Fogel [3] also elaborates on the importance of building a healthy environment culture, facilitating authority based on meritocracy and communication relying on standardized channels and formats.

While shared OSDP aspects, across different sizes and domains of mature OSS projects, include Infrastructure and Communication aspects, defined by IDs S1-S21, in Table 1, they can differ in governance types, defined by IDs S22-S24. For example, the Linux project adheres to the "benevolent dictator" management practice, where lieutenants are assigned for different parts of the code, but the ultimate decisions are made by Linus Torvalds. The community source governance type for library and related fields software, popularized by Kuali Open Library Environment [9], enables institutions to share development resources and influence development of a software project in a closed source setting, provided that in the later phase the project is open sourced. The government type applied in the Apache Subversion project is base on meritocracy, also referred popularly as "do-ocracy", where roles, authority and promotion is based on the participants' demonstrated knowledge and contributions to a project. We argue that such governance model can be suitable also for a closed source industry setting.

While the adoption of open Source practices can benefit companies [5], there are also some issues it raises. Some of the issues include development of products across organizational boundaries, especially in the companies where the development process is highly hierarchical.

3. Research methodology

3.1. Research approach

The research presented in this study is conducted as focus group meeting [14], [8] and it represents a second phase of the two-phase study as depicted in Figure 1, which depicts the entire research process. The participants of the

Table 1. OSS Framework

Aspect	Category	Subcategory	Id.
Infra-structure	Product Info	Features	S1
		Documentation	S2
		FAQ	S3
		News	S4
		Road Map	S5
		Security	S6
	Code Access	Download location	S7
		Binary package	S8
		Release Notes	S9
	Community Guide	Community Overview	S10
		Community Roles	S11
		Coding Conventions	S12
		Commit Conventions	S13
		Building and Testing	S14
		Debugging	S15
		Mailing Lists	S16
		Bugs/Issues	S17
		Releases	S18
Commu-nication	Standardized	Message	S19
		Channel	S20
		Norm	S21
Manage-ment	Meritocracy	Role	S22
		Promotion	S23
		Authority	S24

focus group meeting were the same individuals that were involved in the execution of the first phase of the study and thus were well acquainted with purpose and details of the study. The meeting discussion was structured around predefined interview questions with overall goal of assessing current level of OSDP implementation within the company and identifying and understanding factors behind any changes in the implementation.

The Case Company is a global market leader in software and hardware production within its field, and its core products are based on an OSS product licensed under GPL. The company has over a thousand employees and, through own and partners' offices, it is present in over 170 world countries. The Case Company is also a significant contributor to a number of different OSS communities.

The main questions that are analyzed in the study are:

RQ1 What differences in the level of implementation of OSDP presented in Table 1 can be noted over the past two years in the Case Company?

RQ2 What are the underlying reasons for the change in the OSDP implementation levels?

The main research question, RQ2, tries to understand and answer why certain practices are introduced and other practices are not introduced. In order to understand this, RQ1 focuses on what has happened at the case company during the last two years.

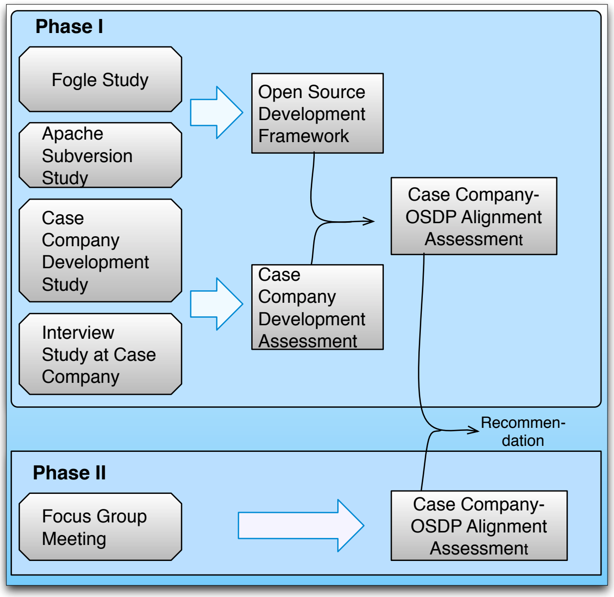
Phase I
Fogle Study
Apache Subversion Study
Case Company Development Study
Interview Study at Case Company
Open Source Development Framework
Case Company Development Assessment
Case Company-OSDP Alignment Assessment
Recommen-dation
Phase II
Focus Group Meeting
Case Company-OSDP Alignment Assessment

Figure 1. The two phase research process

In order to answer the questions it is necessary to understand the status before the two years started and what has happened during the two years, and why these changes have been made. Phase I (see Figure 1) of the study answers what practices were introduced two years before Phase II. Phase I, as presented in [12], was conducted by studying the alignment of the case company and Open Source practices, and to what extent developers thought that practices would be feasible to introduce in their environment. This was done by observing the case company and conducting interviews six persons with roles covering project management, technical lead, architect, and developer. Phase I was carried out over a period of 2 months with the first author visiting the case company during this time. Some further details about the conducted research are presented when the research validity is discussed in Section 3.2.

Phase II was, as described above, conducted through as a focus group meeting. In order to understand what has happened after Phase I, and by that being able to answer RQ1 and R2, a number of discussion questions were phrased and used at the focus group meeting:

1. How often do you use intra company online resources to discuss or solve project related tasks(e.g. online communication, knowledgbase, ongoing projects)?

2. How are change requests handled?

3. Can developers or smaller teams independently choose task to work on?

4. Which aspects of OSDP need to be implemented at

greater level and why?

In order to facilitate discussion in an efficient way, the following format was followed. For each discussion question participants were given some time to think about the question and write down their answers on an enclosed questionary. After formulating individual answers, each participant shared his answer with the group, which presented bases for a discussion. The individual answers were then collected by the researchers. During the discussion, one of the researchers also took notes.

The participants answers were transcribed, systematized based on commonality of their responses, and further analyzed by comparing them to the notes taken by one of the researchers. Based on this, a report was developed with summaries for each of the interview questions. During the analysis the results were also compared to the results from Phase I. The summaries are presented in Section 4 where they are also analyzed and compared to the results from the first phase of the study.

3.2. Validity

In this section the validity of the research is analyzed with respect to the types of validity threats typically present in qualitative studies: construct validity, internal validity, external validity, and reliability [14], [15]. As this study is a continuation of the two phased study, the validity assessment includes both of the phases.

The *construct validity* is concerned with the relationship between the subject of the study and what is measured, in this case the alignment of OSDP of mature OSS communities and software development practices of the Case Company. In order to properly identify mature Open Source development practices, the work by Fogel [3] was used and the result was verified by studying the Apache Subversion [1] project. To assess the development practices within the Case Company the first researcher spent two months in the company, studying the company's processes and examining online communication trails and documentation.

Thus, a prolonged involvement [15] was applied in order to improve the validity of the research. The results of the documentation study were discussed and validated with the Case Company senior employees through a set of six structured interviews, i.e. member checking. The results were also reviewed by the second researcher who did not spend time in the company, i.e., peer debriefing. This also reduces the possible bias that the first researcher might have developed with a prolonged involvement. It also means that research triangulation was applied which also increases validity of the research. The participants included in Phase II of the study were the same individuals as the ones that participated in Phase I of the study, and thus were well acquainted with the subject of research and previous work completed. The participants formulated the answers to the questions themselves, and participated in the discussion that followed which gave the opportunity to discuss and clarify their written answers even further. There exists possibility that participants were not representative sample, but the chances for this are very small.

The *internal validity* is concerned with causal relations. Since the nature of this study is to compare and analyze development practices, the causal relations are not seen as a threat of the study.

The *external validity* is related to the ability to generalize the results of the this study. The OSDP as defined in the paper can be relevant for future analysis. The Case Company studied is large software and hardware company, a world leader in its field, where the main products are built around OSS products. Hence, the participants are experienced in working with mature OSS communities. The competition in the market is typical. Hence, the findings of this research might be relevant to other large software companies that consider implementation of OSDP internally. It provides a framework of characteristics present in OSDP and an insight on benefits and challenges on implementing OSDP within a company setting.

The *reliability* aspect of validity is concerned with the aspect of data and analysis dependence of the underlying research on the researchers. The study was conducted as a prolonged, two phase, structured case study, with the analysis, interviews, and focus group meeting conducted in a structured way.

4. Results

4.1. RQ1

Research Question RQ1 concerns the changes in alignment of OSDP and software development practices of the Case Company over the past two years. To answer RQ1, the results of the first phase and second phase of the study are compared. The results are grouped under the three aspects as outlined in Table 1.

4.1.1 Infrastructure

The web portal is well structured and contains documents with information on organization structure, administrative information, roles and responsibilities, information on development processes, methods, standards, past, and ongoing project related information, code repository, use-net groups, and training manuals. Development processes and the methodology are well defined, with a project management process which is best categorized as a set of sequential steps The coding standards are clearly spelled out in the documentation. There exists ongoing project documentation mostly with information on project management plans, allocated resources, assigned tasks, and task completion.

In the first phase of the study only two interview study participants indicated that they use the portal in their daily work, the architect and the senior project manager. The two interviewees used it for the purpose of updating project management plans or technical documents. At the same time developers, code block architects, and technical leads indicated that they used the portal very little in their daily project related tasks. They also agreed that the documents on the portal were not well organized and that much of the documentation they were interested in was out of date.

In the second phase of the study all focus group participants indicated that they use the company portals several times on daily bases. The portal is used to get up-to-date information on current bugs, fixes, and releases, to prioritize backlogs, to review design and architecture, to engage in communication about software projects, to follow mailing lists for different groups and projects, and as development wiki.

4.1.2 Communication

The internal portal also hosts infrastructure necessary to carry out discussions on various topics and create searchable archives.

In the first phase of the study the majority of the interviewees agreed that a majority of inefficiencies and issues they encounter in their daily work are related to inadequate communication. Most thought that better communication would lead to more efficiency at work. They expressed that usage of electronic communication in a standardized form would be desirable, especially if it would create searchable archives which could later on be referenced for problem solving purposes, similarly to how they would search the internet to understand why programs produce certain error codes and how such issues could be resolved. On the other hand, the majority of the interviewees agreed that it is much more time-efficient and easy to "go and talk" to a person about a problem, recognizing that in this way no written trail on the problem would be left. While there exist non-standardized means to communicate electronically, some interviewees said that the majority of developers refrain from using it, partly due to past experience, where questions and issues brought up through electronic discussions were not addressed in a manner that would facilitate such discussion.

In the second phase of the study all of the participants indicated that they engage in online communication for the purposes of solving ongoing issues, discussing ongoing project work, and searching for information through communication logs, mailing lists, on daily bases. The participants also noted that the increase in online communication was necessitated by opening of a new, global distributed development sight. They pointed out that in the distributed development environment the "go and talk to a person" option was no longer viable and they also recognized the benefits of having the communication archives of the discussions available as they provided a searchable information trail. This has also encouraged different area knowledge experts to produce wiki documents on the most frequently asked questions, which reduced time inefficiencies in repeating the same information.

4.1.3 Management

In the fist phase of the study it was found that the organizational structure and roles and responsibilities within the R&D resemble roles which can be found in OSS communities. Hence, besides developers, there are technical leads, code block maintainers, code block architects, and architects. The code block maintainers code block architects and can be seen as fulfilling the roles of module maintainers and release manager (e.g. [7]) in the open source community. The majority of developers were positive towards the idea of being able to select tasks they would work on from a pool of tasks in the similar manner as this is done in OSS communities. However, interviewees that were in manager positions indicated that this might not be feasible as much time would then need to be spent on managing conflicts for those developers that could not choose tasks or were assigned to less interesting tasks. All interviewees agreed that task deadlines are needed, but sometimes too tight deadlines tend to negatively affect quality, as there then exists a tendency to put in much functionality without properly testing it.

Five of the interviewees thought that the number of formal meetings held was excessive. They expressed that if more time was put in planning of the meeting and appropriate selection of the attendees, the meetings might be less frequent and more efficient. Interviewees at more advanced technical position believed that there was a tendency to involve them into projects too early or too late.

Code block architects and code block maintainers noted that in practice their roles overlap with the role of technical lead. Such overlapping roles on the project are conflicting, as technical lead is perceived to be more of a project driver, while code block architects and maintainers are considered to be expert of a product or a part of it with a sole role of making sure that underlying product development is in line with overall architecture.

In the second phase of the study there were no changes noted with respect to the development roles which continued to be aligned with roles observed in mature OSS communities. The development teams are specialized in a part of the platform and do not have the opportunity to initiate changes independently without consulting the platform owner. However, in case the changes are small, e.g. not affecting common APIs, not conflicting with customer requirements, and development resources are available, teams can independently implement the changes. An increased usage of online communication channels has positively affected the management aspect resulting in less unnecessary meetings, and greater information dissemination resulting in involving appropriate technical resources in early stages of project planning. However, the communication is still restricted to development resources, with platform owners serving as links to other project stakeholders or end customers. Hence, there is no direct feedback loop between the development teams and end customer, as is the case in OSS communities.

4.2. RQ2

Research question RQ2 concerns the underlying reasons for the changes in the OSDP implementation levels. Based on the results of the focus group meeting, the underlying reasons for change in level of the OSDP implementation practices is opening of a new distributed development site. This necessitated greater level of online communication has taken up the same characteristics as the ones observed in OSS communities. The increase in online communication manifested itself in greater usage of mailing lists, project wiki pages, and other online resources, e.g. for project management, bug reports, etc., thus transferring some of the work from 'live' interactions to an online milieu. This results in the creation of searchable archives, which further increased the development efficiency.

5. Discussion

The level of the alignment of the Case Company development practices with OSDP has increased over a two year period, with the change being driven by the need to achieve greater development efficiency across distributed development sites.

Basing its core software and hardware products on OSS and with development teams highly experienced and versed in OSDP, the Case Company has mimicked characteristics of OSDP as presented in Table 1. The company has replicated many of the roles present in OSS communities, such as the role of code block architects, technical leads, and head architects, and they have implemented an online portal and setup usenet groups. Development resources are aligned with OSDP in respect to the common understanding of technical issues and value of standardized practices in design, coding, testing, and development stages. Hence, in both stages of the study, experience gained from through participation in the OSS community process were transferred back into the company.

Due to the lack of planned effort to encourage company-wide usage of an online communication milieu, in the first phase of the project it was observed that such resources are used scarcely. While a majority of developers preferred having searchable communication archives, a starting place where one could go to find out if there exists more information about an investigated issue, they perceive that communicating electronically instead of “face-to-face” is less efficient. The interviewees also indicated a reluctance to take part in open electronic discussions and such discussion are not formally encouraged or enforced to any degree. Face-to-face communication reduces the amount of time one needs to spend searching through archives and can also ask “on-demand” for further clarification of an issue. On the other hand, 'face-to-face' communication can be less efficient in case the resources one needs to talk to are not currently available. The second phase of the study showed significantly improved usage of company portals since they enabled more efficient communication and development between the existing and a new globally distributed sites. Transferring some of the 'face-to-face' communication to an online milieu created a searchable communication archives, and improved the documentation process, e.g. creating more documentation on the system and “F.A.Q. lists”. The more transparent, online, nature of discussions on ongoing projects helped involve relevant resources in early stages of project planning, thus reducing the number of unnecessary meetings and ensured that relevant inputs are acquired early on.

The greatest misalignment still is evident in the way work is assigned and projects are managed. There is still a closed feedback loop between developers and outside company partners and customers, and much of this communication is channelled through higher, management level roles, such as platform owners. Developers only make up-stream change request, without involving platform owners in cases where the changes are minor and not affecting common APIs, provided that there resources available.

In the studied case, it seems like the matter of adopting OSDP is highly motivated and carried out by technical personnel that has recognized its benefits through experience with OSS communities. The management structures need to be further educated on the OSDP so an appropriate adoption process can be found and implemented, as this was the case presented in earlier studies, e.g. at HP [11] and Lucent [4].

6. Conclusions

The results of the prolonged case study show that as a company expends and acquires geographically distributed development sites, adoption of OSDP is preferred and a natural way to engage all software product stakeholders in the most efficient way. This is a relevant finding, as with current need for highly skilled work force, many companies struggle with hiring appropriate human resources in one location, and are forced to either open new sites or outsource some part of development.

Unlike the previous studies which show how systematically planned OSDP implementation is carried out, we show what can happen when such process is carried out in a less planned way, driven by individuals in highly technical roles with extensive experience in working under the OSS process. Characteristics of such approach have shown that OSDP are implemented to a higher degree in a form of infrastructure, and less in a form of communication and management practices. Hence, there exist technical roles modeled around the software product, such as head architect and code block maintainer. There also exist standardized development practices and processes facilitating cross project work. The Case Company portal is created with the purpose of resembling an OSS community online milieu, but in practice, during the first phase of the study, the content of the portal was not well organized, complete, or up-to-date. However, with a new, geographically distributed development site, the second phase of the study observed that the online resources were used more, and thus more resembling OSDP. The greater transparency brought through increased online communication and work has also ensured that appropriate resources are involved in earlier stages of project planning, thus also reducing unnecessary meetings. Repetitive, time consuming tasks, such as 'go-and-talk-to expert' were reduced by having online resources, such as up-to-date documentation and a project wiki.

The prolonged case study presented in this research shows benefits and challenges of implementing OSDP within a closed company setting, when the adoption is not carried out in a systematic and planned way, but rather driven by employees in highly technical roles with experience of working under the OSS communities. We believe that there are more case companies undergoing the same challenges, especially as the software industry increases usage of OSS products and related business models. For this reason, more studies of similar type are needed, not only to raise awareness to the possible problems, but primary to better plan the adoption process so its full benefits can be taken advantage of.

References

[1] ApacheSubversion. Apache subversion open source project. http://subversion.apache.org/, 2012.

[2] B. Fitzgerald. The transformation of open source software. *MIS Quarterly*, 30(3):587–598, 2006.

[3] K. Fogel. *Producing Open Source Software: How to Run a Successful Free Software Project*. O'Reilly Media, first edition, Feb. 2013.

[4] V. K. Gurbani, A. Garvert, and J. D. Herbsleb. A case study of a corporate open source development model. In *Proceedings of the 28th International Conference on Software Engineering*, ICSE '06, pages 472–481. ACM, 2006.

[5] V. K. Gurbani, A. Garvert, and J. D. Herbsleb. Managing a corporate open source software asset. *Commun. ACM*, 53(2):155–159, 2010.

[6] M. Höst and A. Oručević-Alagić. A systematic review of research on open source software in commercial software product development. *Information & Software Technology*, 53(6):616–624, 2011.

[7] C. Jensen and W. Scacchi. Role migration and advancement processes in ossd projects: A comparative case study. In *29th International Conference on Software Engineering (ICSE'07)*, pages 364–374, May 2007.

[8] J. Kontio, J. Bragge, and L. Lehtola. The focus group method as an empirical tool in software engineering. In F. Shull, J. Singer, and D. I. K. Sjøberg, editors, *Guide to Advanced Empirical Software Engineering*. Springer, 2008.

[9] I. Kuali Foundation. Open library environment. http://www.kuali.org/ole, 2016.

[10] J. Lindman, M. Rossi, and P. Marttiin. Applying open source development practices inside a company. *IFIP International Federation for Information Processing*, 275:381–387, 2008.

[11] C. Melian and M. Mähring. Lost and gained in translation: Adoption of open source software development at hewlett-packard. In B. Russo, E. Damiani, S. Hissam, B. Lundell, and G. Succi, editors, *IFIP International Federation for Information Processing*, volume 275, pages 93–104. Springer, 2008.

[12] A. Oručević-Alagić and M. Höst. A case study of open source development practices within a large company setting. *ICSET 2014 International Conference on Software Engineering and Technology, Istanbul, Turkey, Sep 29-30, 2014*, 2014.

[13] E. S. Raymond. *The Cathedral and the Baazar*. O'Reilly Media, Inc., 2001.

[14] C. Robson. *Real World Reserach*. Blackwell Publishing, 2:nd edition, 2002.

[15] P. Runeson, M. Höst, A. Rainer, and B. Regnell. *Case Study Research in Software Engineering*. Wiley, 2011.

[16] W. Scacchi. The future of research in free/open source software development. In *FoSER*, pages 315–320, 2010.

[17] K.-J. Stol, M. A. Babar, P. Avgeriou, and B. Fitzgerald. A comparative study of challenges in integrating open source software and inner source software. *Information & Software Technology*, 53(12):1319–1336, 2011.

[18] K.-J. Stol and B. Fitzgerald. Inner source–adopting open source development practices in organizations: A tutorial. *IEEE Software*, 32(4):60–67, July 2015.

[19] The Apache Software Foundation. Hadoop. http://hadoop.apache.org, 2016.