**Term Paper**

Open Source and IP in the Digital Society

WS2021

Anh Truc Lam

Anh-Thu Tran

**Abstract**

Nowadays, companies are more open to invest into FLOSS projects as the outcome of these projects are promising in the aspects of innovation, cost reduction and resources extension. The reputation of OSS communities are perceived positively as they produce quality software as well as foster new ideas and innovation in companies. In this term paper written in the course “Open Source and IP in the Digital Society” at the Technical University Berlin, we investigate on the reasons for companies in the tech field to invest into FLOSS projects. The primary goal of this term paper is to present an overview of companies’ commercial objectives and deliver five main concepts, which are: Software Quality, Outstanding Technical Support, Business Opportunity, Business Model and External Innovation, based on the investigation of Zhou et al. (2016, pp. 7–8). Based on these concepts, we derive reasons and explain them. We utilized scientific literatures and build a case study based on two Facebook OSS projects: PyTorch and React.

We analyzed 6 papers related to this topic. Our findings reveals that…which focuses on the topics…

This study provide …and should fill the research gap of…and correspond to the research field of OSS projects.

Inhaltsverzeichnis

[1 Introduction 2](#_Toc61791872)

[2 Theoretical Foundations 3](#_Toc61791873)

[2.1 Definition of Free/Libre Open Source Software (FLOSS) Projects 3](#_Toc61791874)

[2.2 Open Source Projects as main driver in companies 3](#_Toc61791875)

[2.3 Open Source Projects in IT-Company Facebook 6](#_Toc61791876)

[2.4 Presented Case Study 8](#_Toc61791877)

[2.4.1 Role of Pytorch in Open Source Projects 8](#_Toc61791878)

[2.4.2 Role of React in Open Source Projects 8](#_Toc61791879)

[3 Methodology 10](#_Toc61791880)

[3.1 Research Design 10](#_Toc61791881)

[3.2 Defining the Case Study 11](#_Toc61791882)

[3.3 Literature Search Strategy 12](#_Toc61791883)

[3.4 Analysis 13](#_Toc61791884)

[3.4.1 Literature Analysis 13](#_Toc61791885)

[3.4.2 Case Study Analysis 14](#_Toc61791886)

[3.5 Model of Commercial Objectives 15](#_Toc61791887)

[4 Results 17](#_Toc61791888)

[4.1 Findings 17](#_Toc61791889)

[4.1.1 Overview of Reasons for Companies to Invest into FLOSS Projects 17](#_Toc61791890)

[4.1.2 Case Study: Reasons for Companies to Invest into FLOSS Projects 24](#_Toc61791891)

[4.1.3 OSS Project Trends 26](#_Toc61791892)

[4.2 Discussion 27](#_Toc61791893)

[4.3 Conclusion 29](#_Toc61791894)

[5 Appendix 29](#_Toc61791895)

# **Introduction**

In the age of digitization, open source software (OSS) become increasingly important to companies commercial objectives, which target the question of how to the align OSS projects with commercial practices (Zhou et al., 2016, p. 3). With the increasing number of participations, capabilities and other ressources, open source projects become tremendously complex and interrelatable as it demands more profressionality and management in OSS development (Germonprez et al., 2018, p. 2). Currently, OSS projects face two contradictory values and approaches. On the one hand, OSS is still building on the “[…] traditions and ideals of the free software movement” and egalitarian perspectives (Germonprez et al., 2018, p. 14). But on the other hand, there is a urgent need of professionality of corporate participation, their long-term investment and support, which is strategically important component in companies supply chain (Germonprez et al., 2018, p. 14). This has the reason because companies aim to reduce costs, seek the potential to receive external innovation and raise their profit by participating in open source communities (Zhou et al., 2016, p. 2). The best-known example is the Linux, Apache, Wordpress and its open source project ecosystems, which has an significant role in the evolution of open source projects (Gastil & Davies, 2020, p. 4). However, the current situation reveals many risks as companies are confronting with various challenges to enter, maintain and support OSS projects. One of the risks are the financial challenges to sustain such projects, which raise questions of how to invest into long-term projects as it is critically important to maintain, for example, contributors, who does not belong to a company, and prevent them to leave the OSS project by chance (Germonprez et al., 2018, p. 14). Additionally, the companies participation is fraught with commercial control, which could deprive the communities rights from their code commit privilege (Zhou et al., 2016, p. 2). Several tech-companies like Microsoft, Google and Facebook have already recognized these opportunities and took participation into the open source community in order to develop and provide their own software such as Microsoft’s ASP.NET framework and Facebook’s OSS react JavaScript library (Kochhar et al., 2019a, p. 1). To investigate the reasons for companies to invest into OSS projects, we took the company Facebook and its OSS projects in our investigation. This term paper proceed with the following structure. In section 2, we present definition and concepts of OSS projects. In section 3, we describe our methodology and analysis. After that, we present the results in section 4 and discuss our result to come to a conclusion in section 5.

# **Theoretical Foundations**

This section provides a theoretical foundation of important definition and description of peer-produced goods, OSS and Open Source Projects with regard on IT-companies. Further, general frameworks, specific companies and technologies related to OSS projects will be presented.

## Definition of Free/Libre Open Source Software (FLOSS) Projects

At the beginning, FLOSS was simply coined as Free Software (FS). In 1998 the term Open Source Software (OSS) was introduced in order to help users clarify, that its software products are not necessarily provided for free, but its source code is freely accessible (Ballhausen, 2019, p. 82). The terms FS and OSS share the same license terms, which are also referred to as the four freedoms. Thus, in order for software to be licensed as FLOSS it needs to meet four criteria (Ballhausen, 2019, p. 83). According to Logothetis & Stylianidis (2016, p. 30), the first grants the user the freedom to run the program for any purposes the user deems fit. The second freedom allows the user to lookup any source code of the program and adapt its implementation to his needs. Users also should have the freedom to reallocate copies of FLOSS in order to assist others. Lastly, any user wields the freedom to allow access to copies of improved code in order to create benefit to the general public (Logothetis & Stylianidis, 2016, p. 30). As software itself is protected under copyright law regardless, if a software is licensed as FLOSS, the four licenses outlined above simply define the distribution terms for FLOSS (Ballhausen, 2019, p. 83). It remains to mention, that it is essential for companies to set up proper FLOSS governance mechanism in order to comply with the licensing requirements and to reduce licensing risk. These help to stay consistent with operational strategy by considering participants’ background and motivation as well as align stakeholder’s interests (Kemp, 2010, p. 1). The terms FS, OSS and FLOSS can be used interchangeably. However, for this paper we focus on the term OSS.

## Open Source Projects as main driver in companies

In various papers, we find associations of the term open source projects in companies with the term corporate investment, which emphasizes how rapid the changes of open source itself through companies investment was (Germonprez et al., 2018, p. 2). Generally, there exists two types of commercial involvement of a company in OSS (Wagstrom, 2009, p. 8). A firm which is focused in building an active community in order to profit on its related services is called *community-focused*. If a firm is relying on product sales it is called *product-focused*. With these underlying findings, Zhou et al., 2016 (pp. 7–8) developed the model “Dimensions of the community involvement models” as shown in Figure 1. As a company firstly needs to verify the common objective of the project and the firm’s strategy as well as assessing the opportunity to create future revenue. Zhou et al. (2016, pp. 7–8) separated this endeavour into the dimension *commercial objective*. Next, in order to work within the project community framework companies need to create the underlying environment for the community to engage. For instance, they need to agree on intellectual property (IP) and legal issues, how to allocate resources, and how to integrate their employees into the community (Zhou et al., 2016, pp. 8–9). This aspect is picked up as community involvement action dimension. (Figure 1)



Figure 1. Dimensions of the community evolvement models (Zhou et al., 2016, p. 8)

The authors identified and assigned serveral concepts to each of the two dimensions, highlighted in yellow by going through academic literature. Zhou et al. (2016, pp. 8–9) then illuminates each concept coloured in green by setting up seven questions, which tries to answer what fuels the companies involvement and how the companies try to reach their goals within the OSS project, coloured in pink (Zhou et al., 2016, pp. 8–9).

Zhou et al. (2016, p. 7) investigated the purpose for a company to invest into OSS project and summarized the five key findings in the dimension *Commercial Objectives*:

|  |  |
| --- | --- |
| **Concepts from Literature** | **Description**  **(Zhou et al., 2016, pp. 7–8)** |
| 1. Software Quality | Companies utilize external knowledge involvement in order to increase quality of OSS projects. |
| 2. Outside Technical Support | Companies present their code base via OSS to raise outside feedback and technical support. They take advantage of creative ideas and capabilites of external contributors. |
| 3. Business Opportunity | Companies pursue a strategic and long-term goal by keeping free access to their open source platform and cultivate business opportunities, such as middle ware software or related services. |
| 4. Business Modell | Companies nurture the open source community by strongly engaging with it. By aligning their business model with the OSS community or supporting community engagement through own employees, companies legitimize themselves to take advantage of newly developed products and services. |
| 5. External Innovation | Companies with fewer resources can benefit by OSS initiatives of larger institutions. They can harness and rely on software quality and continued improvement. |

Table 2‑1 Concepts from Literature - Description

From this model we assessed, that the commercial objectives dimension largely contains answers, which are covering our research question, such that we recycle the concepts for further analysis, namely Software Quality, Outside Technical Support, Business Opporuntity, Business Model and External Innovation (Zhou et al., 2016, p. 8). A description of each concept can be retrieved from Table 1 above. Here the first question is: Is the OSS product critically important for the company’s business? (Q1).

Critically important can be understood in two perspectives, which is “[…] one, gaining profit directly from the OSS product; two, the OSS product is strongly associated with (or greatly helps to gain) profit.” (Zhou et al., 2016, pp. 7–8). We deem the remaining concepts stemming from the second dimension unfit to answer our research question and have excluded this part of the model in our paper. It remains to mention, that the paper of Zhou et al. (2016) was published within the past five years, during the creation of our paper, such that we can rely on its currentness and ultimately ensure the quality of our conclusions.

## Open Source Projects in IT-Company Facebook

In the past two decades OSS development proved themselves as it has produced a large number of highly reliable projects, such as Linus Mozilla browser, MySQL database and the Hadoop framework (Kochhar et al., 2019b, p. 1). Thus, the OSS framework remains an attractive model to build and deploy software. An important tool for developers to contribute to OSS projects is Git (Kochhar et al., 2019b, p. 1). Git is closely following different versions of a software and is also described version control system (VCS). Its intrinsic features are the faciliation of distributed development and the capability to handle over thousand developers. Hence, social coding websites such as GitHub or Gitlab offering Git are enabling over 40 million users to collaborate and partake in OSS projects. Further, even large software companies, such as Google and Microsoft are using GIT as their main development platform and started to open source part of their proprietary software as well (Kochhar et al., 2019b, p. 1).

Observing the open source landscape one notices, that Facebook, a social networking service, is involved in OSS projects as well (Facebook, 2020). This company provides a social networking site, also named Facebook, which enables users to connect and share information easily with family and friends. Currently Facebook is the world’s largest social network, with more than 2.3 billion users worldwide (Wikipedia, 2021). The company engages within numerous projects covering various large areas ranging from Artificial Intelligence, Web Technology and Operating Systems to Security etc. (Facebook, 2020). A member of the company’s open source team shares, that Facebook move towards open source comes naturally as in its main mission the company tries to create a world community. Second to that the firm tries to foster innovation and create better software, as Facebook is facing unique development challenges it has to solve. Here Open source serves the company producing better software and work more transparently (Flory, 2018, p. 1).

We want to take a look at the company Facebook in light of the concepts chosen above. We estimate, whether these can be utilized or need to reformulated to a reasonable extent.

|  |  |
| --- | --- |
| **Concepts from Literature** | **Application** |
| 1. Software Quality | We measure the metric “Rate of delivery”. This indicates, that the higher rates of software version delivery is the better quality software for customers (AltexSoft, 2017). We follow the question : *“How often the version of a software comes with improvements that impacts the user and the gap period between the previous version date?”* |
| 2. Outside Technical Support | This means if contributors, who are not employed in the same company as where the software was developed, has contributed to this project. We follow the question: *“Where are the top 10 contributors on Github employed at?”* |
| 3. Business Opportunity | As Facebook’s OSS projects try to solve their own unique problems, we follow the question: “*Does the open source software save costs or create revenue?”* |
| 4. Business Modell | We need to verify, if a OSS projects potentially provides new revenue streams for Facebook. We follow the question: *“Has the introduction of this software lead to more business opportunites?”.* |
| 5. External Innovation | We follow the question: *“Are other companies utilizing products stemming from advantages by these software companies* |

## Presented Case Study

To approach the research question, we want to illuminate reasons why Facebook is engaging in OSS projects based on scientific literatures and GitHub repositories. In this section, we identify and explain two open source projects provided by Facebook. We base our main focus on the Facebook’s open source projects: Pytorch and React.

### Role of Pytorch in Open Source Projects

The Pytorch project released in 2016 is contributing in the field of Artificial Intelligence (AI) by providing a Python library for deep learning (Facebook, 2020). It gained popularity in the research community as its performance is similar to other deep learning libraries, although by making use of the widely adopted computer language Python familiar to many data scientists due to its simplicity. The objective is to put researchers first by aiding them in reducing difficulty in creating new models, handling and wrenching data (Paszke et al., 2019, p. 2). Further, on 2nd of May 2018 Facebook announced PyTorch 1.0, bridging research and production. Developers still enjoy the possiblity to tweak their AI models quickly and experiment freely, but they can also rely on the library’s performance, when lifting their models up onto production level. Due to the widespread approval of many AI researchers the framework has been downloaded over 1.1 million times and has been the second-most cited deep-learning framework in April 2018 (Jia, 2018, p. 1).

### Role of React in Open Source Projects

The React project released in 2013 is contributing in the field of web technology (Wikipedia, 2020b). It is another open source library of Facebook, which utilizes a JavaScript framework to help reduce complexity in developing user interfaces in the web development field (Facebook, 2020). This framework introduces an innovative take in viewing interchangable content, such as advertisements, on websites, which were initially not fully welcomed by the community as it went against the best practises of JavaScript itself. Facebook faced technological issues with the general workflows and best practises in web development at that time. Although the problems were not unique to Facebook, it was them who took the charge and provided a solution, which not only benefitted themselves but others as well. React was thus created for a single reason of solving how a website is dealing with displaying data (Gackenheimer, 2015, p. 4).

# **Methodology**

In this section, we provide an overview about our steps of investigation in the research design. We explain our research question and our case study choice. Additionally, we describe our literature search strategy, which included a selection process. Based on that, we explain how we analyze the data and how we applied these data on the model.

## Research Design

At the beginning of the course lecture, we defined our main research question:

*“Which reasons does the IT-company have to contribute to FLOSS projects?” (Q1)*.

Due to a lack of resources we need to limit our scope of research on this. Therefore, we specify our investigation and divide Q1 into two sub-questions in the following:

*“Is the OSS product criticially important for the company’s business?” (Q2)* and

*“Which FLOSS-Trends can be identified in Facebook?” (Q3)*.



Figure 1 - Research Design Steps

Our main goal is to provide reasons why companies invest into OSS projects, which leads to several challenges on investigations in this term paper. First, there is a broad range of intensive knowledge with regard on OSS projects, which makes it difficult to focus on a specific scope of technologies and OSS projects (Facebook, 2020). Further, as we decided to focus on two OSS projects as described in chapter 2.4, there are no possibilities to validate our theories with interviews based on experts opinion of the company Facebook due to a lack of time and human ressources. We decided on a data-driven approach, which included scientific literatures and OSS projects on Facebook’s GitHub repositories. Our investigation is summarized in the following steps:

1. We defined the main research questions and divided it in two sub-questions, which gave us a clearer focus on our research direction.

2. We chosed OSS projects of the IT company Facebook as our case study subject. We made our first screen on their current projects and selected two projects.

3. Then, we conducted a systematic literature review to build a theoretical foundation of current investigations of related fields.

4. Based on theories found, we derived an overview of reasons why companies invest into OSS projects based on the model of Zhou et al. (2016, p. 8).

## Literature Search Strategy

We conducted the systematic literature based on the guideline of …(Autor). We summarized our steps in Figure 3. First, we identified digital libraries ACM Digital Library, IEEE Digital Library and Springer Link and evaluate them as situable for our literature review because they possess a broad repository for computer science research (Kaur et al., 2020, p. 4). We took ScienceDirect as an additional digital library. We build the key word search string based on our first literature screen and subjective perception: *(“FLOSS” OR “Open Source Software”) AND ("company" OR "enterprise") AND "facebook"*. We applied this search string in ScienceDirect, ACM Digital Library and IEEE Digital Library. For the digital database SpingerLink, we made a change on the symbols in the search string due to their websites capability to read the search string properly: *Facebook AND FLOSS AND "Open Source Software" AND (Company OR Enterprise).* By varying different key words, for example by adding “firm”, “industry” and “project” the results often includes non-relevant papers in our research field. The previous presented search strings turned out to be suitable because of the scope of literature and its content we collected. We found 1002 papers. Second, we screened the paper by applying predefined exclusion and inclusion criteria. We restricted the search for papers to peer-reviewed, open access and free access, journals, English language and from the year 2015 to 2020. According to Kaur et al. (2020, p. 4), it is important to filter paper with the focus on the field of Computer Science to gain a better result. Lastly, by reading the title and abstract we checked for the egibility of those papers, which left us 8 papers. After we search for other paper manually, we did not include additional paper to our collection. Finally, we found 8 papers in total.



Figure 3 - Selection Process

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Literature Review**  **Key word search string:** (“FLOSS” OR “Open Source Software”) AND ("company" OR "enterprise") AND "facebook" | | | | |
| **1. Identification** | **ScienceDirect**  n=605 | **ACM**  n=331 | **IEEE**  n = 22 | **Springer**  n = 44  'Facebook AND FLOSS AND "Open Source Software" AND (Company OR Enterprise)' |
| **2. Screening** |  |  |  |  |
| 2.1 Peer-Reviewed | n=25 | Only journals  n=35 | / | / |
| 2.2 Year:  2015-2020 | n=21 | n=28 | n = 11 | n = 28 |
| 2.3 Language:  English | / | / | / | n = 26 |
| 2.4 Full text or Open Access | n=7 | Media format: PDF  n=28 | / | N=20 |
| 2.5 Subject Area: Computer Science | n=4 | / | / | N=11 |
| **3. Eligibility** | | | | |
| 3.1 Titel and abstract reviewed | n=2 | n=3 | n=1 | n=5 |
| 3.2 Inclusion criteria | / | / | / | / |
| **4. Additional Papers Included** | n=0 | n=0 | n=0 | n=0 |
| **Papers in total** | n=2 | n=3 | n=1 | n=2 |

Table 3‑3 - Literature Review Overview

## Defining the Case Study

We built and conducted a case study based on Facebooks’ OSS projects to get deeper insights into the reasons for companies to invest into OSS projects. We focus on two main IT technologies: AI und web technologies. Further, these two technologies have a broad research field, thus we put the focus on two specific cases. By limiting to two OSS projects of Facebook, we choose the AI project *PyTorch* and the of frontend web technologies project *React* (Facebook, 2020). We described them in section 2.4.

In this case study, we collected data such as software version, project repositories and contribution histories on GitHub. The data collection can be found in the appendix. After collecting and analyzing data of the two OSS projects, we applied them on the model “Dimensions of the community involvement models” based on Zhou et. al (p. 8, 2016). We considered this approach as a good construct to get a deeper insights into the topic and to answer our research question. We report the results in chapter 4.1.2.

## Case Study Analysis

We have picked these two projects as they at first seemingly share the common objective to help developers improving on their capability to operate efficiently in their respective fields. The nature of each project is generally to minimize the complexity in development. This already provides social motivation to developers to engage in these OSS projects to gain reputation and help the community learn. However, the projects mentioned are living within the ecosystem of Facebook and are not out of scope of their business acumen, thus hinting to further relations this company has with these OSS project, which lastly leads to our research question we try to answer in this term paper.

In the Table 5‑1, we collected data on the top 10 contributors of React and PyTorch listed in the GitHub statistics. We identified the user name of the top 10 contributors automatically listed on the statistics extracted from the OSS projects on GitHub insights. Next, we identified the occupation of contributors which can be easily found in GitHub profile or LinkedIn. React’s top 10 contributors make up half of the commitments (6893 commitments, 50%). Even though there are 5.259.696 users and 1.527 contributors, only a few of people contributes a significant amount of commits on the React OSS project (Quelle GitHub).

In comparison to React, the OSS project Pytorch has 5,1 million less users, but Pytorch has more than the double of commits and similar but less count of contributors. Pytorch has approximately hundred contributor more than React, even though the Pytorch project started 4 years later than React. Ein Diagramm Vergleich hier wäre nice!

2.) Beschreibung wie wir mit den case study Zahlen aus GitHub umgehen, zB statische Berechnung oder iwas einfacheres?

# Irgendwo sollte hier eine leere GitHub Such-Übersichtstabelle auftauchen, zB welche Daten wir uns angeschaut haben zB Commitments, Jahre, Mitgliederzahl, Häufigkeit.

# Sollen wir hier statische Regression der Projekt Beteiligung zeigen?

(4) We collected project repository data and analyzed the affiliation history of project participants, as described in Section 3.5.

(5) We used generalized linear models and survival analysis to quantify how differ- ent types of commercial involvement impact contributor inflow and retention and report the results in Section 4.2. 🡪 (Zhou et al., 2016, p. 4)

We use project repository data to quantify two key variables that may be affected by a company’s involvement: the inflow of new contributors and the retention of existing developers. In Section 3.5.1, we retrieve and process historical data from VCSs. In Section 3.5.2, we identify internal and external developers. In Section 3.5.3, we identify different developer roles. 🡪 (Zhou et al., 2016, p. 10)

🡪 so könnte man ungefähr die Daten in GitHub analysieren, aber wir wollen ja schauen wie wir die Fragen beantworten können, zB schauen ob Angestellte mehr Commitments machen als Freelancer

## Model of Commercial Objectives

The model “Dimensions of the community involvement models” developed by Zhou et al., 2016 (pp. 7–8) as described in chapter 2, lays our foundation to analysis and understand our literature and case study more in detail. We limit the model to the dimension “Commercial Objectives” and we derived two table in the following in which we classify our results into five concepts of commercial objectives. We created seven columns in which we categorize our papers and case study in: paper title, author, year and five concepts from literature. According to the approach of Zhou et al. (2016, p. 10), we marked records in following two tables if the paper and Facebook OSS projects include the information related commercial objectives. We have different source to retrieve additional information, for example, on Facebook Developer blogposts (Jia, 2018, p. 1)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Literature** | | **Commercial Objectives** | | | | |
| **Concepts from Literature** | | | | |
| **Paper titel** | **Author (Year)** | **C1** | **C2** | **C3** | **C4** | **C5** |
| [1] xxx | Autor X, 2016 | x |  |  | x |  |
|  |  |  | x |  |  | x |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# **Results**

## Findings

In this section, we provide our findings of the literature review and the case study, which will be presented as an overview of reasons for companies to invest into OSS projects and recent OSS projects trends.

### Overview of Reasons for Companies to Invest into FLOSS Projects

We found 6 scientific papers, which investigate on open source projects in companies, which we listed in Table 4‑1 and classified them into five concepts: Software Quality, Outstanding Technical Support, Business Opportunity, Business Model and External Innovation. The following Table 4‑1 provides an overview of our findings:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Literature Review** | | **Commercial Objectives** | | | | |
| **Concepts from Literature** | | | | |
| **Paper titel** | **Author (Year)** | **C1** | **C2** | **C3** | **C4** | **C5** |
| [1] Understanding community participation and engagement in open source software Projects: A systematic mapping study | Rajdeep Kaur, Kuljit Kaur Chahal, Munish Saini, 2020 | x |  |  |  |  |
| [2] Eight Observations and 24 Research Questions About Open Source Projects: Illuminating New Realities | Matt Germonprez, Georg J.P. Link, Kevin Lumbard, Sean Goggins (2018) | x | x | x | x |  |
| [3] Digital Democracy: Episode IV - A New Hope\*: How a Corporation for Public Software Could Transform Digital Engagement for Government and Civil Society | John Gastil, Todd Davies (2020) |  |  | x | x |  |
| [4] Moving from Closed to Open Source: Observations from Six Transitioned Projects to GitHub | Pavneet Singh Kochhar, Eirini Kalliamvakou, Nachiappan Nagappan, Thomas Zimmermann, Christian Bird (2019) | x | x | x | x | x |
| [5] Fifteen Years of Open Source Software Evolution | Francis Bordeleau, Paulo Meirelles, Alberto Sillitti (2019) | x |  | x | x |  |
| [6] Open Source Technology Development | Kevin Crowston (2016) | x | x |  |  |  |
| **Legend**  C1: Software Quality  C2: Outstanding Technical Support  C3: Business Opportunity  C4: Business Model  C5: External Innovation | | | | | | |

Table 4‑1 - Overview of Reasons for Companies to Invest into OSS Projects

**C1: Software Quality:**

According to Kaur et al. (2020, p. 12) and Germonprez et al., (2018, p. 13) the importance of a healthy and active community is essential to software quality. Kaur et al. (2020, p. 12) state, that communities practically raise junior developers by giving helpful answers. Whereas Germonprez et al. (2018, p. 13) argue, that also senior contributors must given orderly respect. This needs to be prevented in order to reduce fluctation and ultimately assure software quality. These findings are mirrored in the research results by Kochhar et. Al (2019, p. 9) and Croswton (2016, p. 480), where they state, that a large community greatly reduces bugs, due to the four-eye principle as well as relying on voluntary developers, who can take over simple and miniscule tasks, which might be overlooked by the employees engaged in OSS, due to their scarce time..

, the assurance of software quality and productivity depends on the active participation of the contributors. Thus, it is important to support and motivate the developers, especially, the beginners in open source communities to get involved and to avoid flucation. Such actions to maintain the contributors in the future by providing fast and polite responses to inexperienced developers, providing an experienced mentor and also a friendly environment (Kaur et al., 2020, p. 12). However, they do not investigate OSS in companies specifically but they investigated on the participation and engagement in OSS projects in general (Kaur et al., 2020, pp. 4–5). [1]

In another paper, Germonprez et al., (2018, p. 13) described that the health of open source projects is an indicator for software quality. A healthy open source projects environment involves a stable and successful environment, in which contributors feels valued and acknowledged. To measure the health of open source projects companies can use analytics tools. But since there were not any consensus about which metrics should be measured to acknowledge the health of open source projects, the Linux Foundation’s Community Health Analytics Open Source Software (CHAOSS) standardized the procedure of how to measure metrics with regard on the health of open source metrics (Germonprez et al., 2018, p. 13). [2]

[Truc]

According to Kochhar (2019, p. 9) interview on SQ. Survey over 58% quality of software has increased. Reasons: less time constraint of non employees to take part of code clean up. Quality is already high among ms software products. 4eye check on every single change

(Bordeleau et al., 2019, p. 63) states that OSS is part of almost any software product due to the popularity of powerful and high quality libraries

(Crowston, 2016, p. 480)

Because of the community’s ability to fix bugs and add features, proponents also

claim that OSS enables higher code quality. This effect has been summarized as

Linux’s law: with enough eyes, all bugs are shallow, meaning that bugs can be fixed

more quickly than with limited developer base that has time to debug only a limited

number of problems, which requires prioritizing bug fixes, even not fixing bugs that

affect only a few people. Furthermore, having the code open enables it to be

audited, potentially increasing security compared to closed software that is not

openly reviewed. However, software being open is no guarantee that it will in fact

be audited, as illustrated by the discovery of a bug in OpenSSL, a package that had

only a few active developers.

**C2: Outstanding Technical Support:**

Germonprez et al. (2018, p. 2) investigated on open source project ecosystem of the Linux Foundation as they describe that foundation support play a major role in the evolution of open source projects (Germonprez et al., 2018, p. 2). [2] This has been validated by interviews taken by Kochhar et. al (2019, p. 9), where he assessed, that outstanding technical support ist critical for the stability of the OSS project, due to the sheer number of users deploying its software on various platforms. Based on the reducing fluctations argument in C1, Crowston et. al (2016, p.480) strengthens the viewpoint that a larger vibrant community is leading to greater support as a higher engagement in OSS projects leads to better software and feedback on improvements and lastly more satisfied users. This feedback-loop validates the reliabilty of OSS projects on outstanding technical support.

(Kochhar et al., 2019b, p. 9)

Some open source tools have a large user base, which

increases the support for these tools in terms of new features,

improving the existing ones or finding bugs. With

such community support tools are modified to run of various

systems and platforms. One developer commented,

“The open source workforce is standardized and several tools

which are well-known. If they are not maintained by us, they

are maintained by somebody because they are publicly released”

(D1) while another commented “It is very unusual where

open-source infrastructure breaks whereas closed source is fragile

and complicated” (D1). However, developers also expressed

their dissatisfaction about some inter

(Crowston, 2016, p. 480)

Feedback Loop

In addition to the direct link from inputs to mediators to outputs, the IMOI model

recognizes that there is a feedback loop from the outputs to the inputs of a project.

For example, high-quality (e.g., modular) code (a project output) should make the

codebase easier to maintain and so facilitate additional contributions (an input).

User satisfaction with the project (an output) is important to retaining developers,

while the success of a project (an output) may increase its visibility and ability to

attract new developers (an input). Contrariwise, a project that is struggling may find

that difficulties in development cause developers and users to leave for more

promising projects, thus further complicating development, leading to a downward

spiral.

**C3: Business Opportunity:**

In our definition a business opportunity defines companies, which set a strategic and long-term goalto benefit from open source by providing open access to its platform and gaining new business opportunities as described in chapter Table 2‑1. For example, Wikipedia set in the early era of social media platform a strategic goal to keep their platform open and provide access in order to increase their company size mentioned by Gastil & Davies (2020, p. 5). [3]

According to Germonprez et al., 2018 (pp. 14–15), open source projects need to have a secure long-term investment in order to maintain software quality. There is a urgent need in innovation of market mechanism to coordinate the development of open source projects for example through crowd marketing or funding campaign (Germonprez et al., 2018, pp. 14–15). These authors clearly indicates that companies needs to align their use of OSS and internal innovation process, which means that companies have to perform internal actions to deal with open source project health risks (Germonprez et al., 2018, p. 18). We assign this paper to C3 because the authors describes that companies have to set a strategic and long-term goal, in this case a long-term investment to deal with future risks. [2]

(Kochhar et al., 2019b, p. 6)

A project in an organization might bring new customers

or business opportunities for other projects when these two

projects are somehow interrelated. If an open source project

runs on a platform provided by the organization, users of

the project would need the platform to run their applications.

Thus, open sourced projects can indirectly generate

revenue for the organization.

(Bordeleau et al., 2019, p. 65)

Software commodization

Value of data

the value in the software business is continuously

moving upwards transforming the basic infrastructure and applications

into a commodity. Nowadays, leading applications include a relevant set of

features powered by machine learning algorithms that are based on the analysis

of a huge amount of data that is not available to the community while

software is. For this reason, the business models are changing moving towards

the data that will be able to provide competitive advantages to companies

that own them. This is a relevant problem for the open source communities

that currently do not have the ability to collect and exploit such amount of

data preventing the creation of cutting edge applications.

**C4: Business Model:** In our definition in Table 2‑1, a business model aims to promote proactive engagement with open source communities to develop new products and services.Since OSS projects become increasingly commercialized by large organizations, the tradionational view on OSS projects became decreasingly important (Germonprez et al., 2018, p. 11). The intention to make profit from these projects and should serve the company’s supply chain instead of keeping the OSS projects as the “social constructions of freedom to the forefront” as it was traditionally (Germonprez et al., 2018, p. 11). [2]There is a huge competion of how to make the most effective use of open source resources. For example, Gastil & Davies (2020, p. 5) mentioned that Facebook is a succesful business model because it stands out comparing to other companies like Friendster or MySpace in the early era of social media platform. The most striking point is that Facebook won loyal users and the most skilled programmers by putting the relationship aspects of people’s real life in the foreground, roll ads slowlier at the beginning, used venture capital to build a huge community base and showed the community the information they are interested in (Gastil & Davies, 2020, p. 5). Facebook used its resources efficiently, aligned its srategy with the communities requirements and attracted users, which at the same time, are the ground and success of Facebook. [3]

[Truc]

Kochbar overlapping with BO

Bordeleau overlapping with BO

**C5: External Innovation:** New contributors involvement changes the OSS projects dynamics by gaining more ideas and skills while solving technical issues and thus, enhance innovation (Kaur et al., 2020, p. 12). However, we did not cross “C5: External Innovation” for this paper because we search for more specific information on how smaller companies benefit from innovation of greater companies, which was missing here. [1]

(Kochhar et al., 2019b, p. 4)

Historically, all six projects were developed within the

organization and the vast majority of external users used

to consume those products. By excluding these potential

open source developers from the discussions, the projects

were missing out on valuable feedback and experience that

external developers can bring with them. While there were

people active in the community, there was a significant barrier

to contribute. To ensure that Microsoft is developing the

right product for its customers, it was considered important

to involve community members. A respondent mentioned:

“The community is one part of making sure that we are delivering

the best value to our customers.” (M2) Suggestions from the

community members can provide directions to the technology/

product. “Those guys don’t represent the whole community

but they are the ones who set the trends. They can tell you the

way to design software to be fruitful and productive. You should

go in this direction and they set the standards for everyone else to

follow.” (M1) Ultimately, involving open source developers

helps in overall growth of the community.

Furthermore, open sourcing the code can help external

developers have a say as well as help companies build trust

with users and developers, which is likely to attract more

contributors. One manager indicated, that such a strategy

allows “To make ourselves one with the community and participate

more.” (M1). Over 97% of the respondents agree that the

project was open-

### Case Study: Reasons for Companies to Invest into FLOSS Projects

Additonally, we categorize our findings by analyzing the Facebook’s OSS project React and PyTorch as our case study into the five concepts as described in the previous section as well. The overview is provided in the following Table 4‑2.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Case Study** | | **Commercial Objectives** | | | | |
| **Concepts from Literature** | | | | |
| **Open Source Project** | **Company (Year)** | **C1** | **C2** | **C3** | **C4** | **C5** |
| [1] React | Facebook ? Jahr | x | xo | x | o | x |
| [2] PyTorch | Facebook and Pytorch ? Jahr | x | xo | x | o | x |

Table 4‑2- Case Study Overview of Commercial Objectives

To evaluate our case studies with Facebook OSS projects PyTorch and react, we need another definition of the five concepts C1-C5 in order to suit the OSS project requirements and our expectations. These five concepts are described in chapter 3.4

**C1 Software Quality:** According to PyTorch (2020), the latest version 1.7.0 release was on the 27th October 2020. In comparison the previous version 1.6.0 was announced on the 28th July 2020. The span between the release dates is only 4 months, so we can say that the quality of the software is often worked on and thus follows a professional quality standard. [2]

React. After initial public release of version 0.3.0 on 29 May 2013, it reached its first orderly version by moving the decimal point past the zero reaching version 15 on 7April 2016 <https://en.wikipedia.org/wiki/React_(web_framework)> Since then it grew into version 17.0.1 with multiple versions months apart inbetween. We estimates, that the software is widely adopted and the projects has enjoyed continuing developments to keep SQ.

**C2: Outside technical support:** We found out that 8 of the top 10 contributors where employed in Facebook while one contributor was employed at Google but was a former employee at Facebook. The second ranked contributor in Table 5‑1 was not found on social media platforms. However, we assume that all of the top 10 contributor has a relation to the company Facebook whether in the past or present time.

We find similarities with the React project here as well. Here, the top 10 contributors could be identified to be Facebook employees only one has left the company since.

**C3: Business Opportunity:** According to Zhou et al. (2016, p. 8), an OSS product is critically important for the company’s business when it fulfills to cases. First, is gaining direct profit from the OSS product. Second, is to have a strong association with profit or receiving help to gain profit (Zhou et al., 2016, p. 8). There is no evidence found that Pytorch is critically important for the company’s business by gaining a direct profit from the OSS product. However, Pytorch is partnered with large enterprises like Tesla, Amazon and Uber etc. This is an indication that Pytorch is associated with great profit (Wikipedia, 2020a). [2] 🡪 React profit genauso begründen!

We arrive at the same results for the React project. Several major website utilize the React framework namely, New York Times, PayPal, Netflix, Uber, Airbnb and many more. The cost saved by being able to deploy easy to handle performant software to to achieve best user experience interacting with the respective website doesn’t need to be mentioned [https://www.ideamotive.co/blog/amazing-examples-of-react-web-development]

**C4: Business Modell:** …

None for React

**C5: External Innovation:** …

For React, we have outlined in C3 that a many major website rely on React. We can deduct by the github repo and this finding, that the web development task for named websites was heavily relied on the OSS project initiated by FB.

|  |  |
| --- | --- |
| 4. Business Modell | This means if a software was developed by promoting proactive engagement with open source communities to develop this software. We follow the question: “Was this software created to solve a business model problem? For example expanding ressources. |
| 5. External Innovation | describes if the software has impacted smaller companies which benefit from the innovation of the software |

### OSS Project Trends

# Diagramm hier wäre nice!

# Kann man hier einen Trend erkennen oder sogar vorhersagen? Predictive Analytics als Methode denkbar? Ansonsten nur eine kleine Tabelle machen und kurz erklären.

# Trend von was genau? Projekt-Trend in welche nun mehr investiert werden? Und in welche nicht? Wäre cool hier zwei drei Linien im diagramm zu haben, wo gezeigt wird, dass zB weniger in web technologies investiert wird als in AI, oder vllt auch gleich viel?

**Trend wird hier genau beschrieben:**

The new realities of open source projects are shifting who benefits from the wealth generated in these projects. We observed many new realities (e.g., changes of membership from volunteers to paid employees to a lack of effective market mechanisms). These new realities are interrelated and cannot be solved by a single open source project or organization, but provide new forms of engagement in the design and distribution of software. New mechanisms of practice and research can be developed to incentivize the development of healthy projects that produce secure software, incentivize the engagement of volunteers who are not affiliated with an organization, and continue to leverage the traditional and successful open source development model. 🡪 (Germonprez et al., 2018, p. 18).

**Trend in innovation:**

As open source project engagement becomes increasingly domesticated and stable — it becomes more closely aligned with corporate innovation. While there may be free agents within open source projects, they exist on a considerably more limited scale when open source project engagement is a means for getting corporate work done. This does not mean that these engagements are not open source project engagements, but it does mean that the membership of these open source projects is changing. In short, the licensing of the project remains open, but the member practices around the work on these projects are materially distinct from earlier characterizations of open source project research. 🡪 (Germonprez et al., 2018, p. 12)

## Discussion

(2 Pages)

While analyzing the scientific literatures and the two Facebook OSS projects, we try to answer the questions:

*“Which reasons does the IT-company have to contribute to FLOSS projects?” (Q1)*

“*Is the OSS product critically important for the company’s business?” (Q2)*

Hier Gründe aufzählen und beschreiben, warum Unternehmen in OSS projects investieren.

We limit our investigations on the commercial objectives of companies and find out that C1 zB has the highest number of crosses in our finding table x.

This indicates that…

However, another paper mentioned that…

**Warum haben wir Pytorch und react gewählt?**

**Was haben wir daraus für Gründe erkannt, warum Unternehmen in OSS projekten investieren?**

We saw C1 zB as the most important reason for company to invest into OSS projects.

* Das besondere wäre in unserem Term Paper, dass unser Modell angewendet werden kann, um Gründe zu finden warum ein IT Unternehmen in FLOSS porjects intervestiert, ohne dabei auf die leute zu zugehen und Interviews durchzuführen.

**Unterschiede zu der Intensität der Commits und den Entwicklungsjahren von React und Pytorch geben:**

**….**

Although we do not have a complete knowledge of the companies’ intentions, we could learn about company actions, visions, goals, market share, and values based on the information gathered from company websites, news articles, personal blogs, commit comments, and other documents. Therefore, we conducted an extensive Internet search for the materials relevant to this study: (Zhou et al., 2016, p. 9)

In dem paper wird nur gesagt, dass OSS durch den US staat gefördert werden sollte indem die bevölkerung ihren teil dazu beiträgt und vom government gefunded und gemanged wird. Es gibt keine direkten bezug von OSS projects im unternehmen, aber allgemein werden diese unternehmen wie facebook immer wieder erwähnt und sozusagen als vorreiter bzw. vorbild genommen wie man OSS projects erfolgreich durchzieht

The reasons to build public software in the United States are thus both negative and reactive to recent events on one hand, as well as positive and of long standing on the other. We wish to stop the threats that commercial- driven social media pose to our democratic culture (a negative reason), but also to build a healthier civic and community life than we have ever had. Positive motivations like those have spurred various efforts to innovate over the past 20 years, encouraged by the surprising success of free and open source software (e.g., Linux, Apache, Firefox, Wordpress, and Drupal) and the very democratically operated Wikipedia, powered by the MediaWiki software, which is licensed under the GNU General Public License v2+. (Gastil & Davies, 2020, p. 4)

## Conclusion

(1 Page) (+ Research Gap)

**Aim & Short summary:** …

**Method**: We identified two OSS projects invested by Facebook in the field of AI and web technologies. We collected scientific literature, repository data and histories based on GitHub Facebook projects participation. To extract relevant information we used generalized linear models (?) to measure project participation and built a two dimensional-model to provide a guideline how to identify companies reason to invesrt in FLOSS projects generally.

**Result:** …

# Appendix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rank** | **GitHub User ID** | **Facebook employee** | **Non Facebook employee** | **Commits** |
| **PyTorch** | | | | |
| 1 | ezyang | x |  | 1.456 |
| 2 | gchanan | / | / | 1.078 |
| 3 | apaszke |  | x | 870 |
| 4 | jerryzh168 | x |  | 734 |
| 5 | soumith | x |  | 734 |
| 6 | zdevito | / | / | 623 |
| 7 | Yangqing | x |  | 601 |
| 8 | colesbury | x |  | 582 |
| 9 | smessmer | x |  | 552 |
| 10 | zou3519 | x |  | 544 |
| **React** | | | | |
| 1 | gaearon | x |  | 1.487 |
| 2 | bvaughn | x |  | 1.288 |
| 3 | sophiebits | x |  | 875 |
| 4 | zpao | x |  | 821 |
| 5 | acdlite | x |  | 778 |
| 6 | sebmarkbage | x |  | 731 |
| 7 | trueadm | x |  | 429 |
| 8 | petehunt |  | x, works as CEO at Smyte, but was a former Facebook employee | 205 |
| 9 | chenglou | x |  | 152 |
| 10 | vjeux | x |  | 140 |
| **URL (access date: 15.01.2021)**  Pytorch: <https://github.com/pytorch/pytorch/graphs/contributors>  React: <https://github.com/facebook/react/graphs/contributors> | | | | |

Table 5‑1 - Top 10 Contributors of PyTorch and React

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Start Year** | **Languages** | **%** | **Users** | **Contributors** | **Commits on Master Branch** | **Top 10 Contributor and Commits** |
| 2013 | JavaScript | 95,20% | 5.250.696 | 1527 | 13.775 | 6893  (50%) |
| HTML | 2,00% |
| CSS | 1,20% |
| C++ | 0,80% |
| TypeScript | 0,30% |
| CoffeeScript | 0,30% |
| Other | 0,20% |

Table 5‑2 - React OSS Project on GitHub (2020)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Start Year** | **Languages** | **%** | **Users** | **Contributors** | **Commits on Master Branch** | **Top 10 Contributor and**  **Commits** |
| 2016 | C++ | 52,80% | 53.373 | 1.699 | 32.371 | 7748  (24%) |
| Python | 33,70% |
| Cuda | 6,00% |
| C | 3,90% |
| Cmake | 1,30% |
| Objective-C++ | 0,60% |
| Other | 1,70% |

Table 5‑3 - Pytorch OSS Project on GitHub (2020)

**Literature**

AltexSoft. (2017). *What Software Quality (Really) Is and the Metrics You Can Use to Measure It*. https://www.altexsoft.com/blog/engineering/what-software-quality-really-is-and-the-metrics-you-can-use-to-measure-it/

Ballhausen, M. (2019). Free and Open Source Software Licenses Explained. *Computer*, *52*(6), 82–86. https://doi.org/10.1109/MC.2019.2907766

Bordeleau, F., Meirelles, P., & Sillitti, A. (2019). Fifteen Years of Open Source Software Evolution. In F. Bordeleau, A. Sillitti, P. Meirelles, & V. Lenarduzzi (Eds.), *Open Source Systems* (Vol. 556, pp. 61–67). Springer International Publishing. https://doi.org/10.1007/978-3-030-20883-7\_6

Crowston, K. (2016). Open Source Technology Development. In W. S. Bainbridge & M. C. Roco (Eds.), *Handbook of Science and Technology Convergence* (pp. 475–486). Springer International Publishing. https://doi.org/10.1007/978-3-319-07052-0\_29

Facebook. (2020). Open Source projects by Facebook. *Facebook Open Source*. https://opensource.facebook.com/projects

Flory, J. W. (2018). *A look inside Facebook’s open source program*. https://opensource.com/article/18/1/inside-facebooks-open-source-program

Gackenheimer, C. (2015). *Introduction to React*. Apress. https://doi.org/10.1007/978-1-4842-1245-5

Gastil, J., & Davies, T. (2020). Digital Democracy: Episode IV—A New Hope\*: How a Corporation for Public Software Could Transform Digital Engagement for Government and Civil Society. *Digital Government: Research and Practice*, *1*(1), 1–15. https://doi.org/10.1145/3342194

Germonprez, M., Link, G. J. P., Lumbard, K., & Goggins, S. (2018). Eight Observations and 24 Research Questions About Open Source Projects: Illuminating New Realities. *Proceedings of the ACM on Human-Computer Interaction*, *2*(CSCW), 1–22. https://doi.org/10.1145/3274326

Jia, B. (2018). Announcing PyTorch 1.0 for both research and production. *Facebook for Developers*. https://developers.facebook.com/blog/post/2018/05/02/announcing-pytorch-1.0-for-research-production/

Kaur, R., Kaur Chahal, K., & Saini, M. (2020). Understanding community participation and engagement in open source software Projects: A systematic mapping study. *Journal of King Saud University - Computer and Information Sciences*, S1319157820305139. https://doi.org/10.1016/j.jksuci.2020.10.020

Kemp, R. (2010). Open source software (OSS) governance in the organisation. *Computer Law & Security Review*, *26*(3), 309–316. https://doi.org/10.1016/j.clsr.2010.01.008

Kochhar, P. S., Kalliamvakou, E., Nagappan, N., Zimmermann, T., & Bird, C. (2019a). Moving from Closed to Open Source: Observations from Six Transitioned Projects to GitHub. *2019*, 1–22. https://doi.org/10.1109/TSE.2019.2937025

Kochhar, P. S., Kalliamvakou, E., Nagappan, N., Zimmermann, T., & Bird, C. (2019b). Moving from Closed to Open Source: Observations from Six Transitioned Projects to GitHub. *IEEE Transactions on Software Engineering*, 1–1. https://doi.org/10.1109/TSE.2019.2937025

Logothetis, S., & Stylianidis, E. (2016). BIM Open Source Software (OSS) for the documentation of cultural heritage. *Virtual Archaeology Review*, *7*(15), 28. https://doi.org/10.4995/var.2016.5864

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., … Chintala, S. (2019). *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 12. https://papers.nips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

PyTorch. (2020). *PyTorch 1.6 released w/ Native AMP Support, Microsoft joins as maintainers for Windows*. https://pytorch.org/blog/pytorch-1.6-released/

Wagstrom, P. A. (2009). *Vertical Interaction in Open Software Engineering Communities*. https://patrick.wagstrom.net/thesis/wagstrom-phd-thesis-single.pdf

Wikipedia. (2020a). PyTorch. In *PyTorch*. https://en.wikipedia.org/wiki/PyTorch

Wikipedia. (2020b). *React (web framework)*. https://en.wikipedia.org/wiki/React\_(web\_framework)

Wikipedia. (2021). *Facebook*. https://en.wikipedia.org/wiki/Facebook

Zhou, M., Mockus, A., Ma, X., Zhang, L., & Mei, H. (2016). Inflow and Retention in OSS Communities with Commercial Involvement: A Case Study of Three Hybrid Projects. *ACM Transactions on Software Engineering and Methodology*, *25*(2), 1–29. https://doi.org/10.1145/2876443