

# Turnover of Companies in OpenStack: Prevalence and Rationale

**YUXIA ZHANG** and **HUI LIU**, School of Computer Science and Technology, Beijing Institute of Technology

**XIN TAN**, School of Computer Science and Engineering, Beihang University, State Key Laboratory of Software Development Environment

**MINGHUI ZHOU** and **ZHI JIN**, School of Computer Science and Technology, Peking University, Key Laboratory of High Confidence Software Technologies, Ministry of Education

**JIAXIN ZHU**, Institute of Software, Chinese Academy of Sciences, University of Chinese Academy of Sciences

To achieve commercial goals, companies have made substantial contributions to large open-source software (OSS) ecosystems such as OpenStack and have become the main contributors. However, they often withdraw their employees for a variety of reasons, which may affect the sustainability of OSS projects. While the turnover of individual contributors has been extensively investigated, there is a lack of knowledge about the nature of companies' withdrawal. To this end, we conduct a mixed-methods empirical study on OpenStack to reveal how common company withdrawals were, to what degree withdrawn companies made contributions, and what the rationale behind withdrawals was. By analyzing the commit data of 18 versions of OpenStack, we find that the number of companies that have left is increasing and even surpasses the number of companies that have joined in later versions. Approximately 12% of the companies in each version have exited by the next version. Compared to the sustaining companies that joined in the same version, the withdrawn companies tend to have a weaker contribution intensity but contribute to a similar scope of repositories in OpenStack. Through conducting a developer survey, we find four aspects of reasons for companies' withdrawal from OpenStack: company, community, developer, and project. The most common reasons lie in the company aspect, i.e., the company either achieved its goals or failed to do so. By fitting the survival analysis model, we find that commercial goals are associated with the probability of the company's withdrawal, and that a company's contribution intensity and scale are positively correlated with its retention. Maintaining good retention is important but challenging for OSS ecosystems, and our results may shed light on potential approaches to improve company retention and reduce the negative impact of company withdrawal.

**CCS Concepts:** • **Software and its engineering** → **Open source model**; **Programming teams**; • **Human-centered computing** → **Empirical studies in collaborative and social computing**;

This work was sponsored by the National Natural Science Foundation of China (62141209, 62172037, 61802378, 61825201, and 62192731).

Authors' addresses: Y. Zhang and H. Liu (corresponding author), School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China; emails: {yuxiazhang, liuhui08}@bit.edu.cn; X. Tan, School of Computer Science and Engineering, Beihang University; State Key Laboratory of Software Development Environment, Beijing, China; email: xintan@buaa.edu.cn; M. Zhou (corresponding author) and Z. Jin, School of Computer Science and Technology, Peking University, and Key Laboratory of High Confidence Software Technologies, Ministry of Education, Beijing, China; emails: {zhm, zhijin}@pku.edu.cn; J. Zhu, Institute of Software, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, Beijing, China; email: zhujiaxin@otcaix.iscas.ac.cn.



This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2022 Copyright held by the owner/author(s).

1049-331X/2022/07-ART75

<https://doi.org/10.1145/3510849>

Additional Key Words and Phrases: Software development, open source ecosystem, commercial participation, company withdrawal, survival analysis

#### ACM Reference format:

Yuxia Zhang, Hui Liu, Xin Tan, Minghui Zhou, Zhi Jin, and Jiaxin Zhu. 2022. Turnover of Companies in OpenStack: Prevalence and Rationale. *ACM Trans. Softw. Eng. Methodol.* 31, 4, Article 75 (July 2022), 24 pages. <https://doi.org/10.1145/3510849>

## 1 INTRODUCTION

Open source has become the *de facto* way to build software—not only in the software domain but also across diverse industries [21]. As companies use open-source code to build their own commercial products and services, they see the strategic value of contributing back to those projects [14]. Therefore, companies task their employees to contribute to the projects, with the idea of gaining expected benefits [55]. **Open source software (OSS)** projects, especially large ones, no longer rely on individual contributors, but rather on vast companies [26]. For instance, more than 500 companies contributed over 85 percent of code to the Linux kernel in 2017 [37]. With such a great contribution, companies involved in these ecosystems have a significant influence not only on their development but also on their sustainability.

Once contributions are accepted, the corresponding contributors are expected to maintain them over the long term [46]. This expectation is especially true for large contributions, new features, or standalone code, such as a driver for a specific piece of hardware [46]. Prior research indicates that frequent developer turnover may result in loss of productivity and code quality [17, 45] and even affect the project’s survival probability [34]. In OSS projects in which companies participate intensively, the impact might be more serious when companies decide to withdraw their teams (always  $\geq$  one developer). Researchers have shown that the withdrawal of dominant companies has caused the failure of some OSS projects [78].

Previous work has primarily focused on turnover and retention at the individual level rather than the company level. These studies examined a series of factors that affect the possibility of a newcomer becoming a long-term contributor, e.g., technical skills [40], social ability [59], and environment [79]. Researchers also measured the negative impact of developer turnover on OSS projects [10, 17, 34]. Studies on commercial participation have mainly focused on motivations, strategies to engage, collaboration, and impacts on OSS ecosystems [27, 68, 74, 80], leaving a knowledge gap on the turnover of companies.

Assessing the prevalence and rationale of company turnover in OSS projects is of prime importance because such knowledge is required to monitor the health of an OSS community and to minimize company withdrawal, especially when projects rely heavily on commercial participation. To this end, this article investigates company withdrawal in OSS ecosystems. To achieve this research goal, we investigate approximately a decade of development evolution of OpenStack, one of the fast-growing OSS ecosystems that are increasingly attracting academic attention [33, 63, 75]. Given the 266 companies that have withdrawn from OpenStack that we identified and validated, we formulate the following research questions to guide our study:

**RQ1: How common do companies withdraw their employees from OpenStack?** We answer this RQ from two aspects: (1) the number of withdrawn companies per version and (2) each version’s turnover rate of joined and sustaining companies. We find that the number of leaving companies is increasing over versions and even surpasses the number of joining companies in the 14th version, ending the uptrend of contributing companies. More than half of the companies that joined in a version will withdraw later, and approximately 12% of companies in each version will exit in the next version.

**RQ2: To what degree did the withdrawn companies contribute to OpenStack?** To answer this RQ, we first calculate the distribution of companies' contributions to obtain a general understanding of companies' contribution performance before their withdrawals. Then, we compare the contribution performance of the withdrawn companies with the companies that joined in the same period and sustain in OpenStack. We find that the withdrawn companies used to make limited contributions to OpenStack: among the 266 withdrawn companies, the median number of contributed developers and commits are one and six; the median participated projects and versions are three and two. Compared to the companies that joined in the same period but that are still contributing, the withdrawn companies make fewer contributions (i.e., 0.3 times) to a similar scope of repositories in OpenStack.

**RQ3: What are the signals indicating that companies are going to withdraw?** To obtain the answer to this RQ, we conduct email surveys to obtain companies' withdrawal reasons and use survival analysis to validate the factors that may affect a company's turnover. We find that the factors affecting company withdrawal are complex and diverse, and the most common reasons are that the companies achieved, or failed to achieve, their goals. We find that business integration vendors and development infrastructure vendors have a higher probability of withdrawing. However, the contribution intensity, scale, and being a partial solution vendor are negatively associated with withdrawal.

In summary, our main contributions are as follows:

- A comprehensive understanding of company turnover in terms of their withdrawal frequency, contribution degree, and reasons.
- A framework to study company turnover in OSS ecosystems, including an approach to identify withdrawn companies and a survival model for indicating the likelihood of a company's withdrawal.
- Recommendations for the three OSS parties on better commercial participation.

To the best of our knowledge, this study is the first to explore company withdrawal from OSS.

The remainder of this article is structured as follows. We review related work in Section 2 and introduce our method in Section 3. We present answers to each of the three research questions in Section 4. We discuss the implications for research and practice in Section 5. We present threats to the validity of our reported findings in Section 6 and conclude the article in Section 7.

## 2 RELATED WORK

We discuss two groups of related work. One is commercial participation in OSS, and the other is developer turnover in OSS.

### 2.1 Commercial Participation in OSS

Companies have become intensively involved in OSS projects in recent years [80] and play an increasingly important role in their development. Thus, a number of efforts have been spent on understanding commercial participation in OSS [6, 26, 50, 80]. Researchers start by examining the motivations of companies by conducting a series of case studies of OSS projects [8, 11, 14]. On the one hand, researchers explored companies' motivations by comparing them to individual developers, and they found that companies focus less on social motivations such as reputation and learning benefits but emphasize economic and technological reasons instead [6, 29]. On the other hand, some work has aimed at understanding the business strategies around company participation in OSS [14, 15, 66]. For example, Daffara [14] explored 120 firms that derive their main revenue stream from OSS and clustered them into six business strategies, such as consulting, platform providers, and twin licensing. One of our recent studies [74] took a new perspective

on companies' business strategies by combining commercial objectives with contribution performance and found eight unique contribution models.

Multiple companies are now investing significant efforts in one OSS ecosystem [26], so researchers are motivated to investigate the relationship and interaction among companies. For example, Teixeira et al. investigated collaboration among companies in OpenStack and found that companies tend to form alliances when making contributions [63]. They also found that transparency and weak intellectual property rights in OSS allow a focal company to transfer information and resources more easily among its multiple alliances [63]. Focusing on the same project, our recent study [75] found three collaboration patterns of companies: intentional collaborations (including supply and consumption, distribution-oriented ally, and service delegation), passive collaborations, and isolated fashion. We also found a positive association between a company's position in the collaboration network and its productivity [75].

Furthermore, a few studies have investigated the impact of commercial participation in OSS projects. Zhou et al. [80] studied how commercial involvement influences the onboarding and retention of developers in three OSS projects, and they found that (1) the high intensity of commercial involvement was associated with a decrease in external inflow but with improved retention and (2) a shared control mechanism was associated with increased external inflow. Similarly, we found that a company's domination is positively associated with the quality of issue reports and the productivity of contributors [73]. In another recent work [74], we also found that the diversity of involved companies in an OSS project is positively associated with the number of volunteers. Valiev et al. [64] found that the involvement of companies has a significant effect on the sustainability of projects in PyPI ecosystem.

## 2.2 Developer Turnover in OSS

Developers are the backbone of OSS projects [44]. Their turnover has been extensively studied in OSS communities, where contributors are free to join or leave at any time. Considerable research effort has been invested in investigating how to attract newcomers [40, 60]. Researchers have identified a series of barriers faced by newcomers when making their first contribution to an OSS project, such as trouble deciphering the source code [59], being ignored [35, 59], developing tests for the patch [67], communication [59], and even the process of submitting the contributions [59, 67]. A survey study conducted by Lee et al. [40] found that most newcomers did not have the prior motivation to become long-term contributors, and the rest may face many challenges in their initiative activities in OSS projects due to technical factors, e.g., programming skills. Despite the barriers newcomers may face, researchers have also studied the mechanisms that are established by OSS projects to support the onboarding of newcomers. For example, Tan et al. [60] explored the easy-task recommendation mechanism (i.e., good first issue) in GitHub, and they found that (1) this mechanism has been increasingly adopted by the projects in GitHub; (2) although some newcomers successfully solved the recommended issues, most of them are one-time contributors; and (3) various problems, e.g., recommending insufficient and inappropriate issues, affect the effectiveness of this mechanism. A recent study conducted by Foundjem et al. [22] also investigated the process and impact of newcomer onboarding in OpenStack, and they found that onboarding has a significant correlation with increasing gender diversity and patch acceptance rates and has a significant negative correlation with the time until a contributor's first contributions.

The other studies focused on the reasons for the turnover and retention of developers in OSS projects. Hynninen et al. [32] found that low organizational commitment can cause the departures of developers from an OSS project. Yu et al. [71] suggested that developers' turnover can be partially explained by their dissatisfaction with OSS communities because personal expectations play a role in project retention. A study from Schilling et al. [56] unveiled that the level

of development experience and knowledge has a positive association with developers' retention. By modeling developers' initial behavior, Zhou and Mockus found that developers' early willingness and environment affect their chances of becoming long-term contributors [79]. Lin et al. [41] explored five OSS projects and found that developers have a higher likelihood of persisting in software projects when they (1) start contributing to the project earlier, (2) mainly modify instead of creating files, and (3) mainly code instead of dealing with documentation. Constantinou and Mens [9] conducted an empirical study on two OSS ecosystems and found that developers tend to have a higher probability of abandoning an ecosystem when they (1) do not engage in discussions with other developers; (2) do not have strong social and technical activity intensity; (3) communicate or commit less frequently, and (4) do not participate in either technical or social activities for long periods of time. A recent study conducted by Miller et al. [44] identified that lacking time is also a key reason for disengagement in OSS projects. Only one company withdrawal-related study conducted by Homscheid and Schaarschmidt [31] investigated the drivers that explain the turnover intentions of OSS developers paid by companies. However, they mainly focus on the individual level and do not convey any factors that affect a company's decision to withdraw. These studies identified a series of factors that can be used to understand the turnover and retention of developers in OSS.

The negative impact of developer turnover on software engineering has also been studied. Mockus [47] found that developers leaving a project can harm the quality and productivity of software development, possibly because of the lost experience and knowledge. Based on an analysis of five OSS projects, Foucault et al. [17] observed a negative effect of external turnover on software quality, which is consistent with the findings revealed by Mockus [45] on an industrial project. Rigby et al. [52] and later Nassif et al. [49] profiled the knowledge loss induced by developer turnover and provided tools to help large projects assess the risk of developers who are going to leave a software project. Izquierdo et al. [34] employed survival analysis and found a relationship between the lifetime of contributors and its impact on the continuity of an OSS project. Along similar lines, a recent work by Constantinou and Mens [10] found a positive relationship between the specialization of the leavers and the risk they bring to the OSS ecosystem.

Companies may decide to stop their developers from contributing, which is the kind of decision that directly affects an OSS ecosystem's sustainability [80]. Despite substantial studies on commercial participation and developers' turnover in OSS, the nature of company withdrawal remains unclear. This article bridges this gap by conducting an empirical study of OpenStack. Our study complements the literature by investigating the prevalence and rationale of companies' withdrawal and the factors that may indicate the reasons for a company's withdrawal. To the best of our knowledge, this study is the first attempt to systematically characterize the phenomenon of companies' withdrawal.

### 3 STUDY DESIGN

#### 3.1 Dataset Construction

We introduce how we construct our dataset, including the standard we follow to select projects, the approach to collect and clean data, and how we identify the withdrawn companies.

**3.1.1 Project Selection.** As an open infrastructure, OpenStack was founded in July 2010 by NASA and Rackspace (a large IT web hosting company [18]). Over time, OpenStack has become a collection of OSS projects for building and managing cloud computing platforms, such as Nova, Swift, and Neutron [18]. OpenStack follows a six-month, time-based release cycle [62]. By January 2021, OpenStack released 22 versions and comprised over 20 million lines of code contributed



Table 1. Dataset Overview

#Repositories	#Commits	#Developers	#Companies	#Versions
1,292	338,035	9,653	490	18

by more than 100,000 contributors from 194 countries, and received support from hundreds of companies [19].

The reasons for selecting OpenStack as the case study are as follows: (1) it is widely investigated by research communities [26, 41, 74, 75]; (2) it is a large OSS ecosystem with thousands of repositories, which guarantees the generality of our findings to some extent; (3) it is a highly active and mature ecosystem that has been actively developed for a decade, ensuring a sufficiently long commit history; and (4) its ecosystem involves different types of companies [74] that can be used to investigate their withdrawals. We expect this heterogeneity (i.e., varying from startups to high-tech giants in different sectors) to be a fruitful source for discovering diverse withdrawal reasons.

**3.1.2 Data Collection and Cleaning.** We reuse the processed data from our existing study [75], which investigated companies' collaboration in OpenStack using the version control data. The dataset includes 18 versions and was carefully cleaned, including merging developer identities and identifying the affiliations of developers and commits by using the OpenStack community member profiles.<sup>1</sup>

Although the reused data have a high level of accuracy (i.e., 93%) in the identification of developers' affiliations, we find some problems with the company names, e.g., both the abbreviation and the full name of one company appear in the dataset. Because our study of company turnover is sensitive to the company name, we design an extra step to merge the multiple affiliations of the company. More specifically, we obtain 602 distinct affiliations in the dataset. We manually "googled" each affiliation plus "OpenStack" to ensure that the affiliation represents a company entity. We find that 125 affiliations are problematic, among which 112 affiliations are merged with the existing ones. More specifically, (1) 68 affiliations are represented by the abbreviations of companies' full names. We merge the 54 repeated abbreviations (the corresponding full names appear in the dataset) and replace the remaining 13 abbreviations with their companies' full names. (2) Forty-eight affiliations are units of 48 companies, such as "Taobao" in "Alibaba". All 48 related companies can be found in the dataset, so we change the 48 affiliations to their corresponding companies' names. (3) Nine affiliations are updated because they are variants of the companies' names in our dataset, such as "bcom" to "b<>com".<sup>2</sup>

After merging multiple affiliations of companies, we update the affiliations of the developers whose prior affiliations are problematic. Table 1 summarizes the dataset after cleaning, covering 1,292 Git repositories, 338,035 commits, 490 companies, 9,653 developers, and 18 complete versions.

**3.1.3 Identification of Withdrawn Companies.** Due to the irregular nature of the contributions made to OSS projects, it can be difficult to discriminate contributors who are waiting for time to contribute again from others who simply withdrew from a project [1], and the same challenge is faced when identifying withdrawn companies. Previous research defines individual leavers as those contributors whose last commit was made before a fixed time, such as 180 days ago [17, 41], 60 days ago [58], or 365 days ago [34]. However, the contribution frequency of companies may

<sup>1</sup>Each profile has an "Affiliations" field, containing all the names of the companies that employed the developer to work on OpenStack and the corresponding time periods for those affiliations [20].

<sup>2</sup>A French research institute of technology dedicated to the future of hypermedia.

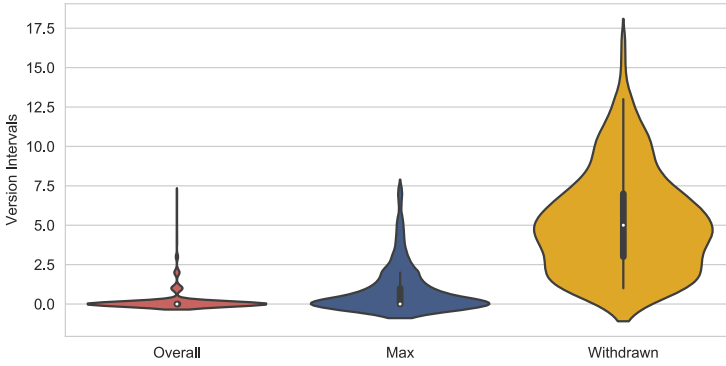


Fig. 1. Distribution of contribution intervals and maximum intervals of all companies and withdrawn companies' latest intervals.

vary due to different business strategies [74]. If we take a fixed timespan, the characteristics of different companies will be neglected.

We calculate the contribution interval of the companies to see the differences in contribution frequencies. For example, if a company contributes to the second version first and then contributes to the fourth version, the interval between the two contributions is 1 (i.e.,  $4 - 2 - 1 = 1$ , only the third version is escaped.). The left two violin plots in Figure 1 show the distributions of the contribution intervals and the maximum contribution intervals (i.e., number of versions) of all companies involved in OpenStack. The median intervals of the two distributions are both zero. More specifically, approximately 89% of all companies' intervals are equal to zero, and 60% of all companies' maximum contribution intervals are equal to zero. This indicates that most companies continuously contribute to OpenStack. For companies that contributed to multiple versions of OpenStack, approximately 40% (143 of the 361 companies) of them have maximum contribution intervals greater than zero, and that range from zero to seven. We can see that companies do have different contribution frequencies, so we cannot take a fixed timespan to determine the withdrawal of companies. Therefore, when judging whether the companies are withdrawn or not, we take into consideration their history of contribution intervals in the case of those companies that contributed to more than one version of OpenStack; for the companies that contributed only once, we take the median contribution interval, i.e., zero, of all the companies as standard. More specifically,

- For the companies (361 out of 490, approximately 74%) that contributed to more than one version, we calculate the maximum contribution interval of each company by version.
- For the companies (129 out of 490, approximately 26%) that only contributed to one version, we take the median interval, i.e., zero, of all the companies as the standard to determine the withdrawal.
- We calculate the latest interval of each company by the difference between **18** (the maximum version in our dataset) and the latest contribution version.
- If a company does not contribute to the 18th version and the latest interval exceeds its historical maximum interval (or the standard interval if contributed only once), we deem it to have withdrawn.

Among the 490 companies, we identified 320 companies that did not make contributions to OpenStack in the 18th version. After comparing the latest intervals of the 320 companies to their maximum intervals (or the median interval of all companies for those companies that contributed

only once),  $266 (54\% = \frac{266}{490})$  companies are identified as withdrawn companies. The right violin plot in Figure 1 shows the distribution of the latest intervals of the withdrawn companies. The median value of the latest intervals is *five*. We also conduct a manual validation of the withdrawn companies, and 34 out of 38 respondents acknowledge their companies' withdrawal. This suggests that the accuracy of our identification is approximately 89% (more details can be found in Section 3.4.1).

### 3.2 Measuring Turnover Rate of Companies

Inspired by the existing work about developer turnover [17, 46, 80], we propose the turnover rate of companies to assess the frequency of company withdrawal in OpenStack. More specifically, we calculate two types of turnover rates for each version<sup>3</sup>: (1) **Turnover of the joined companies** refers to the proportion of companies that joined in version  $v$  but withdrew later from OpenStack to the total number of companies joined in version  $v$ . (2) **Turnover of the sustaining companies** refers to the proportion of companies withdrawn in version  $v+1$  to the total number of companies contributing commits in version  $v$ . The first turnover rate stands in a historical view to see the relationship of companies' joined times and withdrawals. The second turnover rate is to see the change in company numbers between versions. The analysis is based on the dataset described in Section 3.1.

### 3.3 Characterizing Contribution Performance of Companies

To understand how important the withdrawn companies were to OpenStack, we need to measure the degree of their contribution to OpenStack before the withdrawal. We borrowed two metrics (i.e., contribution intensity and extent) from our previous study [74] to measure the contribution performance of the companies:

- Contribution intensity (abbreviated as **CI**) measures the degree of a company's contributions to OpenStack compared to other companies. A company's **CI** is defined as a ratio of the contributions contributed by the company to the total contributions of OpenStack. Higher values are better. The contributions are calculated in commit terms. The formula of how to calculate a company's **CI** is as follows:

$$CI(c, v) = \frac{\#commits_{c,v}}{\sum_i \#commits_{i,v}}, \quad (1)$$

where the numerators  $\#commits_{c,v}$  represent the number of commits contributed by company  $c$  to OpenStack in version  $v$ . The denominators represent the total number of commits in version  $v$ .

- Contribution extent (abbreviated as **CE**) measures the scope of a company's contributions to OpenStack. A company's **CE** is defined as a ratio of the number of repositories contributed by the company to the total number of repositories in OpenStack. As with **CI**, higher values are better. The formula is as follows:

$$CE(c, v) = \frac{\#repositories_{c,v}}{\#repositories_v}, \quad (2)$$

where  $\#repositories_{c,v}$  represents the number of repositories contributed by company  $c$  in version  $v$ . The denominators represent the total number of repositories in version  $v$ .

<sup>3</sup>Since we cannot determine company withdrawal of the last version in our dataset, we discard the statistics of the 18th version.



Similar to [74], we take the median values of a company's CI and CE in all the versions in which it has participated as its overall CI and CE, respectively. For the sustaining/withdrawn companies in each version, we take the median<sup>4</sup> of the companies' CIs and CEs to represent the coordinates of contribution intensity and extent, respectively.

### 3.4 Discovering Companies' Withdrawal Signals

We aim at understanding the reasons for companies' withdrawal from OpenStack and to identify the factors that predict the probability of withdrawal. Targeting the goals, we first conduct e-mail surveys with developers to obtain companies' reasons for withdrawal. Then, we use survival analysis to quantitatively explore the factors that may affect company turnover.

**3.4.1 Email Survey.** Although the turnover literature (more details can be found in Section 2) provides several factors that may affect developers' potential disengagement, there have been few studies on the actual reasons why companies withdraw from OSS. Therefore, we aim at bridging this gap by conducting an open-ended survey among companies that recently withdrew from OpenStack. We analyze the self-reported reasons provided by the core developers from each company to determine whether different companies withdraw for different reasons.

In Section 3.1.3, we have identified 266 companies that left OpenStack from the first version to the 18th version. To find knowledgeable developers to represent their companies, we select the top five developers (ranked by the number of contributed commits<sup>5</sup>), whom we deem to have deep insights regarding their companies' strategies in OpenStack, as representatives of their companies. We identify 455 candidates to survey. More specifically, we first ask them to confirm whether their company has left OpenStack. If the answer is "yes", we ask for their views on why their companies withdrew from OpenStack. If the answer is "no", we invite them to explain why their companies recently did not contribute commits to OpenStack. More details of the questionnaire can be found in the appendix [72]. We send the questionnaires to the candidates through emails. A total of 222 e-mails were bounced due to delivery problems. The reasons might be that the e-mail domain blocks lists or the e-mail addresses are abandoned due to job-hopping. After 20 days, we obtained 38 responses from 37 distinct companies, resulting in a response rate of 16% ( $\frac{38}{455-222}$ ). Of the 38 answers, four respondents indicated that their companies (four in total) had not completely abandoned OpenStack and would make contributions when necessary in the future. This suggests the difficulties of identifying withdrawn companies, and our method can reach an accuracy of greater than 89% ( $\frac{37-4}{37}$ ).<sup>6</sup>

We analyze the answers using thematic analysis [13], a common method for analyzing qualitative data. It involves the following steps: (1) initial reading of the answers, (2) generating the initial codes for each answer, (3) searching for themes among the proposed codes, (4) reviewing the themes to find opportunities for merging, and (5) defining the final themes aiming to identify the "essence" of what each theme is about. Steps (1)–(4) are performed independently by the first two authors. The final inter-rater reliability is 93%. In cases where conflicting decisions are made, a sequence of meetings is held to reach an agreement and to assign the final themes (step 5).

**3.4.2 Survival Analysis.** After investigating the withdrawal frequency, contribution degree, and reasons for withdrawal of companies, we obtained several factors that may be related to company

<sup>4</sup>We took the median value for a more reliable representation [69], while the mean value also presents the similar performance.

<sup>5</sup>Note that approximately 91% of the withdrawn companies (241 out of 266) have fewer than five developers.

<sup>6</sup>Thirty-seven companies are found in the 38 responses.

withdrawal. To validate these factors, we need to model to what degree the hypothesized factors, such as commercial goals, can predict the later withdrawal of companies.

Survival analysis, a popular method originated from medical sciences [38], can statistically quantify the occurrence probability of an event. More specifically, it analyses activities over time defined by one starting and one terminating event and considers cases that are still in progress [23, 38]. It is true for hundreds of companies in OpenStack that are sustaining simultaneously during the study. Survival analysis has been widely used to study problems in software engineering [4, 5, 25, 54, 80]. For example, Lin et al. [41] applied survival analysis to examine the impact of four factors on the duration of developer contributions. Therefore, we adopt survival analysis to investigate the relationship between the factors mentioned in the preceding section and companies' withdrawal. More specifically, survival analysis consists of a set of methods that allow for modeling the probability that an event occurred under different situations. Because of the time-varying factors, we identified (e.g., contribution intensity), the time-dependent Cox model is more appropriate [23, 76] and is considered in this article. We use the "survival" package in R [61] to fit the model.

As the data collection in this phase is not yet fully automated, and given the huge manual effort needed to obtain the value of companies' commercial objectives, we have so far only been able to assemble a dataset of moderate size, i.e., 60 withdrawn companies, as well as for an equal-sized "control" group of companies that did not withdraw,<sup>7</sup> and both are randomly selected. The observing event in this study is *company withdrawal*. The observation period is from the start of the first version to the end of the 18th version. More introductions of the factors can be seen in Section 4.3.2. With this design, we fit a survival model to estimate which factors are statistically useful for indicating companies that are going to withdraw.

## 4 RESULTS

### 4.1 RQ1: How Common do Companies Withdraw their Employees from OpenStack?

We answer this RQ from two aspects: (1) the numbers of withdrawn companies per version; (2) the turnover rates of companies in OpenStack.

As shown in Figure 2, the red bars represent the number of companies joined in OpenStack, the blue bars represent the number of companies withdrawn from OpenStack, and the black plot shows the number of companies that contributed commits (i.e., sustaining) to OpenStack per version. The left y-axis corresponds to the red/blue bars, and the right y-axis corresponds to the black plot. The horizontal axes represent the versions from six<sup>8</sup> to 17. We can see that the number of new joiners is larger than the withdrawn companies before the 13th version. Therefore, the number of companies that are still involved is increasing from the sixth version to the 14th version. With fewer joiners and more withdrawals that even surpassed the number of joiners in the 14th and later versions, the increasing trend has disappeared. It has been proved that the productivity of experienced developers is always higher than newcomers [60]. Therefore, the general experience of the developers in OpenStack is decreasing, although the number of sustaining companies in the later versions (i.e., since the 11th version) are almost stable. Because more experienced companies withdrew when compared to the joining ones in the later versions. This suggests a dangerous signal because the sustainability of a growing OSS ecosystem relies on its experienced contributors [17].

Furthermore, for each version, we calculate the *Turnover of the joined companies*, as shown in Figure 3 in red. We can see that the turnover ranges from 0.33 to 0.68. In almost all the versions, more than half of the joined companies withdrew. *Turnover of the sustaining companies* is

<sup>7</sup>Survival analysis requires that the size of observations is 10-15 times the number of factors, and the treatment group has a similar size of control group [57].

<sup>8</sup>Note that we discard the first five versions of OpenStack because of the instability in the initial phase of the project [73].

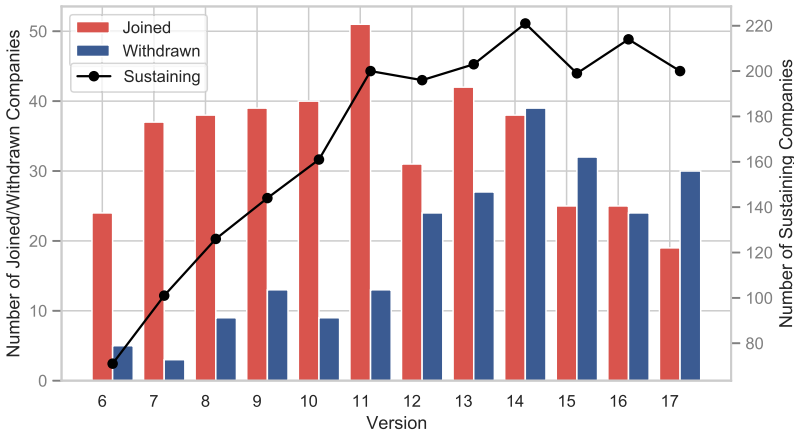


Fig. 2. Evolution of company turnover.

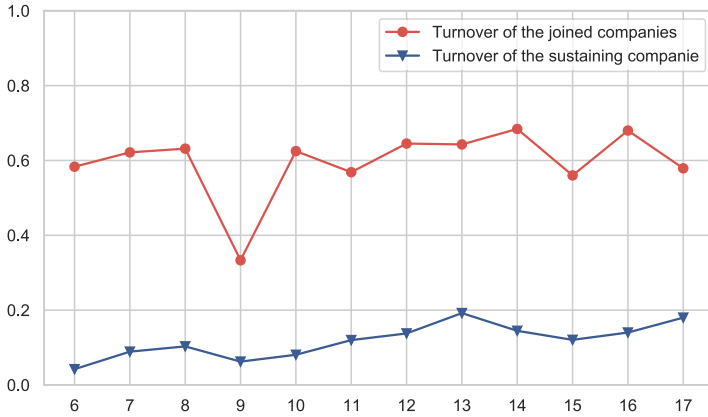


Fig. 3. Turnover of joined and sustaining companies.

presented with the blue line in Figure 3. We can see that the values range from 0.04 to 0.19 with a stable uptrend, and the median value is 0.12. It suggests that approximately 12% of the companies that sustain in the current version will withdraw in the next version.

*Summary:* The number of leaving companies is increasing over versions and even surpasses the number of joining companies in the later versions, ending the uptrend of contributing companies. More than half of the companies that joined in a certain version will withdraw later, and twelve percent of the companies that contribute to each version will withdraw in the next version.

#### 4.2 RQ2: To What Degree did the Withdrawn Companies Contribute to OpenStack?

Existing studies have investigated the impact of developer turnover and found that frequent developer turnover may lead to loss of productivity and code quality [17, 45] and even affect its survival probability [34]. The same impact might occur when companies withdraw from OSS because both are essentially developer losses. Since exploring the exact impact of companies' withdrawal on OSS

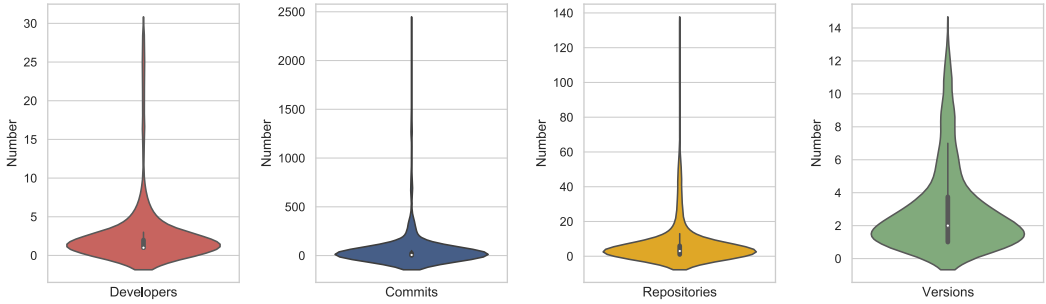


Fig. 4. Distribution of companies' history contributions before withdrawing.

can be a completely new study and is beyond the scope of this study, we intend to simply understand the importance of the withdrawn companies by measuring the degree of their contribution to OpenStack before the withdrawal.

To answer this RQ, we first calculate the distribution of the number of employees who are assigned to OpenStack and the number of commits contributed by these employees to obtain a general understanding of the historical contribution performance of the companies before their withdrawal. We also count the scope of their contribution (i.e., the number of repositories they contributed to) and duration (i.e., the number of versions they participated in) of the companies before their withdrawal. Then, we compare the contribution performance of the withdrawn companies to that of the companies that joined in the same version<sup>9</sup> and are still contributing to OpenStack.

Figure 4 shows the violin plots with the distribution of the number of developers, commits, repositories, and versions of the companies before their withdrawal. The number of developers ranges from 1 to 28, and the median is one. Although most of the withdrawn companies assigned only one developer to OpenStack, more than 700 developers from 266 companies were withdrawn from OpenStack. In the commit aspect, the range expands from 1 to 2,302 with a median of 6.5. The number of repositories ranges from one to 129 with a median of three. This indicates that one-half of companies are focusing on more than three repositories in OpenStack. In the version aspect, the value ranges from 1 to 13 with a median of two. We can see that more than half of the withdrawn companies contributed to OpenStack in multiple versions. Considering the difficulties of joining in an OSS ecosystem [22, 60] and the diverse capabilities of the companies, the phenomenon of more than two hundred companies withdrawing is to be noted, although each one may assign only one developer and contribute six commits to three repositories in limited versions.

We explore the history of the contribution performance of the withdrawn companies by comparing them with the companies that have stayed and that joined in the same version. Figure 5 shows the differences in the contribution intensity to the same version of OpenStack (indicated by the values in the X-axis) between the final withdrawn companies (represented by blue bars) and the companies that have stayed (represented by red bars). The values are calculated by the metrics defined in Section 3.3. It is obvious that the contribution intensity of the companies (whether sustaining or withdrawing) decreased over time. The reason might be that the companies that joined in the initial stage tend to play a core role in OpenStack, and the long-term companies contributed the most commits to OpenStack. To make the intensity comparison more obvious, we add a black line in Figure 5 to show the ratio of sustaining companies' intensity to withdrawals' in each version. We can see that the contribution intensities of sustaining companies are larger than those

<sup>9</sup>The contribution of companies may be affected by the evolution of OpenStack [73]. Therefore, we control the time variable by selecting the companies joined in the same version.

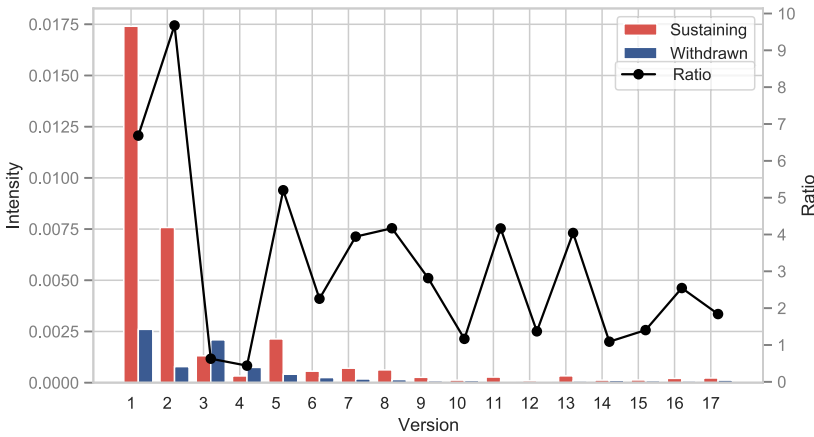


Fig. 5. Contribution intensity of sustaining companies and withdrawn companies.

of withdrawn companies in almost all ( $88\% = \frac{15}{17}$ ) versions, i.e., the ratio values in Figure 5 are larger than one. On average, the contribution intensity of sustaining companies is approximately 2.7 times that of withdrawn companies. Furthermore, we utilize the Mann-Whitney U test [48] to determine whether the contribution intensity of sustaining companies and withdrawn companies are significantly different. The p-value of the test is 0.037 (less than 0.05), indicating a statistically significant difference between the contribution intensity of the two groups of companies. The effect size of the test is  $-0.36$ , which is a medium effect according to Cohen's classification of effect sizes [24]. This indicates that companies with a lower contribution intensity tend to have a higher withdrawal rate.

Figure 6 presents the differences in the extent of the contribution between withdrawn companies (represented by blue bars) and sustaining companies (represented by red bars), where the companies participated in OpenStack in the same version (indicated by the values in the X-axis). Similarly, the black line in Figure 6 shows the ratio of the contribution by companies that have stayed to those that have withdrawn in each version. We can see that the contribution extent of the companies (whether sustaining or withdrawn) decreased over time. The reason might be the rapid growth in the number of repositories (i.e., from 12 to 1,143). More importantly, we can see that the contribution extent of the sustaining companies is not always larger than the extent of withdrawals, and the contribution extent of the two groups is close in most versions (the ratios in Figure 6 range from 0.7 to 1.2 in nine versions). On average, the contribution extent of sustaining companies is approximately 1.05 times the contribution extent of withdrawn companies, i.e., the difference between sustaining companies and withdrawn companies is slight from the perspective of contribution extent. Similar to the contribution intensity, we also conduct the Mann-Whitney U test [48] regarding the difference between the contribution extent of the sustaining companies with the withdrawn companies. As expected, the results show no significant differences (p-value = 0.63, effect size  $r = -0.086$ ).

*Summary:* In general, the withdrawn companies made limited contributions. In particular, by the median, they contributed one developer and six commits to three repositories in limited versions before the withdrawal. Compared to the companies that joined in the same period and are still sustaining, the withdrawn companies tend to have a weaker contribution intensity, but the extent of their contribution is similar.

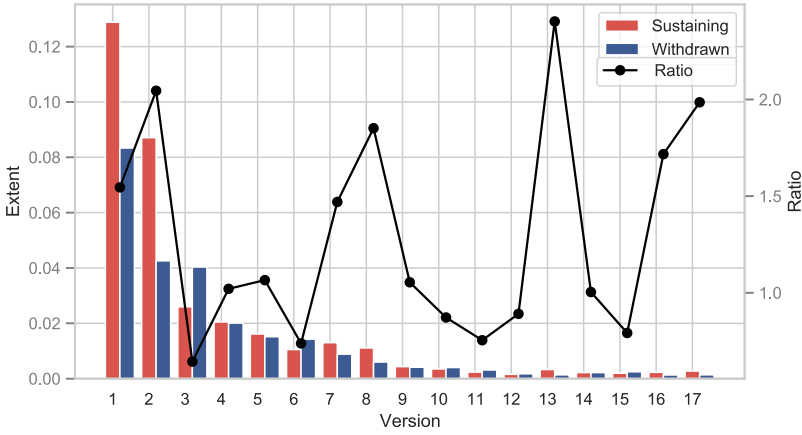


Fig. 6. Contribution extent of sustaining companies and withdrawn companies.

Table 2. Reasons of Company Withdrawal from OpenStack

Categories	Reasons	# Responses
Company	Commercial goal achieved	11
	Commercial goal failed	11
	Acquired	3
	Closed	1
Community	Dominance by other companies	3
Developer	Job hopping	3
Project	Difficult maintenance	3
	Roadmap conflicts	2

### 4.3 RQ3: What are the Signals Indicating that Companies are Going to Withdraw?

**4.3.1 Reasons for Company Withdrawal.** The respondents mentioned diverse and comprehensive reasons as to why their companies withdrew from OpenStack. The thematic analysis of the e-mail responses reveals eight reasons classified into four categories. Note that respondents (six in this case) may cite multiple reasons. We synthesize all the reasons that emerged from the survey in Table 2.

The first category includes the reasons for withdrawal from the **Company** side, pointed out by 21 different companies. More specifically, one of the most mentioned reasons is “Commercial goal achieved” (i.e., 11 responses). For example, one respondent says “When we started using OpenStack, it needed more work to be usable. As it became more mature, this was no longer necessary, and we were able to effectively use it without dedicating scarce personnel time to bug fixes or feature development...”. The other most mentioned reason is “Commercial goal failed” with 11 responses. For example, one respondent says “Our commercial efforts to make a public cloud were ultimately unsuccessful...”. The last two reasons from the company aspect are being “acquired” (three companies belong to this type) and “closed” (only one company belongs to this type).

The second category indicates the reasons for withdrawal from the **Community** side. More specifically, three respondents complain that OpenStack is dominated by other companies. For example, one respondent says: “... We no longer believed we could compete in the IAAS space using OpenStack given the direction these large contributors were taking it. . . . More focused on



pleasing traditional hardware and appliance vendors than simplicity... so we decided to get rid of it and as a result, leave the IT infrastructure market..."

The third category contains the withdrawal reasons from the *Developer* side. Three respondents indicated that their companies' withdrawals are due to job-hopping of the core or solo employee(s), who were responsible for OpenStack-related business. For example, "I moved to Canonical and another employee to Red Hat, and the rest were not OpenStack savvy and eventually dropped it."

The last category includes two withdrawal reasons from the *Project* side. One is a complaint about OpenStack's maintenance difficulties. Three respondents mentioned this reason. For example, "The major takeaway for leadership was staggering difficulty maintaining a production-grade OpenStack cluster. OpenStack did a very poor job abstracting complexity away. Upgrading it every six months became daunting busywork until the stack was just frozen and eventually replaced with a new setup." The other is the roadmap conflict between companies and the project of OpenStack. Two companies mentioned these reasons. For instance, one company simply tells that "It was deprioritized by the company because it was not aligned with the product roadmap."

**4.3.2 Results of Survival Analysis.** We identify the factors that may indicate companies' withdrawal from the answers for RQ2 and RQ3.1. As the answers for RQ2 suggest, the contribution intensity of the withdrawn companies is less than that of the companies that joined in the same version and are still involved. Thus, we hypothesize that **H1: companies that make more contributions have higher survival rates**. Companies hold different commercial goals when participating in OSS ecosystems [74, 80]. In addition, two reasons for company withdrawal, i.e., goal achieved or failed, indicate that **H2: how companies use OpenStack to achieve different goals may relate to their survival rates**. Two respondents mentioned the relationship of companies' scale and their withdrawal, e.g., "We are no longer going to use OpenStack as it proved too hard for a **small** IT team to maintain." Therefore, we also consider the scale of the company as a factor and hypothesize that **H3: companies of a larger scale tend to have higher survival rates**. Since companies also complain that their withdrawals are because of other companies' domination in OpenStack, we take domination as a factor and assume that **H4: the degree of a project being dominated has a negative impact on the survival rates of the companies participating in it**. As a result, we identify four factors that may indicate company withdrawal.<sup>10</sup> It has been found that turnover is detrimental to OSS projects [17, 34, 77]. Therefore, it is of interest to investigate the possibility of using explanatory factors (i.e., **contribution intensity, commercial goals, scale, and domination**) to predict company withdrawal in advance.

For each version, we measure a company's contribution intensity following the equation  $CI$  defined in Section 3.3. For the domination factor, we follow the previous work [73], including the following two steps. (1) For each repository in a specific version, we measure its degree of domination by the ratio of the contributions made by the company with the most contributions to the total contributions received by the repository. (2) For each company in a specific version, we choose the domination value of the repository that the company has contributed the most commits to as the value of the domination factor. As pointed out by existing studies [74, 75], the repository with the most commits for each company can present the company's interest in achieving its goals. So we deem the domination in a company's most interested repository may have the biggest impact on its withdrawal.

To identify the commercial goal of a company, we follow the method used in our previous studies [74, 75], where the goals of some companies have been categorized by using thematic analysis [7]. First, we search the Internet (using "OpenStack" and the company's name as

<sup>10</sup>In Section 6, we discuss why the other factors identified in Section 4.3.1 are not considered.

Table 3. Seven Commercial Goals of Companies in OpenStack

Commercial Goals	Description	# Companies
Selling Full Solutions (SFS)	Making profits by providing full cloud solutions to users, including private/ public/ hybrid cloud services, deployment, and maintenance services, etc.	45
Selling Partial Solutions (SPS)	Making profits by providing solutions to users only on the basis of one or two project(s) in OpenStack.	10
Integrating Business (IB)	Integrating OpenStack with their own business	25
Selling Complementary Services (SCS)	Making profits by providing complementary services, e.g., consulting and training services around OpenStack.	10
Usage (Us)	Using OpenStack in their production environment	25
Community oriented (CO)	Living symbiotically off an open source ecosystems	2
Development infrastructure vendor (DIV)	Providing development infrastructure for OpenStack	3

keywords) and collect the first 20 results. We also collect documents from the marketplace page on the official OpenStack website [19] regarding the products, services, or solutions offered by companies. Then, the first two authors independently perform deductive coding [16], i.e., apply the existing codes (from [74, 75]) to the collected records to identify the goal of a company toward OpenStack. We find a high level of agreement between the two coders with a Cohen's kappa coefficient [39] of 0.87, which shows high inter-rater reliability. After coding, the two authors discussed their disagreements to reach a consensus. We synthesize all the commercial goals that emerged from the 120 companies (i.e., 60 withdrawn companies plus 60 sustaining companies) in Table 3.

We take the number of employees to represent each company's scale, which is obtained by visiting the *About* page of companies' official websites or searching companies' names on LinkedIn and Crunchbase.<sup>11</sup> Since some companies only offer a scale range, we manually define five degrees by combining the existing criteria [2] of determining company scale and the distribution of company employees collected in this study: one refers to " $\# < 10$ "; two refers to " $10 \leq \# < 100$ "; three refers to " $100 \leq \# < 1,000$ "; four refers to " $1,000 \leq \# < 10,000$ "; and five refers to " $\# \geq 10,000$ " ( $\#$  refers to the number of employees of a company).

Before constructing the Cox model (as introduced in Section 3.4.2), we have investigated the distribution of the numeric variables, i.e., *CI* and *Domination*. For *CI* variable, which is detected with skewed distributions, we simply remove the top 1% of values (10 in 987 records and 977 remains) as high-leverage outliers by applying the method described by Patel [51] and applied by Valiev et al. [64]. Then we log-transformed *CI* to satisfy the modeling assumptions. We also apply the **variance inflation factor (VIF)** to detect multicollinearity problems [42] for the reliability and stability of the fitted model. The final regression equation is

$$\text{Surv}(tstart, tstop \text{ Status}) \sim \text{Log}(CI) + \text{Goal} + \text{Scale} + \text{Domination},$$

where *tstart* and *tstop* in the response are used to set the time interval for observing each company, and we treat each version as a time interval. *Status* in the response is a company's survival status in the time range. *CI*, *Goal*, *Scale*, and *Domination* are predictors, and we treat *Goal* and *Scale* as categorical variables in the model. The results of the fitted model are shown in Table 4.

<sup>11</sup>A platform for finding business information about companies [12].

Table 4. Coefficients of the Model (n = 977, #events = 260)

	coef	exp(coef)	p-value
log(CI)	-0.289	0.749	2.28e-14 ***
SPS	-1.18	0.307	0.002 **
IB	0.880	2.41	2.48e-07 ***
DIV	1.87	6.47	6.11e-14 ***
Scale: four	-1.91	0.148	5.71e-05 ***
Scale: five	-1.77	0.171	2.01e-10 ***
Domination	-0.0177	0.982	0.950

\*Only significant commercial goals and company scales are shown.

We follow Johnson's recommendation to use a p-value of 0.005 for statistical evidence instead of the commonly used value of 0.05 because using the latter value often leads to unreproducible results [36].

As expected, *CI* is significantly associated with increased survival rates, meaning that companies with more contributions tend to have higher possibilities of becoming long-term contributors when other factors are held constant. The *integrating business* goal and *development infrastructure vendors* have a higher possibility of withdrawing compared with the full solution-oriented companies. However, companies *being partial solution vendors* tend to have a higher survival rate. As for the scale predictor, the results show that two categories of company scale are significant (i.e., p-values are close to zero). Specifically, large companies with 1,000 to 10,000 (i.e., scale: four) employees and more than 10,000 (scale: five) employees are associated with higher survival rates. For example, holding the other factors constant, having more than 10,000 employees reduces the hazard ratio of withdrawing by a factor of  $\exp(\text{coef}) = 0.17$ , or 83%, when compared with the small ones with no more than ten employees. It may suggest that large companies have higher risk tolerance when involved in OSS ecosystems. Surprisingly, *Domination*, as a key factor in our model and derived from our survey results, does not show statistical significance. The reason might be that its potential good effects on OSS, i.e., positively associated with the productivity of contributors and the quality of issue reports [73], may offset the company's rejection of domination. Finally, the p-values for all three overall tests (likelihood, Wald, and score) are less than  $2e-16$ , indicating that the model is significant. Besides, *Concordance*<sup>12</sup> of the model is 0.86, indicating that the model explains the observed data well and has a good predictive ability.

**Summary:** The factors affecting company withdrawal are complex and diverse. We categorize eight types of reasons from four aspects identified by email surveys. The most common reasons are that the company goal was achieved or failed to be reached. We find that business integration vendors and development infrastructure vendors have a higher probability of withdrawing. However, the contribution intensity, scale, and being partial-solution vendors are negatively associated with the company's withdrawal. The effect of other companies' domination on company withdrawal is not significant.

<sup>12</sup>The most used measure of goodness-of-fit in survival models [23, 28].

## 5 DISCUSSION

We offer practical implications of our findings for OSS communities, companies, and researchers.

**OSS communities.** Although the effect of commercial participation in OSS has not been thoroughly investigated, companies (and their employees) play a crucial role in some large OSS ecosystems that rely heavily on companies. For instance, companies made approximately 80% of the contributions to OpenStack [74], and over 85% of code in the Linux kernel was contributed by more than 500 companies in 2017 [37]. Therefore, company turnover is crucial to the sustainability of this kind of OSS ecosystem. We have validated several factors that are significantly related to company turnover when answering RQ3. Researchers have already found that high turnover is harmful to the development of software projects [10, 17]. Therefore, it is necessary to investigate how to improve company retention in the OSS projects where companies participate intensively. We have found that 21% of the companies attributed their departure to the OpenStack community or its projects, i.e., projects that were dominated by other companies or that were difficult to maintain, as well as roadmap conflicts. Although we did not find a significant relationship between those project/community-related factors and companies' survival rate, there are a few companies that might face difficulties in continuing their contributions or are less loyal when OSS ecosystems are dominated by other companies, need hard maintenance, or change roadmaps frequently.

To achieve a better retention rate, it might be helpful to get companies more engaged in the projects by (1) Increasing the diversity of commercial involvement in the development of OSS projects. Specifically, the metric of company domination we proposed can be used by OSS communities to monitor their projects' domination degree. Once the degree of domination is larger than 50% (an empirical value [73]), the OSS communities may need to take further assurance. (2) Minimizing maintenance costs by optimizing release planning. As pointed out by the respondents in Section 4.3.1, it is important to design good complexity abstracting mechanism to ease the difficulty of upgrading to the latest distributions of the OSS projects. (3) Considering the needs of all parties when making the roadmap, instead of blindly leaning toward the dominant companies. Prior studies [74, 75, 80] have categorized companies into different types by combining their commercial objectives toward the target OSS projects and contribution performance. When designing the roadmap of a project, considering the appeals of all the companies might be impossible, however, the characteristics of different categories of companies can serve as an operational alternative.

There are some uncontrollable factors that lead to a company's withdrawal, including reasons from the side of the company (e.g., commercial goal failed) and those from the developer's aside (e.g., job hopping). Prior studies found that development productivity and code quality will be damaged when turnover occurs [17, 45]. Therefore, the OSS community needs to pay more attention to companies that may be planning to leave and take measures in advance to reduce the negative impact of withdrawal. The survival model proposed in this study with a concordance of 0.83 has the potential to be used to predict the possibility of a company's turnover in a specific version. For example, companies with a decreasing contribution intensity, being a business integrator, or having a small size tend to have a higher possibility of withdrawing. OSS communities can automatically detect (and help retain) the companies that may be about to leave and pay more attention to the maintenance of these companies' contributions.

For OSS projects where the majority of contributors are either volunteers or employees from nonprofit organizations, such as OSS foundations or academics, commercial participation may have a negative influence. For example, researchers found that the involvement of companies negatively affects the sustainability of projects in PyPI ecosystem [64]. It means that the impact of company withdrawal on OSS projects may sometimes be positive. Therefore, OSS communities need to carefully weigh the pros and cons of commercial participation from diverse aspects,

e.g., software development, resource supply, and sustainability of OSS ecosystems, and then take appropriate measures to deal with company withdrawal, i.e., accepting its exit, finding an alternative, or trying to retain them.

**Companies.** The results for RQ1 show that approximately 12% of companies that sustain each version will exit in the next version. In the end, 266 companies (54%) withdrew from OpenStack, with one of the most common reasons reported being the failure of meeting the commercial goal. Quitting halfway not only threatens the sustainability of OSS ecosystems but also wastes the time and talent of the companies themselves. Therefore, it is a good idea for a company to conduct a detailed investigation on the OSS projects they intend to participate in and formulate a reasonable participation strategy. More specifically, companies should pay more attention to the degree of alignment between their own priorities and the roadmap of OSS projects, keeping a balance between the company's profit and the community's interest. Researchers have categorized several classic contribution models from large OSS ecosystems [74, 75, 80]. These models, combining commercial objectives and the contribution performance of different companies, can be used as a guide to help companies develop their participation strategies.

**Researchers.** As presented in Section 4.3.2, we identify several factors (e.g., contribution intensity and commercial goals) that relate to companies' transition to withdrawing from the ecosystem. Thus, by anticipating these transitions, it may be possible to prevent companies from departing. However, the effectiveness of the predictors is not comprehensively evaluated. Researchers can build on top of our results and develop open research questions to deeply understand and improve company retention. This study explores company withdrawal only from the perspective of the overall OSS ecosystem. OSS ecosystems always include more than one project. A company may stay in the overall ecosystem but withdraw from a specific project. From the perspective of projects, further studies of company withdrawal should also be conducted in the future.

## 6 THREATS TO VALIDITY

We discuss threats to the validity of our study by following common guidelines for empirical studies [53, 70].

**Construct Validity.** We are interested in investigating company withdrawal from OSS ecosystems. To achieve this goal, we focus on its frequency, importance, reasons, and prediction. We believe that these questions have a high potential to provide unique insights and value for practitioners and researchers.

**Internal Validity.** The first threat relates to data preparation. On the one hand, the accuracy of identifying the withdrawn companies is a fundamental concern for the validity of our results. Different from the existing methods of identifying individual leavers, we additionally take into account the characteristics of the contributions of different companies, i.e., calculating the maximum contribution interval of each company. We conducted a validation, and 34 respondents from 38 responses agreed on our identification, indicating high accuracy of 89%. Although three respondents indicate their companies did not withdraw from OpenStack, all the three companies have not contributed to OpenStack for more than four years. It may prove the existence of a natural gap between "making no contributions" and "admitting withdrawal". Another disagreement on our identification is because of the respondent's affiliation mistakes. As reported by [74], the accuracy of identifying developers' affiliations by combining OpenStack' member profiles and email domain is close to 94%. A more precise method of identifying developer affiliation is needed before applying the findings of commercial participation in OSS into practice. Given the uptrend of commercial participation in OSS and the knowledge gap on company withdrawal, we believe that such data, even with a slight flaw, are worth studying. On the other hand, our study investigates only the main way companies contribute to OSS, i.e., modifying its source code. This activity is



an approximation, as companies can assign their employees to make other contributions, e.g., reviewing and reporting issues. Other types of contributing activities may reveal that some companies that considered withdrawing might be persistent contributors to the project. Company turnover based on multiple activities is also an interesting topic to investigate in further studies.

The second internal threat relates to the survey validity. As pointed out by existing studies [3, 44, 65], survey results might be affected by a selection bias: companies that did not respond may have had different reasons for disengaging. To address this, a more comprehensive investigation is needed in the future. In addition, developers, who are selected to represent their companies, may unconsciously or deliberately self-censor in their responses, providing socially acceptable reasons rather than real reasons—a common concern in turnover research [30, 44]. Our study reduces this threat by building a survival model on historic trace data rather than self-reported answers.

The last internal threat relates to the survival model validity. The statistical power of the survival model might be limited by the small sample size, which was affected by manually obtaining the commercial goals of companies. Among the eight reasons that lead to company withdrawal in Section 4.3.1, we discard analyzing four reasons when selecting the factors for survival analysis. Three reasons, i.e., acquired, closed, and developer job-hopping, are abandoned because their impact on the company's withdrawal is apparent. The discarded reason "roadmap conflict" is difficult to measure because the related information is always inaccessible. For the reason "Difficult maintenance", we follow the existing study [43] and measure maintenance difficulties by the number of bugs fixed in each version. When fitting the survival model, maintenance difficulty is collinear with other factors and against the requirements of the Cox regression, so we discard it. The remaining factors' VIFs range from 1.11 to 1.56, indicating no collinearity exists in our Cox model.

As a common threat [44], the factor operationalization in the survival model cannot capture the complete concept to be measured. Although we referred to the measurements from existing studies and experimented with different operationalization of our factors to ensure robustness and construct validity, one needs to be careful when generalizing the results beyond the specific operationalization in this study. Besides, this study mainly focuses on companies' withdrawal from the perspective of an OSS ecosystem, i.e., OpenStack. Sometimes practitioners, e.g., team leaders in an OSS repository, may want to learn about how many companies are leaving the repository in an OSS ecosystem, that is also an interesting topic and we leave it in our future work.

**External Validity.** We purposely select the OpenStack ecosystem because it can well represent a large and active ecosystem with intense involvement from diverse companies. Yin [70] emphasized that case studies are generalizable to theoretical propositions and not to populations or universes. The method we used to investigate company turnover in OSS, e.g., quantitative analysis, survey, and survival analysis, can be used to identify and verify more factors that affect the duration of commercial participation in other OSS ecosystems. Furthermore, we perform our case study on OpenStack with thousands of projects and hundreds of companies from different domains. Hence, we expect our findings to be generalized to other similar OSS ecosystems. In the future, we plan to conduct a study on more OSS ecosystems from different domains and scales.

## 7 CONCLUSION

Although commercial involvement in OSS development is still increasing, company withdrawal remains a knowledge gap in the literature. This article conducts an empirical study on OpenStack to understand how common company withdrawal is, to what degree withdrawn companies have made contributions, and what the reasons behind withdrawal are. We find that the number of withdrawn companies is increasing over time and even surpasses the number of new ones in the later versions, ending the uptrend of sustaining companies. More than half of the companies that joined in a certain version will withdraw later, and twelve percent of the companies sustaining



in each version will exit in the next version. In general, the companies that have withdrawn have made limited contributions but should be noted because of the difficulty of joining an OSS ecosystem. We categorize eight types of reasons for companies' withdrawals. The most common reasons are goal achievement or failure. Through the survival analysis, we find that the factors affecting companies' withdrawal are complex and varied. The survival model we conducted may be used to predict the retention probability of a company. Therefore, the OSS community can take related measures in advance for its sustainability. To facilitate replications or future work, we provide the data, scripts, and other resources used in this study online [72].

## ACKNOWLEDGMENT

We are grateful to the OpenStackers who answered the survey.

## REFERENCES

- [1] Daniel Bégin, Rodolphe Devillers, and Stéphane Roche. 2017. Contributors' withdrawal from online collaborative communities: The case of openstreetmap. *ISPRS International Journal of Geo-Information* 6, 11 (2017), 340.
- [2] G. Berisha and J. S. Pula. 2015. Defining small and medium enterprises: A critical review. *Social Science Electronic Publishing* 1, 1 (2015), 17–28.
- [3] Jelke Bethlehem. 2010. Selection bias in web surveys. *International Statistical Review* 78, 2 (2010), 161–188.
- [4] Christian Bird, Alex Gourley, and Prem Devanbu. 2007. Detecting patch submission and acceptance in oss projects. In *Proceedings of the 4th International Workshop on Mining Software Repositories*. IEEE Computer Society, 26.
- [5] Christian Bird, Alex Gourley, Prem Devanbu, Anand Swaminathan, and Greta Hsu. 2007. Open borders? immigration in open source projects. In *Proceedings of the 4th International Workshop on Mining Software Repositories (MSR'07: ICSE Workshops 2007)*.
- [6] Andrea Bonaccorsi and Cristina Rossi. 2006. Comparing motivations of individual programmers and firms to take part in the open source movement: From community to business. *Knowledge, Technology & Policy* 18, 4 (2006), 40–64.
- [7] Virginia Braun, Victoria Clarke, Nikki Hayfield, and Gareth Terry. 2019. Thematic analysis. *Handbook of Research Methods in Health Social Sciences* (2019), 843–860.
- [8] Peter G. Capek, Steven P. Frank, Steve Gerdt, and David Shields. 2005. A history of IBM's open-source involvement and strategy. *IBM Systems Journal* 44, 2 (2005), 249–257.
- [9] Eleni Constantinou and Tom Mens. 2017. An empirical comparison of developer retention in the rubygems and npm software ecosystems. *Innovations in Systems & Software Engineering* 13, 2 (2017), 101–115.
- [10] Eleni Constantinou and Tom Mens. 2017. Socio-technical evolution of the Ruby ecosystem in GitHub. In *Proceedings of the 2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering*. IEEE, 34–44.
- [11] Kevin Crowston, Kangning Wei, James Howison, and Andrea Wiggins. 2012. Free/Libre open-source software development. *Acm Computing Surveys* 44, 2 (2012), 1–35.
- [12] Crunchbase. 2020. Discover Innovative Companies and The People Behind Them. Retrieved from <https://www.crunchbase.com/>.
- [13] Daniela S. Cruzes and Tore Dyba. 2011. Recommended steps for thematic synthesis in software engineering. In *Proceedings of the 2011 International Symposium on Empirical Software Engineering and Measurement*. IEEE, 275–284.
- [14] Carlo Daffara. 2007. Business models in FLOSS-based companies. In *Proceedings of the Workshop Presentation at the 3rd Conference on Open Source Systems*.
- [15] Linus Dahlander and Mats Magnusson. 2008. How do firms make use of open source communities? *Long Range Planning* 41, 6 (2008), 629–649.
- [16] Jennifer Fereday and Eimear Muir-Cochrane. 2006. Demonstrating rigor using thematic analysis: A hybrid approach of inductive and deductive coding and theme development. *International Journal of Qualitative Methods* 5, 1 (2006), 80–92. DOI : <https://doi.org/10.1177/160940690600500107>
- [17] Matthieu Foucault, Marc Palyart, Xavier Blanc, Gail C. Murphy, and Jean-Rémy Falleri. 2015. Impact of developer turnover on quality in open-source software. In *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*. 829–841.
- [18] OpenStack Foundation. 2019. Introduction: A Bit of OpenStack History. Retrieved 28 Dec. 2019 from <https://docs.openstack.org/project-team-guide/introduction.html>.
- [19] OpenStack Foundation. 2019. OpenStack Website. Retrieved 28 Dec. 2019 from <https://www.openstack.org/>.
- [20] OpenStack Foundation. 2020. OpenStack Foundation: Member Directory. Retrieved 2 Jan. 2020 from <https://www.openstack.org/community/members/>.

- [21] The Linux Foundation. 2021. Participating in Open Source Communities. Retrieved 3 Feb. 2021 from <https://www.linuxfoundation.org/en/resources/open-source-guides/participating-in-open-source-communities/>.
- [22] Armstrong Foundjem, Ellis E. Eghan, and Bram Adams. 2021. Onboarding vs. diversity, productivity, and quality - empirical study of the openstack ecosystem. In *Proceedings of the 2021 IEEE/ACM 43rd International Conference on Software Engineering*. 1033–1045. DOI : <https://doi.org/10.1109/ICSE43902.2021.00097>
- [23] John Fox. 2002. Cox proportional-hazards regression for survival data. *An R and S-PLUS Companion to Applied Regression* 2002 (2002), 1–18.
- [24] Catherine O. Fritz, Peter E. Morris, and Jennifer J. Richler. 2012. Effect size estimates: Current use, calculations, and interpretation. *Journal of Experimental Psychology: General* 141, 1 (2012), 2.
- [25] Mathieu Goeminne and Tom Mens. 2015. Towards a survival analysis of database framework usage in Java projects. In *Proceedings of the IEEE International Conference on Software Maintenance & Evolution*.
- [26] Jesus M. Gonzalez-Barahona and Gregorio Robles. 2013. Trends in free, libre, open source software communities: From volunteers to companies. *it-Information Technology* 55, 5 (2013), 173–180.
- [27] Dietmar Harhoff, Joachim Henkel, and Eric Von Hippel. 2003. Profiting from voluntary information spillovers: how users benefit by freely revealing their innovations. *Research Policy* 32, 10 (2003), 1753–1769.
- [28] Frank E. Harrell Jr, Kerry L. Lee, and Daniel B. Mark. 1996. Multivariable prognostic models: Issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in Medicine* 15, 4 (1996), 361–387.
- [29] Joachim Henkel. 2006. Selective revealing in open innovation processes: The case of embedded Linux. *Research Policy* 35, 7 (2006), 953–969.
- [30] Peter W. Hom, Thomas W. Lee, Jason D. Shaw, and John P. Hausknecht. 2017. One hundred years of employee turnover theory and research. *Journal of Applied Psychology* 102, 3 (2017), 530.
- [31] Dirk Homscheid and Mario Schaarschmidt. 2016. Between organization and community: Investigating turnover intention factors of firm-sponsored open source software developers. In *Proceedings of the 8th ACM Conference on Web Science*. 336–337.
- [32] P. Hynninen, A. Piri, and T. Niinimäki. 2010. Off-site commitment and voluntary turnover in GSD projects. In *Proceedings of the 2010 5th IEEE International Conference on Global Software Engineering*. 145–154. DOI : <https://doi.org/10.1109/ICGSE.2010.25>
- [33] Daniel Izquierdo, Nicole Huesman, Alexander Serebrenik, and Gregorio Robles. 2018. Openstack gender diversity report. *IEEE Software* 36, 1 (2018), 28–33.
- [34] Daniel Izquierdo-Cortazar, Gregorio Robles, Felipe Ortega, and Jesus M. Gonzalez-Barahona. 2009. Using software archaeology to measure knowledge loss in software projects due to developer turnover. In *Proceedings of the 2009 42nd Hawaii International Conference on System Sciences*. IEEE, 1–10.
- [35] Carlos Jensen, Scott King, and Victor Kuechler. 2011. Joining free/open source software communities: An analysis of newbies’ first interactions on project mailing lists. In *Proceedings of the 44th Hawaii International Conference on System Sciences*.
- [36] Valen E. Johnson. 2013. Revised standards for statistical evidence. *Proceedings of the National Academy of Sciences* 110, 48 (2013), 19313–19317.
- [37] Corbet Jonathan and Kroah-Hartman Greg. 2017. 2017 Linux Kernel Development Report. Retrieved 3 Feb., 2021 from <https://www.linuxfoundation.org/2017-linux-kernel-report-landing-page/>.
- [38] J. P. Klein and M. L. Moeschberger. 2010. *Survival Analysis: Techniques for Censored and Truncated Data*. Springer.
- [39] J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics* 33, 1 (1977), 159–174. Retrieved from <http://www.jstor.org/stable/2529310>.
- [40] Amanda Lee, Jeffrey C. Carver, and Amiangshu Bosu. 2017. Understanding the impressions, motivations, and barriers of one time code contributors to FLOSS projects: A survey. In *Proceedings of the 2017 IEEE/ACM 39th International Conference on Software Engineering*. IEEE, 187–197.
- [41] Bin Lin, Gregorio Robles, and Alexander Serebrenik. 2017. Developer turnover in global, industrial open source projects: Insights from applying survival analysis. In *Proceedings of the IEEE 12th International Conference on Global Software Engineering*. 66–75.
- [42] Edward R. Mansfield and Billy P. Helms. 1982. Detecting multicollinearity. *The American Statistician* 36, 3a (1982), 158–160.
- [43] Vishal Midha, Rahul Singh, Prashant Palvia, and Nir Kshetri. 2010. Improving open source software maintenance. *Journal of Computer Information Systems* 50, 3 (2010), 81–90.
- [44] Courtney Miller, David Gray Widder, Christian Kästner, and Bogdan Vasilescu. 2019. Why do people give up flossing? A study of contributor disengagement in open source. In *Proceedings of the IFIP International Conference on Open Source Systems*. Springer, 116–129.

- [45] Audris Mockus. 2009. Organizational volatility and developer productivity. In *Proceedings of the ICSE Workshop on Socio-Technical Congruence*.
- [46] Audris Mockus. 2009. Succession: Measuring transfer of code and developer productivity. In *Proceedings of the IEEE International Conference on Software Engineering*. 67–77.
- [47] Audris Mockus. 2010. Organizational volatility and its effects on software defects. In *Proceedings of the 18th ACM SIGSOFT International Symposium on Foundations of Software Engineering*. 117–126.
- [48] Nadim Nachar. 2008. The mann-whitney U: A test for assessing whether two independent samples come from the same distribution. *Tutorials in quantitative Methods for Psychology* 4, 1 (2008), 13–20.
- [49] Mathieu Nassif and Martin P. Robillard. 2017. Revisiting turnover-induced knowledge loss in software projects. In *Proceedings of the 2017 IEEE International Conference on Software Maintenance and Evolution*. IEEE, 261–272.
- [50] W. Oh. 2007. Membership herding and network stability in the open source community : The ising perspective. *Management Science* 53, 7 (2007), 1086–1101.
- [51] W. Ray, Jagdish Patel, C. Kapadia, and D. Owen. 1977. Handbook of statistical distributions. *Journal of the Royal Statistical Society. Series A (General)* 140, 3 (1977), 383. DOI : <https://doi.org/10.2307/2344937>
- [52] Peter C. Rigby, Yue Cai Zhu, Samuel M. Donadelli, and Audris Mockus. 2016. Quantifying and mitigating turnover-induced knowledge loss: Case studies of Chrome and a project at Avaya. In *Proceedings of the 2016 IEEE/ACM 38th International Conference on Software Engineering*. IEEE, 1006–1016.
- [53] Per Runeson and Martin Höst. 2009. Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering* 14, 2 (2009), 131.
- [54] Ioannis Samoladas, Lefteris Angelis, and Ioannis Stamelos. 2010. Survival analysis on the duration of open source projects. *Information & Software Technology* 52, 9 (2010), 902–922.
- [55] Mario Schaarschmidt, Gianfranco Walsh, and Harald F. O. von Kortzfleisch. 2015. How do firms influence open source software communities? A framework and empirical analysis of different governance modes. *Information and Organization* 25, 2 (2015), 99–114.
- [56] Andreas Schilling, Sven Laumer, and Tim Weitzel. 2012. Who will remain? an evaluation of actual person-job and person-team fit to predict developer retention in floss projects. In *Proceedings of the 2012 45th Hawaii International Conference on System Sciences*. IEEE, 3446–3455.
- [57] David A. Schoenfeld. 1983. Sample-size formula for the proportional-hazards regression model. *Biometrics* 39, 2 (1983), 499–503. Retrieved from <http://www.jstor.org/stable/2531021>.
- [58] Pratyush N. Sharma, John Hulland, and Sherae Daniel. 2012. Examining turnover in open source software projects using logistic hierarchical linear modeling approach. In *Proceedings of the IFIP International Conference on Open Source Systems*. Springer, 331–337.
- [59] Igor Steinmacher, Tayana Conte, Marco Aurélio Gerosa, and David Redmiles. 2015. Social barriers faced by newcomers placing their first contribution in open source software projects. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. ACM, 1379–1392.
- [60] Xin Tan, Minghui Zhou, and Zeyu Sun. 2020. A first look at good first issues on GitHub. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 398–409.
- [61] R Core Team. 2013. R: A language and environment for statistical computing.
- [62] José Apolinário Teixeira and Helena Karsten. 2019. Managing to release early, often and on time in the OpenStack software ecosystem. *Journal of Internet Services and Applications* 10, 1 (2019), 7.
- [63] Jose Apolinario Teixeira, Salman Qayyum Mian, and Ulla Hytti. 2016. Cooperation among competitors in the open-source arena: The case of OpenStack. In *Proceedings of the International Conference on Information Systems*.
- [64] Marat Valiev, Bogdan Vasilescu, and James Herbsleb. 2018. Ecosystem-level determinants of sustained activity in open-source projects: A case study of the PyPI ecosystem. In *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ACM, 644–655.
- [65] Francis Vella. 1998. Estimating models with sample selection bias: A survey. *Journal of Human Resources* 31, 1 (1998), 127–169.
- [66] P. Wagstrom, J. D. Herbsleb, R. E. Kraut, and A. Mockus. 2010. The impact of commercial organizations on volunteer participation in an online community. In *Proceedings of the Academy of Management Annual Meeting*.
- [67] Jing Wang, Patrick C. Shih, Yu Wu, and John M. Carroll. 2015. Comparative case studies of open source software peer review practices. *Information and Software Technology* 67, C (2015), 1–12. DOI : <https://doi.org/10.1016/j.infsof.2015.06.002>
- [68] Joel West and Scott Gallagher. 2006. Challenges of open innovation: The paradox of firm investment in open-source software. *R&d Management* 36, 3 (2006), 319–331.
- [69] Taro Yamane. 1973. *Statistics: An introductory analysis*. (1973).
- [70] Robert K. Yin. 2017. *Case Study Research and Applications: Design and Methods*. Sage Publications.

- [71] Yiqing Yu, Alexander Benlian, and Thomas Hess. 2012. An empirical study of volunteer members' perceived turnover in open source software projects. In *Proceedings of the Hawaii International Conference on System Sciences*.
- [72] Yuxia Zhang, Hui Liu, Xin Tan, Minghui Zhou, Zhi Jin, and Zhu Jiaxin. 2021. Online appendix to "Turnover of Companies in OpenStack: Prevalence and Rationale". Retrieved 3 Mar. 2021 from <https://github.com/YuxiaZhang-BIT/Dataset-CompanyTurnover>.
- [73] Yuxia Zhang, Xin Tan, Minghui Zhou, and Zhi Jin. 2018. Companies' domination in FLOSS development—an empirical study of openstack. In *Proceedings of the ICSE'18 Companion: 40th International Conference on Software Engineering Companion*. IEEE.
- [74] Y. Zhang, M. Zhou, A. Mockus, and Z. Jin. 2019. Companies' participation in OSS development—an empirical study of openstack. *IEEE Transactions on Software Engineering* (2019), 1–1. DOI : <https://doi.org/10.1109/TSE.2019.2946156>
- [75] Yuxia Zhang, Minghui Zhou, Klaas-Jan Stol, Jianyu Wu, and Zhi Jin. 2020. How do companies collaborate in open source ecosystems? an empirical study of openstack (ICSE'20). In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*. Association for Computing Machinery, New York, NY, 1196–1208. DOI : <https://doi.org/10.1145/3377811.3380376>
- [76] Zhongheng Zhang, Jaakko Reinikainen, Kazeem Adedayo Adeleke, Marcel E. Pieterse, and Catharina G. M. Groothuis-Oudshoorn. 2018. Time-varying covariates and coefficients in Cox regression models. *Annals of translational medicine* 6, 7 (2018), 121–130.
- [77] Minghui Zhou and Audris Mockus. 2010. Developer fluency: Achieving true mastery in software projects. In *Proceedings of the 18th ACM Sigsoft International Symposium on Foundations of Software Engineering*. ACM, Santa Fe, New Mexico, 137–146.
- [78] Minghui Zhou and A. Mockus. 2015. Who will stay in the FLOSS community? modeling participant's initial behavior. *IEEE Transactions on Software Engineering* 41, 1 (2015), 82–99.
- [79] Minghui Zhou and Audris Mockus. 2015. Who will stay in the FLOSS community? modeling participant's initial behavior. *Software Engineering, IEEE Transactions on Software Engineering* 41, 1 (2015), 82–99. DOI : <https://doi.org/10.1109/TSE.2014.2349496>
- [80] Minghui Zhou, Audris Mockus, Xiujian Ma, Lu Zhang, and Hong Mei. 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 (2016), 13.

Received June 2021; revised December 2021; accepted January 2022