2022

Modeling the Effects of Diversity and Corporations on Participation Dynamics in FLOSS Ecosystems

Olivia Newton
University of Central Florida

MODELING THE EFFECTS OF DIVERSITY AND CORPORATIONS ON PARTICIPATION DYNAMICS IN FREE/LIBRE AND OPEN SOURCE SOFTWARE ECOSYSTEMS

by

OLIVIA B. NEWTON
B.S., University of Central Florida, 2013
M.S., University of Central Florida, 2017

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the School of Modeling, Simulation, and Training,
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2022

Major Professor: Stephen M. Fiore

ABSTRACT

A multitude of societal issues associated with the development of technology have emerged over the years including, but not limited to: insufficient personnel for maintenance; a lack of accessibility; the spread of harmful tools; and bias and discrimination against marginalized groups. I propose that a systems perspective is necessary to identify potential leverage points in technology production systems to influence them towards increased social good and evaluate their effectiveness for intervention. Toward this end, I conducted a mixed-methods study of a widely-adopted approach in tech production, free/libre and open source software (FLOSS) development. A survey was distributed to elicit responses from FLOSS project contributors to characterize their perceptions of diversity and corporate involvement as they relate to participation decisions and information gathering activities in online platforms. To complement this, an analysis of data from FLOSS projects on GitHub was completed to model participation dynamics. Survey results indicate that contributors attend to information that is used to infer group diversity and information about corporate decision making related to FLOSS systems. Furthermore, the influence of this information on participation decisions varies on the basis of economic needs and sociopolitical beliefs. Analyses of eighteen project ecosystems, with over 9,000 contributors, reveal that projects with no to some corporate involvement generally have broader contributor and user bases than those that are owned by a company. Taken together, these findings suggest that the internal practices of companies involved in FLOSS can be perceived as opaque and controlling which is detrimental to both the expansion of a project's contributor base and for increasing diversity across FLOSS ecosystems. This research highlights the need to differentiate projects on the basis of corporate involvement and community ethos to

design appropriate interventions. A set of recommendations and research propositions are offered to improve inclusivity, equity, and sustainability in tech development.

EXTENDED ABSTRACT

The development of technology has evolved significantly and permeated many aspects of our lives. Concurrently, a multitude of societal issues associated with the development of technology have emerged including, but not limited to: insufficient personnel for maintenance; a lack of accessibility; the spread of harmful tools; and bias and discrimination against marginalized groups. It is therefore critical to model technology production to identify leverage points to influence the direction of development towards social good. I propose that a systems perspective, integrating existing theory and research on knowledge production and contemporary technology development, is necessary to not only identify potential leverage points but also evaluate their effectiveness for intervention. Toward this end, I conducted a mixed-methods study of a widely-adopted approach in tech production, free/libre and open source software (FLOSS) development, which arguably lowers barriers to participation and affords increased innovation. This study thus entails two components: survey research and ecosystem analysis. A survey was distributed to elicit responses from FLOSS project contributors to characterize their perceptions of diversity and corporate involvement as they relate to participation decisions in addition to associated information gathering activities in online platforms. To complement this, an analysis of data from FLOSS projects hosted on GitHub was completed to model participation dynamics and their relationship with group and ecosystem level factors. Survey results indicate that contributors attend to (1) information that is used to infer group diversity in projects and (2) information about corporate decision making related to FLOSS systems. Furthermore, the influence of this information on their participation decisions varies on the basis of their economic needs and their sociopolitical beliefs regarding authority and control. Analyses of eighteen project ecosystems, with over 9,000 contributors, reveals that projects with no to some corporate

involvement generally have broader contributor and user bases than those that are owned by a company. Taken together, these findings suggest that the internal practices of companies involved in FLOSS can be perceived as opaque and controlling which is detrimental to both the expansion of a project's contributor base and for increasing diversity across FLOSS ecosystems. This research highlights the need to differentiate projects on the basis of corporate involvement and community ethos to design appropriate interventions. Based on these findings, a set of recommendations and propositions for future research are offered to improve inclusivity, equity, and sustainability in tech development.

Para mi mamá.

ACKNOWLEDGMENTS

As someone who values community and collaboration, I would be remiss to not recognize the many people who made this research possible. First, I extend my eternal gratitude to my advisor and committee for providing invaluable guidance and sharing their expertise which helped shape the form and value of my research. I would like to thank my advisor, Dr. Steve Fiore, for all that he has done to ensure that I have access to opportunities that underpin success in academia. More importantly, I want to thank Steve for inspiring me: his passion for open and collaborative scholarship inspired a similar passion in me. To my committee, I thank you for believing in my potential. I especially want to thank my committee for their graciousness and patience as I worked to complete a dissertation in the midst of a pandemic.

Second, I would like to thank peers who have extended their support and friendship to me throughout the course of my graduate career. These women, in a multitude of ways, have shown me that I am not alone, that I belong here, and that I am capable (in no particular order): Kimberly Stowers, Karla Badillo-Urquiola, Jihye Song, Samaneh Saadat, and Cintya Larios.

Third, I give my deepest thanks to my loved ones for providing that which has sustained me. Gracias mamá por tu inmenso amor; thank you for instilling in me a love of learning. Thank you, dad, for encouraging my creative endeavors. Thank you, Jonathan and Felix, my brothers, for always laughing with me. Thank you to my best friends, Cassandra and Caity, for being my biggest cheerleaders. Thank you to my goddaughter, Abby, for modeling joy and compassion like no one else could. Thank you, Tyler, for understanding me when I needed it the most.

Last, but not least, I would like to thank free/libre and open source software project contributors, and the folks who created and maintain the open information systems that enabled

the completion of my research. My hope is that this dissertation serves to improve these systems for them and everyone.

Portions of the research described in this document have been published in the Proceedings of Human Factors and Ergonomics Society International Annual Meeting (Chapter 4) and submitted in the form of a preprint to arXiv (Chapter 5). The presentation which corresponds to this document is available on Google Slides.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

AI	Artificial intelligence
ANOVA	Analysis of Variance
API	Application programming interface
CAS	Central Authentication Service
CE	Community Edition
CI	Corporate involvement
CSCW	Computer supported cooperative work
DARPA	Defense Advanced Research Projects Agency
ES	Expected shortfall
FLOSS	Free/libre and open source software
FOSS	Free and open source software
FSF	Free Software Foundation
GNU	GNU's Not Unix
GPL	General Public License
HCI	Human-computer interaction
ICE	Immigration and Customs Enforcement
IT	Information technology
IRB	Institutional Review Board
KaR	Knowledge at risk
KL	Knowledge loss
OSS	Open source software
RTA	Reflexive thematic analysis

SE	Software Engineering
STEM	Science, technology, engineering, and mathematics
QA	Quality Assurance
UCF	University of Central Florida

CHAPTER 1. INTRODUCTION

Collaboration is a key method for addressing complex problems and producing innovative solutions to meet pressing societal needs (Fiore, 2008; Salas et al., 2012; Stokols et al. 2008). Collaboration takes many forms—from small groups working together in close proximity to large, distributed collectives engaged in bursts of interdependent work. Ultimately, the value of collaboration for complex problems is derived from the blend of unique perspectives, knowledge, and skills of the individuals working together to address them (Uzzi et al., 2013). Collaboration has evolved into new forms with the nature of labor changing within and beyond organizational boundaries over time. One such example is open collaboration, in which "goal-oriented yet loosely coordinated participants interact to create a product (or service) of economic value," as is seen in free/libre and open source software (FLOSS) development (S. S. Levine & Prietula, 2014, p. 1416). Understanding collaboration in the context of software development projects is important because, not only is this form of work increasing, but the products and services that are created in them are becoming more broadly used. As a knowledge work domain, FLOSS development can be viewed as a space of social epistemology, where knowledge is produced through collective effort and the direction that such production takes is shaped by social relations (Fuller, 1988). FLOSS projects are thus a means through which collectives can construct knowledge, innovating on and improving the technologies used by people to learn, work, and express themselves.

The development models associated with FLOSS have grown into a dominant paradigm and already made a significant impact on labor as evidenced by its large community of participants and widespread adoption by organizations. Although there are a number of FLOSS licenses that impose differing constraints, they generally emphasize some degree of software

freedom (Rosen, 2005). Through its origins, it is grounded in the principles of freedom, innovation, and shared resources, and its community is described as valuing equality, sharing, and reciprocity with the aim of fostering innovation and advancing the commons (Germonprez et al., 2013; Rajanen & Iivari, 2015). The various technical, social, and educational services made widely available via FLOSS-based technologies illustrate the types of shared resources that these projects contribute to society.

Online platforms integrating workflow systems with social network features to support collaborative work in FLOSS projects have, arguably, lowered barriers for participation in software development (McDonald & Goggins, 2013). But, relevant to participation, research on FLOSS projects shows that team formation (Majumder et al., 2012), task curation (Sarma et al., 2016), and developer onboarding (Steinmacher et al., 2019) are still significant challenges. Furthermore, like other STEM fields, FLOSS projects are affected by issues related to social bias and a lack of diversity and inclusivity (Terrell et al., 2017). These social issues are linked to the barriers that influence the participation differences between demographic groups observed in FLOSS projects (Balali et al., 2018). For example, differences in participation have been found in studies of contributor gender: analyses reveal that fewer women than men participate in FLOSS development (El Asri & Kerzazi, 2019) and turnover-like processes are higher for women in that they tend to have a shorter duration of participation compared to men in FLOSS projects (Qiu, Nolte, et al., 2019).

The demographic imbalances in participation observed in work domains engaged in technological development can have cascading consequences on society more broadly. At each stage of development, from conceptualization and design to release, subjective decisions and interpretations are made with respect to what can and will be developed (Tufekci, 2015). These

decisions and interpretations are based on the knowledge and experience of the individuals involved in the development of the technology. According to the situated knowledge thesis (Haraway, 1988), "what one knows or experiences reflects one's social, cultural, and historical location" (Schmaus, 2015, p. 245). As such, the inclusion of groups that have been traditionally excluded from software development has implications for broader societal issues and labor questions with respect to equity in knowledge production, and the direction and application of technological advances.

Recently, researchers described FLOSS development as in the midst of a transformation brought on by increased corporate engagement that affords an opportunity for researchers to positively impact “the experience of human labor and social planning”, to “encourage skillful, diverse, inclusive global work” (Germonprez et al., 2019, p. 2,4). It is perhaps no surprise then that social diversity is an increasingly popular topic of discussion across technology-centered fields and FLOSS development is no exception to this trend (e.g., Aue et al., 2016; Daniel et al., 2013; El Asri & Kerzazi, 2019; Ortu et al., 2016; Terrell et al., 2017; Vasilescu et al., 2015).

The goal of this dissertation was to conduct a study of membership change in FLOSS projects that can inform strategies for the formation of collaborations that are diverse and inclusive. Because group composition has implications for the direction of technological developments and participation in FLOSS projects (Hilderbrand et al., 2020), it is important to understand membership change and the factors that drive it, including corporatization. An analysis of the relationship between these phenomena necessitates a systems approach to understand relevant factors and identify leverage points, or places in the system where interventions can be applied to effect change (Meadows, 1999). Towards this end, I model the relationship between membership change and social diversity in FLOSS ecosystems, and the

effects of increased corporate involvement on diversity in development projects. The primary contribution of this research is thus the evaluation of online platforms and group composition as leverage points for membership change dynamics in FLOSS ecosystems.

As a foundation for this research, I first review literature on membership change and FLOSS projects. For the topic of membership change, I summarize theories of organizational behavior and group work and related methodological frameworks linking it to collaborative processes and outcomes. Work in FLOSS development is distributed not only among groups of people but also across a variety of technologies, from forges and "social coding platforms" to communication channels and question and answer forums. Research on FLOSS projects thus spans across a number of disciplines but is primarily located within the overlap between software engineering (SE), human-computer interaction (HCI), and computer supported cooperative work (CSCW). Many of these, though, draw on concepts and theories from the social sciences literature, and "thick descriptions" (Geertz, 1973) of FLOSS have been produced in the humanities. This review of the literature is followed by a description of results from three studies of membership change in FLOSS projects.

CHAPTER 2. THEORETICAL AND EMPIRICAL ASPECTS OF MEMBERSHIP CHANGE

Membership change[1] has been studied extensively by organizational and group researchers to understand the phenomenon and its effects on collaboration in a variety of work domains. The term membership change is used to describe the departure of members from a group, the addition or arrival of new members to a group, and, to a lesser extent, the retention of members in a group. Membership change is typically used interchangeably with the term turnover, although the latter term is often used to describe the rate at which group members leave and are replaced by newcomers. Turnover can be either voluntary (i.e., the individual makes the decision to leave) or involuntary (e.g., the individual is let go by employer) (Dess & Shaw, 2001; Shaw et al., 1998). Membership change has been empirically linked to differences in collaborative processes and outcomes across different types of groups. In this chapter, I review organizational and group research literature to provide a foundational understanding of membership change as it has been studied to date.

Major lines of research in this area of study include the measurement of membership change, quantification of loss associated with turnover, and modeling the relationship between membership change and collaborative processes (e.g., communication) or outcomes (e.g, performance). Empirical studies of turnover have focused on joining and leaving behavior, but also the retention of members in an organization or group. Following the trend of distributed, self-organized collaboration, researchers have also identified the ways that turnover affects collective processes and outcomes in these forms of collaborative work. In general, across

[1] This broad term generally does not include team dissolution at the end of a project or team fracture in which collaborators are uninterested in working with each other due to, for example, prior conflict.

studies of turnover, it is observed that the addition of newcomers necessitates infrastructure and personnel support for onboarding and enculturation, and that the loss of members results in additive, multiplicative, or exponential effects on organizational or group outcomes, depending on the operationalization of the loss.

Measuring Membership Change

Perhaps the most common approach to studying turnover and its effects relies on the calculation of rates and proportions to quantify the amount of membership change observed during a given period of time. Researchers have also suggested more dynamic approaches to analyze the phenomenon by, for example, modeling the temporal dispersion of membership change and revealing patterns of membership change as they emerge over time (Hausknecht & Holwerda, 2013). Analyzing temporal patterns can, for example, show that turnover oscillates under certain conditions, with high and low rates observed at different points in time (see Figure 1). Alternatively, the analysis of temporal patterns may reveal that turnover steadily increases or decreases over time. These patterns of course have different implications for different types of groups and tasks, but they can provide insights about the group (e.g., their cohesiveness), or the project (e.g., its health). These insights can, in turn, inform the identification of opportunities for intervention, or strategies to mitigate and augment the effects of expected turnover.

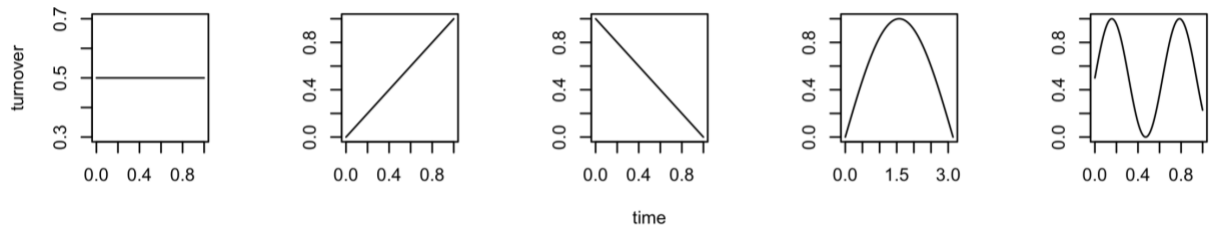


Figure 1 Potential patterns of turnover over time: constant, increasing, decreasing, inverted U, and oscillating.

Types of Turnover and Loss

Much of the research on turnover and its effects has been guided by human capital theory. This theory posits that the amount of human capital acquired by an organization explains observed organizational outcomes, specifically productivity, and that the negative effect of turnover is greater when leavers have high levels of organization-specific knowledge, skills, and abilities (Dess & Shaw, 2001). Human capital theorists thus argue that the loss of personnel through turnover should result in decreases in productivity and hypothesized effects differ on the basis of specificity and the location of the loss as it relates to the organization. Generally, when turnover in personnel occurs, there is a potential loss of different forms of capital (see Figure 2 for types of capital). Anklam (2005) describes the differing forms of capital for studying organizations:

> "*Human capital* [consists of] the capabilities of the individuals required to provide solutions to customers. *Structural capital* [consists of] the capabilities of the organization to meet market requirements. *Customer capital* [consists of] the value of the organization's relationships with the people with whom it does business. *Social capital*

consists of the stock of relationships, context, trust, and norms that enable knowledge-sharing behavior" (p. 9).

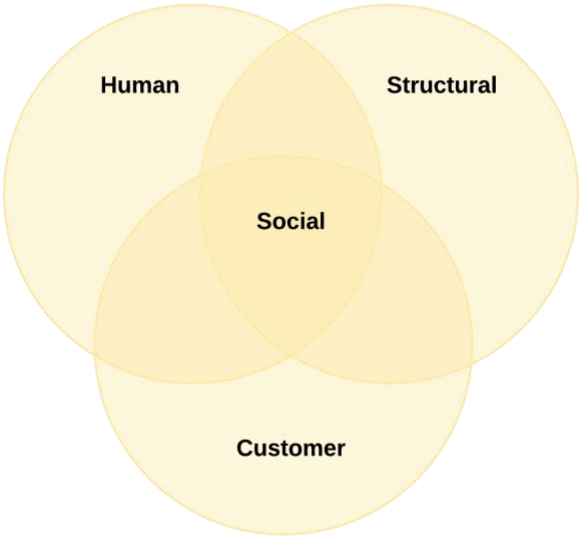


Figure 2 Differing forms of capital associated with organizations as described by Anklam (2005).

Given this, by taking into account social ties and influence, we see that turnover also results in the loss of *social capital* (Chou & He, 2011; Dess & Shaw, 2001). The distinction between human capital and social capital is important for the quantification of losses associated with turnover and evaluating the effects of those losses. Levine and Choi (2004) state that "there are differences between human and social capital losses in terms of the degree of performance erosion (e.g., additive or exponential) likely to result from high turnover levels [and] the type of intermediate performance or outcome variables each is likely to affect is expected to be different" (p. 449). An alternative approach taken by researchers focuses primarily on more task-oriented aspects for the quantification of losses resulting from turnover. This work provides a

foundation for the systematic assessment of turnover by emphasizing three properties of turnover: member proficiencies, that is, the skills and abilities of group members who left, joined, or remained; positional distribution, or the status and role, of group members who left; and the temporal distribution of members leaving the group (Hausknecht & Holwerda, 2013).

Modeling and predicting who will leave an organization or group is valuable for several reasons. In addition to characterizing the levels and types of membership change observed in a group, modeling turnover can inform the quantification of risk associated with member loss, and managers' strategies to mitigate the negative effects and augment the positive effects of turnover. Research in this area finds that employees leave organizations for a number of reasons, from the state of the economy (i.e., whether or not other jobs are available), organizational factors (e.g., leadership), to individual non-work-related factors (e.g., family), and individual work-related factors (e.g., values and satisfaction) (Mobley, 1982). Some research has also investigated the effect of performance requirements on turnover (e.g., emotional labor; Mesmer-Magnus et al., 2012).

Bao and colleagues (2017) investigated turnover in commercial software projects through an analysis of developers' monthly reports in information technology (IT) companies. Specifically, they were interested in identifying factors that predicted developer turnover in these companies. For prediction, they evaluated different machine learning classifiers. As input for the classifiers, they extracted 67 factors from developers' monthly reports that fall under one of six broad categories: Working Hours of Each Month; Overall Statistics of Working Hours; Statistics of Task Report; Readability of Task Report; Project Statistics of Each Month; and Overall Project Statistics. They found that the amount of content in a report and variance in working hours of developers overall and in the first month were the most important factors in predicting

developer turnover. They also found significant differences in these factors between developers who stay with the company and developers who leave the company. They speculated that the importance of variance in working hours overall suggests an instability in the developer's "work state" whereas the importance of variance in working hours in the first month suggests that the developer's experience of the working environment in the project shapes their perception of and, in turn, commitment to the project.

Team Assembly

A related but distinct research domain—team assembly—focuses on predicting the likelihood and success of collaboration. Team assembly is most clearly related to joining behavior in studies of membership change as it is concerned with identifying individuals who will come together to collaborate on a task[2]. Relevant research thus seeks to elucidate the factors that drive team assembly, including the selection and acceptance of newcomers in a previously formed collaborative group, and the success of the newly formed group. Although not explicitly stated in the team assembly literature, this line of research is related to onboarding; that is, how new members are encultured in the organization or group. Generally, team assembly describes the mechanisms and member characteristics that lead to collaborations, whether they be novel or repeated.

In terms of the mechanisms that drive team assembly, factors like reputation, popularity, and social ties have been found to predict the formation of a team (Table 1; Lungeanu et al., 2015). These mechanisms, in particular those that are classified as compositional and relational mechanisms, primarily represent individual preferences with respect to collaboration. For

[2] Team assembly is also conceptually related to team viability and team fracture. Prior team fractures constrain new assemblage of teams that includes members who experienced conflict in previous interactions (Whiting et al., 2019).

example, in the context of scientific collaborations, researchers prefer to collaborate with researchers who have seniority, a history of high performance, and are similar to them with respect to gender and institutional affiliation (Lungeanu et al., 2015). Additionally, in a comprehensive study of scientific and artistic teams, Guimera and colleagues (2005) found that the balance between team size and coordination needs is associated with the success of the team. With a sufficiently large team, member roles and expertise can become more specialized such that work can be more effectively distributed across members.

Table 1 Theoretical mechanisms of team assembly, adapted from Lungeanu et al. (2015).

Mechanism Type	**Description**	**Examples**
Compositional	Characteristics of individuals: internal preferences for collaborators, based on similarity (homophily), tenure, and record	Seniority, prior performance, gender inertia, institution affiliation inertia
Relational	Characteristics of relations: social ties and dynamics (preferential attachment)	Prior success, mutual friends, popularity
Ecosystem	Characteristics of the ecosystem: neighborhood connectivity	Coherency/fragmentation, redundancy

Team, or group, formation is another related research area. In the organizational sciences, team formation describes the processes by which members learn about each other and develop shared knowledge (Kozlowski et al., 1999). In data mining research, team formation describes a type of problem associated with team assembly in a social network. To address this problem, data scientists focus on creating algorithms that can reveal the member composition that lends itself to the emergence of effective collaborations while simultaneously reducing the costs associated with such collaborations. This type of research feeds into the development of

recommender systems as data scientists apply computational methods to identify the best potential collaborators to form a team within a social network. Thus far, research on group formation in computer supported cooperative work has focused on group composition, self-presentation, assembly mechanisms, recruitment, organizing structures, and group culture (Harris et al., 2019). Team formation in FLOSS projects in online platforms remains a particularly difficult problem to solve (Majumder et al., 2012).

Membership Change and Collaboration

Turnover has been linked to various performance outcomes and team processes (Argote et al., 2018; Levine & Choi, 2004) but the observed effects vary depending on the type of outcome or process under analysis. Research shows that turnover tends to have a negative effect on coordination and productivity but also introduces benefits in the form of newcomer innovation and creativity (Argote et al., 2018; Espinosa et al., 2007a; Huang & Cummings, 2011). In this section, I review research spanning several topics, including the effects of turnover in different types of groups, the effects of turnover properties on collaboration (e.g., newcomer versus leaver proficiencies), and factors that moderate the effects of turnover on collaboration.

Researchers propose that the effects of turnover depend on both the group's taskwork and the structure of the group. Intuitively, turnover is expected to have more harmful effects when a group works interdependently rather than independently—if each group member works on their own, then their work is less likely to suffer as a result of a member leaving the group. Furthermore, groups with low structure are more susceptible to the negative effects of turnover when compared to groups with high structure (Argote et al., 2018; Levine & Choi, 2004). Membership change in distributed, self-organized collaborations is also associated with negative

outcomes. In these types of groups, turnover is linked to lower levels of social integration, team learning behavior, and task flexibility (van der Vegt et al., 2010).

Turnover can have effects on social integration, or group members' affinity to, and satisfaction with, the group. This influences their engagement and investment in the team and its shared goals. As such, when social integration is high, teams are more effective. Team learning can also be affected by turnover but may depend on task flexibility. Team learning describes the actions taken by members to learn and help each other learn (e.g., asking questions and discussing issues), whereas task flexibility refers to the ability of group members to do their own and others' tasks (i.e., ability to transition from one task to another task). Pointing to the importance of the group's adaptiveness and proactivity for collective outcomes in self-organized groups, team learning behavior and task flexibility have been found to mediate the negative relationship between turnover and collective productivity (van der Vegt et al., 2010).

Research in this area also finds that member proficiencies can explain collective performance differences observed across groups and organizations. The ability and status of newcomers predicts collective performance, where high newcomer ability and status are associated with higher performance (Levine & Choi, 2004) and low newcomer and remaining member ability paired with high leaver ability are associated with lower performance (Hausknecht & Holwerda, 2013). This is because oldtimers are more likely to accept the suggestions of newcomers with high ability and status, leading to a more effective integration of newcomer contributions. This research exemplifies the importance of accounting for differences in member proficiencies, or skills, when analyzing the effects of turnover on collective performance (Hausknecht & Holwerda, 2013).

Other research has studied turnover effects in more controlled laboratory studies. Argote et al. (2018) conducted an experiment to analyze the effects of communication networks and turnover on shared knowledge structures[3] and performance outcomes. To do this, they manipulated the communication network (fully-connected versus centralized) and turnover in groups of university students working together on tasks drawn from the cooperative quadrant of McGrath's (1984) circumplex model. They found that the structure of the communication network mediated the effects of turnover, specifically observing that "perfectly centralized communication networks [. . .] reverse[d] the negative effects of turnover in groups" (p. 202). Furthermore, they observed high dyadic communication frequency in the centralized networks. Argote et al. suggest that centralized networks impose an explicit coordination logic that is ultimately beneficial for the integration of newcomer contributions. Their results provide evidence for the need to examine the dyadic interactions and turnover on collaboration. Given that turnover can serve as a "means to introduce innovation and creativity into mandated structures or centralized networks", they suggest the adoption of a centralized structure in addition to the "frequent rotation of team members" can lead to both increased innovation and the maintenance of efficiency.

Summary

Collaboration dynamics driven by turnover in teams can have varying effects on group processes and the achievement of shared goals. These effects range from those detrimental to collaboration to those providing sources of creativity and innovation. Generally, human capital

[3]Specifically, transactive memory systems, a type of shared knowledge structure that captures how expertise is distributed and coordinated within a group by knowing who knows what; it includes the relevant knowledge within the individual and the processes employed by group members to process information and build knowledge.

loss can result in negative effects that are additive, whereas social capital loss is associated with negative effects that are exponential. However, the association can be complicated in that an inverted-U shape relationship may exist between turnover and performance. This suggests that there is a possible optimal level of membership change. Because of this, research on turnover in modern work domains still has much to uncover with respect to what drives these performance changes.

When considering process changes, collaboration and communication structures can alter, magnify, or reduce the effects of turnover as research using network-based approaches reveals that these types of structures can be related to different outcomes. As an example, centralization and hierarchy can be leveraged for efficient coordination. Research on turnover typically focuses on local outcomes (e.g., individual or group productivity), but the increasingly distributed and open nature of work necessitates a broader perspective. Further, there is much that still needs to be learned about the global or societal outcomes resulting from turnover (e.g., changes in labor and the nature of work in an industry). Based on this, I argue that the distributed and decentralized nature of contemporary work, in combination with the increased transparency of organizations' inner workings, afford a unique opportunity to study the relationship between membership change and social issues in labor and production.

In the next section, I discuss FLOSS development as a contemporary form of work that has already begun to transform notions of labor and collaboration and holds significant potential for improving equity. I describe the characteristics of this evolving work environment and what is similar and different about this in relation to traditional forms of collaboration. The goal is to contextualize the aforementioned concepts associated with membership change in this work

environment to set the stage for exploring an integrated set of research questions advancing our understanding of FLOSS development.

CHAPTER 3. FREE/LIBRE AND OPEN SOURCE SOFTWARE DEVELOPMENT

As discussed at the outset, FLOSS development is a relatively new form of distributed work that takes advantage of a number of social and technical factors to change collaboration on complex projects. The FLOSS paradigm can be distinguished from other development paradigms in several ways but primarily in terms of its licensing and the makeup of its contributor base (Crowston & Howison, 2005). In its ideal form, the distribution, use, and modification of FLOSS is available to practically any individual with the adequate motivation and skill. Contributors to FLOSS projects may be developers paid by an organization or a volunteer with personal investment in producing working software. Over the past two decades, online platforms, like SourceForge and GitHub, have been created to support the distributed, collaborative work that characterizes modern software development projects. These platforms aim to facilitate collaboration in software development through an integration of workflow tools (e.g., version control systems and issue tracking) and social network features (e.g. following users).

FLOSS projects, by virtue of their nature and the use of these platforms, can be a transparent workspace, enabling empirical studies of collaboration in software engineering. Collaboration in FLOSS projects can be challenging as a result of the varying levels of expertise held by contributors and the sheer number of contributors. Some amount of turnover among contributors in FLOSS projects is desirable as it reflects interest in the software and growth in the community. However, turnover in these projects is also associated with knowledge loss and decreases in productivity (Nassif & Robillard, 2017; Newton et al., 2019; Rigby et al., 2016). In this section, I synthesize research on contributors to FLOSS projects and membership change among this contributor base. First, I describe the different classification schemes that have been used by researchers to study contributors to FLOSS projects. Then I summarize research on

onboarding and the social and technical challenges faced by newcomers in open source projects. Following this, I review the methods and findings of research on membership change in FLOSS development, specifically focusing on its relation to coordination, knowledge loss, and diversity.

Free/Libre and Open Source Contributors

Contributors to FLOSS projects have been characterized and differentiated from each other in a number of ways. Quantitative research tends to classify participants on the basis of their activity (e.g., high versus low), status and influence (e.g., technical role and/or popularity), and commitment (e.g., paid versus volunteer). Demographic information, including gender and geolocation, has also been used to capture differences between project contributors. Surveys, interviews, and ethnographic studies of FLOSS contributors also reveal that they differ and can be classified on the basis of their values and goals (Barcomb et al., 2018; Coleman, 2013). One consistent theme emerges across all of these classification methods: the FLOSS contributor base is far from homogenous (Bach & Terry, 2010; Coleman, 2013; Singh, 2012).

Types of Contributors

Much of the literature on contributors to FLOSS projects draws on the "onion model" and core-periphery framework to differentiate types of participants (Figure 3; Barcomb et al., 2018; Crowston & Howison, 2005). The members of the core and periphery are distinguished from each other on the basis of their level of activity and commitment to the project, where the core is the most active group of, sometimes paid, developers in the project and the periphery, while potentially contributing frequently, is less involved in the project. The core group is also much smaller than the periphery. Some older research using commit-based measures of developer role suggest that the core evolves over time, meaning that the core group constantly undergoes membership change. But this is inconsistent with more recent research using network-based

measures of developer role (Table 2) which suggest that the core group is actually relatively stable over time whereas the periphery group is volatile and characterized by short tenure in projects (Joblin, Apel, & Mauerer, 2017; Joblin, Apel, Hunsen, et al., 2017).

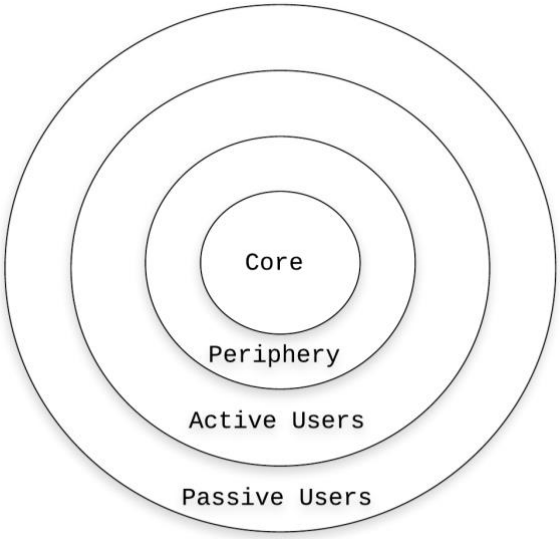


Figure 3 The onion model of free/libre and open source software development projects, adapted from Crowston & Howison (2005).

Table 2 Joblin et al.'s (2017) open source contributor types based on network structure.

Type	**Description**
Core	Active developers* with degree in the upper 20th percentile
Peripheral	Active non-core developers with non-zero degree
Isolated	Active developers with a zero degree
Absent	Inactive developers

*A developer is considered active if they make ≥ 1 commit(s) (i.e., they modify project files).

Barcomb et al. (2018) characterize different types of volunteer contributors in FLOSS, developing a framework of episodic volunteering informed by interviews with community members. Specifically, they gleaned information from interview responses to characterize volunteers' motivations, social norms, perceptions of community, satisfaction, and commitment. Episodic framing is contrasted with the more frequently used core and periphery distinction, where volunteers in FLOSS may be either *habitual* or *episodic* with respect to their contributions to a project (Table 4). Whereas the core-periphery framework emphasizes the amount of code contributed by developers, the habitual-episodic framework focuses on temporal aspects of contributions by volunteers, both in terms of length and frequency. Given that the core-periphery framing treats the volunteer group as relatively homogenous, Barcomb et al. suggest that this framing can help account for the heterogeneous nature of the volunteer group in FLOSS projects, including the fact that volunteers make contributions that are not limited to source code changes.

Table 3 Volunteer contributor types based on temporal nature of participation (Barcomb et al., 2018).

Type	**Description**
Habitual	Volunteers who "make continuous or successive contributions"
Episodic	Volunteers who "contribute infrequently and/or for a short duration"

FLOSS contributors can also be differentiated from each other in terms of their values and goals (Coleman, 2013) and social or demographic information (e.g., gender; Terrell et al., 2017). Contributors' geographic location and origins introduce national and regional differences, including socio-political ideologies. For example, in an ethnography of free software contributors, Coleman (2013) notes that:

> "southern European hackers have followed a more leftist, anarchist tradition than their northern European counterparts. Chinese hackers are quite nationalistic in their aims and aspirations, in contrast to those in North America, Latin America, and Europe, whose antiauthoritarian stance makes many—though certainly not all—wary of joining government endeavors" (p. 19).

Research is needed to understand how these differences vary across projects. On the one hand, understanding the relationship between these differences is important from a theoretical perspective to account for how they affect process and performance changes. On the other hand, accounting for these types of differences is critical for the achievement of a diverse, global workforce, especially given that these differences are determinants of the projects and communities that developers and other FLOSS contributors will choose to join.

One issue for the classification of participant types as it relates to membership change is the identification of role transitions. Role transitions within a group or organization may be considered instances of internal turnover in which an individual leaves a particular unit to join another one. Studying role transitions is complicated by differences in who is considered part of the group, team, and/or community. Researchers may focus solely on code contributors, leaving out non-coding participants who make crucial contributions to FLOSS projects as noted by Balali et al. (2018). To some extent, this is a constraint imposed by online platforms and workflow systems: the behavioral traces of non-code contributors contain less information that can be used to appropriately model individuals and groups. However, this also reflects a power dynamic between code contributors and non-code contributors in FLOSS projects. Although

some FLOSS projects may be meritocratic[4], decision making is often limited to the project founder or a small number of trusted contributors. For example, a study by Rajanen and Iivari (2015) revealed that usability specialists lacked power over the decision making of FLOSS developers, and their attempts to contribute resulted in conflict and/or exclusion from the project. Motivating developers to integrate ideas from others, including researchers, into the software project may therefore require alternative approaches. This might include harnessing the power of the user base to sway decision making and priorities in FLOSS projects (Nidy & Kwok, 2005).

What's In A Name?

To study and understand FLOSS development, it is important to draw attention to the distinctions in the labels used by contributors and communities. In this section, I describe the origins of the free and open software movements, and the fractures between them. The term 'FLOSS' is used to capture two groups: the free software movement and the open source movement. Free software and its communities are principally concerned with and motivated by notions of freedom. Free software is primarily associated with the Free Software Foundation (FSF)[5] and copyright licensing (e.g., the GNU General Public License) which provides a set of essential freedoms: "freedom to run, copy, distribute, study, change and improve the software" (*What Is Free Software?*, 2022). The free software movement is thus not centered on creating software that is free of charge, rather it promotes the creation of software which gives its users freedom. This was Richard Stallman's, the founder of FSF's, primary concern in using and distributing software, and as scholars have noted, he was not guided by a sense of justice

[4] This is an arguable point as research shows that meritocratic ideals are not always upheld in FLOSS projects (Ehmke, 2014; Nafus, 2012; Terrell et al., 2017).
[5] https://www.fsf.org/

(Coleman, 2013). The open source model and the Open Source Initiative[6], a non-profit corporation, emerged as a reaction to the "free software" language and similarly promotes its values through licensing, which provide "more liberties" (Elliott, 2003, p. 46). The FSF makes it clear, however, that their philosophy does not align with the values ascribed to the open source movement: practicality, popularity, and success (*Why Open Source Misses the Point of Free Software*, 2022). These values are seen as benefiting business and sideline free software as a guiding principle. The additional liberties provided by some open source licenses allow companies to control the modification and deployment of software resulting in software that is not free. While all free software is open source, not all open source software is free.

Nonetheless, these fractures in contemporary software development are sometimes overlooked in studies of FLOSS projects. This is complicated by the use of the term FOSS (free and open source software), which drops libre and is viewed as qualitatively distinct from FLOSS by the FSF and associated communities (*FLOSS and FOSS*, 2021) and results in the conflation of the terms in favor of open source[7]. Communities from both movements are thus characterized as holding similar values but it is unclear how accurate this overlapping portrayal is, from the time of initial fracturing and up to now. Researchers describe FLOSS communities and projects as broadly holding values of equality, sharing, and reciprocity with the aim of fostering innovation and advancing shared resources (Germonprez et al., 2013; Rajanen & Iivari, 2015). Yet, developers and leaders in the community have demonstrated bias against particular social groups (Nafus, 2012; Terrell et al., 2017) and some have translated traditional social hierarchies into the

[6] https://opensource.org/

[7] To clarify, FSF and associated communities do not generally use the term FLOSS to describe themselves or the software programs they produce. Instead, they use free/libre software, free software, or libre software.

FLOSS space (Newton & Stanfill, 2019). Furthermore, while multiple ethnographic works have precisely detailed the origins, values, and significance of free software (e.g., Coleman, 2013; Kelty, 2008), less work has been devoted to examining how these characteristics influence differences in the quantitatively assessed phenomena in FLOSS projects, like participation. There are few exceptions to this, including the work carried out by Barcomb et al. (2018) to characterize episodic contribution across FLOSS development with careful attention paid to the sampling of projects, selecting projects according to the dimensions of community size (number of projects) and the community's orientation towards either software vendors or volunteers.

Over a decade ago, researchers began to distinguish between FOSS and OSS 2.0. The latter represents a move away from the conceptualization of FLOSS development as a participatory community which is engaged in volunteer labor and motivated by both personal needs and a desire to improve codebases as a social good, to propose the emergence of OSS development which is dominated by companies and paid labor (Fitzgerald, 2006). This distinction has not necessarily been adopted or applied in a broad manner in research on FLOSS or OSS, and there have been renewed calls to characterize OSS as a space in which corporate engagement is the norm and paid developers produce the majority of contributions (Germonprez et al., 2019). In early and more recent works, these changes are described as a transformation which has oriented open collaboration in software development towards commercial goals and needs. These distinctions may be based on real observations, but it is also the case that the free software movement continues to exist in these spaces, and associated projects are not cleanly separated or removed from in studies of OSS development, and in particular as it occurs in platforms like GitHub. This introduces limitations with respect to the validity of claims made based on big data, but also presents an opportunity to explore how computational methods can be

applied in a way that leverages the benefits of big data for improved qualitative classification of FLOSS projects.

Joining and Onboarding

For nearly two decades, there has been extensive research on onboarding in FLOSS because joining and specialization are significant challenges for contributors. Early work emphasizes the types of behavior—referred to as "joining scripts"—that newcomers should engage in when attempting to join a project community (von Krogh et al., 2003). These joining scripts are based on activity; newcomers are expected to maintain some level of activity and participate in particular types of activity (e.g., reporting bugs) if they want to be accepted into a project's community. Recent work takes a more nuanced approach to onboarding in open source projects, distinguishing between different types of barriers faced by newcomers and identifying ways that project managers can lower barriers and support participation. Specifically, this research identifies technical and social challenges for newcomers attempting to join open source projects in addition to the challenges faced by project managers to identify appropriate candidates in the newcomer pool for particular tasks (e.g., Gousios et al., 2016; Steinmacher et al., 2019).

Barcomb et al. (2018) find that social ties may play an important role in community participation given that study participants note that their contribution to FLOSS projects is oftentimes the result of an invitation. In terms of perceptions of community, the authors propose that codes of conduct can be useful in helping potential contributors assess their fit within a particular FLOSS community. They also propose that non-code contributors be thanked and credited in projects to promote satisfaction and continued participation. At least one study participant noted that some patterns of activity are particularly challenging for community

managers, specifically identifying burstiness—too much of it—as a reason for discontinued participation: "I'm absolutely losing volunteers in the community IT side because there's nothing keeping people involved because it's too 'bursty' in nature" (p. 12). These researchers point out that, while it is known that volunteers' decision to participate in and commit to FLOSS projects is influenced by social norms and association with positive causes, this topic has not yet received much attention.

Recently published work by Fronchetti et al. (2019) and Qiu et al. (2019) examines the signals that social coding platform users rely on when seeking opportunities to participate in FLOSS development. Fronchetti et al. (2019) constructed a prediction model to investigate the factors that predict developer onboarding in open source projects and found that popularity (measured in stars), the time to merge pull requests, and number of programming languages were the highest ranked predictors. Among these factors, popularity was the strongest predictor of developer onboarding. The importance of number of programming languages is consistent with theorizing suggesting that developer familiarity with programming language contributes to task difficulty and is thus a barrier to contribution (von Krogh et al., 2003). The presence of a code of conduct was the lowest ranked factor in Fronchetti et al.'s prediction model. Although this conflicts with the proposition offered by Barcomb et al. (2018), it should be noted that this was a binary variable in which the differences between codes of conduct are collapsed. In reality, codes of conduct vary in detail and foci and, while gaining popularity, are not yet common in open source projects (Tourani et al., 2017). Additionally, some projects do not simply lack a code of conduct—they instead adopt a No Code of Conduct[8]. This is problematic in that an explicitly

[8] https://github.com/domgetter/NCoC

stated ‘lack of code of conduct’ implies a particular norm for behavior that could be attractive or repulsive depending on contributor values.

Qiu et al. (2019) employed a mixed-methods approach to similarly study the signals used by contributors in choosing projects, and characterize those signals by investigating their observability in GitHub. They found that contributors rely on many signals: the level of activity in the project, the popularity of the project, the disposition of issue and pull request handlers, the presence of issue and pull request templates and labels, and the presence of an organized, detailed README. These signals vary by the degree to which they are observable, or discoverable, with some signals necessitating multiple observations and/or actions on the part of the potential contributors. The README file, which typically includes introduction and instructions and sometimes the purpose and status of a project (Prana et al., 2018), is one of the most directly observable and easily discoverable signals in a project as it is generally located on the main webpage. In contrast, the full contents of a project’s code of conduct are unlikely to be included in the README or the main webpage. Furthermore, README files, while generally task-oriented, also convey information about social norms in the project, including implicit approval of or alignment with particular views (Newton & Stanfill, 2019). This may help explain the low ranking of code of conduct in Fronchetti et al.’s prediction model for developer onboarding.

Social and Technical Challenges

Once contributors have decided that they would like to join a project, they face new challenges to overcome if they hope to be successful in the onboarding process. Contributors to FLOSS projects frequently identify social issues as significant impediments to their ability to participate. Gousios et al. (2016) described social challenges faced by contributors in FLOSS

development and noted that newcomers benefit from explicit guidelines for communication and the contribution process. Indeed, the presence of a code of conduct in a project may not predict developer onboarding (Fronchetti et al., 2019), but it has been linked to the retention of newcomers (Tourani et al., 2017). Considering that newcomers identify the lack of participation guidelines as a barrier to contribution (Gousios et al., 2016), it stands to reason that the use of a code of conduct facilitates onboarding as it makes explicit the rules and norms of contribution in a project and/or the values held by those managing the project. In interviews with the creators of popular codes of conduct, Tourani and colleagues (2017) found that value-based phrasing in a code of conduct may dissuade some individuals from participating, while rule-based phrasing can support conflict resolution in FLOSS projects. Related to the socialization of newcomers, some FLOSS projects exhibit a gift culture in which newcomers are expected to offer gifts in the form of features or modules in order to be accepted by its current members (Rajanen & Iivari, 2015; von Krogh et al., 2003).

In surveys and interviews with developers and other project contributors, Steinmacher et al. (2019) identified four types of social barriers: reception issues, newcomers' communication, finding a mentor, and cultural differences. With the exception of finding a mentor, each of these barriers is explicitly linked to communication and all four barriers are associated with the social norms that govern interaction in FLOSS projects. Challenges emerging around newcomer communication behavior in particular increased with the size and age of the project, providing evidence of the interdependence between technical barriers present in FLOSS projects and the social barriers faced by newcomers when interacting with project contributors. Early work on joining behavior in FLOSS projects proposed a contribution barrier for newcomers that arises from the complexity of the technology under development (von Krogh et al., 2003). The

contribution barrier is a multifaceted conceptualization of task complexity that takes into account code complexity, programming language difficulty, software interfacing, and coordinative complexity (Table 4). This notion of technical barrier can serve to differentiate between modules, or components, being evaluated within the larger software architecture, within and between FLOSS projects. Newcomers typically select modules with lower technical barriers when choosing where and what to contribute to within a project.

Table 4 Four facets of the contribution barrier proposed by von Krogh and colleagues (2003).

Facet	**Refers to...**	**Explanation**
Code complexity	the structure and form of the source code, in terms of how easeful or difficult it is to modify	Algorithm difficulty varies based on the purpose and implementation of the module
Programming language difficulty	both the inherent difficulty of the language and its subjective difficulty in terms of developer's familiarity with the language	Different languages may be used in different modules developers may have to learn a new language in order to contribute
Software interface	the presence or absence, and usability, of software interfaces, particularly in the context of adding to the software architecture	Interfaces can serve as a means to ease exploration and understanding of the modules, and how they are connected
Coordinative complexity	the degree of interdependence between the module and other modules in the software architecture	Modules may function independently, or may be "intertwined", requiring more or less knowledge about the entire software architecture

A significant factor then in technical contribution barriers are thus the result of task complexity. In software development, complexity is measured using quantity-based (e.g., lines of code) or structure-based methods (program flow) and, importantly, is associated with software quality. Specifically, more complex code is typically more difficult to understand and maintain

(von Krogh et al., 2003), making the software more susceptible to errors (Pawade, Dave, & Kamath, 2016).

To deal with the challenges of onboarding developers, various recommender systems have been proposed to, for example, identify mentors (Canfora, Di Penta, Oliveto, & Panichella, 2012) and curate tasks (Sarma et al., 2016) for newcomers. Ultimately mentorship in FLOSS projects requires labor on the part of core developers to guide, instruct, and collaborate with less experienced developers (Joblin, Apel, & Mauerer, 2017). For FLOSS projects, research suggests that having a subset of developers devoted specifically to onboarding is important for successful newcomer integration, but this is not a common practice (Balali et al., 2018). In their development and evaluation of a recommender system for mentor identification, Canfora et al. (2012) found that the proportion of communication between a newcomer and a mentor, compared to other factors, resulted in the best performance. Conversely, mentors' commit activity was associated with much poorer performance. The authors interpret this finding as evidence that highly active developers are either unwilling or unable to mentor newcomers.

Another important facet of onboarding in software development is the division of labor and specialization (von Krogh et al., 2003). While division of labor is typically conceptualized in terms of amount of work, specialization refers to the types of tasks that an individual is well suited for in the project and is related to the emergence of modularity in software (Joblin, Apel, & Mauerer, 2017). Specialization can thus be contrasted from task flexibility, or generalization. Feature gifts may serve as a mechanism through which newcomers can work towards specialization—this matters in particular for project outcomes in FLOSS development because specialization is associated with "the efficiency of innovation" (von Krogh et al., 2003, p. 1230). In FLOSS projects, specialization can be quantified by examining the location of code changes—

a developer who makes changes to a single component, or module, of the code base has a higher degree of specialization compared to a developer who makes changes to all of the components of a code base. Technical barriers and expertise are thus intricately linked in FLOSS projects.

Expertise

Expertise in software development is a blend of well-defined and ill-defined problem solving (Simon, 1973). From a well-defined standpoint, programming expertise draws from deep semantic knowledge and schematic knowledge structures integrated through a form of syntactic rules (D´etienne, 2002). But from an ill-defined standpoint, programming requirements are often incomplete and under-specified, meaning there are many possible solutions. Research shows that expert developers spend more time making program requirements explicit (Batra & Davis, 1992) and are more likely to include design requirements in their code (Cevalier & Ivory, 2003). In this way, experts help reduce ambiguity to the problem environment and reduce the uncertainty in these ill-defined problem spaces[9].

The complexity of work in modern software development is intensified by the fact that coding is a collaborative endeavor (Kilamo, Leppänen, & Mikkonen, 2015). Expertise in software development is thus a blend of individual and team cognitive processes (Fiore et al., 2010). Importantly, though, expertise is associated with more collaborative competencies. For example, research finds that expert programmers have superior communication and collaboration competencies (e.g., Kelley & Caplan, 1993), engage in more collaborative learning and teaching

[9] Although the reduction of ambiguity in this context is essential for the development of exact solutions (i.e., the creation of code), the presence of ambiguity in this form of work potentially enables the proliferation of solutions due to varied interpretations of the problem and in turn discussion around approaches and implementations of solutions (cf. McMahan & Evans, 2018), particularly in the space of FLOSS where developers can easily engage with each other's work due to the open nature of the code. This has implications both for evaluations of expertise and the emergence of technological innovation.

behaviors (Curtis et al., 1988), and are more likely to review code and consult with their peers (Sonnentag, 1995). When comparing top and average software developers, research showed that interpersonal networking ability adds to general cognitive ability (Kelly & Caplan, 1993; Riedl et al., 1991). Such differences between experts and non-experts persist in open source projects. FLOSS experts are characterized by more than just programming skills; they also share their knowledge and seek out assistance from others when needed (Baltes & Diehl, 2018). These findings suggest that researchers need to more fully study teamwork competencies in relation to task competencies and group composition and how they influence collaborative work outcomes.

Expertise in software development can be reflected in hours or years of experience (Dieste et al., 2017), or in high performance, sometimes realized through lines of code generated (Sonnentag, 1995, 1998). Although expertise and experience are not the same, tenure—or length of participation—can be a proxy for expertise in FLOSS development (Vasilescu et al., 2015). Studies with student and professional software developers have produced mixed findings, showing that experience has an inconsistent relationship with productivity and quality (Dieste et al., 2017) and self-assessed expertise (Baltes & Diehl, 2018). However, in studies of FLOSS projects, measures of differences in knowledge, perspectives, and experience positively predict productivity and project success (Vasilescu et al., 2015; Daniel, Agarwal, & Stewart, 2013).

Together, these studies have thoroughly mapped the space of barriers to contribution, distinguishing between the social and technical, identifying linkages between them, and specifying relevant interaction dynamics between mentors and newcomers. Some of this work has described expertise, typically in terms of how it emerges and can be observed in FLOSS projects but also relates it to onboarding and mentorship, albeit to a lesser extent.

Leaving

As a final consideration of membership change, "leaving" addresses the more explicit departure of contributors. This, too, can have varied ramifications for collaboration and cognition in FLOSS development. Early research on turnover in open source projects suggested that, in these types of projects, membership change is quite common in the core group of contributors (i.e., the most active 20% of all committers to the project) (Robles & Gonzalez-Barahona, 2006). However, this finding is not consistent with network-based measures of developer role showing that the core group is highly stable whereas the periphery group is volatile (Joblin, Apel, & Mauerer, 2017; Joblin, Apel, Hunsen, et al., 2017). Past research relies primarily on commit-based measures of developer role in which developers who are most active or produce the majority of commits to the code are classified as the core and all others belong to some part of the periphery. More recent work submits that the use of network-based measures of developer role provide a more accurate classification of project contributors by accounting for inter-developer relationships and points to studies showing that commit-based measures are not always consistent with developer perceptions of contributor role (Joblin, Apel, Hunsen, et al., 2017).

Relevant to this dissertation, both gender diversity (variation in the gender of a project's contributors) and tenure diversity in a FLOSS project predict turnover in medium-sized and large teams (Vasilescu, Posnett, et al., 2015; Vasilescu, Serebrenik, et al., 2015). In studies of project outcomes, high turnover rates are associated with lower software quality and productivity, and increases in defects as newcomers lack sufficient expertise and exhibit different levels of activity (Foucault et al., 2015; Mockus, 2010). This may be because "project survivors are likely to lack the understanding of the design and structure of the abandoned code" (Rigby, Zhu, Donadelli, &

Mockus, 2016, p. 1007). In other words, when developers leave a project, the knowledge associated with the code they have authored is lost and this loss impairs newcomers' ability to contribute to the project.

Types and Rates of Turnover

Robles and Gonzalez-Barahona (2006) found that FLOSS projects are subject to high instability with respect to the contributor base. Furthermore, the organizational structure of these projects differs from traditional projects in that it is highly dynamic. One of the primary goals of their research was to characterize the core group of contributors in libre software projects. They hypothesized that two types of group composition are plausible: a code god scenario and a regenerative scenario. In the code god scenario, the project is led by a group of developers from the beginning and this group remains with the project over time, ensuring its success. Implicit in this is the idea that the loss of this group would likely result in the death of the project. In the regenerative scenario, the project is led by different groups at different times in the project lifetime. Based on their analyses, Robles and Gonzalez-Barahona (2006) assert that the majority of the projects in their sample can be characterized as having a regenerative core group. In other words, most projects "have multiple core groups over time" (p. 283). The remaining projects exhibit either a code gods composition or a mixed composition (regenerative and code gods). The results of this research suggest that a regenerative/regeneration process in the core can be evaluated against other types of change in organizational structures. Additionally, the researchers cite work that considers a half-life parameter[10] for contributor groups in software projects; this could potentially be implemented in a computational model of turnover in FLOSS projects.

[10] The half-life parameter is "defined as the time required for a certain group of contributors to fall to half of its initial population" (Robles & Gonzalez-Barahona, 2016, p. 274).

In one of the more in-depth analyses of this phenomenon, Foucault et al. (2015) analyzed activity in five large open source software projects written in one of four programming languages to uncover the relationship between developer turnover and software quality. In defining turnover, they distinguish between *internal* and *external* turnover, where internal turnover describes the "movement of developers within [a] project" and external turnover describes the "arrival and departures of developers from the project" (p. 829). The former of these two types of turnover reflects roles changes, or internal mobility, within the project modules; this, in addition to the analysis of turnover's effects on productivity, distinguishes Foucault et al.'s work from prior research. The notion of internal turnover is conceptually related to membership succession—"the flow of members through organizational positions"—as proposed in the organizational sciences literature (Trow, 1960, p. 259). Rather than quantify turnover in terms of individual contributors, Foucault et al. quantified turnover in terms of the amount of source code attributable to newcomers, leavers, and stayers (i.e., code churn). To assess software quality, they used a measure based on the density of bug-fixing commits and differentiate between development bug fixes and post-release bug fixes. To derive turnover rates, the researchers assessed the suitability of different period sizes and decided on a 6-month period. In their analysis of turnover at the project level, they find two types of phases in the lifespan on software projects. The enthusiastic phase is characterized by a relatively higher number of newcomers to leavers while the alternating phase is observed when "either the number of newcomers or leavers is higher than the other one" (p. 835). Overall, Foucault et al. observed that the number of newcomers and leavers was very high (≥ 80% of developers). Additionally, they observed very little role change (i.e., few newcomers become stayers). To examine the relationship between turnover and software quality, they used Spearman correlation tests and found positive, strong

correlation between external newcomer activity and bugfix density, but no statistically significant correlation between external leavers activity on bugfix density. They also showed a positive, strong correlation between stayers and bugfix density. Relevant to the power dynamics between code and non-code contributors (Rajanen & Iivari, 2015) described in the previous section on participation, Foucault and colleagues note that their "metrics assume that the only way developers contribute to a project is by modifying its source code" (p. 837).

Coordination

The organizational sciences have long studied how it is that individuals and groups coordinate their interdependent activity in support of shared goals. Membership changes pose a particular challenge to coordinating collaboration effectiveness. Research on turnover in the open source projects has shown that there are differences in turnover and activity levels between core and periphery contributors, and this has implications for coordination. Joblin et al. (2017) constructed developer networks using historical data from the version control systems of ten large open source projects and modeled developer turnover with a discrete state Markov model. Specifically, they assigned a discrete state to every developer in the project for each time window to examine group stability. They found that core developers had a low probability of transitioning to the periphery and were unlikely to leave the project. This is in contrast to periphery developers who had a high probability of leaving the project and were unlikely to transition to the core. With respect to team size, their results suggest that relatively large projects with respect to team size require higher levels of explicit coordination, like communication, to maintain activity levels over time. Relevant to team size and coordination requirements, their analysis of network topology revealed that scale-free structure emerges when project growth exceeds 50 developers. At the same time, they observed notable differences in how groups of

developers were organized: the core group was hierarchically organized while the periphery group was not hierarchically organized. This observation can be linked to Argote et al.'s (2018) research. In summarizing their findings, Argote et al. conclude that hierarchical organization is better suited for performance in dynamic environments. Given that a hierarchical structure emerges among a subset of contributors in open source projects, differences in the nodes or network component that connect the hierarchically-organized and randomly-organized nodes are likely significant determinants for the effect of structure in a project or ecosystem on collaborative outcomes.

Knowledge Loss

One of the reasons that turnover is detrimental to productivity is because it results in knowledge loss; that is, knowledge about the work, how it was done and why, is lost when turnover occurs. Rigby and colleagues (2016) sought to understand how the risk of knowledge loss associated with turnover in FLOSS development could be modeled, predicted, and, ultimately, managed by a software team through a case study of two mature projects: Avaya and Chrome. They contend that the results of their study can enable information risk mitigation strategies and that successors can be assigned to files that are likely to be abandoned as a result of turnover. To identify abandoned files' authors, these researchers used a blame-based approach rather than a commit-based approach as the "blame function present in version control systems determines the person who last changed a line of code" (p. 1008). To quantify risk of developer turnover, they operationalized knowledge loss as the "source files abandoned after developers leave" a project (p. 1006). From this, they measured: loss distribution, expected loss, knowledge at risk (KaR), and expected shortfall (ES). The latter two measures—KaR and ES—provided likelihood and severity information about large losses and are based on techniques used to assess

financial risk. Rigby et al.'s analysis showed that the loss distribution in Avaya and Chrome did not follow a normal distribution which has implications for modeling and mitigating the effects of turnover. In their KaR and ES calculations, they found that between approximately 4% of total number of files are at risk for abandonment at a 5% chance of occurrence in a quarter. The ES calculation produced a mean of 797 files for Avaya and 709 files for Chrome for losses occurring approximately 5% of the time; while this is a relatively small percentage of total files, in large projects, it translates to hundreds of files. Rigby et al. argue that their findings demonstrate the utility of their approach for identifying developers and code that pose high risk and helping newcomers direct their contributions to meet project needs. If this approach is indeed effective for the identification of at-risk developers, then it may have potential utility for pinpointing opportunities to intervene and mitigate human capital loss.

Nassif and Robillard (2017) sought to replicate and extend research by Rigby et al. (2016) by conducting additional analyses and including more software projects in their analysis. The additional analyses were used to evaluate: 1) knowledge loss in different time period lengths; 2) the differential effects of knowledge loss through the weighting of abandoned files; and 3) the distribution of knowledge loss across modules in addition to the persistence of abandoned files across time periods. In their analysis of quarterly loss distribution, they observed the same general distribution across projects in their data set, with the exception of Chromium and Gitlab CE. The use of weighted knowledge loss per quarter did indeed produce different distributions compared to unweighted knowledge loss, revealing that a large number of the files that were abandoned were small rather than large. They also observed that, for most projects, abandoned files were localized to a small set of folders or were more evenly distributed across folders. Lastly, in the analysis of abandoned-file persistence, they observed that only one project,

Apereo CAS, quickly removed abandoned files from their folders whereas between a quarter and half (25-50%) of abandoned files in the other studied projects persisted for on average at least two years. In their limitations section, they noted that they assumed that core and periphery developers left projects at the same rate. Joblin et al.'s (2017) research on the evolution of developer coordination shows that the core group is stable and has a low likelihood of undergoing internal or external turnover. This is in contrast to the periphery, which is highly volatile. The Markovian modeling approach used by Joblin et al. might thus have utility for improving and extending Nassif and Robillards' work. The results of these studies add to the body of evidence that the effects of turnover differ on the basis of properties like positional distribution.

Group Diversity

Vasilescu and colleagues (2015) examined the effects of diversity on productivity and turnover in teams on GitHub. Although diversity can be conceptualized in any number of ways within software development teams, they focused on contributors' gender, commit tenure (across GitHub), and project tenure (within a given project). Productivity was measured by the number of commits to the main repository and its forks, and turnover was measured as "the fraction of the team in a given quarter that is different with respect to previous quarter" (p. 3794). They also controlled for several factors: team size, project forks, quarter index, overall project activity, project age, tenure median, and comments. To determine the effects of gender and tenure diversity, Vasilescu and colleagues used multiple linear mixed-effects models. Additionally, to analyze team productivity, they used piecewise, or segmented, regression models. This resulted in three models, one for each of the following: teams with less than eleven members (small), teams with between eleven and thirty members (medium), and teams with more than thirty

members (large). They found that gender diversity had a significant, positive effect on productivity across team sizes and tenure diversity had a positive, significant effect for medium and large teams. In other words, as gender diversity increased, so did team productivity. They failed to establish a relationship between gender diversity and turnover but it is worth noting that this type of diversity is low in GitHub and that most teams have no discernable gender diversity (i.e., they are identified as teams made up by only men or only women; Vasilescu et al., 2015). However, they did find that tenure diversity had a significant, positive effect on turnover. These effects were small but stable. Overall, this research provides evidence for the positive effect of gender and tenure diversity on productivity and, more generally, the value of diverse perspectives for collaborative outcomes. Although gender diversity is associated with positive outcomes in FLOSS projects, it is not frequently present in projects and in the cases that it is, it is observed to be very low (i.e., contributor groups are largely made up of men with just a few women). This is due to the generally low participation of women in FLOSS development.

Gender Differences in Participation

HCI and software engineering researchers have examined gender differences in participation within FLOSS collaborations to both confirm that women are indeed underrepresented in this space and to characterize the differences in experience that may be contributing to low and brief participation. Qiu et al.'s (2019) research demonstrates that there are quantifiable gender differences in the length of time before a contributor departs from a project, specifically showing that women are more likely to disengage before men. Some research has begun to explore the presence and effects of bias experienced in open source projects. This work is insightful given that FLOSS developers have some level of awareness of potential collaborators' gender in social coding platforms (Vasilescu, Posnett, et al., 2015) and

the salience of gender can influence interactions in social spaces (McNicol, 2013). Indeed, Terrell et al., (2017) found that contributions made by women whose gender is identifiable are accepted at lower rates when they are not "insiders" and that they are accepted at higher rates than those of men when their gender is not identifiable, indicating that negative evaluations of women's contributions are driven by bias, rather than quality of work. Imtiaz and colleagues (2019) investigated the effects of different types of bias that may be experienced by women and influences their observed behavior in social coding platforms. They found that the contributions of women were centralized to a smaller set of projects compared to men. Furthermore, their results showed that, in communication, women were less likely to use profanity and less likely to express positive or negative sentiment, thus remaining relatively reserved in their interactions compared to men.

Summary

Research related to turnover in FLOSS projects largely focuses on either joining behavior and onboarding newcomers or leaving behavior and the loss of human capital. Topics in this body of work are multifaceted and include contributor characteristics (e.g., motivation), task roles and role transitions, barriers to participation, coordination requirements, risk assessment, and the effects of diversity. Several themes emerged in this literature. First, the project contributor base is heterogeneous when considering dimensions like motivation but relatively homogenous when considering dimensions like gender. Second, turnover and code churn is expected and, to a certain degree, desirable in FLOSS projects, specifically among periphery contributors. Third, communication and social norms are significant components of the barriers to participation experienced by newcomers. Relatedly, FLOSS projects benefit from having roles dedicated to mentorship and coordination. Fourth, different types of diversity have been

empirically linked to turnover in open source projects. Additionally, women experience discomfort, bias, and harassment in FLOSS projects (Terrell et al., 2017) and, perhaps unsurprisingly, gender diversity is relatively uncommon in FLOSS development. Lastly, there is some evidence that volunteers' decision to participate in and commitment to an open source project is influenced by social norms and its association with positive causes. But, to date, the relationship between membership change, group diversity, and alignment between the values and goals of the organizations, communities, and individuals has received little attention from researchers. In the next two chapters I discuss work I have completed to redress some of these gaps by studying a number of the sociotechnical factors associated with collaboration dynamics and outcomes in FLOSS projects.

CHAPTER 4. GROUP COMPOSITION AND TECHNICAL BARRIERS

In this chapter, I describe a case study of membership change and group outcomes in FLOSS development. My collaborators and I focused on group and technical factors, specifically examining how task complexity and the distribution of expertise in a group moderate the effects of turnover on productivity. The majority of this material was reported in Newton et al. (2019) *Expertise and Complexity as Moderators of Turnover-induced Knowledge Loss in Open Source Software Development*.

In prior work, we adapted a set of theoretical issues associated with team cognition to discuss their utility in the study and measure of teamwork in open source projects (Newton et al., 2018). Building on this work (Newton et al., 2018), we studied how collective levels of productivity in open source projects are altered by differing amounts of expertise and developer turnover. Additionally, in line with theory and research in the cognitive and organizational sciences (i.e., cognition in the face of complexity; Espinosa et al., 2007; Klein et al., 2003), we examined how task complexity interacts with these factors. The goal of this research was to examine how differences in turnover-induced knowledge loss, task complexity, and the distribution of expertise predicted productivity levels in a large, mature open source project. Our research questions were as follows:

- RQ 1. Group Composition: How do varying levels of expertise affect productivity?
- RQ 2. Contribution Barriers: How do varying levels of task complexity affect productivity?
- RQ 3. Knowledge Loss: How do varying levels of turnover-induced knowledge loss affect productivity?

- RQ 4. Moderators of Knowledge Loss: Do variations in group composition and contribution barriers moderate the effects of knowledge loss on productivity?
    - RQ 4.1 Does expertise moderate the effects of knowledge loss on productivity?
    - RQ 4.2 Does task complexity moderate the effects of knowledge loss on productivity?

We thus modeled the relationships between these factors using data extracted from the GitLab Community Edition (CE) project, a web-based FLOSS used by developers to collaborate on code. We hypothesized that, in predicting productivity, there would be a significant main effect of: the distribution of expertise (H1); task component complexity (H2a) and task coordinative complexity (H2b); and knowledge loss (H3). We also expected that expertise and task complexity would moderate the effects of knowledge loss on productivity. Specifically, we hypothesized that there would be a significant two-way interaction between expertise and knowledge loss (H4), coordinative complexity and knowledge loss (H5a), and component complexity and knowledge loss (H5b), in addition to significant three-way interactions between expertise, complexity, and knowledge loss (H6a, H6b; see Figure 4 for visualization of main effects and interactions).

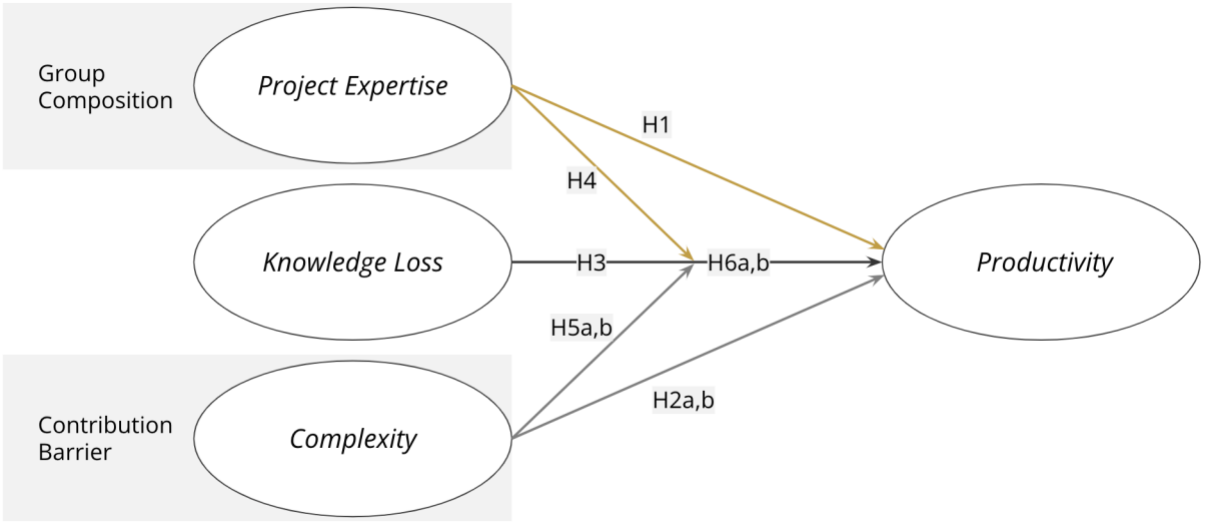


Figure 4 Diagram of hypothesized variable relationships. It was expected that: high levels of knowledge loss would be associated with low levels of productivity; high levels of task complexity would be associated with low levels of productivity; high levels of expertise would be associated with high levels of productivity.

Method

Because we were interested in extending existing work through an integration of concepts from the cognitive and organizational sciences with computational social science techniques in the study of knowledge loss in open source projects, we analyzed a subset of the data included in the replication package for a recently published study of knowledge loss in open source projects (Nassif & Robillard, 2017). We selected the GitLab CE project for our analysis. GitLab CE is a web-based FLOSS that is used by developers to collaborate on code. It provides tools and features that support the development lifecycle, including source code hosting, issue tracking, code integration, and software releases. In other words, GitLab CE is software created to support collaborative software development. We selected this project for analysis because it exhibited a more uniform distribution of knowledge loss across folders compared to other projects provided

by the researchers. The data set for the GitLab CE project spans from October 2011 (i.e., when the project was created) to November 2016. The data was partitioned into 21 quarters by Nassif and Robillard (2017), but our analysis was limited to the first 17 quarters because turnover information was not available for 2016. Across this subset of the data, there were a total of 349 unique project contributors and eight folders. We used a project folder as the unit of analysis as it is the level at which collaboration around code emerges at differing levels of complexity. This allowed us to examine whether differences in the distribution of expertise and task complexity influence productivity.

Variables

We used project tenure as a proxy for *project expertise* and quantified differences in how tenure was distributed within project folders with the Gini coefficient. Tenure was measured as the number of quarters since the developer's first commit to the project. The Gini coefficient is a statistical measure of dispersion that ranges between 0 and 1 and is commonly used to measure inequality in a distribution (Dorfman, 1979). In this case, a value closer to 1 indicates high inequality in the distribution of expertise and a value closer to 0 indicates more equality. As such, a high Gini coefficient suggests that most contributors to a folder were relative newcomers and only a few were experts.

We used two unique measures of complexity in the data set. Task complexity is defined in a number of different ways, sometimes varying based upon discipline. We follow the definition used often in the organizational sciences put forth by Wood (1986) as it has been used in studies of software teams (e.g., Espinosa et al., 2007). Here, task complexity is defined along two complementary dimensions: component and coordinative (Wood, 1986). Component complexity addresses the number of distinct acts associated with a task as well as the number of

cues or items that need to be processed. Coordinative complexity addresses the degree to which task variables need to be integrated for successful task completion. This is one way that the presence of task interdependencies can be operationalized in studies of collaborative work. Variations in task complexity have been empirically linked to differences in information request decisions (Topi et al., 2005), the use and effectiveness of cognitive artifacts for decision making (Speier, 2006), and the value of member familiarity for performance in distributed software teams (Espinosa et al., 2007b). To quantify *coordinative complexity*, we calculated the ratio of lines of code to the number of code contributors in a folder. A folder with many lines of code and many contributors produces a small ratio, which is indicative of high coordinative complexity. In contrast, a folder with many lines of code but few contributors results in a large ratio, which is indicative of low coordinative complexity. For *component complexity,* we calculated the ratio of lines of code to the number of files in a folder. A folder with many lines of code and many files has high component complexity and a folder with many lines of code but few files has low component complexity.

As discussed, prior research has analyzed the effect of turnover on collective outcomes in open source projects (Foucault et al., 2015) and modeled differences among types of developers (Joblin, Apel, & Mauerer, 2017), and the distribution of abandoned program files (Nassif & Robillard, 2017). We contend that any knowledge loss resulting from developer turnover can impede performance outcomes over time. Therefore, we define *knowledge loss* as the proportion of developers who have left the GitLab CE project.

Developer productivity within a project can be measured in terms of the number of commits they submit to the project. Commits are used to make changes to code and represent the

actual work activity for developers. To quantify productivity, we averaged the number of commits made by the developers who contributed to each folder in each quarter.

Results

We used multiple linear mixed models to analyze the effects of expertise, complexity, and knowledge loss on productivity. To perform this analysis, we used the statistical computation software R (R Core Team, 2018) and the lmer function from the lme4 package (Bates, Mächler, Bolker, & Walker, 2015), which was developed to fit linear mixed-effects models. Predictors were implemented as fixed effects and productivity as the response variable of interest. As random effects, we included intercepts for folder and quarter. This allowed us to control for the lack of independence of observations in the data (Baayen, Davidson, & Bates, 2008). The *p*-values reported here were obtained using the likelihood ratio test with the anova function from the stats package (R Core Team, 2018). For the models examining a single fixed effect, the test compared the full model with the particular fixed effect against a null model without the particular fixed effect. For the interaction models, the test compared the full model with the interaction terms for fixed effects against the null models without the interaction terms (Winter, 2013). Lastly, all assumptions of the models were checked by examination of residual plots. Tables 5 and 6 provide a summary of the main effects and interaction models, respectively, including model estimates and standard error (se), *t*-values, and effect sizes of the full model (i.e., fixed and random effects).

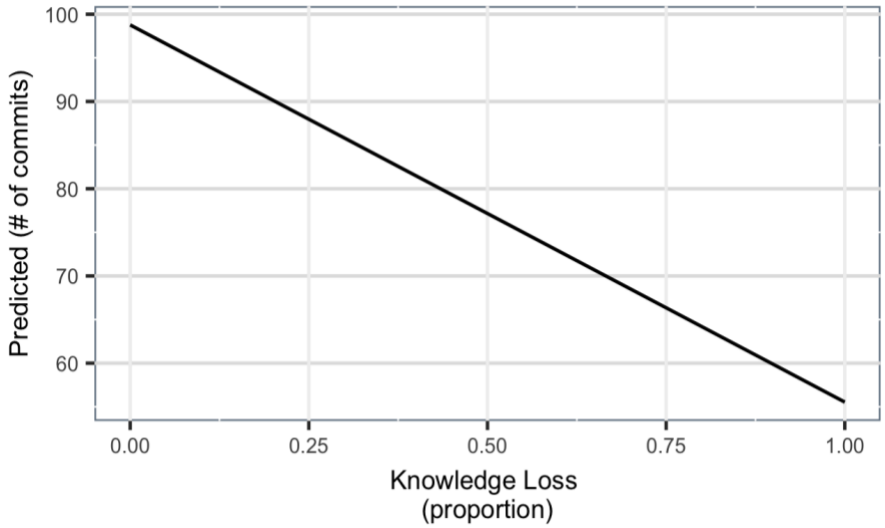


Figure 5 Marginal effects plot for the effect of turnover-induced knowledge loss on productivity.

In support of H1, inequality in the distribution of expertise predicted productivity. Expertise inequality—a larger number of newcomers—had a significant negative effect on the average number of commits completed by developers in a quarter. The effect of coordinative complexity on productivity was not significant (H2a) but, surprisingly, it was associated with a minor increase in average number of commits. In support of H2b, the effect of component complexity on productivity was significant; higher levels of component complexity were predictive of lower levels of productivity. In support of H3, higher proportions of knowledge loss were significantly predictive of lower levels of productivity (Figure 5).

Table 5 Predicting productivity with mixed model linear regressions.

Predictor	***B*** *(± se)*	***t-value***	R^2	ΔR^2 from null model	χ^2
Expertise	-39.64 (±17)	-2.33*	.94	-.00	4.79
Coordinative Complexity	.05 (±0.06)	.73	.94	-.00	0.46
Component Complexity	-.69 (±0.17)	-3.94***	.94	.00	13.61
Knowledge Loss (KL)	-43.23 (±9.17)	-4.72****	.95	.00	18.65

Note: * $p < .05$, ** $p < .01$, *** $p < .001$, **** $p < .0001$

The two-way interaction model for knowledge loss and expertise failed to reach significance (H4), but the negative effect of knowledge loss on productivity diminished as inequality in the distribution of expertise in a folder increased. This suggests that knowledge loss in these cases occurred primarily among newcomers. The two-way interactions between knowledge loss and coordinative (H5a) and component (H5b) complexity were significant. Although software complexity is typically detrimental to maintainability (Pawade et al., 2016), we observed that the negative effect of knowledge loss on productivity was less severe when task complexity was high (Figures 6 and 7). Moreover, the negative effect of knowledge loss on productivity was reversed when component complexity was high; in this case, we observed an increase in the average number of commits made by developers (Figure 7).

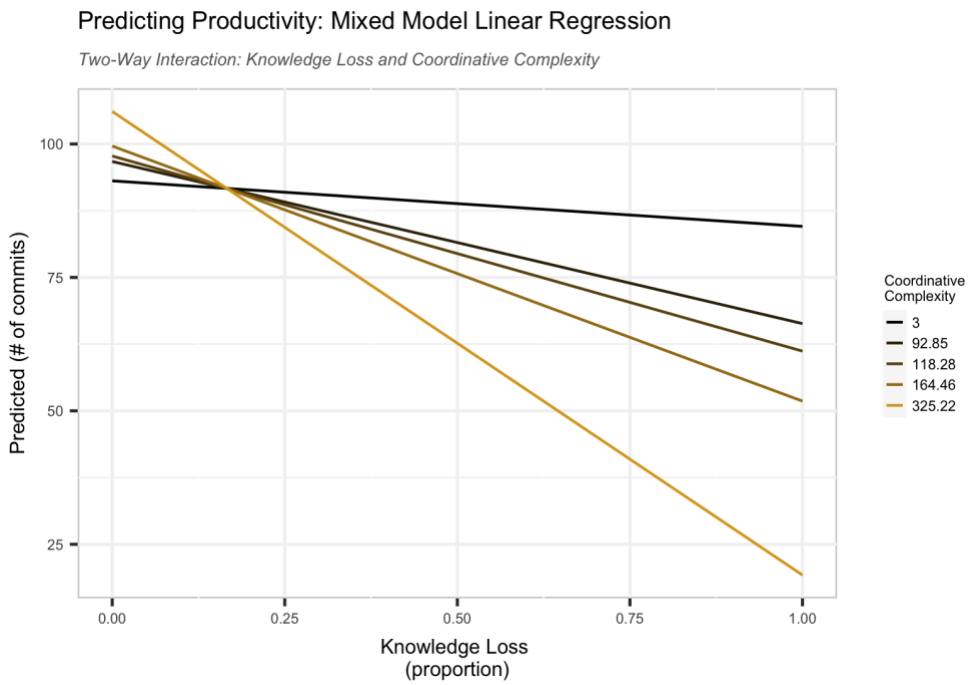


Figure 6 Marginal effects plot for the interaction between knowledge loss and coordinative complexity. Lower values indicate higher complexity. The black line represents the highest level of complexity and the yellow line represents the lowest level of complexity.

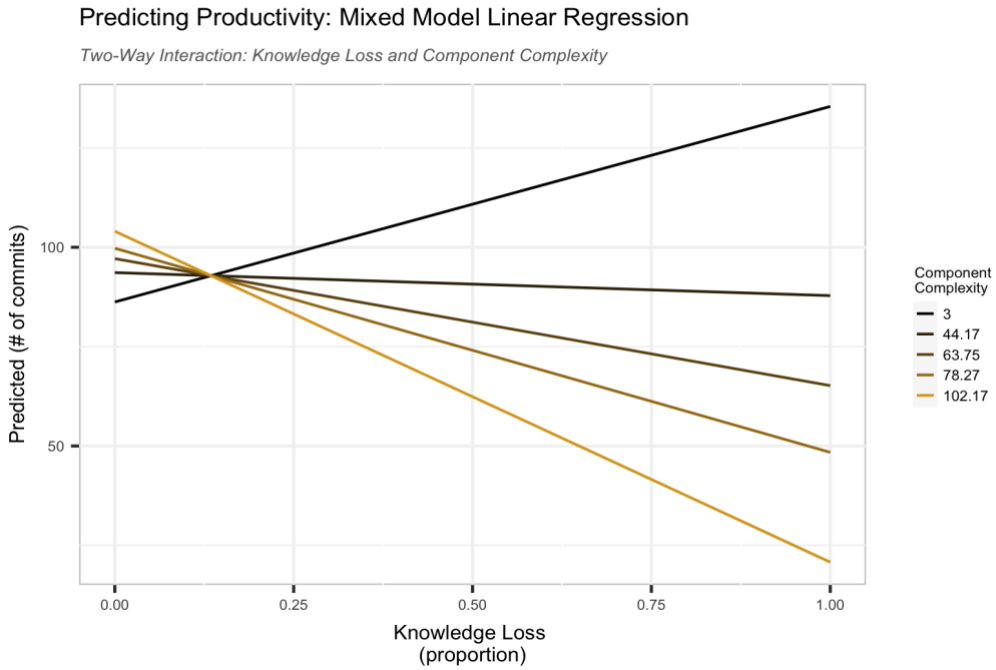


Figure 7 Marginal effects plot for the interaction between knowledge loss and component complexity. The black line represents the highest level of complexity and the yellow line represents the lowest level of complexity.

Table 6 Predicting productivity with mixed model linear regressions.

Interaction	***B*** *(± se)*	***t-value***	R^2	ΔR^2 from null model	χ^2
KL ⨯ Expertise	-124.78 (±73.95)	1.69	.95	.00	2.77
KL ⨯Coord C	-.24 (±0.09)	-2.69**	.95	.00	6.92
KL ⨯Comp C	-1.34 (±0.29)	-4.6****	.96	-.00	18.59
KL ⨯Exp ⨯ Coord C	-.61 (±0.83)	-0.74**	.96	.01	17.93
KL ⨯Exp ⨯ Comp C	2.54 (±2.1)	-1.21****	.97	.02	34.74

Note: * $p < .05$, ** $p < .01$, *** $p < .001$, **** $p < .0001$

The three-way interactions between knowledge loss, project expertise, and task complexity were significantly predictive of productivity (H6a, H6b) and help explain why productivity improved with complexity. In the three-way interaction with coordinative complexity, the negative effect of knowledge loss was reversed as complexity increased (Figure 8). This suggests that, although some abandoned code persisted in the project, the code was continually modified by a steady stream of newcomers. Further, in line with recent research on open source project teams (Jarczyk et al., 2018), a small group of developers managed the task interdependencies introduced by newcomers in the project. We observed a similar effect with respect to the three-way interaction including component complexity. Across levels of component complexity, the negative effect of knowledge loss on productivity was reversed when

the distribution of expertise was highly unequal (Figure 9). Furthermore, the highest levels of productivity were observed in cases of high component complexity. Across these results, our findings indicate that experts in the project were crucial for the maintenance of a complex codebase as they managed the contributions of newcomers who had short-term participation.

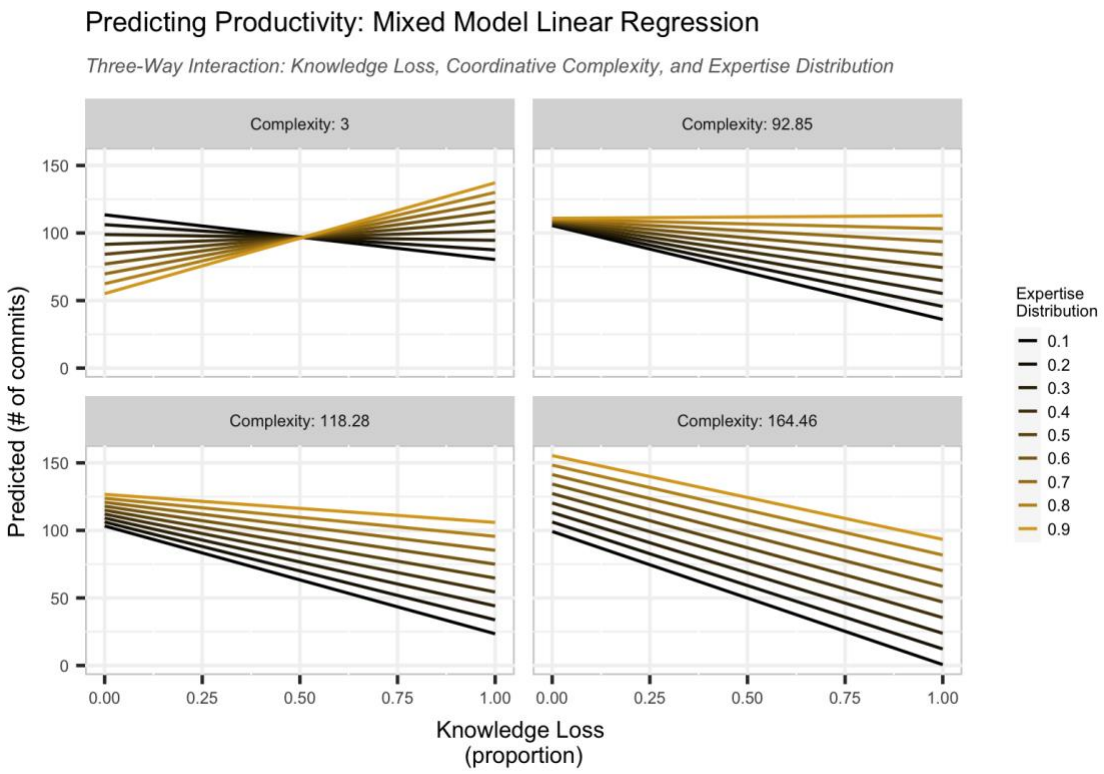


Figure 8 Marginal effects plots for the interaction between knowledge loss, expertise distribution, and coordinative complexity. The top left quadrant shows differences when coordinative complexity is high, and the bottom right quadrant shows differences when coordinative complexity is low.

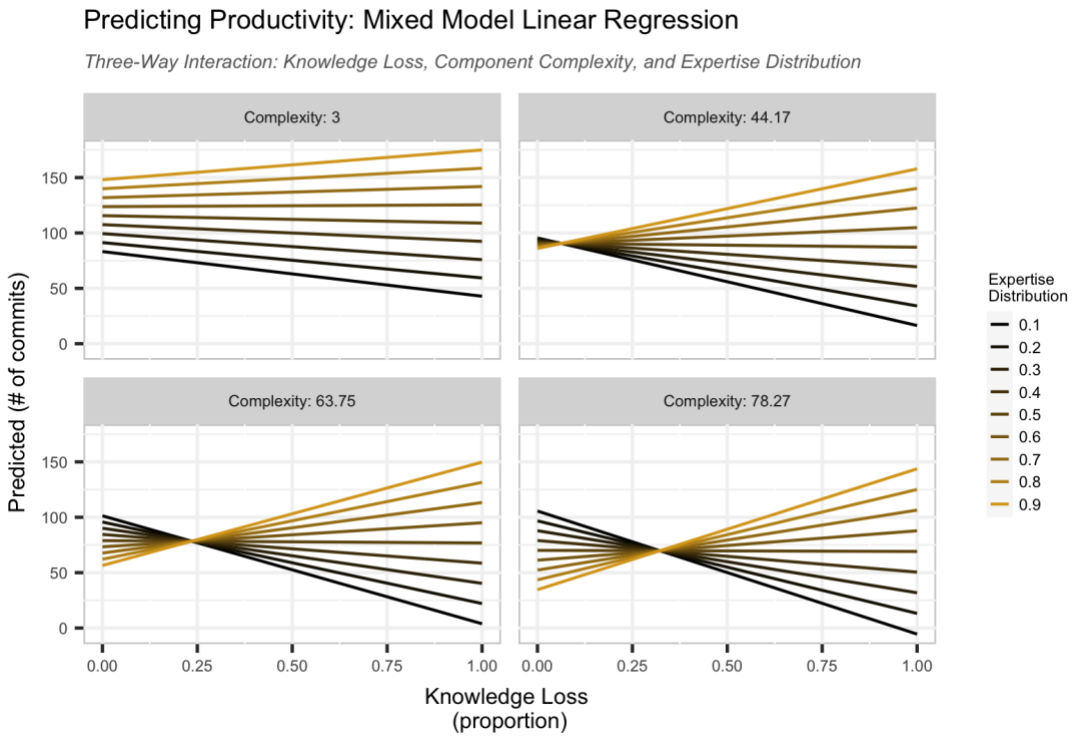


Figure 9 Marginal effects plots for the interaction between knowledge loss, expertise distribution, and component complexity. The top left quadrant shows differences when component complexity is high, and the bottom right quadrant shows differences when component complexity is low.

Discussion

This study finds a nuanced relationship between types of complexity, knowledge loss, and productivity. In line with past research, both knowledge loss and task component complexity had a negative effect on productivity. The two-way interaction model of task coordinative complexity and productivity was not significant. It is possible that this may be due to the lack of precision of the measure used for capturing task interdependencies. Other measures may better reveal the effects of variations in task coordinative complexity as prior research indicates that, for some open source projects, a large number of files are created and modified by only two developers (von Krogh et al., 2003).

The results of the three-way interaction models suggest that, in line with organizational theory, the expertise of remaining members and the positional distribution of turnover determine its effects on team performance (Hausknecht & Holwerda, 2013). Folders that were characterized by high expertise distribution inequality (i.e., few experts and many newcomers) were also those with the highest levels of productivity. This provides evidence that project experts are able to effectively leverage the benefits of turnover to support their own productivity. Additionally, the folders with high expertise distribution inequality may reflect that those particular folders are entry points for newcomers in the project. Building on the work presented here, research can aid in the identification of opportunities for selection and training as well as automated methods for assigning work to apprentice software developers when turnover occurs (Sarma et al., 2016).

Limitations

Although we studied a relatively large organizational structure with hundreds of developers, the generalizability of our results is limited given that our analysis focused on a single, albeit, complex project. Additionally, when quantifying knowledge loss, we did not take into account developer roles. Research on turnover in the FLOSS projects has shown that there are differences in turnover and activity levels between core and periphery contributors (Joblin et al., 2017). Future work can examine other ways to measure knowledge loss, comparing, for example, the differential effects of abandoned files and leavers on collaborative outcomes. This can include an analysis of files and leavers that takes into account their relative importance for the project (e.g., file centrality and developer role, respectively). The observed interaction between expertise, task complexity, and knowledge loss also warrants further investigation both within the GitLab CE project and across other FLOSS projects.

The operationalization of productivity used in this study, while not distinct when compared to other studies of productivity in open source projects, is potentially limited. The number of commits that a developer submits to a repository may have natural variation at the individual for two significant reasons: personal style/work style and organization or organizational/project norms. Interpretations of commit-based measures of productivity are thus limited with respect to assessing improvement or decrements in performance. These types of measures do however provide insights about the effects of varying levels of activity which have been linked to developer's awareness of and participation in open source projects (Fronchetti et al., 2019; McDonald & Goggins, 2013).

Finally, although this study does inform our understanding of how some compositional factors influence outcomes, it did not examine one of the core concepts necessary for examining some of the broader social issues surrounding work in the FLOSS development. That is, it did not consider diversity associated with gender within projects. As such, I next describe a study devised to examine turnover as it relates to gender of contributors in open source projects.

CHAPTER 5. GENDER DIFFERENCES IN TENURE AND TURNOVER

In this chapter, I describe a study of gender differences in tenure and membership change in open source projects. The goal of this research was to (1) investigate the presence of gender differences in tenure and membership change probabilities, and (2) differences in membership change between mixed-gender groups and single gender groups in open source projects. The majority of the material in this chapter was uploaded to *arXiv* as a preprint (Newton & Song, 2022, *Modeling Gender Differences in Membership Change in Open Source Software Projects*) and will be extended with more recent data for submission to a journal.

Research finds that participation among women in FLOSS development is low compared to men (El Asri & Kerzazi, 2019). This issue is more complicated than a lack of women who are interested in and skilled enough to participate in FLOSS development. Recent studies provide evidence that women face particular challenges and biases when attempting to participate (Balali et al., 2018; Imtiaz et al., 2019). In the context of expertise and turnover in FLOSS projects, the data suggest that this occurs even when they are more competent than their male peers (Terrell et al., 2017), and that women are likely to disengage from projects faster than men (Qiu, Nolte, et al., 2019). Owing to these differences in participation, and because of the more general problem of diversity in STEM fields, group diversity has become an increasingly popular topic of discussion across technology-centered fields and also specifically in human-computer interaction (HCI) research (Himmelsbach et al., 2019) and FLOSS research (Aue et al., 2016; Blincoe et al., 2019; Daniel et al., 2013; El Asri & Kerzazi, 2019; Ortu et al., 2016; Terrell et al., 2017; Vasilescu et al., 2015). Germonprez and colleagues (2019) suggest open source development is undergoing a transformation that may reflect "an evolution or a coming crisis in how open source projects are able to encourage skillful, diverse, inclusive global work" (p. 4). We

recognize this transformation as an opportunity to effect change, and aim to contribute to the related body of research through a study characterizing the factors that distinguish projects that are developed by mixed-gender groups and projects that are developed by single gender groups. To do this, we analyzed variable relationships extracted from a longitudinal data set curated by Vasilescu et al. (2015) to promote studies of diversity in FLOSS development. Rather than focus solely on the lack of contributors who are not men in FLOSS projects, we also pay particular attention to those projects that have a mixed-gender contributor base to characterize the membership change dynamics observed within them. Additionally, we extend prior work on diversity and participation by gender groups through an analysis of differences and similarities between project contributors whose gender is inferable via cues in an online space and contributors whose gender is not inferable due to the lack of such cues, or the noisiness of existing cues. Little research on membership change and group composition in FLOSS projects in social coding platforms has explored this topic to date and it leads us to consider the limitations of research on gender in social coding platforms, both in terms of the lines of inquiry that are available using data extracted from social coding platforms and the questionable nature of studying related phenomena through binary categories of gender (Keyes, 2018; Steinhardt et al., 2015).

Recent work takes a nuanced approach to onboarding in FLOSS projects, distinguishing between different types of barriers faced by newcomers and identifying ways that project managers may be able to lower barriers and support participation. Research in this area identifies technical and social challenges for newcomers attempting to join open source projects in addition to the challenges faced by project maintainers to identify and mentor candidates in the newcomer pool for particular tasks (Balali et al., 2018; Gousios et al., 2016; Steinmacher et al., 2019). In a

study conducted by Balali and colleagues, some mentors expressed experiencing "difficulty in creating an inclusive community" (Balali et al., 2018, p. 693), specifically pointing to the issue of correct gender pronoun usage. Furthermore, women contributors shared that they felt "less comfortable with and accepted by their counterparts who are men" (Balali et al., 2018, p. 702). Taken together, these findings reveal a tension experienced by not only cis[11] women, but also contributors who are, for example, trans and/or non-binary. This suggests there remains a set of challenges in defining and identifying contributors on the basis of gender and a gap in understanding with respect to the experiences and inclusion of these contributors in FLOSS projects.

Pointing to the need to consider more tacit factors, research conducted by Fronchetti et al. (2019) and Qiu et al. (2019) characterizes the signals that social coding platform users rely on when choosing a project to contribute to from the options available to them. Fronchetti et al. (2019) constructed a model to investigate the factors that predict developer onboarding in open source projects and found that popularity (measured in stars), the time to merge pull requests, and number of programming languages were the highest ranked predictors; among these factors, popularity was the strongest predictor of developer onboarding. The effect of the number of programming languages in their modeling is consistent with prior theorizing on contribution barriers by von Krogh et al. (2003), in which developer familiarity with programming language contributes to task difficulty and can thus impede participation. Qiu and colleagues (2019) employed a mixed-methods approach to identify the signals used by contributors in choosing

[11] The descriptor cis, or cis gender, indicates that, for an individual there is match between gender identity and sex assigned at birth. A cis woman is thus one who was assigned female at birth and identifies as a woman (Schilt & Westbrook, 2009).

projects, and characterized those signals by investigating their observability in social coding platforms. They found that the FLOSS contributors rely on many signals when selecting projects to join: the level of activity in the project, the popularity of the project, the disposition of issue and pull request handlers, the presence of issue and pull request templates and labels, and the presence of an organized, detailed README.

Last, in interviews with volunteer contributors, Barcomb et al. (2018) found that their participation in open source projects is oftentimes the result of an invitation extended by a known person, providing evidence that social ties and norms play an important role in participation. In sum, although some of the aforementioned studies did not consider diversity as a factor, they add to the repertoire of features that need to be studied if one wants to provide a more complete picture of turnover. Platform features influencing participation range from tacit to explicit, where, for example, signals vary by the degree to which they are observable, or discoverable, with some signals necessitating multiple observations and/or actions on the part of the potential contributors. In other cases, they can involve overt requests to join and may depend on the breadth of connections one has in a community. In total, then, these show the varying factors influencing participation decisions, factors which, in some cases, may differ depending on gender.

Findings across the related areas of research suggest that the low participation of women in FLOSS projects is a complex issue. We contribute to this body of work by reporting on (1) similarities and differences between contributors whose gender can be inferred and contributors whose gender remains obscured or may not otherwise fit within a gender binary, (2) the representativeness of the sample of FLOSS projects with an identifiable mixed-gender contributor base to the population from which it is drawn (i.e., FLOSS projects collaboratively

developed and maintained on GitHub), (3) differences between that sample of projects and a comparable sample of projects that do not have an identifiable mixed-gender contributor base, and (4) a case study of membership change dynamics in a small set of open source projects maintained by a mixed-gender group of contributors. The research we describe in this paper was guided by the following questions:

- **RQ1**. Gender Differences in Platform Tenure: Are there disparities in platform tenure between gender groups?
    - *RQ1.1*: Do women typically have shorter platform tenure compared to men?
        - Based on research examining differences in tenure in software engineering firms (James et al., 2017) and open source projects (Qiu, Nolte, et al., 2019), we expect that women, on average, have shorter platform tenure when compared to men.
    - *RQ1.2*: Is the distribution of platform tenure among project contributors of unidentifiable gender similar to that of either women or men, or is it quantitatively distinct?
- **RQ2**. Representativeness of Mixed-Gender Teams: Are projects maintained by a mixed-gender contributor base quantitatively similar to or distinct from the broader open source project population?
- **RQ3**. Turnover in Mixed-Gender Teams: Does the mixed-gender status of a project team predict differences in turnover?

In addition to examining differences and similarities across the population and samples drawn from it, we conducted a case study of fifteen projects. Our case study was focused on characterizing participation dynamics in projects maintained by a mixed-gender group of

contributors by modeling the probability that contributors, overall and grouped by gender, will join a project, remain present in a project, and/or maintain a period of absence in the project after initial participation. The following questions guided the case study component of our research:

- **RQ4.** Representativeness of Case Study Sample: Is the case study sample quantitatively similar to or distinct from the broader open source project population on GitHub?
- **RQ5**. Project Contributor Transitions: Are contributors more likely to remain in a project after joining?
    - We expect that, overall, contributors have a greater probability of staying with a project than leaving a project after joining (Joblin, Apel, & Mauerer, 2017).
- **RQ6**. Gender Differences in Contributor Transitions: Do transition probabilities differ for contributors on the basis of gender?
    - We expect that men have a higher probability of staying with a project rather than leaving after joining (Qiu, Nolte, et al., 2019).
    - We expect that women have a higher probability of leaving a project rather than staying after joining (Qiu, Nolte, et al., 2019).

Method

For the present study, we used the data set curated and shared by Vasilescu and colleagues (2015). To explore differences in retention and sustained participation on the GitHub platform, we first analyzed platform tenure across three gender groups, with contributors labeled as: woman, man, or unknown. We then analyzed a subset of the data to (1) identify predictors of membership change on the GitHub platform for two samples drawn from the population via mixed effects models, and (2) conduct a case study of membership change dynamics in fifteen projects selected from the mixed-gender project team sample. In this section, we first describe

Vasilescu et al.'s (2015) data set, including the computation and definition of their variables. This is followed by a description of the approach used to sample projects for both the prediction of membership change and characterization of membership change dynamics in our case study sample in addition to the specific techniques used to model turnover and contributor transitions.

Data

The longitudinal data set of 23,493 GitHub projects and 122,014 users was collected and enhanced by Vasilescu et al. (2015) with the goal of enabling studies of diversity in open source projects on GitHub, a platform that provides tools and services to support software development and its management (Begel et al., 2013). Projects were selected from the GHTorrent data dump 1/2/2014 for inclusion by the researchers if the project had at least 2 committers, 10 total commits, and 6 months of history. These selection criteria allowed for the curation of a data set corresponding to active, collaborative open source projects hosted on GitHub, where a project is defined as a base repository and all of its forks, or copies of the base. The data set includes information about both GitHub users (e.g., gender, commit activity, etc.) and projects (e.g., age, main programming language, number of watchers, etc.). To enhance the data set, Vasilescu et al. (2015) applied a username aliasing approach to user data in order to identify project contributors using multiple aliases and merged their information. In addition to this, the researchers used a gender resolution technique to infer the gender of project contributors. This technique leveraged name and location data to probabilistically determine gender and was used to infer the gender of 873,392 users ("32.6% of all users, but 80% of those who disclosed their names"; Vasilescu et al., 2015, p. 515). Vasilescu et al. reported that, for these users, 91% were labeled as men and 9% were labeled as women; the gender of the remaining users was labeled as unknown. Specifically, gender was inferred using the genderComputer tool (Vasilescu, 2014), which uses the

individual's name and location in combination with "transformations, diminutive resolution, and heuristics" to infer gender (Vasilescu et al., 2015, p. 515). User and project data were segmented into quarters (i.e., 3-month periods). 1,136 projects in the data set did not have a main programming language and were excluded from our analyses, resulting in the selection of 22,357 open source projects.

Sample Modeling

We modeled turnover in projects with an identifiable mixed-gender contributor base (i.e., those with at least one woman on the team) and projects that did not have an identifiable mixed-gender contributor base (i.e., those where there were no identifiable women on the team). To do this, we selected a subset of the projects in the data set if, and only if, there was at least one contributor who was identified as a woman in at least one quarter of the project.

From the 23,357 projects in the data set, only 5,539 had at least one woman contributor at one point in time and were labeled as identifiable mixed-gender teams; the remaining 16,818 were labeled as not having an identifiable mixed-gender team. These two groups are unbalanced and, in the interest of comparing teams with similar quantitative characteristics to better understand the relationship between gender and membership change, we applied a sampling technique to select 5,539 projects from the subset of 16,818 projects that did not have identifiable women on the team. We next describe this sampling approach.

Sample Selection

To attain equal samples for the mixed-gender team categorical variable, we constructed a vector for each project and calculated the Euclidean distance between them to select quantitatively similar projects. We created a vector of features for each project that consisted of: number of contributors (12-month period), number of commits (12-month period), number of

forks, number of watchers, and project age. First, we normalized the feature vectors so that each feature's value was in a range between zero and one. The normalization of these features prevented bias toward features with a larger range. Then, for every project labeled as a mixed-gender team, we found the nearest project—where the distance between two projects is the Euclidean distance between their feature vectors—that did not have an identifiable mixed-gender team and downsampled from the 16,818 projects to 5,539 projects. We did not include main programming language in the feature vector because it is a categorical variable that interferes with Euclidean distance calculation.

Sample Coverage

Following the guidance of Nagappan et al. (2013), we evaluated the representativeness of the samples used in our analyses. To do this, we employed their vocabulary and technique for measuring sample coverage, specifically using their algorithm as implemented in statistical computation software R (R Core Team, 2022). This vocabulary and technique were proposed with the aim of improving the generalizability of methods and findings in software engineering research. As a first step, the universe, or population, is defined and projects in that universe are then characterized along one or more dimensions. The set of dimensions is selected on the basis of their relevance to the research topic; these dimensions "define the space of the research topic" within the universe (Nagappan et al., 2013, p. 2). For this research, the *universe*, or population, consists of the open source projects developed by a team (i.e., more than one person) on GitHub at the time of data collection; that is, all of the projects in Vasilescu et al.'s data set. For empirical research on membership change, we contend that, at minimum, the *space* consists of the following dimensions: number of contributors (12-month period), number of commits (12-month period), number of forks, number of watchers, main programming language, and project age. The

selection of these dimensions is based on prior research showing that activity and popularity are related to growth and attraction of newcomers (Fronchetti et al., 2019).

Model Implementation

We constructed mixed effects models to analyze the factors that predict membership change in open source projects. These models were implemented using statistical computation software R (R Core Team, 2022) and, in particular, the *lmer* function in the lme4 package (Bates et al., 2015). The variables included in our modeling are described in Table 7. The ratio of turnover served as the response variable and we included the intercept of a project as a random effect. The *p*-values reported were obtained using the likelihood ratio test with the *anova* function from the stats package. For the models, which included a single fixed effect, the test compared the full model with the particular fixed effect against a null model without the particular fixed effect. All assumptions of the models were checked by examination of residual plots. The primary fixed effect of interest was the mixed-gender status of the team; that is, was the project team labeled as mixed gender or not. Several variables representing different types of tenure were provided in the data set. We only used the GitHub tenure variable as it was most appropriate for our research question concerning participation in the broader FLOSS development taking place in GitHub rather than in a specific project. We modeled the effects of team size, platform tenure disparity, and three indicators of project activity (pull requests, comments, and issues). For team size, the project team consisted of GitHub users who contributed to the project, including committing code, submitting pull requests, contributing to discussion via comments, and reporting issues (Vasilescu, Serebrenik, et al., 2015). Platform tenure disparity was included in our modeling because we contend that it reflects important differences in group composition with respect to status and experience based on activity traces

(Marlow et al., 2013), and has implications for collaboration and turnover (Newton et al., 2019). Tenure disparity was calculated for each project team at each quarter using the Gini coefficient. The Gini coefficient is a calculation initially developed to study income disparity (Dorfman, 1979), and more recently adapted to study disparities in other domains (e.g., in group diversity research; Solanas et al., 2012). The larger a Gini coefficient is, the higher the centralization among a small number of people in a population. Groups with a high value have higher levels of disparity in project tenure (e.g., many relative newcomers and few contributors with long tenure).

Table 7 Project variables included in the mixed effects model. Each of these variables were collected and/or aggregated by Vasilescu et al. (2015) with the exception of mixed-gender team (computed using the has_women variable in their data set) and platform tenure disparity (computed using the github_tenure variable).

Implementation	Variable	Description
Fixed Effects	*Mixed-Gender Team*	Binary value representing the presence of women in the contributor group across project lifetime
	Team Size	Number of contributors (committers, pull request submitters, commenters) in a given quarter
	Tenure Disparity	Calculated using the Gini coefficient, a measure of disparity [0,1], where higher values indicate greater disparity in contributors' tenure
	Pull Requests	Number of pull requests in a given quarter
	Comments	Number of comments in a given quarter
	Issues	Number of issues in a given quarter
Random Effects	*Project*	Each project is assigned its own intercept to account for baseline differences in projects
Dependent Variable	*Turnover Ratio*	Fraction of the team in a given quarter that is different with respect to previous quarter

Case Study

In this section, we describe an approach to model the dynamics of membership change in a small set of projects selected from the mixed-gender team sample. Nagappan et al. (2013) developed a greedy algorithm for the selection of projects that maximizes the coverage of a sample. We used their algorithm to select fifteen projects from the mixed-gender projects sample and use their sample coverage scoring method to evaluate its representativeness.

Contributor Transitions

We adopted an approach used in prior work by Joblin and colleagues (2017) to examine the likelihood of FLOSS contributors joining and leaving a project, and examining differences and similarities on the basis of gender. Joblin et al. used sequential data modeling to evaluate the stability of the contributor base in open source projects. Specifically, they used the discrete state Markov model in which they assigned a state to developers at each time window for a project. We similarly assigned one of two discrete states to project contributors for each quarter of the project lifetime (Figure 10). In each quarter, a project contributor is classified as either absent or present; a contributor was assigned an absent value for each quarter prior to the quarter that they first joined the project and for each quarter after they left the project. For this classification, we used four variables computed by Vasilescu and colleagues: team, which lists the user IDs of contributors who were considered members of the project team; left, the list of users who left the project in a given quarter; joined, the list of users who joined a project in a given quarter; and stayed, the list of users who remained in the project in a given quarter. We thus represented project contributor state transitions as a transition matrix.

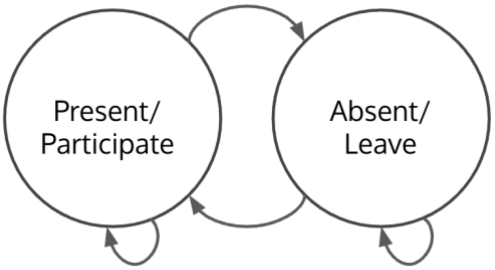


Figure 10 In each quarter, a project contributor is classified as either absent or present, and has some probability of transitioning to a different state or remaining in the same state in the next quarter.

Results

Gender Differences in Platform Tenure

Consistent with findings in research on software engineering organizations (James et al., 2017), in Vasilescu et al.'s data set, we observed that contributors identified as women tend to have shorter tenure in FLOSS projects when compared to contributors identified as men (Table 8). This observation complements work by Qiu et al. (2019) showing that women were more likely to disengage from the platform sooner than men. Furthermore, we observed that contributors of unknown gender similarly have shorter tenure when compared to contributors who were identified as men. Distribution plots show that, compared to both contributors identified as women and of unknown gender, men are more heavily represented across levels of platform tenure (Figure 11). These plots also show that the distributions of platform tenure for women and contributors of unknown gender are nearly identical; both distributions have right skew. The y-axis for each of these plots is different due to the difference in total contributor counts for each gender group.

Table 8 User count and GitHub tenure (in days) by gender.

Gender	User Count (%)	Median GitHub Tenure
Women	5,284	502
Men	57,103	549
Unknown	14813	506

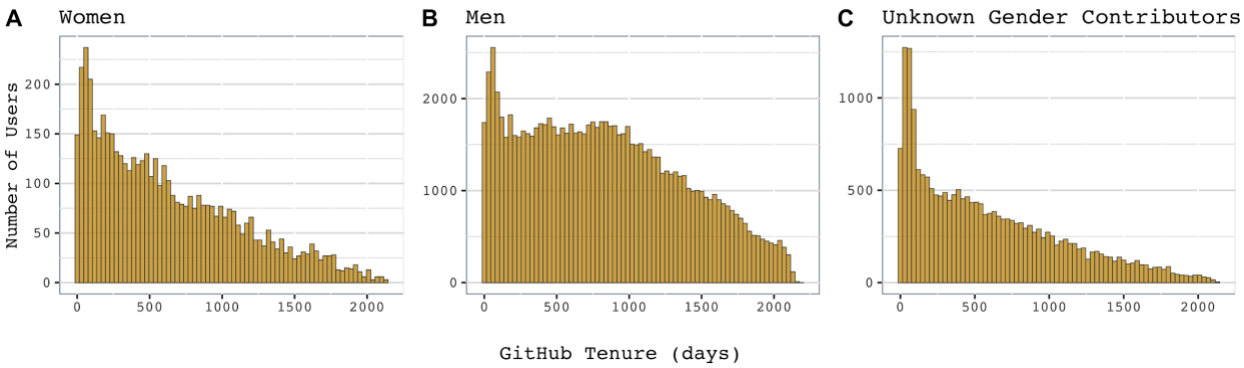


Figure 11 The distribution of platform tenure for GitHub users in Vasilescu et al.'s data set. Most identified women and unknown gender contributors have relatively short tenure on the platform.

To statistically assess differences between these groups, Kruskal-Wallis tests, non-parametric equivalent of a one-way analysis of variance, were applied to users' platform tenure data (Table 9). Women and unknown gender contributors had a lower median tenure compared to men in the platform, and there was an overall significant difference between groups $p < .0000$. There was a significant difference between women and men, and men and contributors of unknown gender. The difference between women and contributors of unknown gender was, however, non-significant. These results suggest that few women joined the GitHub platform at its inception and/or some of the women who joined early on did not remain on the platform for

very long. Again, contributors of unknown gender in the sample exhibit a similar pattern of engagement which may be evidence of other similarities between these groups.

Table 9 Results of Kruskal-Wallis tests. GitHub platform tenure is counted in days; the difference in medians is thus counted in number of days for the two groups being compared.

Groups	**Difference**	***H*-value**	***p*-value**
Women and Men	47	122.80	.0000
Women and Unknown Gender Contributors	4	3.11	.08
Men and Unknown Gender Contributors	43	3198.10	.0000

Table 10 Summary statistics for two groups in sample for mixed effects modeling. Median values are identical for both groups as a result of smart sampling technique.

	Means		Medians	
	has_woman = TRUE All Projects	**has_woman = FALSE** Sample	**has_woman = TRUE** All Projects	**has_woman = FALSE** Sample
Contributors (12 mos)	19.23	9.05	7	7
Commits (12 mos)	18,762.70	1064.54	360	360
Forks	25.76	10.58	5	5
Watchers	52.57	24.70	5	5
Project Age	16.93	17.11	18	18

Model Results

Table 11 provides a summary of the models, including model estimates and standard error (se), *t*-values, and R^2 values. Although each model was statistically significant, the models

with mixed-gender team status and platform tenure disparity as terms have the largest estimates, and the former has the largest R^2 value.

Table 11 Summary of results for mixed effects models. These models include the specified term as fixed effect and project as random effect. Pull requests is abbreviated to PRs.

Model Term	**Estimate (±*se*)**	***t*-value**	R^2	ΔR^2 **from null model**
Gender Composition	-0.020 (±0.003)	-5.17****	.27	-.00
GitHub Tenure	0.160 (±0.007)	23.40****	.25	-.02
Team Size	-0.001 (±0.000)	-19.59****	.25	-.02
Activity: PRs	0.000 (±0.000)	-10.61****	.26	-.01
Activity: Comments	0.000 (±0.000)	-20.32****	.25	-.02
Activity: Issues	-0.002 (±0.000)	-24.35****	.23	-.04

Note: * $p < .05$, ** $p < .01$, *** $p < .001$, **** $p < .0001$

Case Study

In this section, we report the coverage score for the sample used in our case study in addition to contributors' transition probabilities in a project. The purpose of the coverage score calculation is to describe the representativeness of the sample relative to the study population. Results for the fifteen projects that were selected are provided in Table 12. Overall, the sample has low coverage, or low representativeness. This is expected given the small size of the sample relative to the population size. Additionally, the low score appears to be driven primarily by the main language and project age dimensions.

Table 12 Sample coverage scores for case study sample.

	Case Study Sample
Overall Score	0.044
Main Language	0.691
Contributors (12 mos)	0.997
Commits (12 mos)	0.956
Forks	0.997
Watchers	0.991
Project Age	0.644

Contributor Transitions

Median transition probabilities for users in the case study sample are visualized across all projects and for gender groups in Figure 12 and probabilities for each project selected for the case study are given in Table 13. Overall, we observe some stability in these projects: contributors had a higher probability of remaining in the project after joining (9 out of 15 projects), although there are some exceptions. When looking at the transition probabilities for the woman or group of women in a project, it was more likely that a contributor will leave a project after joining (i.e., probability of present → absent is greater than probability of present → present). This is in contrast to the transition probabilities for men, who were more likely to remain in a project after joining (i.e., probability of present → present is greater than probability of present → absent). This suggests the group of men in mixed-gender teams is more stable than the group of women in mixed-gender teams. However, relevant to the difference between the overall trend and the trend among women, all of the projects included in the case study had teams that were made up primarily of men (see Team Size column in Table 13). Even in the two

projects with smaller teams, projects 5 and 11, there were few women, 1 and 2 respectively. These results in addition to the finding that women's contributions tend to be centralized to a smaller number of projects than those of men (Imtiaz et al., 2019) help explain the shorter tenure of women in FLOSS projects and the GitHub platform in general.

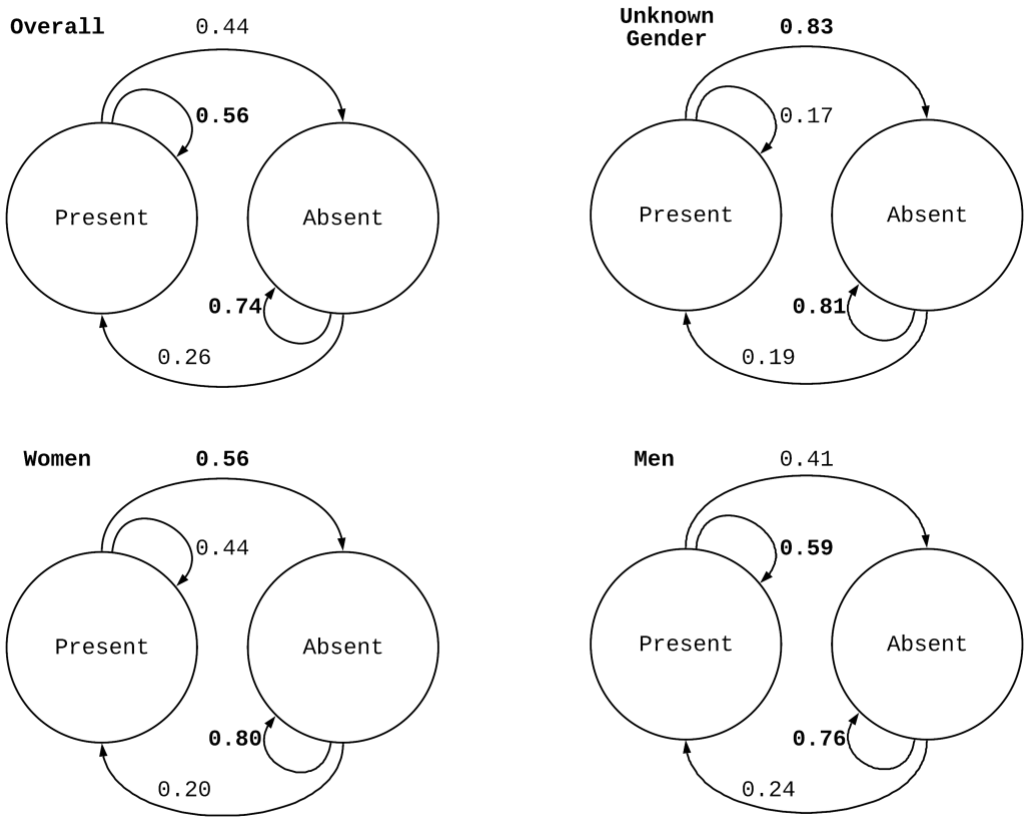


Figure 12 Participation dynamics in GitHub: transition probabilities averaged across all project contributors in case study sample and by gender group.

Table 13 Transition probabilities, across all contributors in a project and contributors grouped by gender in a project. Contributors to projects 3, 4, and 5 were all identified as women or men (i.e., no unknown gender contributors).

PID	**Team Size** *(# women)*	**Language**	**Group**		*Present*	*Absent*
1	145 (6)	Ruby	All	*Present*	0.40	**0.60**
				Absent	0.16	**0.84**
			Women	*Present*	0.00	**1.00**
				Absent	0.13	**0.88**
			Men	*Present*	0.42	**0.58**
				Absent	0.17	**0.83**
			None	*Present*	0.33	**0.67**
				Absent	0.15	**0.85**
2	38 (2)	Python	All	*Present*	0.46	**0.54**
				Absent	0.28	**0.72**
			Women	*Present*	**1.00**	0.00
				Absent	**0.50**	**0.50**
			Men	*Present*	**0.54**	0.46
				Absent	**0.76**	0.24
			None	*Present*	0.00	**1.00**
				Absent	0.33	**0.67**
3	45 (2)	JavaScript	All	*Present*	0.43	**0.57**

PID	**Team Size** *(# women)*	**Language**	**Group**		*Present*	*Absent*
				Absent	0.22	**0.78**
			Women	*Present*	0.00	**1.00**
				Absent	0.20	**0.80**
			Men	*Present*	0.44	**0.56**
				Absent	0.22	**0.78**
			None	*Present*	-	-
				Absent	-	-
4	47 (5)	Java	All	*Present*	**0.69**	0.31
				Absent	0.43	**0.57**
			Women	*Present*	0.33	**0.67**
				Absent	**0.67**	0.33
			Men	*Present*	**0.73**	0.27
				Absent	0.40	**0.60**
			None	*Present*	-	-
				Absent	-	-
5	16 (1)	Python	All	*Present*	**0.85**	0.15
				Absent	0.12	**0.88**
			Women	*Present*	0.00	**1.00**
				Absent	0.20	**0.80**
			Men	*Present*	**0.87**	0.13
				Absent	0.11	**0.89**
			None	*Present*	-	-
				Absent	-	-
6	35 (3)	JavaScript	All	*Present*	**0.54**	0.46

PID	**Team Size** *(# women)*	**Language**	**Group**		*Present*	*Absent*
				Absent	0.26	**0.74**
			Women	*Present*	0.00	**1.00**
				Absent	0.17	**0.83**
			Men	*Present*	**0.59**	0.41
				Absent	0.29	**0.71**
			None	*Present*	0.00	**1.00**
				Absent	0.19	**0.81**
7	25 (4)	JavaScript	All	*Present*	**0.64**	0.36
				Absent	0.26	**0.74**
			Women	*Present*	**0.50**	**0.50**
				Absent	0.33	**0.67**
			Men	*Present*	**0.65**	0.35
				Absent	0.24	**0.76**
			None	*Present*	-	-
				Absent	**0.75**	0.25
8	112 (9)	C	All	*Present*	**0.69**	0.31
				Absent	0.40	**0.60**
			Women	*Present*	**0.68**	0.32
				Absent	0.39	**0.61**
			Men	*Present*	**0.69**	0.31
				Absent	0.42	**0.58**
			None	*Present*	**0.63**	0.38
				Absent	0.19	**0.81**
9	35 (2)	Python	All	*Present*	**0.59**	0.41

PID	**Team Size** *(# women)*	**Language**	**Group**		*Present*	*Absent*
				Absent	0.35	**0.65**
			Women	*Present*	**1.00**	-
				Absent	-	**1.00**
			Men	*Present*	**0.64**	0.36
				Absent	0.33	**0.67**
			None	*Present*	0.33	**0.67**
				Absent	**1.00**	0.00
10	25 (2)	Python	All	*Present*	0.40	**0.60**
				Absent	0.30	**0.70**
			Women	*Present*	1.00	-
				Absent	-	1.00
			Men	*Present*	0.45	**0.55**
				Absent	0.38	**0.63**
			None	*Present*	1.00	-
				Absent	-	1.00
11	15 (2)	Ruby	All	*Present*	0.40	**0.60**
				Absent	0.28	**0.72**
			Women	*Present*	**1.00**	0.00
				Absent	0.50	0.50
			Men	*Present*	0.33	**0.67**
				Absent	0.24	**0.76**
			None	*Present*	-	-
				Absent	-	-
12	47 (1)	Ruby	All	*Present*	**0.58**	0.42

PID	Team Size *(# women)*	Language	Group		*Present*	*Absent*
				Absent	0.23	**0.77**
			Women	*Present*	0.00	**1.00**
				Absent	0.20	**0.80**
			Men	*Present*	**0.60**	0.40
				Absent	0.23	**0.77**
			None	*Present*	0.00	**1.00**
				Absent	0.20	**0.80**
13	39 (1)	JavaScript	All	*Present*	**0.56**	0.44
				Absent	0.39	**0.61**
			Women	*Present*	-	-
				Absent	0.33	**0.67**
			Men	*Present*	**0.65**	0.35
				Absent	0.43	**0.57**
			None	*Present*	0.00	**1.00**
				Absent	0.29	**0.71**
14	98 (5)	C++	All	*Present*	0.48	**0.52**
				Absent	0.16	**0.84**
			Women	*Present*	0.38	**0.63**
				Absent	0.06	**0.94**
			Men	*Present*	0.48	**0.52**
				Absent	0.17	**0.83**
			None	*Present*	0.00	**1.00**
				Absent	0.06	**0.94**
15	129 (3)	C++	All	*Present*	**0.58**	0.42

PID	Team Size *(# women)*	Language	Group		*Present*	*Absent*
				Absent	0.12	**0.88**
			Women	*Present*	**0.50**	**0.50**
				Absent	0.05	**0.95**
			Men	*Present*	**0.59**	0.41
				Absent	0.13	**0.87**
			None	*Present*	0.38	**0.63**
				Absent	0.05	**0.95**

Discussion

In this study we set out to study factors that are related to the low participation of women in open source projects. We analyzed projects and their compositional features to better understand similarities and differences between contributors based upon gender. We found that, in addition to factors already linked to turnover in open source projects, group composition in terms of gender predicted levels of turnover with mixed-gender teams exhibiting lower levels of turnover. The significant effect of gender composition on turnover complements findings from a study by Blincoe et al. (2019) in which developers reported more negative behaviors and experiences in male-only teams. In other words, the unpleasant nature of such experiences in homogeneous teams produces increases in turnover. We analyzed projects and their composition to better understand similarities and differences between contributors based upon gender. Our findings add to the body of evidence showing that participation dynamics vary by gender. This research also extends prior work by revealing similarities in tenure between GitHub users who are women and those whose gender is unknown. Through our case study, we demonstrated the utility of transition-probability models for examining group differences in contexts where

participation is ephemeral yet recurrent. The application of this approach in our study showed that women are not only more likely to leave a project compared to men, but that they are also less likely to return to a project after a period of absence. Furthermore, the participation of GitHub users of unknown gender in a project is very limited—they contribute in a small period of time and their departure from the project is more permanent.

In conducting this study, we considered the ways that the design of the GitHub platform shapes studies of gender in open source projects. First, the platform does not request gender information upon account creation and, as a result, likely diminishes the salience of gender and alters its effects on social interactions (McNicol, 2013). This reflects a particular design choice and, embedded within it, assumptions about the types of, and ways that, individuals use the platform that have significant implications, both for the phenomena of interest and the ways that the phenomena can be studied. Although we do not make a claim to the appropriateness of this design choice, we do contend that it is one that warrants further attention. Second, this design choice results in the need for researchers interested in analyzing differences and similarities between genders on the platform to infer gender based on a set of, arguably, noisy cues, including, for example, displayed name, location, and, as some other researchers have used, picture and email information to locate accounts on other platforms that belong to the same user which can result in misgendering (Keyes, 2018). Related to this, and how gender information is requested or otherwise inferred, we reflected on the observation that research on gender in open source projects has largely relied on binary categories of gender to account for differences and similarities in participation. Platform design, in combination with the gender binary frame applied to study gender differences, has resulted in an analysis of genders as relatively homogeneous groups and the potential exclusion of FLOSS contributors of other marginalized

genders, including in the research presented here. This presents an opportunity for future research to better understand and characterize within group differences and different types of diversity in open source projects (Himmelsbach et al., 2019). Researchers can employ a mixed-methods approach to provide a more comprehensive analysis of the gender binary issue and gender categories.

In the work presented here, we suggest GitHub users whose gender was not identifiable by Vasilescu et al. were potentially not men, but also not necessarily women. While there were quantitative similarities between users who were identified as women and unknown gender users, additional research is needed to accurately characterize the latter group. The group of users whose gender was not resolved may be made up individuals who generally exhibit lower levels of engagement in the platform for different reasons and thus do not provide the type of information that is used to infer gender by researcher. The reasons for lower levels of engagement may be tied to gender (e.g., the bias and harassment, Nafus, 2012), or other factors that are not necessarily specific to gender but reflect other sociodemographic dimensions (e.g., limited time or financial resources for ongoing participation).

Limitations

The data set used for the analysis presented here was collected by Vasilescu et al. from a GHTorrent data dump in 2014. It does not capture changes in the platform user base and participation in open source projects that have occurred since then. This limits the interpretation of results. For example, it is possible that there has been a reduction in the gender imbalance observed in GitHub. Recent studies of gender differences and diversity suggest, however, that this is likely not the case (e.g., El Asri & Kerzazi, 2019; Qiu et al., 2019). We therefore expect the application of our analyses to a more recent data set would likely produce similar findings.

Further, this study provided insights as to ‘what’ was happening in these projects. Follow-on research could additionally include methods that uncover ‘why’ these differences arise (see, for example, Balali et al., 2018; Blincoe et al., 2018; Terrell et al., 2017). The interpretation of results is also limited given the difficulty associated with accurately labeling gender for all users. It is possible that there was some incorrect classification of users. This type of erroneous classification is however unlikely in the case study sample given the selected projects had relatively large team sizes and that relatively few women participate in open source projects on GitHub.

Conclusions

In sum, in this paper, we reported the results of a quantitative study of gender differences in membership change in open source projects hosted on GitHub, a social coding platform. Replicating and extending prior work on gender differences in tenure, we observed that women had shorter tenure when compared to men, and that unknown gender contributors similarly had shorter tenure in the platform. Our application of Nagappan et al.'s (2013) sample coverage scoring approach provides evidence that the open source projects maintained by mixed-gender teams are not quantitatively distinct from those that are maintained by teams who do not have an identifiable mixed-gender composition. The results presented here also suggest the need to further explore the relationship between gender composition, status, and expertise in open source projects, as we found that these factors, in addition to activity levels, predicted turnover. Future work in this area may reveal important insights about the experiences of women as they relate to power dynamics and membership change in open source projects. Lastly, we observe that, although the overall project team is relatively stable, there are gender differences in the movement in and out of open source projects.

Like other research on membership change in FLOSS development, our study focused on modeling factors internal to open source projects (i.e., group composition). Less work has examined how factors exogenous to a specific project may alter turnover within them. Significant changes to the platform and decisions made by corporate stakeholders may have the potential to instigate an exodus of users from the platform. For example, some GitHub users expressed dissatisfaction with the decision to sell the platform to Microsoft and claimed they would leave the platform following the acquisition (Warren, 2018). GitHub has also been criticized by both its users and employees for their continued relationship with the U.S. Department of Homeland Security's Immigration and Customs Enforcement (ICE) (Chan, 2019; Ghaffary, 2020). These actions taken by GitHub decision makers likely influence the choices of some developers to participate in open source development on the platform. Here, an understanding of within group differences could illuminate the causes of diversity issues in open source along other sociodemographic dimensions. Such an understanding however necessitates an analysis of how race, class and status intersect with gender in studies of membership change. Information about these types of changes and decisions are public and can be leveraged to investigate the effects these events on membership change in open source projects.

Summary

The studies described in these chapters (4 and 5) contribute to the growing body of work showing that there exists a complex relationship between diversity, gender, and membership change in open source projects. In study 1, statistical analysis revealed that group composition in terms of platform experience and task complexity moderate the effects of turnover on productivity. In study 2, our analyses confirmed that women not only have shorter tenure than their peers who are men, but also that women have a greater probability of leaving a project after

joining whereas men have a greater probability of staying after joining. In both of these studies, modeling efforts were focused on individual and project factors related to membership change in FLOSS development. I contend that, while providing some insights, this research should be extended to construct a multilevel model that additionally accounts for differences within and across FLOSS ecosystems and the social computing technologies that enable work within them. Finally, as part of this ecosystem analysis, I add that the degree of corporate involvement could also factor into these compositional dynamics within FLOSS development. I turn next to a description of my proposed work, designed to build on these prior studies and help the field better understand the relationships between features in the complex sociotechnical ecosystems of FLOSS.

CHAPTER 6. DIVERSITY AND CORPORATIONS IN FLOSS ECOSYSTEMS

In earlier chapters, I reviewed my prior research examining turnover in FLOSS projects and associated gender differences. In those studies, I modeled group composition, task complexity, and turnover as predictors of productivity (Chapter 4) and examined gender differences in participation and membership change (Chapter 5). In this chapter, I describe an extension of this work through an examination of how social diversity in projects, the broader ecosystems in which they are embedded, and the FLOSS products associated with those projects, are related to membership change. My overarching goal is to begin the development of a multilevel model of membership change dynamics in FLOSS ecosystems. I now turn to a discussion of these factors followed with an integrated set of research questions, and a multi-method approach for addressing these questions.

As detailed in Chapter 3, prior research on membership change in FLOSS development has paid significant attention to the predictive utility of individual and/or project features, providing a rich understanding of turnover in projects and among participants. Less research has attended to broader social and organizational factors related to differences in participation in FLOSS projects. The question is if, and how, more exogenous factors influence what happens within projects and across their respective ecosystems. An analysis of the relationship between these phenomena thus necessitates a systems approach to understand relevant factors and identify leverage points, or places in the system where interventions can be applied to effect change (see Table 14; Meadows, 1999; Abson et al., 2017). Such an analysis can help address important questions about, for example, how increased corporate involvement in FLOSS development is altering the evolution of these ecosystems in ways transforming observable processes (Germonprez et al., 2019).

Table 14 Meadows' (1999) leverage points, adapted from Abson et al., (2017).

Level	Places to Intervene	System Characteristics
Shallow	Parameters (such as subsidies, taxes, standards)	*Parameters*: "relatively mechanistic characteristics typically targeted by policy makers" (p. 32)
	Size of buffer stocks, relative to their flows	
	Structure of material stocks and flows	
	Length of delays, relative to the rate of system change	*Feedbacks*: "interactions between elements within a system of interest that drive internal dynamics" (p. 32).
	Strength of negative feedback loops	
	Gain around driving positive feedback loops	
Deep	Structure of information flows (access to information)	*Design*: "social structures and institution that manage feedbacks and parameters" (p. 32).
	Rules of the system	
	Power to add, change, or self-organize system structure	
	Goals of the system	*Intent*: "underpinning values, goals, world views of actors that shape emergent direction to which a system is oriented" (p. 32).
	Mindset/paradigm out of which the system arises	
	Power to transcend paradigms	

Corporations initially rejected the free and open source paradigm, but, since the mid-2000s, they have maintained a relationship with FLOSS communities, altering development processes and licensing along the way (Fitzgerald, 2006). Collaborations between organizations and communities have resulted in a shift away from the development of standard platforms to the improvement of their usability and features (Germonprez et al., 2013), but it is unclear if and to what extent corporations have constrained or facilitated positive outcomes associated with FLOSS projects (e.g., the production of more equitable technologies; Newton, 2020). This is a significant gap in understanding that should be addressed, especially in consideration of technology corporations' (e.g., Microsoft, Google, and Amazon) effects on public goods and

communities. Noble (2020) describes how such corporations harm public goods through economic policy:

> "These companies also play a key role in the decimation of shared knowledge and education as a public good. While we seek remedies based on evidence and truth that can shape policies in the collective best interest, Big Tech is implicated in displacing high-quality knowledge institutions—newsrooms, libraries, schools and universities—by destabilizing funding through tax evasion, actively eroding the public goods we need to flourish. The "Silicon Six"—Amazon, Apple, Facebook, Google, Microsoft and Netflix—collectively avoided paying $155.3 billion in taxes between 2010 and 2019."

In addition to this, the technologies produced by "Big Tech" corporations are also associated with a host of biases that disproportionately affect marginalized and/or underserved populations. Population bias, and to a lesser extent algorithmic bias, in the collection and analysis of social media data is well-documented in the literature (Ruths & Pfeffer, 2014). For example, in a study comparing algorithm performance for urban and rural populations on Twitter, Johnson et al. (2017) found that, even when controlling for population bias in the training data, algorithmic design led to poorer performance for rural communities compared to urban communities.

The possibility of technology corporations transforming FLOSS development in a way that affords increased diversity in projects, while optimistic, is seemingly disconnected with the reality that organizations involved in contemporary technological development, across industry and academia, lack inclusivity (Clark, 2016; Waxman, 2017). Germonprez and colleagues (2019) contend that open source projects are in the midst of a transformation as the presence and participation of corporations in development efforts continues to increase. They note that such a transformation has far reaching effects "as the change in social construction is accelerating" (p.

4), influencing labor in open source projects and in distributed work domains in general. These statements thus call attention to changes in labor in a knowledge work domain that drives technological development and innovation. The technologies produced in FLOSS development and related software engineering work domains are a common aspect of daily life for many people. Because of this, changes in labor in FLOSS projects have the potential to create cascading effects in software ecosystems and beyond. Germonprez et al. thus prompt researchers to consider the shifting hierarchies in FLOSS development and the potential effects of the aforementioned transformation on diversity and inclusivity in such projects. Diversity and inclusivity are topical issues for FLOSS development that have implications for both participation and production in technological development. Research in this area has demonstrated that individuals who differ along social dimensions (e.g., gender) also differ along other dimensions that influence problem-solving processes and task outcomes in software development (e.g., information processing style; Gralha et al., 2019). The effects of varying group composition in software development are evident in the many cases that software has been unusable for, or otherwise failed, some populations (Hilderbrand et al., 2020).

Germonprez et al. (2019) also speculate that social computing technologies may hold a space of particular importance in addressing problems emerging in this form of complex system—they are a leverage point in the system that can be targeted by researchers. Social computing technologies are a type of collaboration infrastructure that is created and designed by organizations, and thus might be classified as belonging to the "design" realm of system characteristics in Meadows' conceptualization of leverage points. Noble (2020) however would likely argue that altering these technologies, or creating new versions of them is insufficient to effect meaningful change. Instead, focus should be drawn to deeper leverage points, to system

"intent", or "the underpinning values, goals, and world view of actors that shape the emergent direction to which a system is oriented" (Abson et al., 2017, p. 32). In FLOSS development, within and outside of social computing technologies, the primary actors in the system are the individuals who comprise associated communities and technological corporations. Based on this, I evaluate organizations and group composition as leverage points in FLOSS ecosystems. Specifically, I propose an investigation of the links between membership change, social diversity, and corporate involvement in FLOSS projects (Figure 13) hosted on platforms like GitHub. Past research establishes the relatively ephemeral nature of participation for a significant set of contributors in open source projects and, because it alters group composition, it is expected that membership change has implications for diversity in FLOSS ecosystems.

In sum, Chapters 4 and 5 described my studies of turnover in open source projects in which I specifically modeled group composition, task complexity, and turnover as predictors of productivity (study 1) and examined gender differences in participation and membership change (study 2). In this chapter, I extend this work with the aim to address some of the aforementioned gaps through an analysis of the relationship between participation and social diversity in projects, the broader ecosystems in which they are embedded. The goal of this research is to construct a multilevel model of membership change dynamics in FLOSS ecosystems. This has theoretical and practical implications, contributing to theory building on open collaboration in addition to providing insights for FLOSS communities and the organizations they partner with to develop technologies.

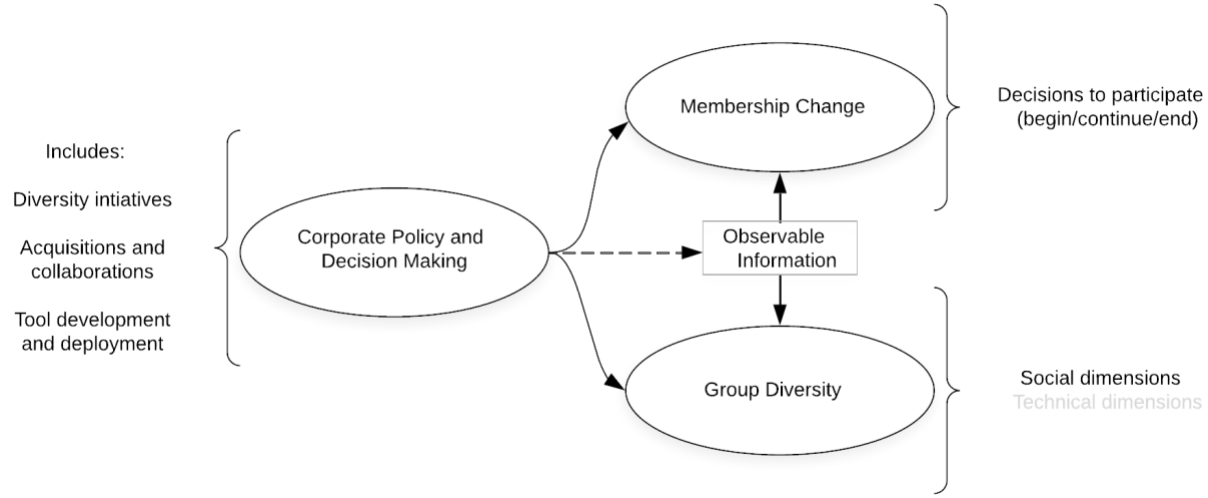


Figure 13 Diagram illustrating the hypothesized relationship between membership change, diversity, and corporate involvement in FLOSS ecosystems.

Approach

This research is a multi-study, mixed-method effort that was conducted in multiple phases. In the first and second phase, study participants and projects were identified and recruited. In the third phase, perceptions and experiences of contributors to FLOSS projects were elicited through surveys. In the fourth phase, project data was used to construct models of FLOSS ecosystems, with a specific focus on the relationship between membership change, social diversity, and level of corporate involvement. In this section, I define the primary concepts of interest: FLOSS ecosystems, social diversity, and corporate involvement. Operationalization of these concepts is fully detailed in the Method section.

Social Diversity

My prior work considered only gender diversity in the context of FLOSS development. I broaden this to additionally examine other forms of social diversity. Diversity will be evaluated

along a set of social dimensions and I specifically draw on the conceptualization of diversity described by Himmelsbach and colleagues (2019), which weaves together social, historical, and technological factors:

> "diversity dimensions cannot merely be conceptualized as simple characteristics of users. Anchored in the tradition of critical diversity studies (e.g. [15], [41], [55], [107] [109]), we see diversity dimensions both as descriptive and evocative [17]. We define diversity as (i) social differences with attributed social meaning [41], that (ii) refer to social inequality [41] and are embedded in a historically evolved social and structural context [17] [109] and (iii) influence how people live [100] and experience technology. Such a critical concept of diversity can serve as a "multidimensional tool for exploring [...] majority-minority relations" [17] and to understand user experience." (p. 2).

These researchers conducted a review of diversity research in the field of human-computer interaction over the course of a decade and observed a significant increase in research examining the dimensions of age, ethnicity and culture, gender and sex, and race. In addition to this set of dimensions, they observed a recent increase in papers that studied mental abilities, religion, and sexual orientation. Studies of social diversity in FLOSS projects have provided insight on between-group differences related to gender and nationality inferred via geographic location (Aue et al., 2016; Vasilescu, Serebrenik, et al., 2015).

Group diversity can be quantified in several ways, taking into account the amount of variety, separation, and/or disparity that is present in the group (Harrison & Klein, 2007), and a number of mathematical techniques are described in the literature (e.g., coefficient of variation, Blau index, Gini coefficient). The specific calculation used for diversity is determined on the basis of the variable type. Approaches like the Gini coefficient and coefficient of variation are

used to quantify diversity for numerical variables whereas approaches like the Blau index are used to quantify diversity for categorical variables. This research contributes to the body of work examining social diversity in FLOSS projects across multiple dimensions and in relation to corporate engagement. Contributors who volunteer to participate in the survey component of this research were asked to share demographic information. Data extracted from platform APIs were supplemented using tools developed by Vasilescu et al. (2015) to label gender and nationality.

Corporatization

Different types of corporation-community relationships have been studied in the context of FLOSS development. The findings suggest that a corporation's management style can introduce problems with FLOSS communities. In early research, Dahlander and Magnusson (2005) proposed three types of firm-community relationships that fall along a continuum of community benefits: symbiotic (firm gains-community gains), commensalistic (firm gains-community indifferent), and parasitic (firm gains-community loses). The symbiotic approach, in which the community benefits from the relationship, is associated with several managerial challenges:

> "(1) respecting the norms and values of the FLOSS communities; (2) using licenses in a suitable way; (3) attracting developers and users; (4) dealing with the resource consumption involved in community development; (5) aligning different interests about the nature of work; and (6) resolving ambiguity about control and ownership" (Dahlander & Magnusson, 2005, p. 491).

While the development and maintenance of a symbiotic relationship introduces these challenges for the corporation, this approach affords the highest levels of influence on the community. In contrast, the commensalistic approach affords low influence and the parasitic approach affords

no influence, and corporations' use of communal resources is judged negatively by the community, contributing to the deterioration of the corporation-community relationship.

In the FLOSS literature, research continues to examine the relationship between communities and corporations that collaborate with them (Germonprez et al., 2017), and the observed and potential effects of the increased presence of corporations in FLOSS development (Germonprez et al., 2019). In such works, the levels at which corporations engage with FLOSS development have been elaborated. First, corporate engagement occurs at the level of economic and business markets to promote open source and proprietary products (corporate-communal markets). Second, corporate engagement exists at the intersection of ideology, knowledge production, and innovation. Here, corporations influence the development and direction of communal resources. Third, corporate engagement is "built on social and material rules within an open source community"; this determines the strategies for the distribution of communal resources (Germonprez et al., 2017, p. 66). Nonetheless, in this literature, there has been no described approach for quantifying the degree or level of corporate involvement in a project ecosystem. It is therefore necessary to develop and evaluate methods for quantifying corporatization in FLOSS ecosystems. As a starting point, and as later described, I qualitatively label the level of corporate involvement for each project and further quantify this at the ecosystem level.

To conclude, this research builds on my prior work examining turnover and knowledge loss in FLOSS projects and related gender differences. That work identified relationships between project features (e.g., task complexity levels) and contributor factors (e.g., group composition), and turnover. I additionally examined how gender differences in participation are related to membership change. I extend this work by both broadening the lens to an ecosystems

level, but also by enriching FLOSS project data, through the use of a survey which covers a set of relevant topics: experiences, attitudes, and values regarding diversity and corporate engagement in FLOSS development; awareness of diversity and corporate engagement in FLOSS development; and information gathering activities related to diversity and corporate engagement in FLOSS development. With this, I study how social diversity in projects and their ecosystems is associated with participation and corporate engagement. Together, this will be used to develop a multilevel perspective on participation dynamics and outcomes in FLOSS ecosystems. This research is guided by the following research questions:

- *Survey Research: Characterizing Contributor Perceptions of Diversity and Corporate Involvement in FLOSS Projects*
    - SR RQ1 What are contributors' perceptions of diversity in FLOSS projects?
        - SR Q1.1 Awareness: Are contributors aware of the diversity of the group in a FLOSS project?
            - SR Q1.1a What types of diversity are contributors aware of?
            - SR Q1.1b What dimensions of social diversity are most salient?
            - SR Q1.1c Does awareness differ between demographic groups?
        - SR Q1.2 Values and Attitudes: Is diversity important to FLOSS contributors?
            - SR Q1.2a What types of diversity are important to contributors?
            - SR Q1.2b What attitudes and beliefs are used to justify the importance/unimportance of diversity in FLOSS projects?
    - SR RQ2. What are contributors' perceptions of corporate involvement in FLOSS projects?

        - SR Q2.1 Experiences: What aspects of contributors' experiences are altered by corporate involvement in FLOSS projects?
        - SR Q2.2 Motives: Do contributor reasons for participation differ in a project with/without corporate involvement?
        - SR Q2.3 Decision Making: Does corporate involvement influence contributors' intent to sustain participation?
        - SR Q2.4 Awareness: Are contributors aware of the actions of corporations subsidizing and participating in FLOSS development?
        - SR Q2.5 Influence: Are contributors responsive to the actions of corporations subsidizing and participating in FLOSS development? (e.g., in the case of Microsoft and its subsidiary GitHub's contract with U.S. Immigration and Customs Enforcement)?
- *Ecosystem Research: Comparing and Contrasting FLOSS Projects on the Basis of Corporate Involvement*
    - ER Q1. Participation: Does contributor participation differ for projects with and without corporate involvement?
        - ER Q1.1 Does contributor tenure differ for projects with and without corporate involvement?
        - ER Q1.2 Does sustained participation differ for projects with and without corporate involvement?
    - ER Q2. Social Diversity: Are differences in social diversity in an ecosystem associated with corporate involvement?

        - ER Q2.1 Which projects have the highest and lowest levels of social diversity in each ecosystem?
    - ER Q3. Project Characteristics: What are the quantitative characteristics of projects that have/lack corporate involvement?

Method

This study contains two components: survey research and a FLOSS ecosystem analysis. The survey was distributed to FLOSS contributors and the ecosystem analysis was applied to GitHub repository data. This research received Exempt status from the University of Central Florida's Institutional Review Board (IRB). In this section, I describe the survey research approach I employed in which a survey was first distributed broadly and then to a targeted group. I further provide a summary of the approach used to analyze survey participant responses. I then describe the steps taken to collect user and behavioral trace data for FLOSS repositories on GitHub. This includes a description of the data source, data wrangling and aggregation, sampling procedure, data enhancement, and statistical tests applied to sample data.

Survey Research

A survey was created and administered via Qualtrics to elicit contributor perceptions of diversity and corporate involvement in FLOSS projects. The contents of the survey are in Appendix A. This survey instrument contains Likert-type, multiple selection, and open-ended questions. The inclusion criterion for this study was contribution to FLOSS development on GitHub. Prior to completing the survey, potential participants were asked to specify the level of corporate involvement of FLOSS projects to which they contributed or had contributed to in the past (Table 15). They then had the opportunity to review an explanation of research approved by the UCF IRB and were asked to provide confirmation of their decision to voluntarily participate

in the study. In an effort to maintain data integrity, a set of measures was implemented in Qualtrics before survey distribution. This included the use of captcha, timestamps, open-ended questions, a honeypot question, and "ballot-stuffing" prevention. The survey was administered in two phases.

Table 15 Representation of FLOSS project types in survey samples.

Level of Corporate Involvement	***Sample***	
	Social Media	**Committer Emails**
Corporations were not/are not involved in any of the projects	3	4
Corporations were not/are not involved in most of the projects	11	11
Corporations were/are involved in approximately half of the projects	14	2
Corporations were/are involved in most of the projects	6	6
Corporations were/are involved in all of the projects	0	7
Total	34	30

In the first phase, a general call for participants was distributed via social media, specifically on my Twitter account[12] and in the GitHub subreddit[13], and a set of listservs, including the NIH Science of Team Science listserv[14] and Listserv For Scientists[15]. It is possible that recipients of the survey invitation distributed it through other channels. The first phase distribution resulted in over 400 responses, but as is common with surveys distributed via social media (Xu et al., 2022), a large portion of this data was unusable due to (1) evidence of automated activity, (2) incomprehensible, incoherent, or plagiarized text responses, and (3)

[12] https://twitter.com/oIivia_n
[13] https://www.reddit.com/r/github/
[14] scitslist@list.nih.gov
[15] scientists@sciencelistserv.org

incomplete responses. Therefore, only 34 complete responses were deemed usable and kept for analysis.

In the second phase, an invitation to participate was sent directly to contributors to the GitHub projects selected as the sample for the ecosystem analysis component of this research. A total of 596 contributor emails were extracted from commits and pull requests associated with the aforementioned GitHub projects. Of these emails, 10 were returned as undeliverable. This distribution resulted in 57 responses (10% response rate); however, only 30 of these responses were complete and kept for analysis. Complete and usable data were therefore collected from a total of 64 survey participants across two samples.

Analysis of Survey Responses

To analyze participant responses to Likert-type and multiple selection questions, aggregation, statistical, and visualization techniques were applied to the data using R (R Core Team, 2022) and in particular with functions provided in the tidyverse (Wickham et al., 2019) and ggplot2 (Wickham, 2016) packages. Demographic information provided by participants was summarized using descriptive statistics and bar plots showing the distribution of responses as relative proportions for each sample. Responses to questions related to study concepts were visualized using the likert package (Bryer & Speerschneider, 2022).

Reflexive thematic analysis (RTA), a contemporary approach for thematic analysis described by Braun and Clark (2019), was applied to participant responses to open-ended survey questions to characterize participants' experiences and dispositions regarding study concepts. Rather than applying predefined themes to the data or framing themes as emerging from the data, themes in RTA are the result of active interpretation on the part of the researcher to derive meaning from the data and determine the codes and themes which provide understanding related

to a study's research questions. The goal of RTA is not to consensus or reliability and it is appropriate for a single coder to complete the analysis. This approach is thus flexible and acknowledges the researcher's "active role in knowledge production" (Bryne, 2022, p. 1393). Six phases comprise RTA: (1) familiarization with the data; (2) generating initial codes; (3) generating themes; (4) reviewing potential themes; (5) definition and naming themes; and (6) producing the report.

Following the guidance offered in the literature, I address underlying theoretical assumptions associated with my application of RTA. I adopted a constructionist epistemology. The recurrence of codes is not the most significant criteria I applied to the data and I instead weighed meaning and meaningfulness as the "central criteria" in coding (Byrne, 2022, p. 1395). Furthermore, my analysis had a primarily experiential orientation, reflecting the feelings and experiences of participants, although in considering implications and future work some shift towards a critical orientation was necessary to interpret participant responses in relation to the social context in which they are embedded. Lastly, I conducted an inductive (i.e., data-driven) analysis in which codes were descriptive and based on explicit meaning, termed semantic coding.

FLOSS Ecosystem Analysis

To examine the complex relationship within and between ecosystems, a complementary set of techniques was applied. Generally, research on free/libre and open source projects relies on statistical and computational modeling techniques to explore and confirm variable relationships. In some of this research, linear mixed models have been used to analyze, for example, the effect of both individual-level and project- or group-level variables on outcomes like productivity and software quality (e.g., Vasilescu, Posnett, et al., 2015). Other studies have used logistic

regression and survival analysis to analyze differences in support quality operationalized as issue closure rate (Jarczyk et al., 2018) and, more recently, to model gender differences in duration of participation (Qiu, Nolte, et al., 2019). Lastly, network analysis has been applied to both project and contributor data in order to characterize them (Aljemabi & Wang, 2018), in combination with Markov modeling to investigate various group and organizational phenomena (e.g., developer coordination; Joblin et al., 2017; Newton & Song, 2022), and to build recommender systems (e.g., for developer onboarding; Liu et al., 2018).

In the present research a combination of these approaches was used, with network methods used to model participation dynamics in ecosystems and regression models used to analyze variable relationships of interest. Network analysis has been applied in studies of knowledge work to better understand dynamics and outcomes in collaboration (e.g., modeling complex systems, examining network evolution, and/or analyzing the relationship between network structure and individual behavior; Schoder et al., 2014). Multi-layered networks were constructed based on project data to model and visualize FLOSS ecosystems. Project ecosystems and participation are represented in these networks via two types of links (project-project and individual-project). These links were used to model membership change in FLOSS ecosystems over time in addition to the compositional and ecosystem mechanisms associated with differences in diversity and FLOSS projects. Data extracted from online sources, including platform APIs and API mirrors (e.g., GHTorrent; Gousios & Spinellis, 2012) were used to construct the models and statistical analysis were applied to measures of network structure in addition to variables representing social diversity and corporate involvement. The steps taken to collect data and carry out these analyses are described next.

Data Sources

Data were collected and aggregated from the GHTorrent mysql-2019-06-01 data dump for sample selection which included data for over 125 million repos. Additional data were extracted from the GitHub REST API using the PyGitHub library[16] and GrimoireLab's Perceval module[17]. These data were supplemented with information from project websites and Wiki pages, when available.

Population Definition and Sample Selection

As a first step for sample selection, the study population was defined. Based on the purpose of the study, only repositories that were devoted to software development and were active were classified as belonging to the study sample. Specifically, GitHub repos were identified as belonging to the study population if they met the following criteria:

1. *Software development focus*: a specified programming language
2. *Continuity of development activity*: at least 5 years old
3. *Sufficient amount of development activity*: at least 50 commits
4. *Sufficient number of contributors*: at least 5 members

These criteria and thresholds were established to ensure that there would be enough data to model study phenomena over time. Additionally, they align with criteria and thresholds used in other studies of FLOSS projects (e.g., Daniel et al., 2018). Based on this, the study population is made up of 2,572 repositories—approximately 0.002% of all repos in GHTorrent data dump.

The next step for sample selection was the aggregation of project data across a set of dimensions. This set of dimensions was selected on the basis of their relevance to the research

[16] https://pypi.org/project/PyGithub/
[17] https://github.com/chaoss/grimoirelab-perceval

topic; these dimensions "define the space of the research topic" within the universe (Nagappan et al., 2013, p. 2). For empirical research on participation in FLOSS projects on GitHub, the space consists of the following dimensions: number of contributors (12-month period), number of commits (12-month period), number of forks, number of watchers, main programming language, and project age. The selection of these dimensions is based on prior research showing that activity and popularity are related to project growth and attraction of newcomers (Fronchetti et al., 2019). The last step was the calculation of the proportion of committer turnover for projects in the study population. The proportion of turnover was calculated based on two year-long periods: P_1= 2017-06-01: 2018-05-31; P_2 = 2018-06-01:2019-06-01. Distributions for these dimensions are provided in Appendix B. These aggregated data then served as input for a sample coverage algorithm (Nagappan et al., 2013) to select 20 GitHub projects from the study population. Two of these projects were removed from the sample: one is a mirror of a software repo hosted outside of GitHub and the other had been deleted from GitHub by the time of data collection. The remaining 18 projects were kept for statistical analysis.

Bot Detection and Removal

This research is focused on the observable behavior of humans in FLOSS projects. As a result, bots were identified and removed from the data set. As in prior research on GitHub (e.g., Golzadeh et al., 2021; Newton et al., 2022), bots were identified based on their username, information in their GitHub profile bio, and/or GitHub's contributor type label (user or bot). All usernames and bios that contained the pattern 'bot' in addition to 'CI', 'automated'/'automation', or 'machine' were extracted in an iterative process (e.g., all accounts selected in 'bot' pattern matching were removed for closer inspection before beginning 'CI' pattern matching). Then, all extracted accounts were manually inspected to remove false positives (e.g., in the case where a

human included bot or robot in their username). Examples of usernames for automated accounts that did not contain bot include “gitter-badger” and “meeseeksmachine”. Through this process, a total of 30 accounts were labeled automated and removed from the data set. Some of these automated accounts were found across multiple projects (e.g., dependabot) whereas others were specific to a project.

Contributor Data Enhancement

Location. Usernames belonging to contributors in the sample projects and their corresponding ecosystem were used to extract location information provided in GitHub profiles. Location information was prepared for analysis using the tmaptools (Tennekes, 2018) and tidygeocoder (Cambon et al., 2021) packages in R. Specifically, functions in these packages were used to define coordinates for every user who included location information in their profile which were then used to assign country and continent-level labels to each of those users.

Identity Merging. Previous research on GitHub projects has produced a set of best practices for studying participation and contribution in FLOSS development. This includes handling the possibility that a single user may have multiple accounts or emails which are associated with their activity on the platform. As a result, it is important to identify those users with multiple accounts and merge them before analyzing platform activity. Identity merging was applied using an approach based on heuristics described in Vasilescu et al. (2015).

Gender Resolution. As in prior research on gender in GitHub, the genderComputer tool[18] was used to infer the gender of project contributors based on the name and location data they provided in their online profile. User information extracted for the sample projects was

[18] https://github.com/tue-mdse/genderComputer

additionally manually inspected to evaluate the tool's output. Manual inspection of GitHub and social media profiles revealed that errors in inferring gender were due to the presence of nicknames (e.g., Oli, Dani, and Juaky) and gender-neutral names (e.g., Taylor and Casey) in the data. This manual inspection was constrained by my language capabilities: I speak English and Spanish fluently, and am thus familiar with nicknames and gender-neutral names in these two languages. However, less than 15 instances of these classification errors were discovered, and researchers who have developed and refined these gender resolution techniques report precision of upwards of 93% (Vasilescu et al., 2015).

Modeling Ecosystems

Ecosystems as a unifying concept have transcended a number of social science disciplines in attempts to capture some form of complex multi-level set of interdependencies. In computational social sciences and related fields, this definition varies depending on the context. The term ecosystem has been applied in studies of FLOSS projects in a somewhat inconsistent manner, although generally it is used to describe a grouping or cluster of projects that are linked to each other on some basis. In a subset of this research, all of the FLOSS projects hosted on a social coding platform comprise an ecosystem (e.g., projects in GitHub or BitBucket; Casalnuovo et al., 2015). Other research specifies an ecosystem along the lines of the software's application domain (e.g., data analytics; Geiger et al., 2018), the main programming language in which the project is written (Constantinou & Mens, 2017), or goal similarity and the presence of technical dependencies between projects (Blincoe et al., 2015; Sarma et al., 2016). Because this dissertation is focused on factors crossing the nature of the work processes and those doing the work in these areas, an approach based on technical dependencies was used to identify FLOSS ecosystems. Specifically, the reference coupling method proposed by Blincoe et al. (2015) was

used for ecosystem generation. This method is based on mentions of dependencies between projects in comment data extracted from platform APIs. The cross-reference patterns reported in Blincoe et al. (2015) were used: User/Project#Num (e.g. rails/rails#123) or User/Project@SHA.

A set of multi-layered networks was constructed, one for each project ecosystem. The network was structured as follows:

- Nodes: projects, individual contributors
- Edges: contributor-contributor (joint work experience), contributor-project (contribution history)

Network variables were analyzed to identify differences in structure at the ecosystem level. Participation was examined at the project and ecosystem level. The specific approaches used varied according to their associated research question. To evaluate the quantitative differences between ecosystems with high and low corporate involvement, network variables were computed and analyzed to identify differences in structure at the ecosystem level, specifically evaluating differences in average degree centrality, edge density, and transitivity (i.e., global clustering coefficient). To compare tenure between project ecosystem types, mixed effects models were created with contributor project tenure as the response and variables representing corporatization as predictors. Contributor tenure was calculated at the quarter-level (3 months) in line with prior work on tenure and turnover in FLOSS projects. To compare sustained participation between project ecosystem types, A survival analysis was applied to contributor data from each ecosystem. Survival analysis allows for the analysis of differences in time until an event occurs which in this case is disengagement from a project ecosystem. This approach is appropriate for censored data (Clark et al., 2003) as the event of interest only occurs in some of the study sample. For example, a contributor may have left the project at a date after data was collected for

this research. To evaluate differences in social diversity as they related to project ecosystem type, statistical models were created with variables representing diversity as the response and variables representing corporate involvement. Nonparametric approaches were applied due to the skewed distributions observed in the data. Lastly, social diversity differences between and within project ecosystem types were analyzed with descriptive statistics and distribution plots that enable the characterization and visualization of variables of interest. Variables computed and used in the ecosystem analysis are described in Table 16.

Table 16 Variables used in ecosystem analyses.

Variable	Description	Level of Analysis	Used For
Corporate Involvement	Categorical with three levels: (3) company owned; (2) receives corporate support; (3) no corporate influence	Project, Ecosystem	All RQs
Contributor Tenure	Numerical: time between first and most recent contribution, in quarters (3 months)	Project, Ecosystem	ER Q1
Tenure Disparity	Gini coefficient (numerical): ranges from 0 to 1, where 0 represents equality and 1 represents complete inequality	Project, Ecosystem	ER Q2
Gender Diversity *Country Diversity*	Blau index (numerical): ranges from 0 to 1, where 0 represents a homogenous group and 1 represents a heterogenous group	Project, Ecosystem	ER Q2
Average Degree [0:n-1] *Edge Density [0:1]*	Network metric (numerical): based on graph of projects and contributors in an ecosystem as nodes, and contribution and joint work experience as edges	Ecosystem	ER Q3

Results

Survey of FLOSS Contributors

This research posits questions related to awareness of, and interest in, group diversity in FLOSS projects, and in particular those on GitHub. I first provide a summary of sample demographics to help contextualize survey responses before moving on to describe perceptions of contributors on diversity and corporate involvement in projects. Because the samples vary on some dimensions that might influence interpretations, they are described separately. In this way, the summary also serves to highlight similarities and differences between the two survey samples. Following the demographic characterization of the study samples, I summarize survey responses related to participants' perceptions of diversity and corporate involvement. As is customary in these kinds of studies (Blincoe & Damian, 2015; Blincoe et al., 2019), the goal is to characterize and describe the survey findings. As the goal is not to find statistically significant differences between groups, the description of results that follows summarizes the data through a reporting of response proportions to Likert-type items and the derivation of themes from free text responses. Plots of Likert-type items present participant responses in the form of stacked bar charts showing the proportional distribution of responses. Responses to multiple selection items are visualized as proportions using horizontal bar charts.

Demographics

Social Media/Listserv Sample. Complete data were aggregated and analyzed from 34 participants. The sample largely self-identified as "male" or a "man" (85%) and the average age was approximately 30 years (Tables 17 and 18). With respect to race and ethnicity, the sample was predominantly White (71%) and of this subset one participant further identified themselves as Hispanic. Five participants identified themselves as Asian and four as Black. Nearly all

participants reported that they reside within North America (91%) and, in particular, the United States (88%). Of the remaining participants, two reside in Europe and one in Asia. This sample is thus relatively homogenous with respect to gender, race/ethnicity, and country of residence.

Relevant to expertise, the majority of participants reported that they work primarily in software development and engineering (34%) or academic and researcher roles (31%); were paid to work in open source development (66%); and had at least three years of experience (65%). Survey participants reported taking on a multitude of roles to fulfill the needs of open source projects. Among these roles, most participants identified themselves as code contributors and integrators, although one participant specialized as a UX design contributor in open source projects.

Committers Sample Demographics. Complete data were aggregated and analyzed from 30 participants. The sample largely self-identified as "male" or a "man" (80%) and the average age was approximately 34 years (Tables 17 and 18). Like the social media sample, most participants identified themselves as White (76%). Among participants who reported their race as White, two identified themselves as Hispanic/Latine, one as Pacific Islander, and one as Arab. Three participants identified themselves as Asian, two as non-White Hispanic, one as non-White Arab, and one as North African. In this sample, Europe was most heavily represented (43%), followed by North America (40%). Four participants reported that they reside in Asia and one participant resides in Oceania. This sample is homogenous with respect to gender and race/ethnicity, but exhibits some heterogeneity when examining country of residence.

The majority of participants reported that they work primarily in software development, engineering, or design (60%) or academic and researcher roles (17%). In contrast with the social media sample, most participants in this sample reported that they were not paid to work in

FLOSS development (68%). Experience in FLOSS development was high in this sample, with 87% reporting having at least three years of experience and 33% reporting that they had more than 10 years of experience. These participants primarily contribute code (90%), document issues (57%), and/or integrate external contributions (37%). However, a small number also reported that they coordinated work (20%) and/or worked on UX/usability (20%).

Both samples were made up of a large portion of participants who attained graduate level degrees (Figure 16) and reported annual incomes above US$70,000 (Figure 17). Together, both samples generally reflect observed demographics described in prior work on FLOSS projects. One exception however is in the social media sample, which was US-dominated—research on FLOSS on GitHub finds that contributors are more globally distributed[19]. With this understanding of the survey sample makeup in place, I turn next to an analysis of survey responses to characterize the perceptions of diversity and corporate involvement in FLOSS projects.

Table 17 Participant counts by reported gender for each sample.

Gender	$N_{Social\ Media}$	$N_{Committers}$
Man	30	24
Woman	3	1
Nonbinary	0	1
Prefer not to say	2	4

[19] The description of FLOSS development as globally distributed while not wholly inaccurate does not capture differences in representation by region, and as they occur between the Global North and Global South. Research on FLOSS development tends to focus on projects and contributors who are distributed across the Global North.

Table 18 Descriptive statistics for participant age in each sample.

Age	$N_{Social\ Media}$	$N_{Committers}$
Minimum	21	20
Maximum	44	52
Mean (SD)	29.51 (±5.81)	34.13 (±8.95)
Median	28	31.50

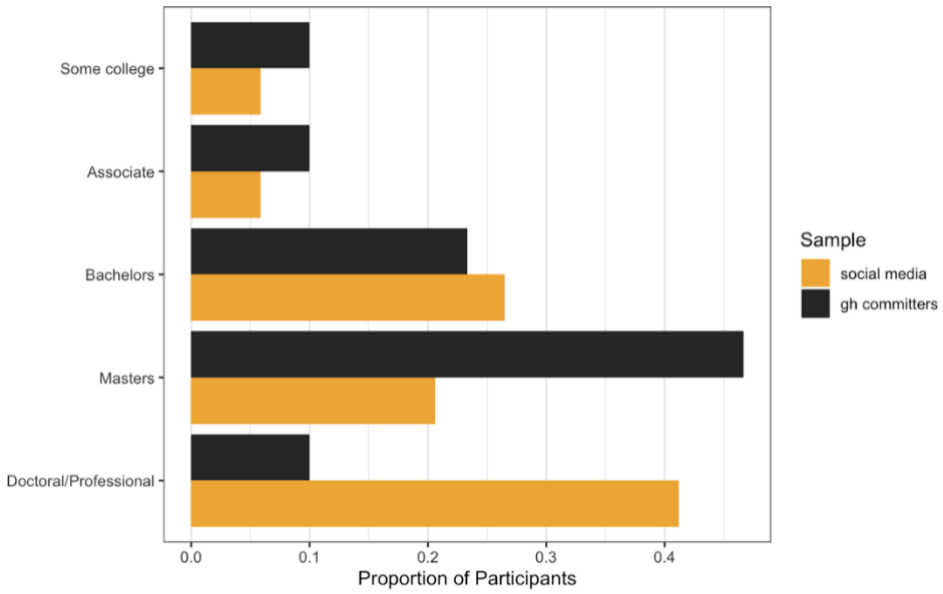


Figure 14 Education levels of survey participants survey.

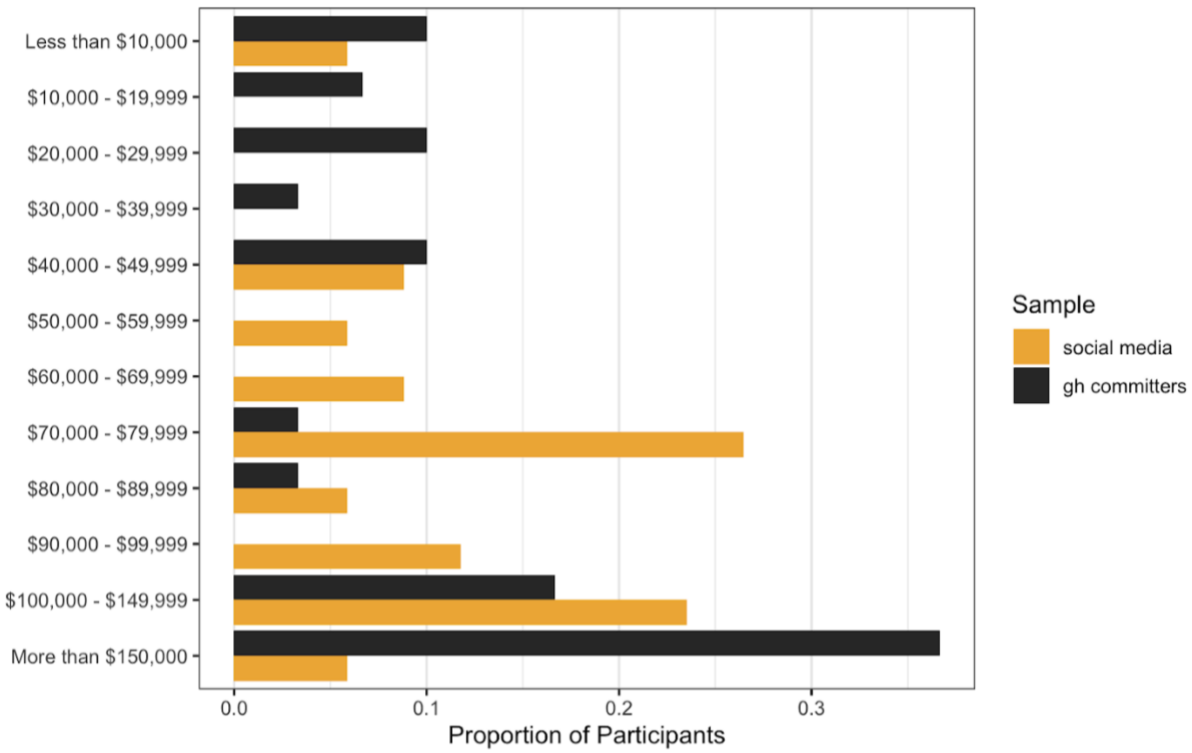


Figure 15 Income levels of survey participants survey.

What Are Contributors' Perceptions Of Diversity In FLOSS Projects?

The overarching goal of the responses collected and described in this section is to characterize contributors' perceptions of diversity, specifically as they relate to awareness, information gathering, values, and contribution decisions. Contributors' perceptions of diversity vary greatly between the two samples when examining responses related to awareness and contribution decisions. Some overlap between survey samples emerges in around the salience of different types of diversity and the importance attributed to diversity.

SR Q.1.1 Are contributors aware of the diversity of the group in a FLOSS project?

Between the two samples, participants varied in the level of awareness of diversity in projects they reported (Figure 18). Participants in the social media/listserv sample were likely to

have some level of awareness whereas participants in the committers sample were more unlikely to be aware of diversity in projects. However, the sample distributions are nearly identical for the ease or difficulty reported for assessing diversity in projects (Figure 19). More participants reported either experiencing some difficulty in assessing diversity than participants who reported an easeful experience in assessing diversity in projects. Over a quarter of participant in both samples did not find the assessment of diversity easy or difficult. Participants were also asked to specify the information sources which enabled awareness of diversity (Table 19) and the majority in both samples reported they relied on user profiles and comments.

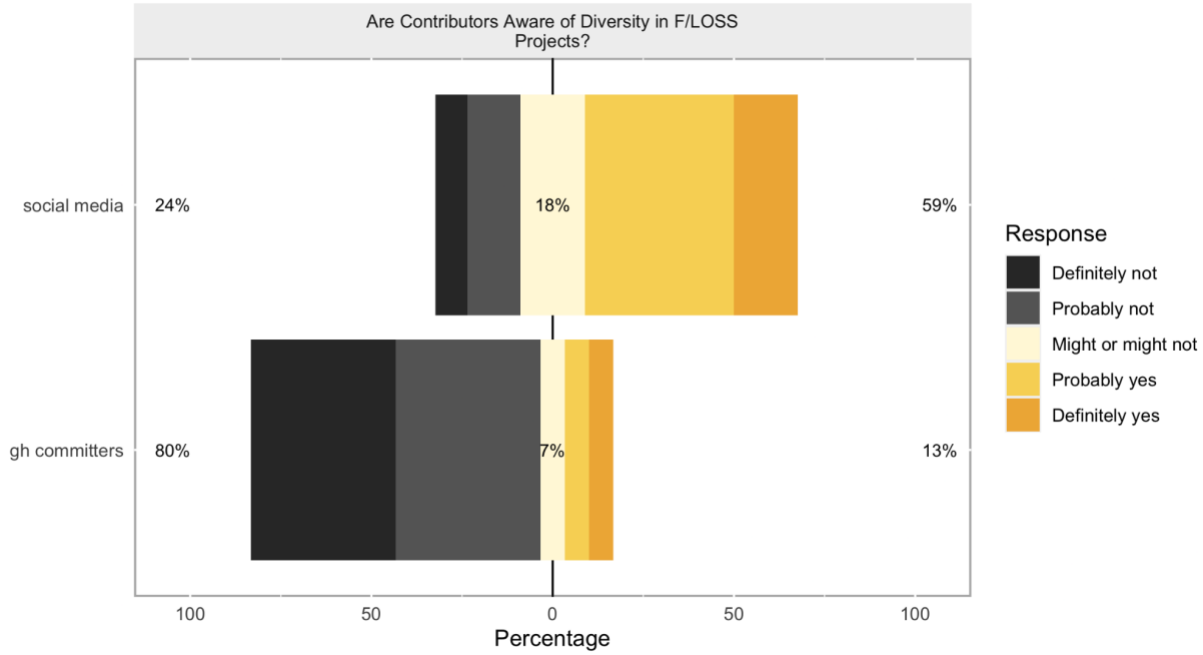


Figure 16 Survey participants' awareness of diversity in FLOSS projects.

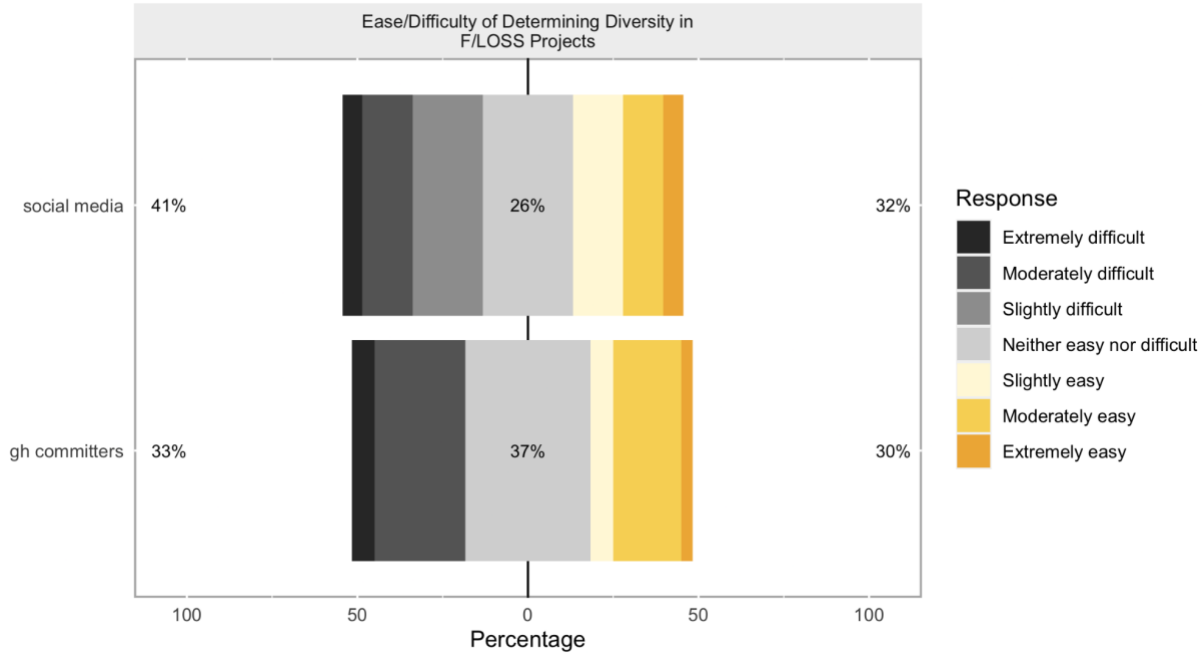


Figure 17 Survey participants' perception of the level of difficulty associated with determining the diversity of FLOSS projects.

Table 19 Sources of information used by contributors to determine group diversity in FLOSS projects.

Information Source	**Sample**
Name	GH Committers
User profiles (e.g., pictures, bio) • Within platform (GitHub) • Outside platform (LinkedIn, Twitter)	Social Media, GH Committers
Comments, discussion • Within platform (GitHub issues, commits) • Outside platform (Slack)	Social Media, GH Committers
Previous joint work experience	Social Media
"Significant" members	Social Media
Community representation at conferences	GH Committers
Organization video calls (e.g., on Zoom)	GH Committers

SR Q1.1a,b What types of diversity are contributors aware of in FLOSS projects? What dimensions of social diversity are most salient?

Although the committers sample responded that they were unlikely to be aware of diversity in the projects to which they contribute, they responded that they were aware of a multitude of types of diversity, including gender, race/ethnicity, and expertise (Figure 20). This apparent contradiction suggests that participants vary in the extent to which they hold explicit and implicit awareness of the social and technical attributes of other contributors. The committers sample responses suggest that they attend to both social and technical characteristics of a project's contributor base. Furthermore, unlike the social media sample, the committers sample responses extended the set of social and cultural dimensions listed in this survey question, specifically adding sexual orientation and language as types of diversity they aware of in FLOSS projects.

Approximately half of the participants in both samples report being aware of gender diversity, and in the committers sample, over 80% of participants report being aware of diversity in expertise levels. While all participants in the social media reported being aware of some type of diversity, just over 20% of the participants in the committers sample reported that they were not aware of any type of diversity in FLOSS projects. The two samples also differed with regard to their awareness of diversity in age and education level, with the social media sample reporting higher levels of awareness.

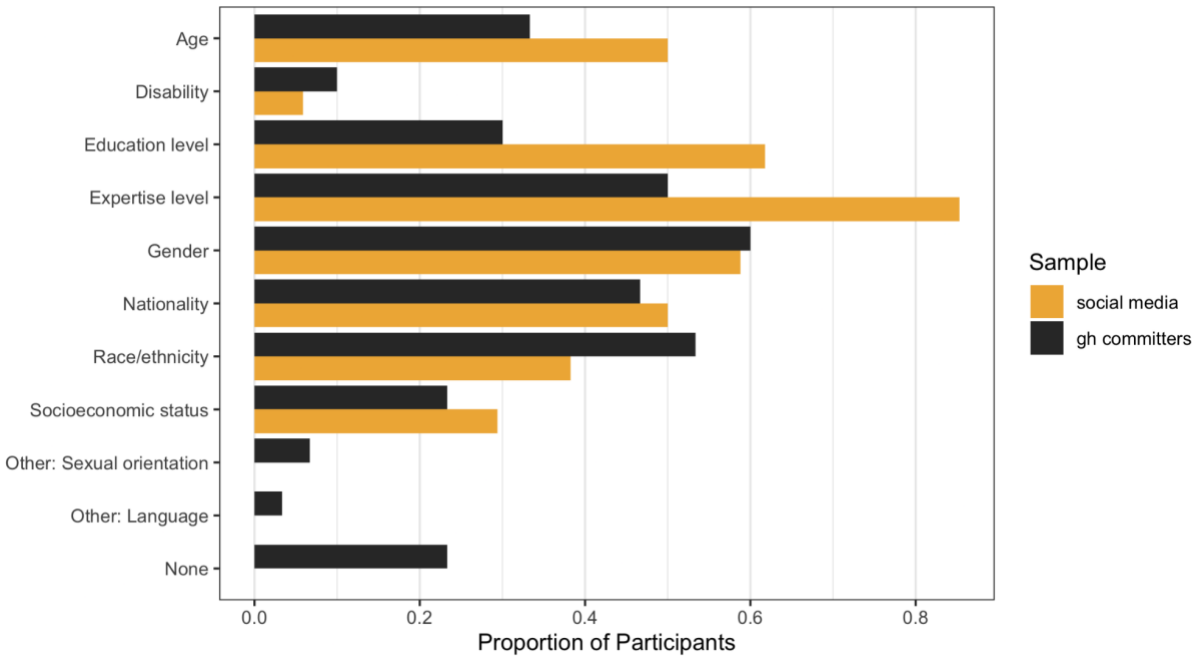


Figure 18 Salient social dimensions in FLOSS projects.

SR Q1.1c Does awareness differ between demographic groups?

Evaluating differences in awareness between demographics is limited given the small N for more typically underrepresented groups in FLOSS projects. Nonetheless, there are some differences in groups that emerge when examining gender and continent (based on stated country of residence). Figure 21 shows that higher proportions of men (N = 54) reported that they were unlikely to be aware of diversity. In contrast, all women (N = 4) and most participants who refrained from providing gender information (N = 6) were more likely to be aware of diversity. The single participant who was explicit in rejecting the gender binary did not provide a concrete response regarding awareness levels. Differences in awareness of diversity by geographical

location are presented in Figure 22. This visualization reveals a difference between American[20] (N = 42) and European (N = 15) contributor groups (the two most highly represented location groups in the data). Participants in the Americas varied in their awareness, although a large number were more likely to be aware of diversity. Participants in Europe overwhelmingly reported not being aware or having a low likelihood of being aware of diversity in FLOSS projects. This observation extends to participants who live in Asia (N = 6), with the exception of one participant who stated they were definitely aware of diversity. Oceania had N = 1; this participant stated they were definitely not aware of diversity. These findings suggest that awareness of diversity may differ on the basis of gender and culture.

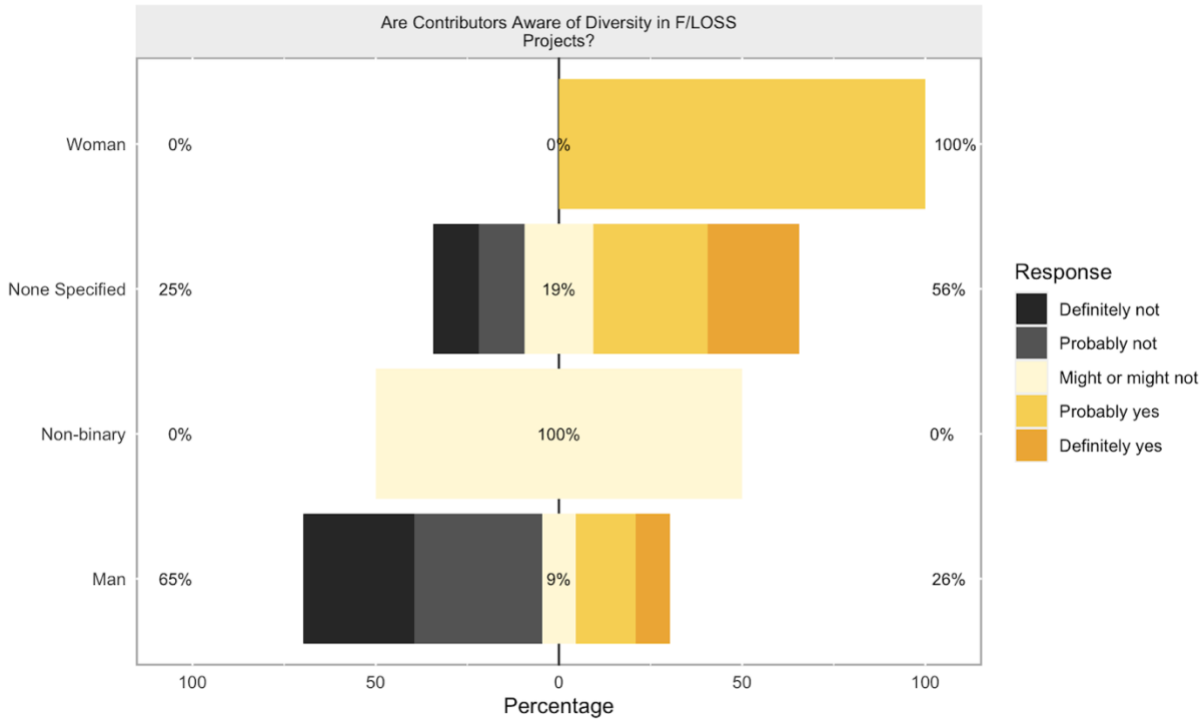


Figure 19 Survey participants' awareness of diversity in FLOSS project broken down by participant gender.

[20] These participants all reporting living in the US or Canada. South America and the Caribbean are therefore not represented in this survey research.

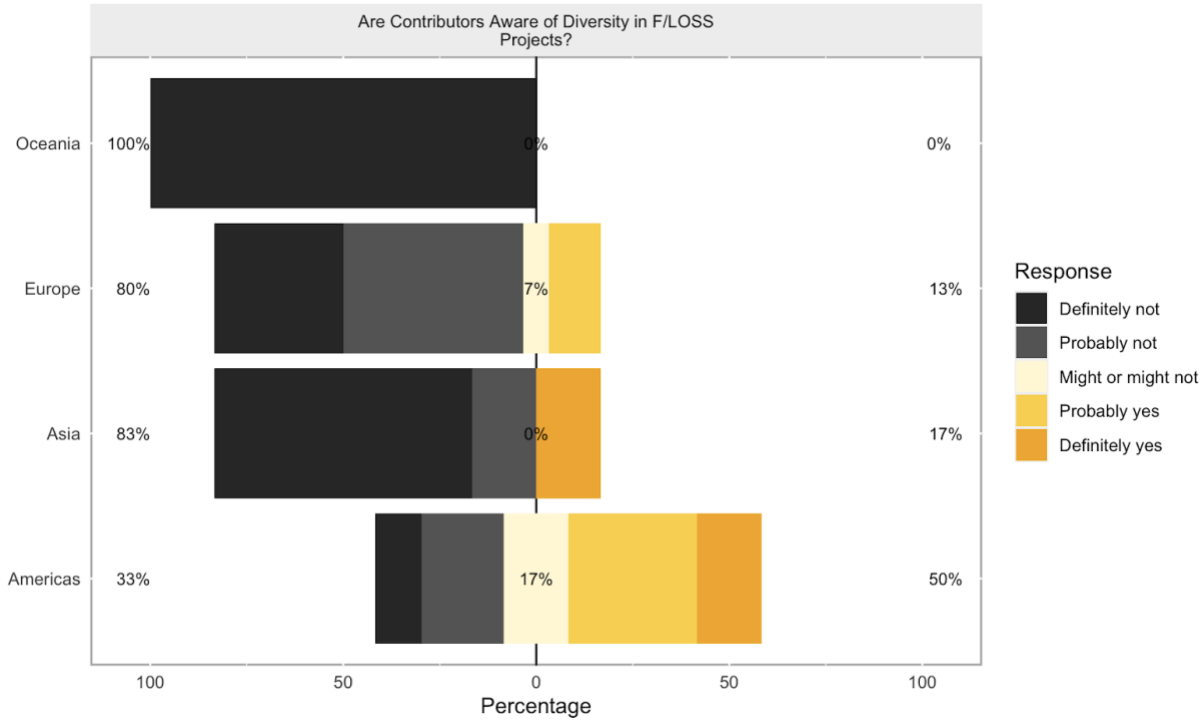


Figure 20 Survey participants' awareness of diversity in FLOSS projects broken down by the continent for which participants specified country of residence.

SR Q1.2a Is diversity important to FLOSS contributors? What types of diversity are important to contributors?

The social media/listserv sample assigned some level of importance to diversity in projects, although some participants reported that diversity was not important to them (Figure 23). Participants in the committers sample were more likely to view diversity as not important or less important. But a small number of participants in the committers sample did report that diversity was particularly important to them. Although a larger number of participants in the social media sample assigned importance to diversity, they prioritize variety in expertise and experience rather than social categories (Figure 24). This is in contrast to the committers sample which more evenly reported valuing social and technical dimensions of group composition. A

few participants in this sample added to the list of dimensions specified, noting that they additionally valued sexual orientation, language, and expertise specialty (not simply level of expertise).

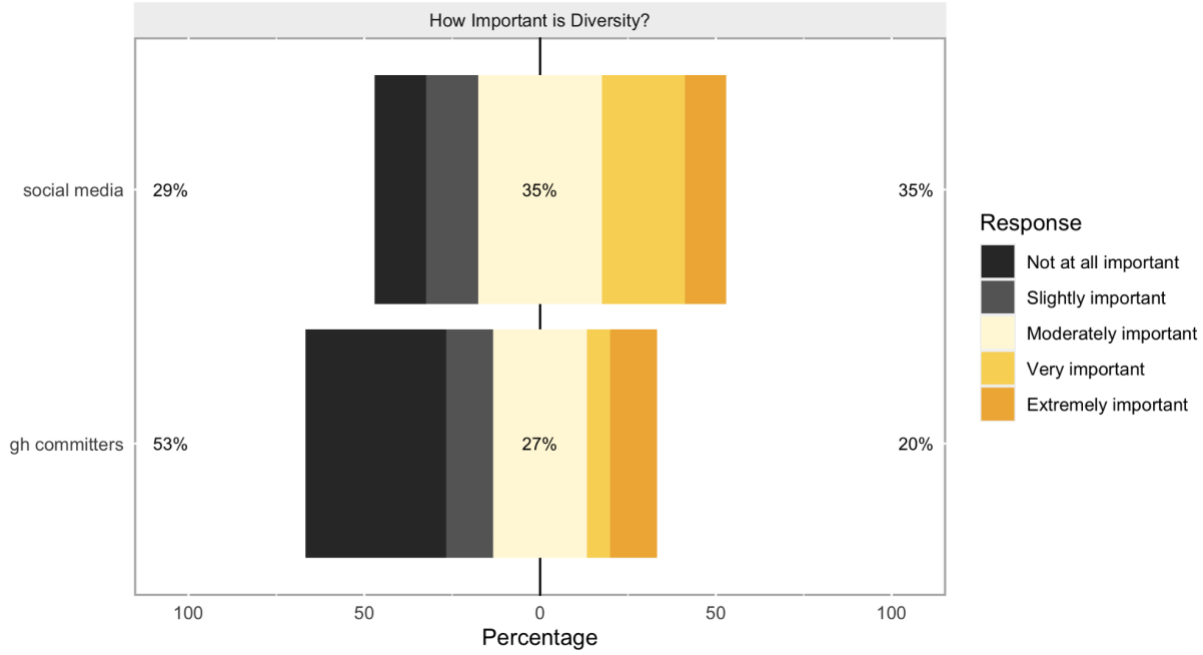


Figure 21 The importance of diversity in FLOSS according to survey participants.

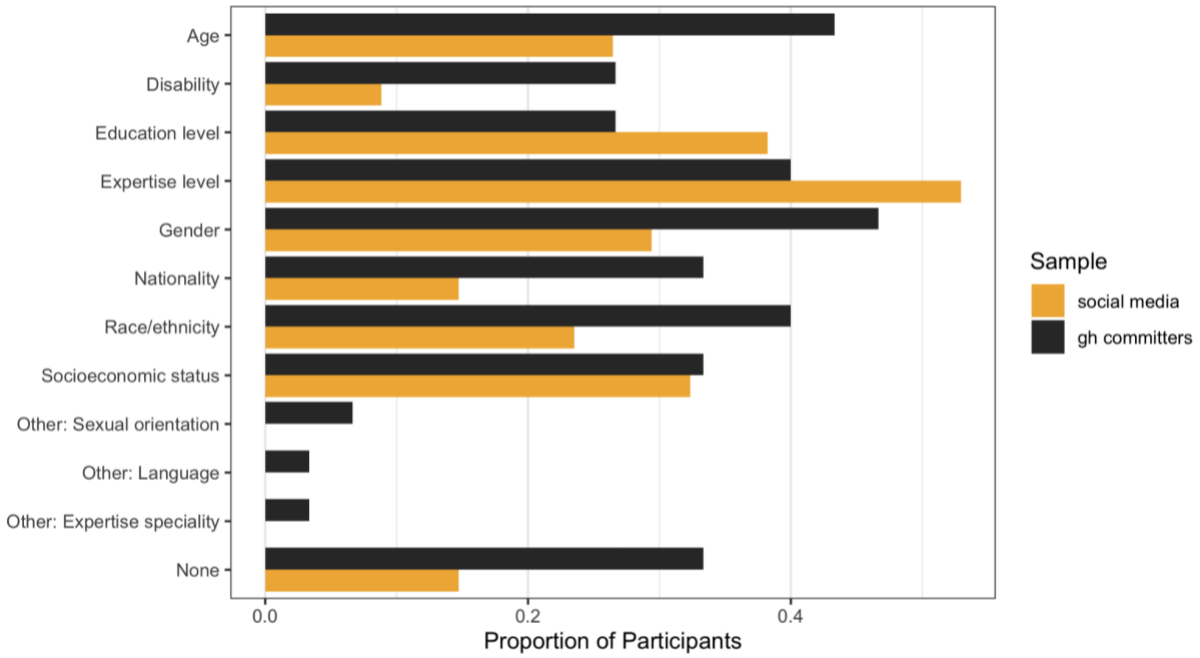


Figure 22 The types of social diversity that matter to survey participants.

SR Q1.2b What attitudes and beliefs are used to justify the importance/unimportance of diversity in FLOSS projects?

Participants were asked to elaborate on their perceptions of diversity, specifically on the importance of diversity in FLOSS development and its effect on their participation decision. The themes identified in free-text responses to this prompt are provided in Figure 23 and reported effects on decision making are visualized in Figure 24. Based on the prompt framing, most participants provided elaborations describing either why diversity was important or why diversity was not important. In terms of why diversity is important in FLOSS development, participants described its value as an *input*, its contribution to development *processes*, and its effects on project *outcomes*. As an input, diversity enables the introduction of new ideas via varied knowledge and perspectives. It also influences the ability of the project team to engage in risk management in the design process through the consideration of use cases and edge cases that

"*ensures software can be as inclusive as possible*" [GH7]. Diversity thus influences decision processes which feed into outcomes as it "*helps improve the product*" [GH22]. In this way, FLOSS contributors see diversity as having value for the development of software, adding technical strengths and extending use to a broader audience.

Moving beyond task-specific benefits, some participants expressed the importance of diversity for the *health of the project and community*, and its association with *moral and personal positions*. With respect to project health, diversity holds value as a means to ensure the project is maintained and continues to grow by expanding the labor pool available for software development tasks. At the community level, participants detail how diversity contributes to the creation of an open and inclusive community through moderation: "*Diverse backgrounds and situations* [...] *help to ensure the community rejects bad actors*" [GH23]. Other participants see diversity as a natural goal of FLOSS—"*fundamentally open source has to be accessible to everyone*" [SM21]—but also a reflection of their own values and commitments ("*I don't want anyone to feel excluded*" [GH27]). Ensuring diversity in FLOSS projects is thus seen as "*part of the work*" [SM22] and holds both intrinsic and social value to contributors. Women in both samples particularly describe the importance of diversity for their personal experiences in addition to project outcomes. They are aware that they are the minority within these spaces and appreciate the opportunity to collaborate with and learn from other women which is uncommon in their experience: "*I'm a woman and the work environment generally is filled with men, I look forward to work more with other women*" [SM1]. This points to the importance and lack of peers and expert mentorship for underrepresented groups in FLOSS projects. Furthermore, these responses provide context to the variety observed in Figure 25.

However, valuing diversity in FLOSS does not necessarily translate to a perceived need for or ability to take action. Multiple participants stated that while diversity is good and has value, it is simply not a priority for them or specifically in FLOSS development. Instead, diversity "*can take a backseat*" [SM12] to technical competence and social compatibility. Furthermore, the lack of diversity in a project is only an issue in certain circumstances: "*I would not find this problematic as long as they are not actively excluding a more diverse set of members*" [SM7]. This suggests that some FLOSS contributors do not see an explicit connection between diversity and potential benefits, technical or social. While some participants felt that it was unnecessary to take action, others lamented that they lacked the tools and resources to evaluate diversity levels in addition to the effectiveness of diversity initiatives: "*It's sometimes hard to judge, and particularly hard to measure improvement. If we want to improve our project's diversity, how do we know if we've managed that?*" [SM34]. While existing communities on GitHub have worked to develop such metrics (e.g., CHAOSS[21]), there exists a lack of awareness around available tools. An added challenge is that diversity initiatives require a form of social labor that might be neglected: "*As a technical effort, that kind of social effort can be unattractive*" [GH4].

Among survey participants who judged diversity as unimportant, elaborations ranged from *apathy* and lack of *practical value*, to an emphasis on the *volunteer nature of work* in FLOSS development. Participants who expressed apathy either stated that they did not care about diversity or that there should be more importance placed on technical skills and code quality. In their view, diversity has no added value for FLOSS. A subset of participants provided more

[21] https://github.com/chaoss/wg-dei

extended elaborations concerning the unimportance of diversity given the characteristics of the work domain. They felt that diversity should not be controlled primarily because FLOSS development is based on labor from volunteers. One participant stated that concerns of diversity could actually result in discrimination against potential contributors whereas another stated that this would lead to the artificial restriction of the contributor base. Related, a third participant stated that diversity is important in places with "*limited seats*" [GH15] (e.g., a company) which is not the case with FLOSS. This reveals an internal conflict and potential paradox in the perception of available space in this work domain: an acknowledgement of the volunteer nature of the work which is not impeded by space constraints and the belief that diversity initiatives necessarily dissolve the unlimited nature of the space. Lastly, participants express that diversity is not important in FLOSS development as it does not confer benefits to disadvantaged groups: "*with pure FLOSS (non-corporate-sponsored), where the work is unpaid and the product is free, participating won't do anything to 'catch up'*" [GH20]. While causal relationships have not been established in the literature, research suggests that this is not exactly accurate as participation in FLOSS development serves as an entry point for work in the broader field and observable traces in online platforms like GitHub serve as a portfolio of sorts, communicating information about skills and experience relevant to career choices (Dabbish et al., 2012).

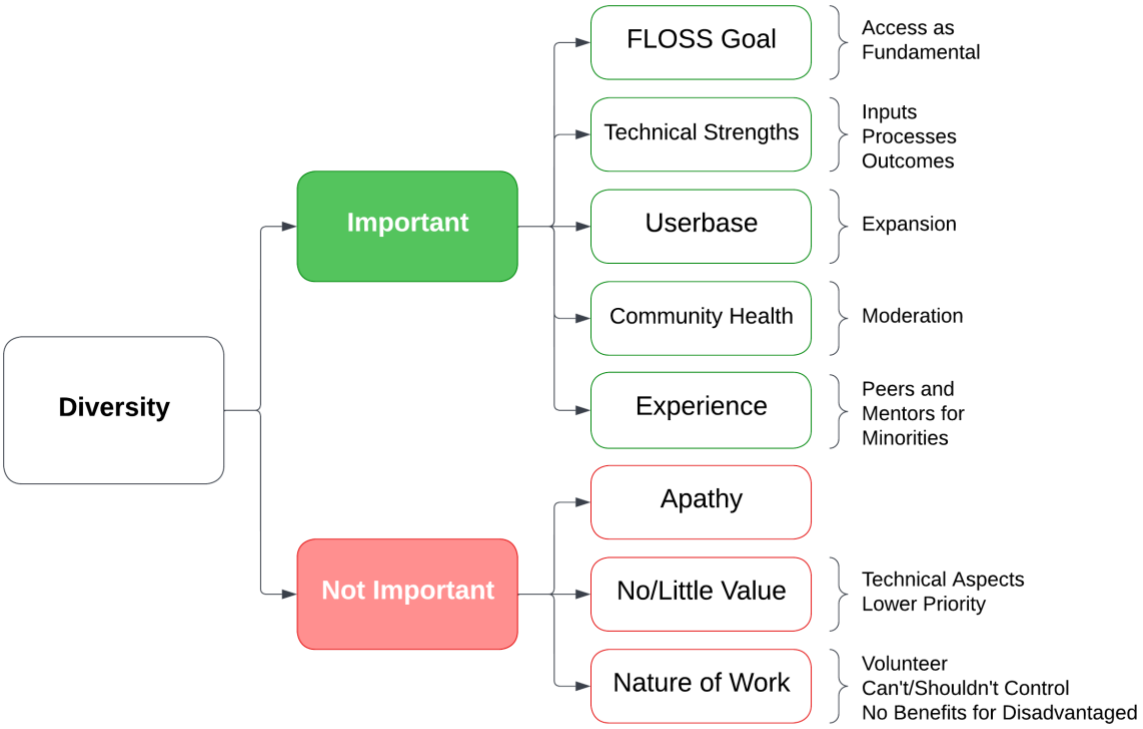


Figure 23 Why diversity matters (or does not) in FLOSS development according to participants.

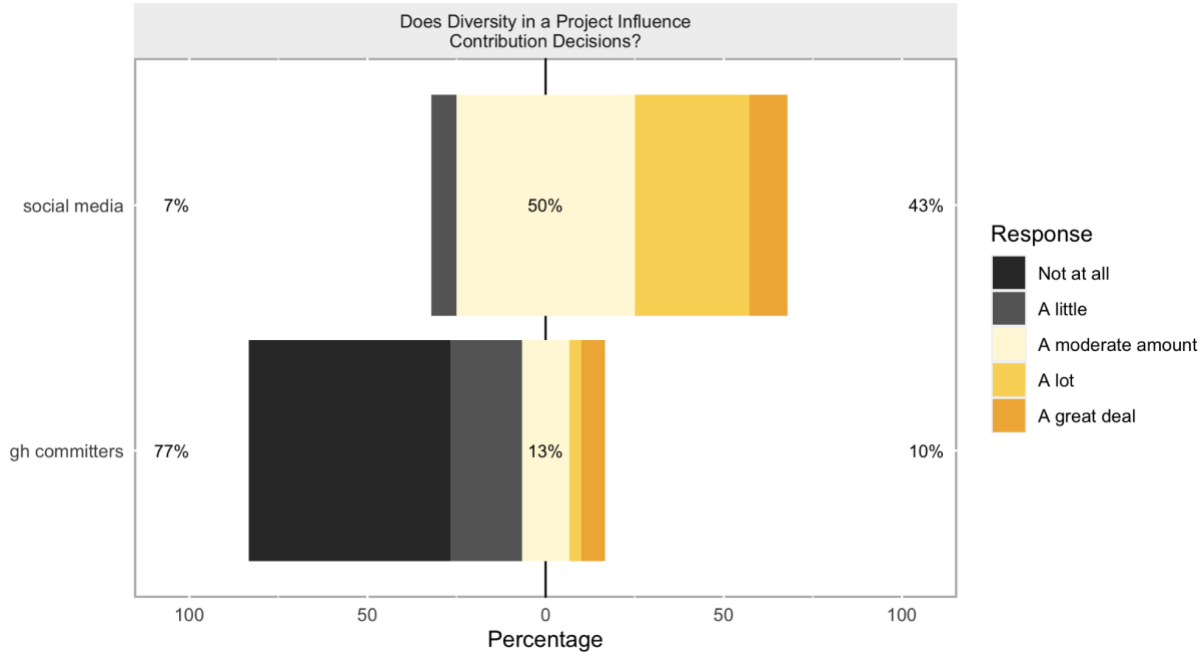


Figure 24 The effect of diversity on contribution decisions according to survey participants.

In sum, FLOSS contributors exhibit a great deal of variety in their beliefs and experiences related to diversity. While there are some commonalities in awareness and information gathering activities, the value and importance of diversity can be used to demarcate subgroups in FLOSS development: individuals who value diversity and want to promote it in the contributor and user; individuals who view diversity as good but reject control mechanisms for the contributor base; and individuals who do not view diversity as good or relevant to FLOSS development. I next describe the differences and similarities between FLOSS contributors on the topic of corporate involvement in projects.

What Are Contributors' Perceptions Of Corporate Involvement In FLOSS Projects?

The goal of the responses collected and described in this section is to characterize contributors' perceptions of corporations, specifically as they relate to experiences, effects on

diversity, contribution decisions, awareness, and information gathering. Like the previous section on perceptions of diversity, responses to Likert-type and multiple selection items are grouped according to the sample source. The results of the RTA reported in this section though primarily reflect differences between groups on the basis of the types of projects in which they generally contribute. Participants who contributed to projects with corporate involvement generally had more favorable views of corporations, and participants who contributed to projects with little to no corporate involvement generally had less favorable views of corporations. For the latter group, these views ranged from complete aversion to reluctance to engage in FLOSS development in the presence of a company.

SR Q2.1,2 What aspects of contributors' experiences are altered by corporate involvement in FLOSS projects? Do contributor reasons for participation differ in a project with/without corporate involvement?

Survey participants in the social media sample mostly rated their experiences between project types as distinct (Figure 25). There was slightly more variety in the committers sample, although in general these participants tended to rate their experiences as differing less often. Survey participants were also asked to draw from their experiences to assess the effect of corporations on diversity in FLOSS projects. While the social media sample seemed to perceive corporations as having a positive impact on group diversity in FLOSS projects, the committers sample varied more in their perceptions of corporations' effect on group diversity. A large portion of the latter sample generally perceived corporations as being neither helpful nor harmful, but, as Figure 26 shows, some participants felt that the effects of corporations on group diversity were more extreme in both directions (i.e., harmful or helpful). These variations in perception do not differ much when participants are grouped based on the types of projects to which they tend to contribute.

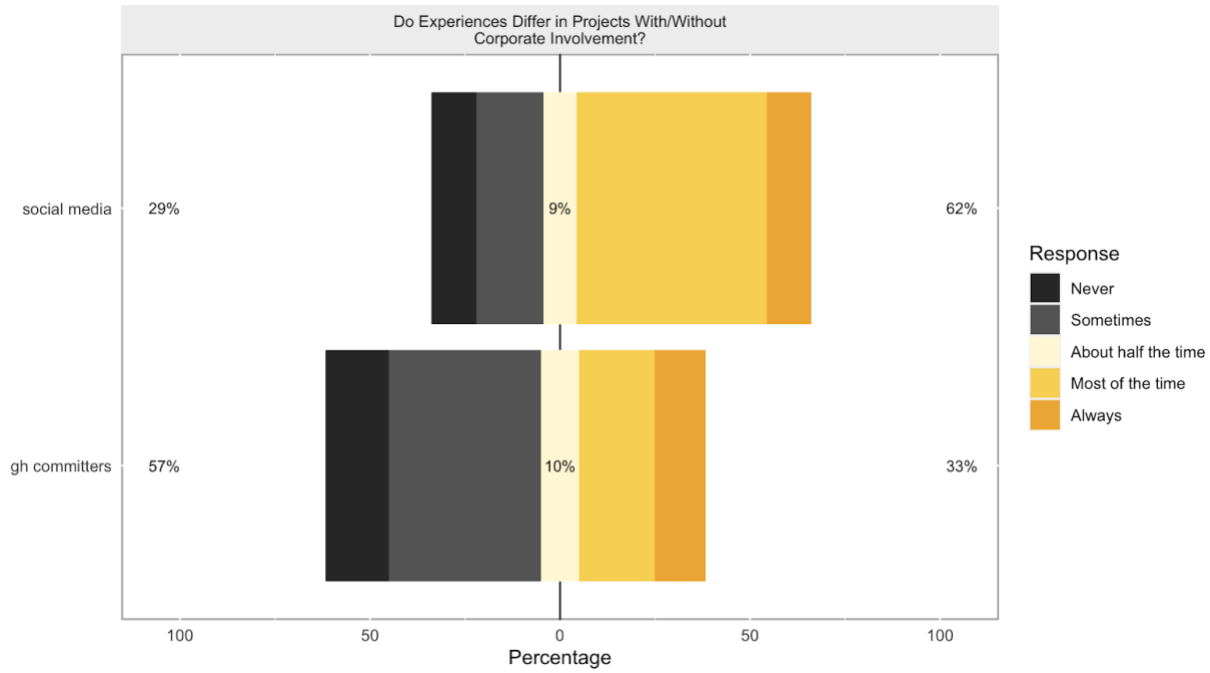


Figure 25 Survey participants ratings of the difference between projects with and without corporate involvement.

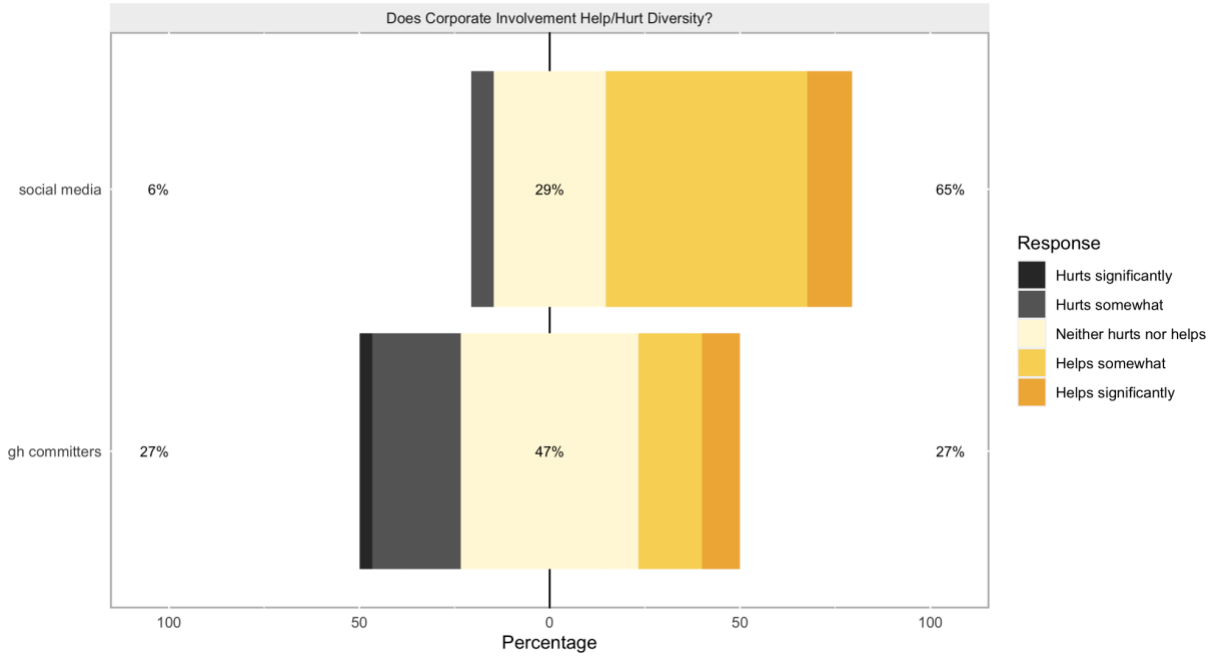


Figure 26 Perceptions of the effect of corporate involvement on diversity in FLOSS projects.

In responding to a prompt about the differences between projects with and projects without corporate involvement, most survey participants framed their experiences as *positive* and/or *negative*, and a small number reported *no difference* (Figure 27). Positive aspects of projects with corporate involvement reflect personal benefits—*career opportunities* and *compensation*—in addition to satisfactory *work support* processes and characteristics. FLOSS projects provide early-career professionals with the opportunity to build relationships for *networking* purposes but also for developing *familiarity* with specific developers and software development tasks. Another important component of corporate involvement for FLOSS contributors is the potential for compensation that is sometimes available to developers within and outside the organization. A common theme across participants who view corporate involvement favorably was the speed and quality of support in associated projects: "*projects with corporate involvement are usually lively and faster*" [SM9]; "*it feels more professional, when dealing with corporates"* [GH24].

Negative aspects of corporate involvement primarily reflected issues with *structure*, *goals*, and *risk*. The structure of corporate-involved projects was viewed as controlling and cumbersome, resulting in an inflexibility around completing taskwork due to top-down constraints. Further, there is a need for increased effort on the part of contributors to determine appropriate processes and meet bureaucratic demands: "*they'll often require CLAs [Contributor License Agreements] and other burdensome impediments*" [GH14]. Participants stated that this extended the time to complete tasks by imposing unnecessary requirements and were likely the result of the software not being a priority for the associated company. Some of these requirements are even seen by contributors as reducing the quality of code within projects. As one participant noted:

> "*most of the time, projects with corporate involvement tend to lead to a lower quality of work and more time consumed as corporations set up initiatives and guidelines we all have to follow which slows everything down"* [SM2].

While corporations may then be associated with faster times with respect to user support, they are also associated with increased time and effort on the part of code contributors with no perceived benefit to software quality.

Other factors which produced distinct experiences and influenced participation include the *goals* of companies involved in FLOSS and potential *risks* of joining projects with corporate involvement. A set of risks identified by survey participants exists at the project level: *abandonment*, *corporate wrongdoing*, *scale*, and *conflict*. These first two risks were seen as potentially producing the same effect on project success:

> "*Because a corporation may be involved in a FLOSS project, there is a concern that if the corporation in question is involved in nefarious dealings (or voluntarily abandons support) that it can severely damage or limit the momentum of a project*" [GH6].

The third type of risk similarly reflects a concern about insufficient resources, but in the opposite case: a FLOSS project with corporate involvement may become popular and grow significantly without the personnel or infrastructure to meet needs at scale. Finally, the fourth risk is linked to control and the foundations of FLOSS, namely licensing, in which participants expressed concerns about legal issues that may emerge from contribution to projects with corporate involvement. In particular, these concerns centered on the possibility that their contributions would become restricted as the intellectual property of the company in question. Each of these risks leads some FLOSS contributors to exhibit hesitancy in joining or completely avoid corporate-backed projects. However, several participants described the importance of

familiarizing oneself with a company's reputation in FLOSS communities as this better enabled the assessment of risk for participation decisions.

Although fewer participants focused on the positive and negative aspects of FLOSS projects without corporate involvement, those who did described the lack of *structure* as producing benefits and consequences. On the positive side, the governance models of non-corporate projects are perceived as fairer and granting greater freedom to contributors. This allows them to improvise as needed to produce code that addresses their own and others' needs. On the negative side, the lack of structure can create coordination issues: "*projects without corporate involvement are more public, but also more disorganized, with mostly no one doing project management*" [GH4].

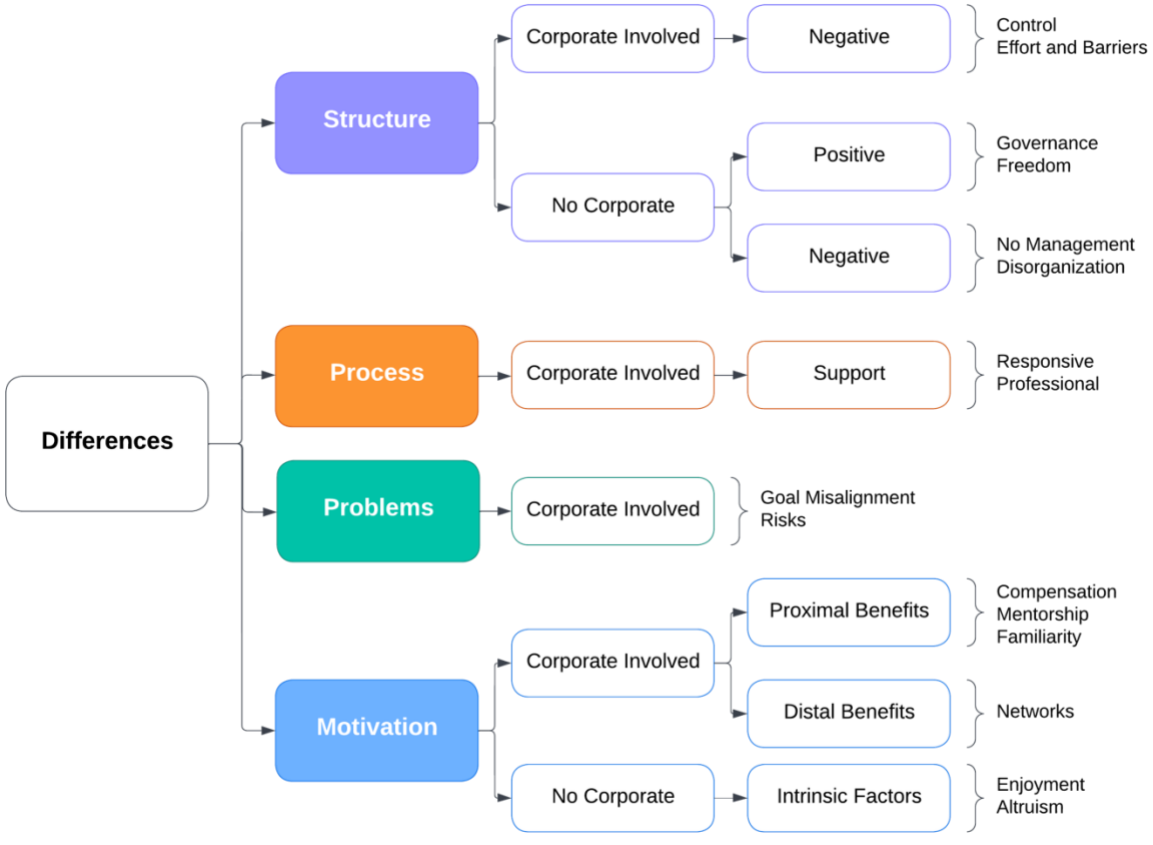


Figure 27 Differences between projects with and without corporate involvement.

SR Q2.3 Does corporate involvement influence contributors' intent to sustain participation?

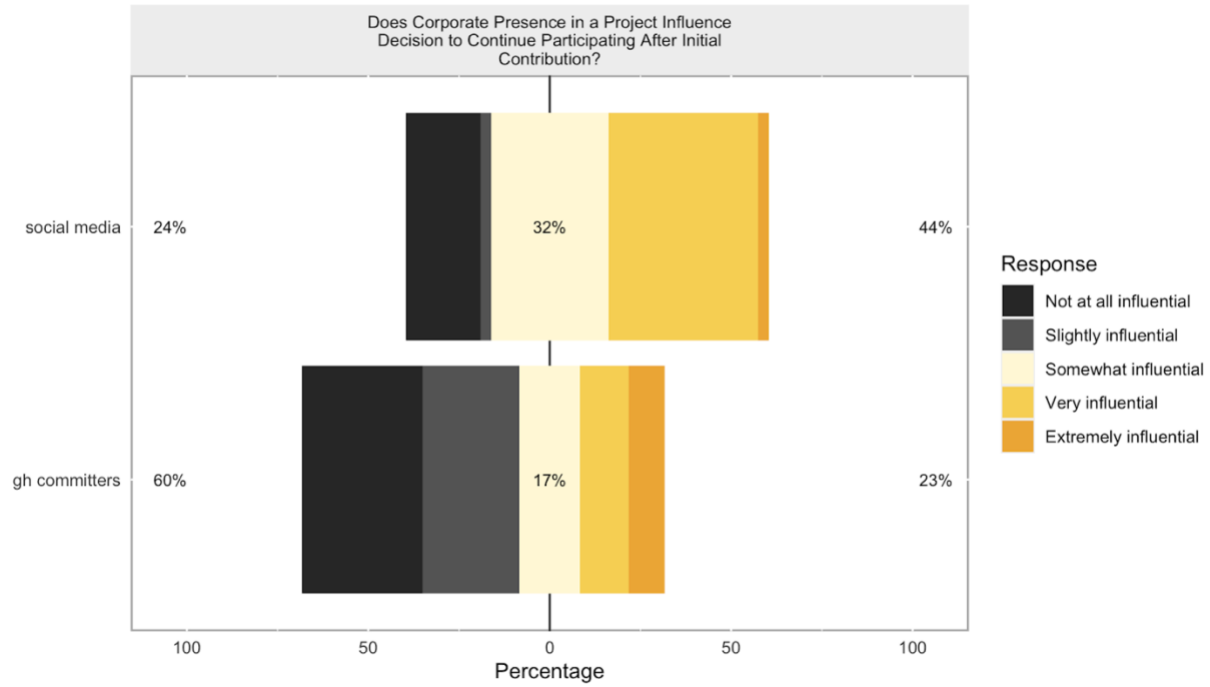


Figure 28 Participant ratings of corporate presence's effect on contribution decisions.

In responding to a prompt about the effect of corporate involvement in a project, or lack thereof, on decisions to continue contributing, responses covered a number of reasons they might choose to end or continue their participation in corporate-involved projects (Table 20). In general, participants' responses reflected the differences they experienced between these types of projects (see overlapping themes between Figure 27 and Table 20). Among reasons to leave a project, participants expressed that they experienced discord with decision makers and a misalignment of values. While some were motivated to continue due to perceived personal benefits, others were demotivated by a lack of collective need and social good:

> "*I'm less interested in helping with projects that already have corporate funding since they have the money to hire someone, while other projects I could work on don't*" [SM15]

and "*It's more of a moral dilemma - Like why should I contribute to the benefit of a corporate whose only motivation is to become richer? I would rather spend my time with FOSS projects whose contributors are driven by passion for programming (most of them) and the prime motivation being to create better software and make everyone's life a bit better*" [SM28].

These participants affirmed their commitment to the communities and values of free software in guiding their contribution decisions over time. Still, some participants clarified that they were more nuanced in evaluating projects: their decisions were informed by how the contribution experience felt to them or based on the observable and established history between the corporation and FLOSS communities. Even in situationally informed decisions, contributors expressed that such nuance had limits as some entities were to be avoided at all costs: "*I [am] sensitive to a small set of organisations with whom I do not want any association*" [SM21].

Table 20 Decision rationale for sustained participation in projects with and without corporate involvement. Corporate involvement is abbreviated to CI in the Project Type column.

Project Type	Decision	Theme Group	Theme	Subtheme
CI	*Remain*	Personal Benefits	Resources	
			Collaboration	
	Leave	Control	Top-Down Decision Making	Resource Allocation
				Payment
		Goals	Misalignment	Ethics
		Motivation	Compensation	
		Legal Reasons	Licensing	Ownership
-	*Situational*	Moral	Improving Code	"Right Thing"
		Experience	Feeling	
		Personal Benefits		
		Community Interaction	Corporate's Reputation	
			Culture	
No CI	*Remain*	Values	Freedom	
			Moral	Shared

SR Q2.4 Are contributors aware of the actions of corporations subsidizing and participating in FLOSS development?

Participants in both samples vary with respect to the degree that they reported being aware of current events associated with GitHub and FLOSS development (Figures 29, 30, and 31). Both samples were for the most part aware of Microsoft's acquisition of the platform and the platform's relatively new AI-based coding tool CoPilot. Both samples also had a blend of

participants who were aware and were not aware of GitHub's association with U.S. Immigration and Customs Enforcement (ICE) and backlash from employees and users. The two samples differed however regarding their awareness of the remaining two events: the announcement of a partnership between DARPA and the Linux Foundation, and controversy at GitHub stemming from a Jewish employee's Slack messages about the threat of Nazis during the January 6 U.S. Capitol attack. In both cases, the social media sample reported higher levels of awareness than the committers sample although most participants across both samples reported no or little awareness of GitHub's handling of its employee's comments. This may be due to the fact that the social media sample was mostly made up of FLOSS contributors living in the US whereas the committers sample was more evenly distributed between the Americas and Europe.

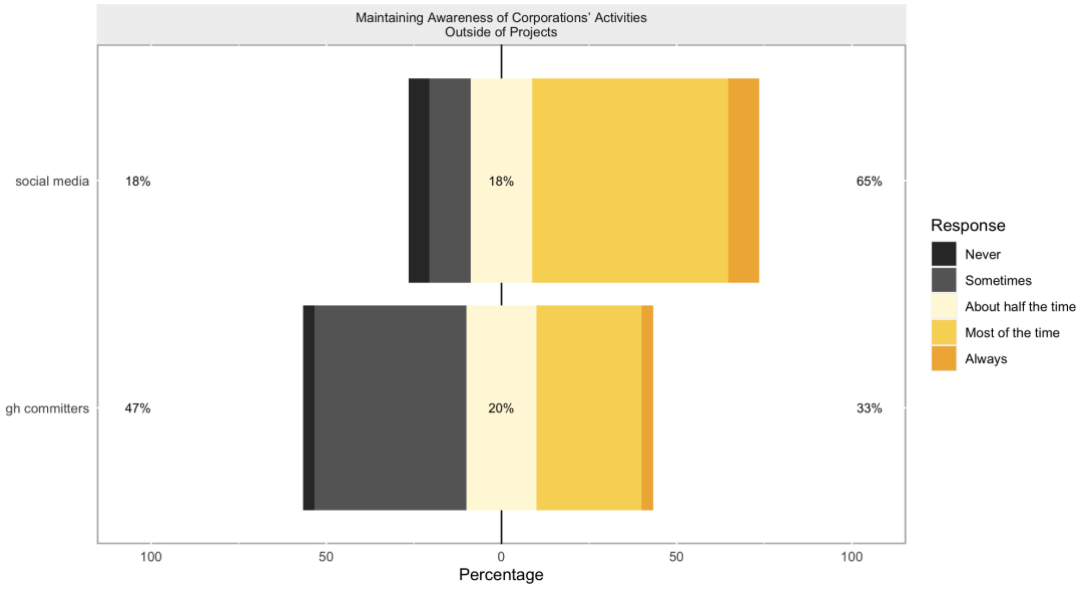


Figure 29 The degree to which participants reported maintaining awareness of corporations' activities outside of FLOSS projects.

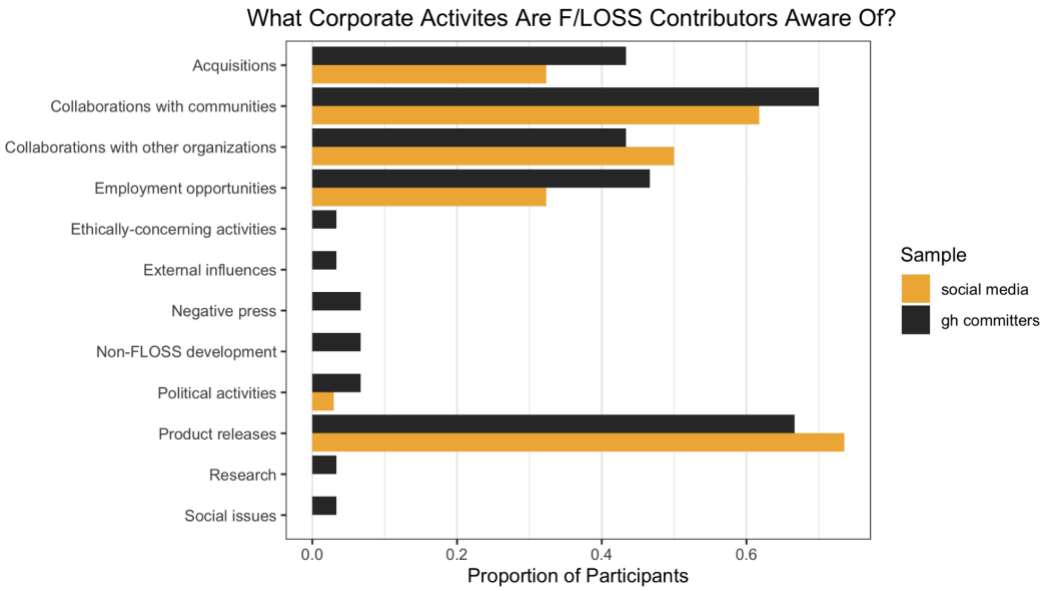


Figure 30 Participant responses regarding the awareness they have of corporate actions and activities. Participants in the committers sample added 'Ethically- concerning activities', 'Negative press', 'External influences', 'Non-FLOSS development', 'Research, and 'Social issues', hence the low proportions for these types of events.

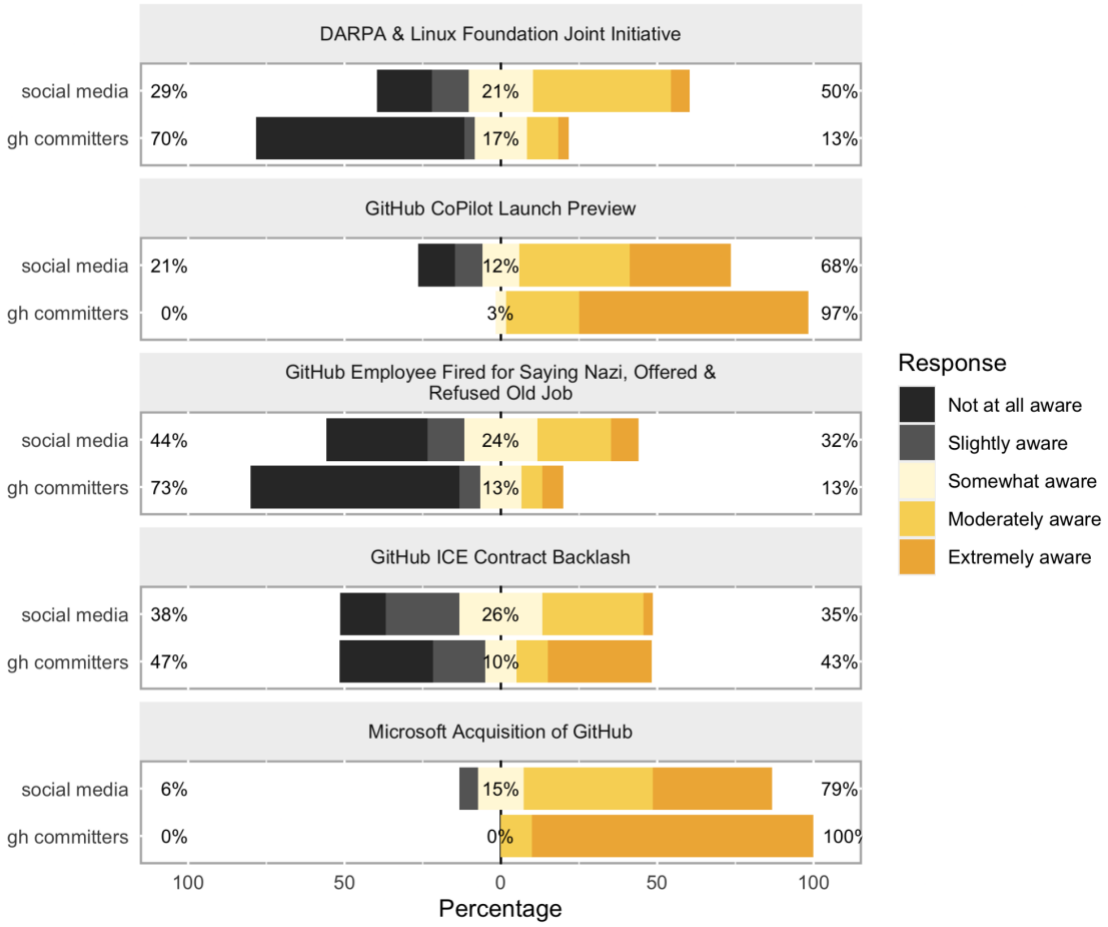


Figure 31 Participants' awareness of FLOSS-related events.

SR Q2.5 Are contributors responsive to the actions of corporations subsidizing and participating in FLOSS development?

Some survey participants report that their participation in projects has been altered as a result of the actions and decisions of organizations involved in FLOSS development. In the social media sample, 35% of participants reported that organizational decisions influenced their decision to participate in FLOSS projects (not GitHub specific) and, in the committers sample, 17% reported that organizational decisions influenced decision to participate in FLOSS projects.

A considerable portion of survey participants report that they have changed the way they contribute to projects as a result of the actions and decisions of organizations involved in FLOSS development. In the social media sample, 50% of participants reported that organizational decisions influenced their use of GitHub, compared to 37% of participants in the committers sample. This subset of the committers sample specified that they chose to end their use of GitHub as a result of events reported in the press, including Microsoft's acquisition of GitHub, GitHub's contract with ICE, GitHub's termination of a Jewish employee, and/or GitHub's launch of CoPilot. One additional participant noted that although they did not end their use of GitHub, they were less likely to recommend the platform to others. In contrast, the social media sample subset varied along this dimension: some specified that events reported in the press led them to end their use of GitHub while others specified that this led them to begin using GitHub or affirmed their decision to use the platform. Similar to the committer sample, some in this subset included Microsoft's acquisition of GitHub, GitHub's contract with ICE, GitHub's termination of a Jewish employee, and/or GitHub's launch of CoPilot as motivators to end their use of GitHub.

Participants had the opportunity to elaborate on their general perceptions of corporate involvement in FLOSS. Among those that chose to provide a response, many elaborated on their responses to corporate actions. There was a common theme of choosing to host or mirror their projects elsewhere following Microsoft's acquisition of GitHub. The majority specified that their individual and organizational projects were migrated to or mirrored on GitLab and one participant noted that the acquisition led them to create their own web platform for hosting FLOSS.

The results of this survey research provide evidence for the effect of corporate activities and decisions on individual decision making related to participation in FLOSS development. These survey results also add to the body of work showing that individuals attend to social information in groups in online platforms and are at least implicitly aware of group differences. There is also some evidence that a subset of FLOSS contributors is motivated by the presence of group diversity in projects but feels unequipped to promote and maintain it. Participants in the sample not only confirmed that there are key differences in individual preferences, experiences, and beliefs with respect to project types and group diversity, they provided a richer understanding of their perceptions through descriptions of the rationale which informs their perspective and in turn guides their decision making. I next describe the results of a complementary approach to the survey study: an analysis of big data extracted from FLOSS project ecosystems in GitHub.

Analysis of FLOSS Ecosystems

In this section, I describe the results of a series of quantitative analyses of participation and project differences based on levels of corporate involvement in FLOSS development. First, I summarize descriptive information for projects in the study sample and their corresponding ecosystems. Then I move on to characterize differences in participation between project types and diversity levels through statistical techniques applied to predict tenure (hypothesis testing) and time until departure (survival analysis). To enhance understanding of these differences, I describe how diversity varies between and within projects. Lastly, I provide an overview of the network characteristics of project ecosystems to reveal similarities and quantitative differences between project types.

Project Descriptives

I next describe high-level characteristics of GitHub projects selected for further analysis and as origins for ecosystem creation. This information is presented in Tables 21-23. These tables are arranged with the projects having the highest levels of corporate engagement at the top and the lowest levels of corporate engagement at the bottom. The sample projects cover a broad set of application domains ranging from computer systems to software development to education and research tools. The primary programming language for these projects falls nearly evenly across six languages, including the most popular language on GitHub: Javascript (Orlowska et al., 2021). There is some variation in licensing, but GNU licenses are the most represented in the sample. The contributor bases of these projects reflect the gender makeup commonly found in most studies of FLOSS on GitHub: contributors identified as men make up the majority and only some of these projects have contributors identified as women.

Table 21 Primary maintainer type, primary language, application domain, and license information for sampled projects. The projects are grouped by level of corporate involvement (CI): 1 = No clear corporate influence; 2 = Receives corporate support; 3 = Owned by a company/for profit corporation.

CI	PID	Maintainer	Domain	Language	License
3	1	Company	API Integration	Java	Apache-2.0
	6	Company	Ecommerce	PHP	GNU-LGPL-2.1
	7	Company	Civic/Environmental Research	Python	Multiple
	8	Company	Media Distribution	Java	Apache-2.0
	11	Company	Software Development, QA	Go	MIT
	13	Company	Communications	Java	None specified
	17	Company	IT, Network Security	Go	Apache-2.0
2	2	Research Group	Computer Systems	C	BSD-2-Clause
	3	Community	Application Development	Javascript	GNU-AGPL-3.0
	5	Community	File Management	C	GNU-GPL-3.0
	14	Community	VFX, Image Processing	C++	BSD-3-Clause
1	4	Community	IT, Server Systems	PHP	GNU-GPL-3.0
	9	Individual	Education	Python	GNU-GPL-3.0
	10	Individual	File Distribution	Javascript	GNU-GPL-3.0
	12	Community	Ecommerce	PHP	MIT
	15	Community	Research, Data Analysis	Python	MIT
	16	Community	Data Analysis	Python	GNU-GPL-3.0
	18	Community	Geocaching	Java	Apache-2.0

Table 22 Project contributor base size and gender makeup.

CI	PID	No. Contributors	No. Women (%)	No. Unknown (%)
3	1	15	2 (13%)	0 (0%)
	6	11	0 (0%)	0 (0%)
	7	9	1 (11%)	1 (11%)
	8	67	4 (6%)	1 (2%)
	11	45	5 (11%)	0 (0%)
	13	24	0 (0%)	2 (7%)
	17	485	33 (7%)	34 (7%)
2	2	40	2 (5%)	2 (5%)
	3	20	0 (0%)	0 (0%)
	5	188	9 (5%)	4 (2%)
	14	161	8 (5%)	5 (3%)
1	4	3	0 (0%)	0 (0%)
	9	5	0 (0%)	0 (0%)
	10	97	6 (6%)	9 (9%)
	12	49	1 (2%)	0 (0%)
	15	49	8 (16%)	1 (2%)
	16	54	5 (9%)	0 (0%)
	18	102	6 (6%)	11 (11%)

Table 23 Contributor gender counts for project sample overall and by level of corporate involvement. Corporate Involvement is abbreviated to CI.

	No CI (1)	Some CI (2)	Company Owned (3)	Overall
Women	27	19	43	89
Men	313	379	539	1231
Unknown	21	11	36	68
Total	361	409	618	1388

Ecosystem Descriptives

Of the 18 projects in the sample, cross-references to other projects in comments were found for nine projects and these projects were used to model participation in FLOSS ecosystems. The names of the projects in the sample are provided in Appendix C. These nine projects are thus embedded in clearly definable ecosystems which enable the analysis of differences between and within ecosystems. The remainder of this section focuses on those nine projects and their ecosystems. Overall counts and statistics for the projects included in the ecosystem analysis are provided in Tables 24 and visualized in Figure 32. Levels of corporate involvement are visualized in Figure 33.

Table 24 FLOSS Ecosystems Data Set. The count column is based on aggregation of data across all ecosystems. The remaining columns are based on data grouped at the ecosystem level.

	Overall	***Ecosystems***		
	Count	**Median**	**Mean**	**SD**
No. Projects	38	2	4.22	3.63
No. Contributors	10,109	294	1,123.22	1,534.88
No. Commits	369,825	8,395	41,091.67	47,180.73
No. Pull Requests	50,584	2,208	5,620.44	6,568.57
No. Issues	24,808	1,179	2756.44	3,434.88

Projects 17 and 18 have the largest ecosystems and differ from each other based on observed level of corporate involvement. The ecosystem for project 17 has relatively high levels of corporate involvement and among the highest activity levels. This project's ecosystem has the highest level of internal activity (i.e., commits), and has the second highest level of external contributions (pull requests). The ecosystem for project 18 similarly exhibits high levels of activity but has relatively low levels of corporate involvement. The third largest ecosystem belongs to project 16 and is much smaller than the two previously described ecosystems but has very high levels of activity, including the highest level of external contributions. This ecosystem does not contain any projects that were identified as being owned by a private company. The remaining ecosystems are generally small and have much lower levels of activity with the exception of project 15's ecosystem which also has no corporate presence.

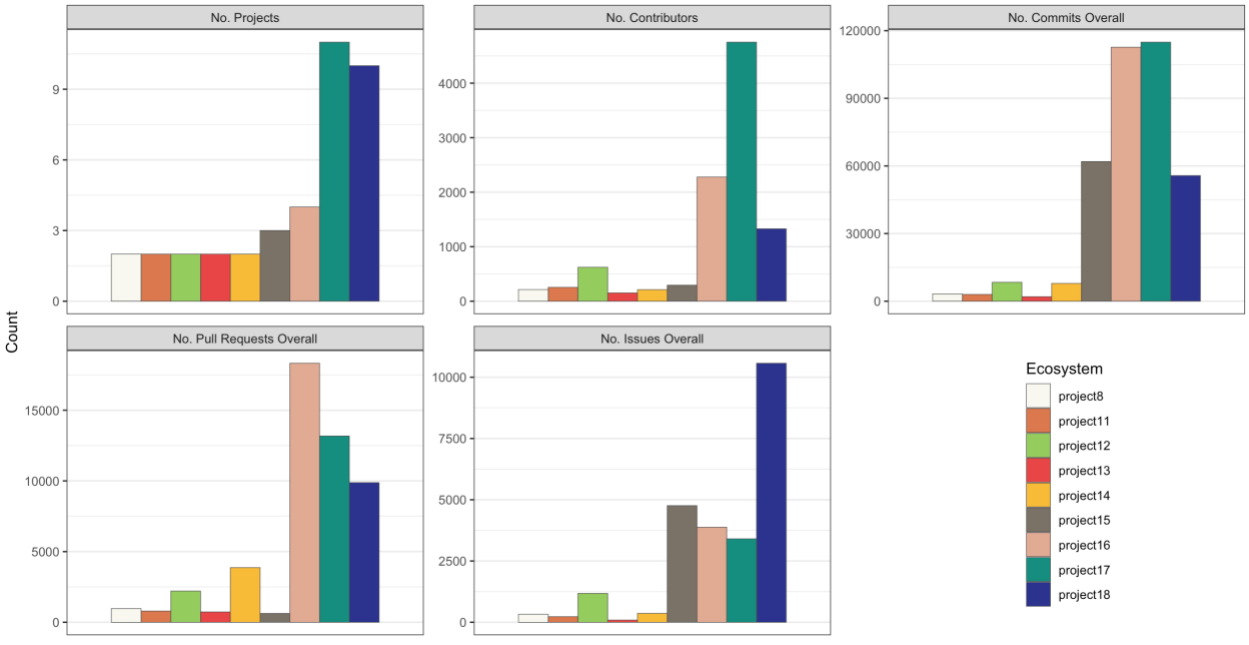


Figure 32 Ecosystem Size and Levels of Activity. Contributor and event counts are based on accounts that were labeled as human (i.e., bot activity is not included in plots or analyses).

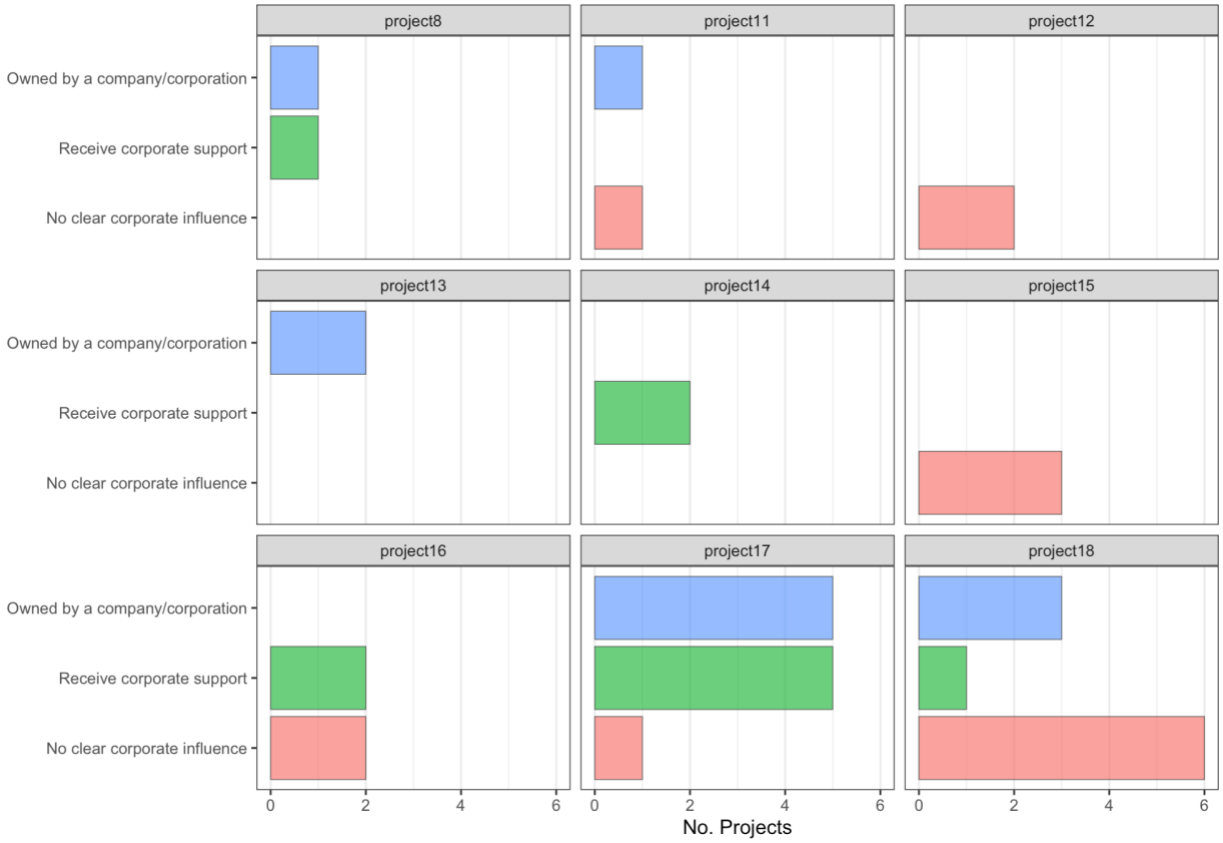


Figure 33 Corporate Engagement in FLOSS Ecosystems. The distribution of types of projects in each ecosystem showing the different levels of corporate presence.

Participation Differences

I now describe the results of the ecosystem analysis as they relate to my research questions. For each question, I characterize the relevant data by providing visualizations of distributions and summary statistics for each variable. Then, I describe how these data address the research question, including the results of statistical tests of group differences, when applicable and appropriate.

ER Q1.1,2 Do contributors' tenure and sustained participation differ for projects with and without corporate involvement?

Contributors' tenure in a project was calculated to examine differences between project types (Table 25). A series of density plots are used to visualize the distribution of tenure in Figures 34-37. This set of plots is used to visualize differences across a set of dimensions: level of corporate involvement, gender, location, and ecosystem size. Density plots, rather than standard histograms, are used to visualize the distribution of tenure because the shape of the distribution is not based on subjective input (i.e., selection of the number of bins or the width of bins for grouping values) which better enables an assessment of the actual distribution shape for the variable(s) of interest. In Figure 35, tenure values are grouped by corporate involvement level and in general the plots show that tenure is right skewed, indicating that the mean is greater than the median regardless of corporate engagement. This is most severe for the midpoint level (receiving corporate support). The distribution for tenure in projects that are owned by a company has two peaks, suggesting that it is bimodal, whereas the distribution has a single peak for projects that have no clear corporate influence and appears unimodal.

Table 25 The number of projects in each project type group and corresponding minimum and maximum values for tenure.

			Tenure (Quarters)	
	No. Projects	**No. Contributors**	**Minimum**	**Maximum**
Owned by a company	15	1,698	1	54
Receives corporate support	11	6,024	1	57
No clear corporate influence	12	3,004	1	56

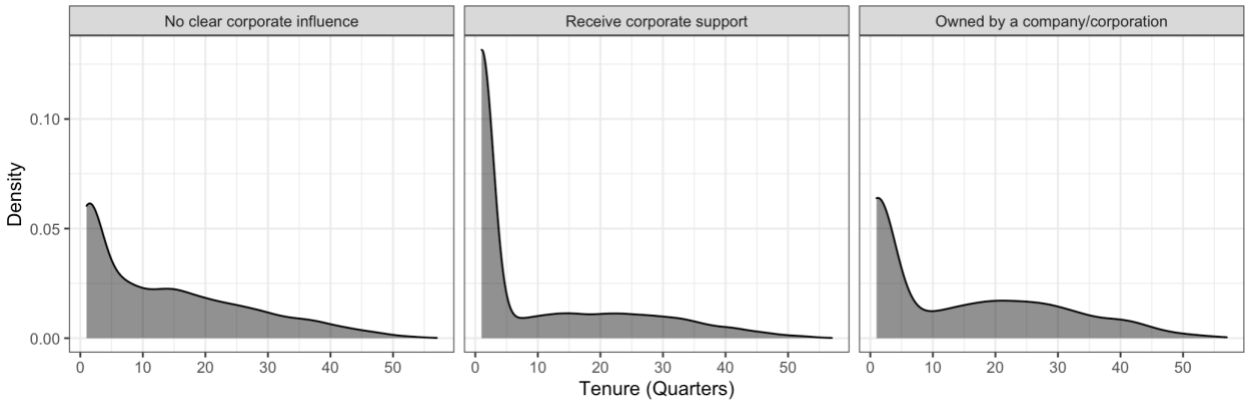


Figure 34 Distribution of Tenure in FLOSS Ecosystems: Density plots for contributor project tenure in quarters (3-month periods) with values grouped by level of corporate involvement.

The distribution of tenure does not appear to vary greatly on the basis of gender as visualized in Figure 35. Instead, the greatest difference again appears to be between levels of corporate involvement. Some differences in the distribution of tenure are present when grouping data based on contributor location as in Figure 36. Unlike the overall distribution of tenure for projects with no clear corporate influence, the distribution of tenure for contributors in this project type who are located in Oceania appears bimodal. The least amount of skew is observed for contributors in Oceania and, to a lesser extent, in Africa; however, these groups are the smallest (N = 69 [<1%] and N = 161 [1%], respectively) compared to the other continent groups. Furthermore, while location information was available for approximately 70% of contributors, the density plot for the largest grouping of contributors is the unknown category (N = 3,558 [30%]) and the distributions for this category follow the distributions observed in Figure 34. The distributions for contributors in the Americas and Europe roughly follow the distributions in Figure 34 as well. The most extreme skew is observed for the group of contributors located in Asia. Lastly, the distribution of tenure varies somewhat when examining ecosystem size (Figure

37) and in particular for two subgroups. The distribution of tenure in projects with no clear corporate influence in one of the largest ecosystems (blue distribution in the first panel in Figure 37) and in projects with high corporate involvement in the smallest ecosystems (grey distribution in the third panel in Figure 37) do not have the right skew that is otherwise observed in these distributions.

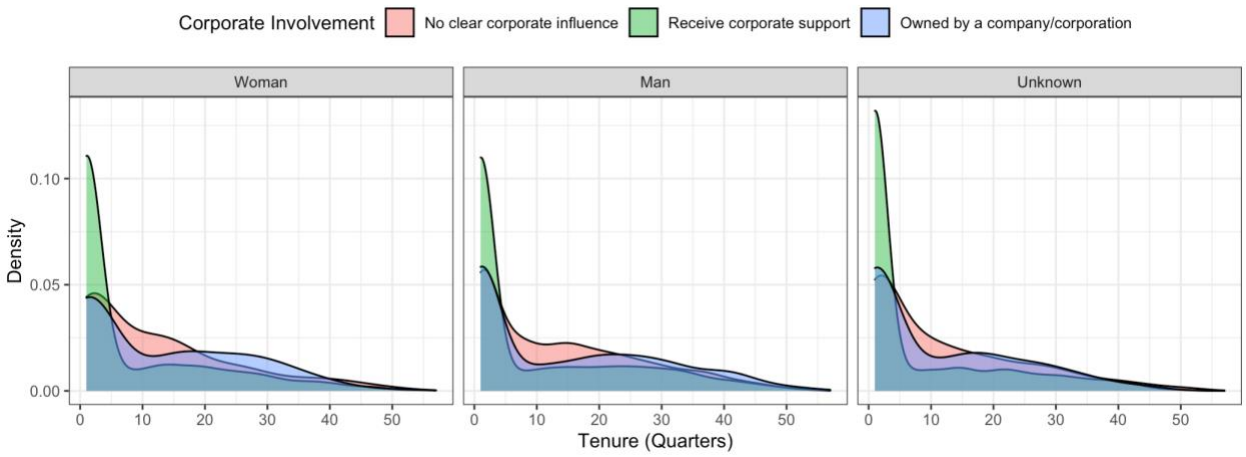


Figure 35 Distribution of Tenure: Density plots for project contributor tenure in quarters (3-month periods) with values grouped by gender and level of corporate involvement.

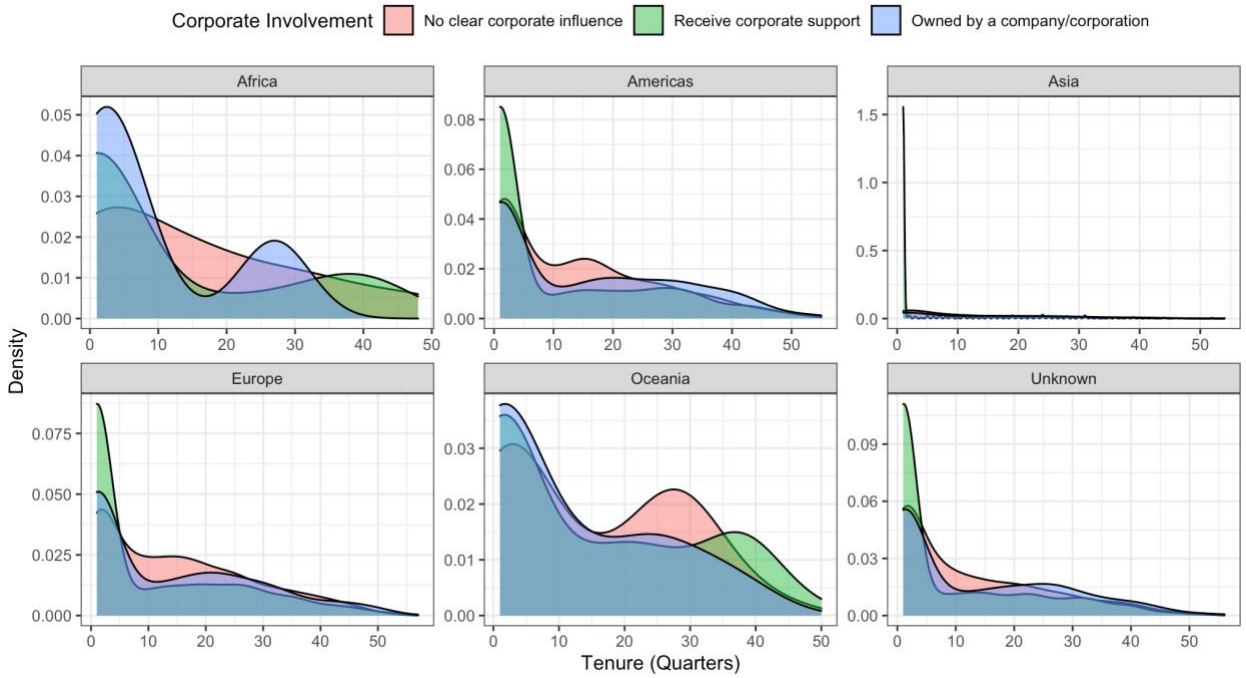


Figure 36 Distribution of Tenure: Density plots for project contributor tenure in quarters (3-month periods) with values grouped by location (continent) and level of corporate involvement. Unlike the other plots, in this set, the y-axis scales vary by facet. This was an intentional decision due to the extreme values observed in the Asia facet which obscured the distributions for the other continent groups.

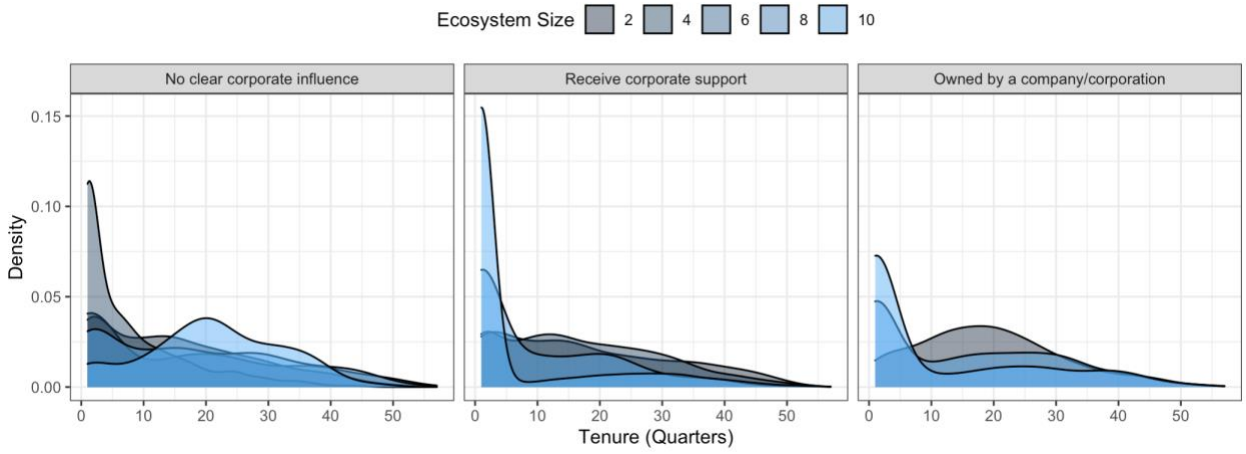


Figure 37 Distribution of Tenure: Density plots for project contributor tenure in quarters (3-month periods) with values grouped by ecosystem size (number of projects) and level of corporate involvement.

To investigate the effect of corporate involvement on tenure, a complementary set of analyses were completed. First, due to the distribution of the data, normality assumptions required for parametric statistical tests were not met. A Mann-Whitney *U* test was therefore applied to the data. The results of the test applied to company-owned and no corporate influence projects are reported here as this is the primary group comparison of interest based on the study's research questions. Additional results for tests comparing these groups to the midpoint group are provided in Appendix D. The results of this analysis suggest that the presence of corporations in projects is significantly associated with differences in tenure. Specifically, results indicate that tenure (measured in quarters) is greater for projects with no clear corporate influence (*Mdn* = 10) than for projects which are company- or corporate-owned (*Mdn* = 9), $U = 3{,}034{,}696$[22], $p = .01$.

Second, a survival analysis was conducted to examine differences in time until departure from a project. This analysis thus serves to provide an additional dimension from which to examine differences in participation. Differences in survival probabilities between project types are provided in Table 26 and visualized in Figure 38. Generally, both groups have a similar survival length as indicated by the end of both curves. However, there is some difference in the first ten quarters, or 2.5 years, of participation: contributors in projects with no clear corporate influence have a better chance of sustained participation. This changes after the first ten quarters, with contributors in both projects having similar chances of sustained participation. Stated another way, the risk for leaving is lower at the beginning for no corporate influence projects—the curve begins at 70% chance of survival—and this risk increases to a similar level observed in company-owned projects after ten quarters. Survival curves generated for project ecosystems,

[22] In Mann-Whitney tests, large *U* statistics result from large sample sizes.

ecosystem size, gender, and location are visualized in Figures 39-42, respectively. Contributors in project 13's ecosystem (entirely company-owned) had the greatest probability of sustained participation, followed closely by project 8 (mixed company-owned and corporate support) and project 11 ecosystems (mixed company-owned and no corporate influence). This suggests that corporate involvement measured at the project level rather than ecosystem level better captures participation differences between project types. The majority (56%) of project ecosystems are small (2 projects) and contributors in these ecosystems have a high probability of sustained participation (Figure 40). But the risk of leaving increases to the risk levels observed for other ecosystem sizes at ten quarters, reflecting the same pattern observed in Figure 38. There is no visibly distinct difference between survival curves when grouping contributors by gender (Figure 41) or location (Figure 42).

Table 26. Survival probabilities by project type at two time points. Survival probabilities at these time points represent the greatest difference (quarter 1) and convergence (quarter 12) between project types. Probabilities across all time points (survival curves) are visualized in Figure 39.

Time (Quarter)	Corporate Involvement	Survival Probability	Std. Error	Lower 95% Conf. Int.	Upper 95% Conf. Int.
1	*None*	0.71	0.01	0.70	0.73
	Complete	0.58	0.01	0.55	0.60
12	*None*	0.45	0.01	0.43	0.47
	Complete	0.46	0.01	0.44	0.48

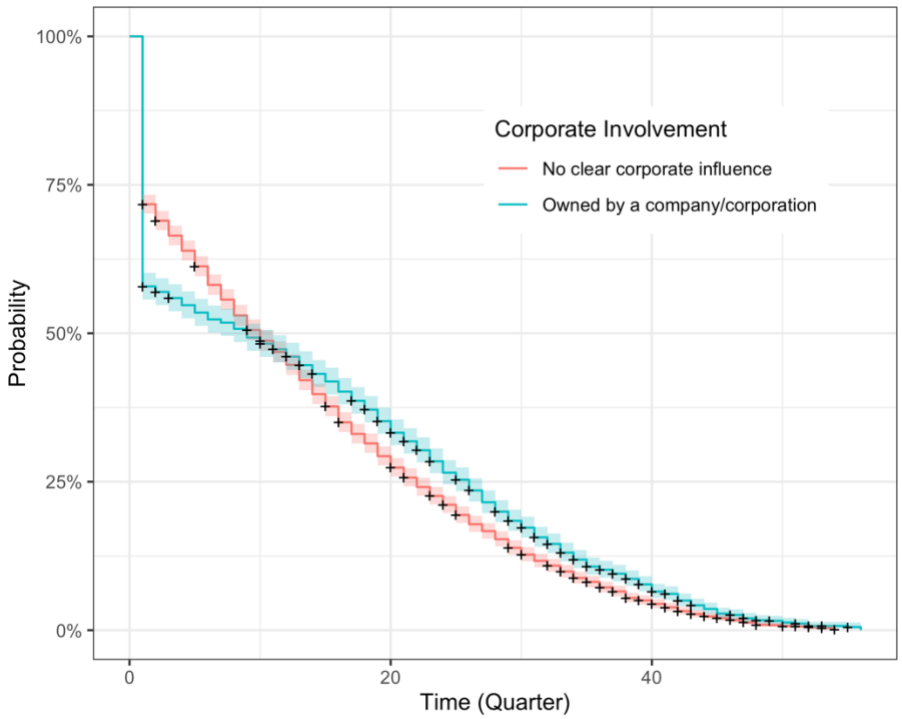


Figure 38 Survival curves based on data grouped by project type.

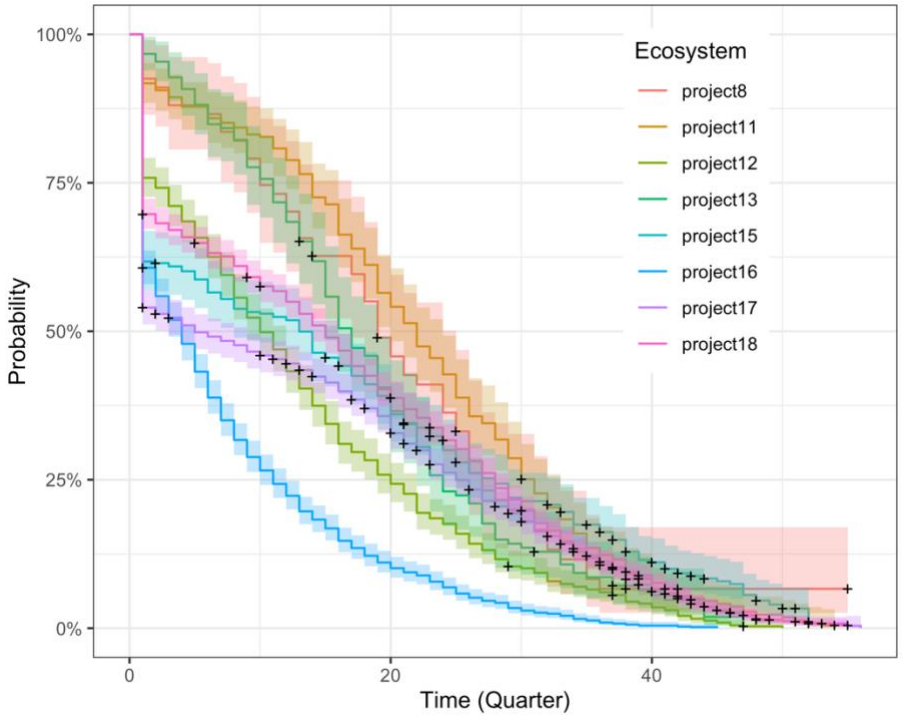


Figure 39 Survival curves based on data grouped by project ecosystem.

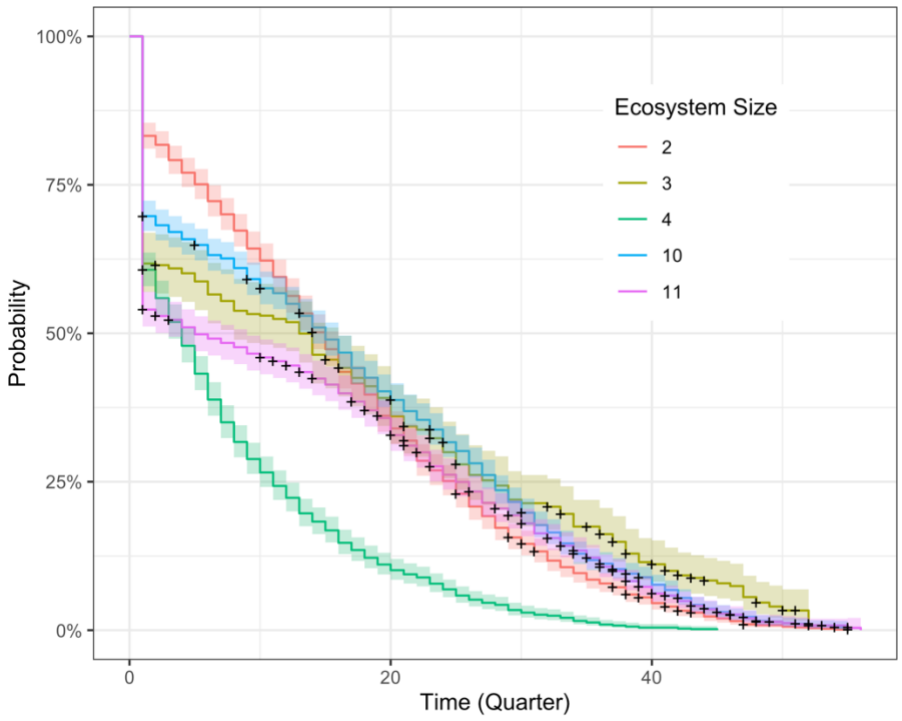


Figure 40 Survival curves based on data grouped by ecosystem size.

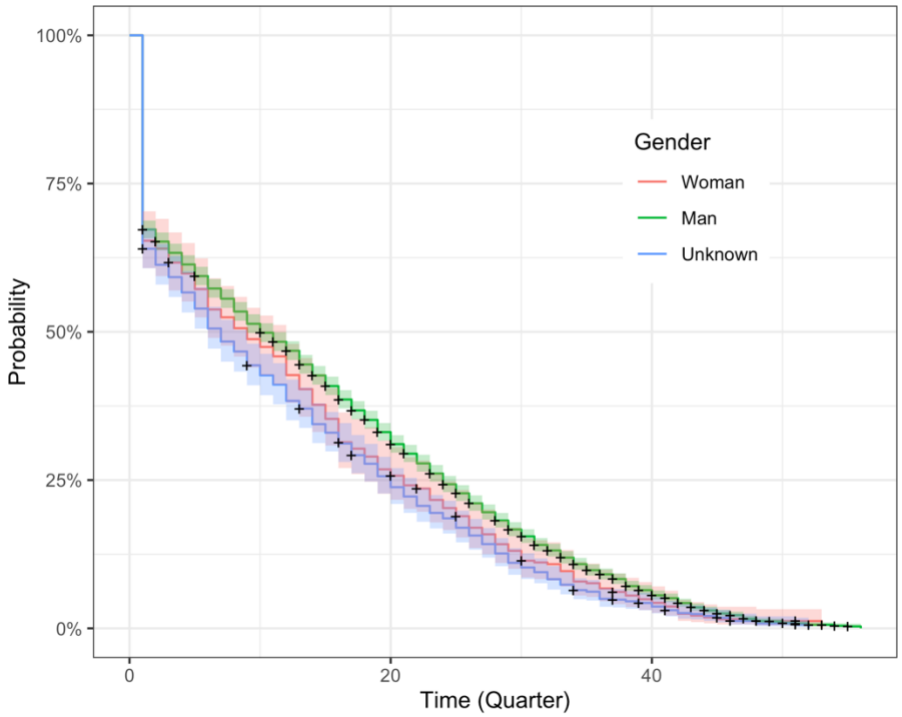


Figure 41 Survival curves based on data grouped by gender.

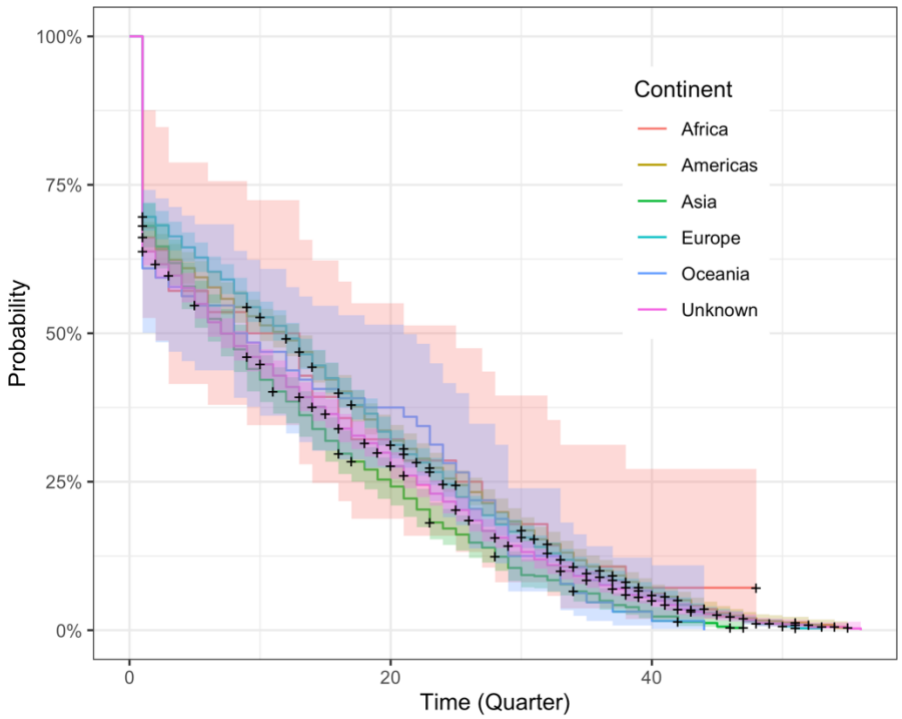


Figure 42 Survival curves based on data grouped by location (continent level).

In sum, the results of hypothesis testing and survival analysis suggest that there are significant and notable differences in participation between projects with and without corporate involvement. Contributors to projects with no clear corporate influence tend to have longer tenure than those who contribute to projects that are owned by a company. Contributors to projects with no clear corporate influence also tend to have a higher probability of sustained participation in the first two years of participation. After two years, the difference in probabilities is reduced such that the risk of leaving is nearly identical between project types. Visualizations of the relationship between gender and participation, and location and participation, do not reveal drastic differences between groups in the sample. In the next section, I describe observed group

composition to characterize similarities and differences in diversity within and between project ecosystems in the sample.

Diversity Differences

ER Q2. Are differences in social diversity in an ecosystem associated with corporate involvement?

Gender and location information in GitHub user profiles was used to examine social diversity in FLOSS ecosystems. These social dimensions of group composition are compared between project types and as they relate to differences in tenure. Contributor gender counts and proportions are provided in Table 27 and Figure 43, respectively. In general, the largest number of contributors is associated with projects that have some corporate support followed by projects that have no clear corporate influence. The proportions of gender groups are consistent across project types. Descriptive statistics for contributor location counts are provided in Table 28. The largest number of different countries is observed in projects with corporate support followed by projects with no corporate influence.

Table 27 Contributor gender counts for project sample overall and by level of corporate involvement. Corporate Involvement is abbreviated to CI.

	No CI (1)	**Some CI (2)**	**Company Owned (3)**	**Overall**
Women	245	471	136	852
Men	2,382	4,982	1,460	8,824
Unknown	524	1,034	254	1,812
Total	3,151	6,487	1,850	11,488

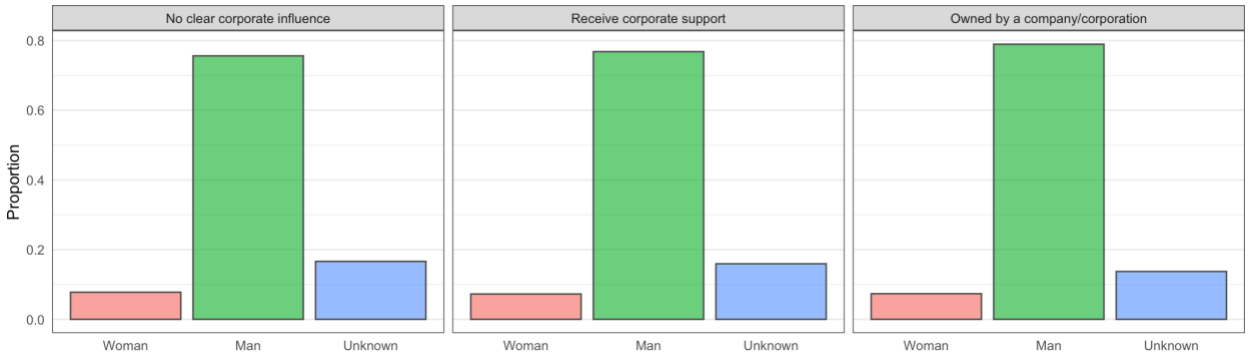


Figure 43 Gender data grouped by project type.

Table 28 Descriptive statistics for the number of countries in sample data grouped by project type and across the entire sample.

	No CI	**Some CI**	**Company Owned**	**All Projects**
Total Count	78	97	62	98
Minimum	2	10	2	2
Maximum	63	86	42	86
Median	19	28	18	24.50
Mean	23.80	35.36	19.58	25.82

Distributions for diversity in projects are provided in Figures 44 and 45 and diversity values are given in Tables 29-30. Gender diversity is generally low in projects (<0.3) while country diversity is high (>0.75). A Welch's unequal variances *t*-test, a modification of the two sample t-test that is appropriate for cases in which there are unequal groups and there is unequal variance between those groups, was applied to the data. No significant difference was found between the two project types of interest, $t(22) = 0.74$, $p = .47$, $d = 0.25$ (lower CI: -0.51; upper CI: 1.01), with no corporate influence projects ($M = 0.16$, $SD = 0.07$) having, on average, similar levels of gender diversity as company-owned projects ($M = 0.14$, $SD = 0.09$). The skewness and

prevalence of similar values (ties) between project types limited the application of statistical tests to country diversity data. Mean and median values for country diversity for project types are provided in Tables 29-30.

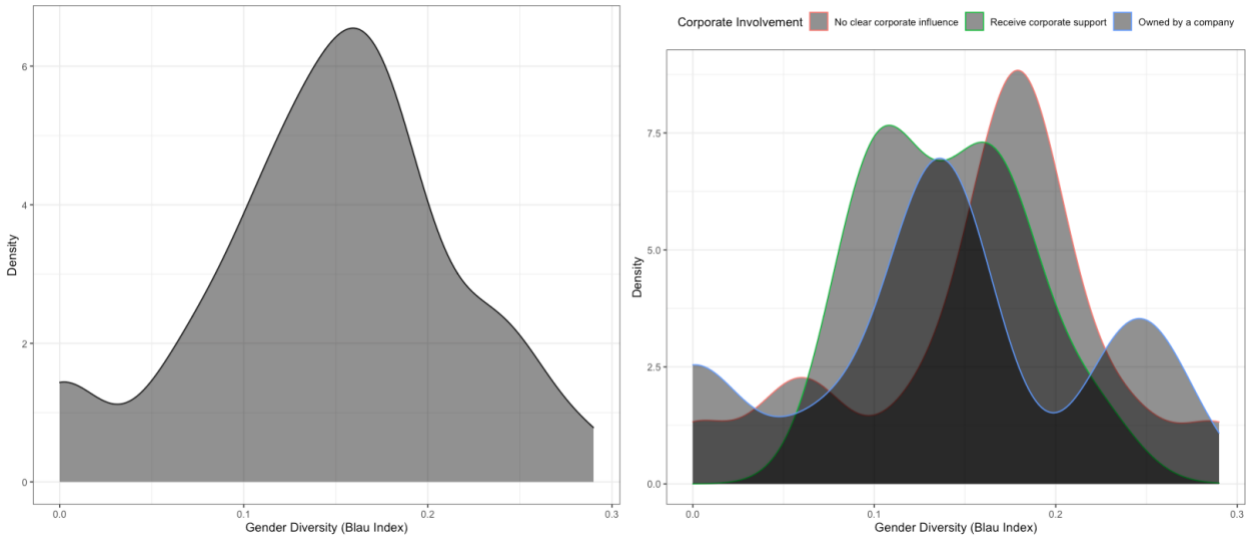


Figure 44 The distribution of gender diversity in the sample (left) and by project type (right).

Table 29 Measures of central tendency for diversity measured using the Blau index [0,1].

	No CI	**Some CI**	**Company Owned**
Mean Gender Diversity	0.16	0.14	0.14
Median Gender Diversity	0.18	0.15	0.14
Mean Country Diversity	0.78	0.84	0.68
Median Country Diversity	0.83	0.86	0.80

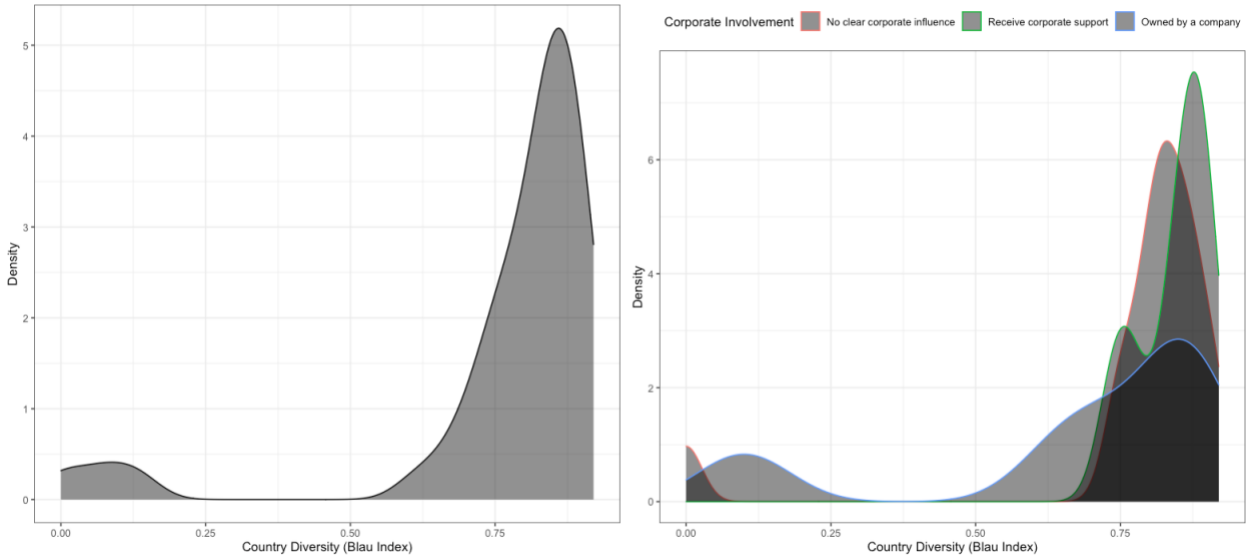


Figure 45 The distribution of country diversity in the sample (left) and by project type (right).

Table 30 Contributor makeup for projects calculated at ecosystem level. Diversity values for gender and country are based on complete data (i.e., users with unknown information were excluded in the calculation of diversity). Both Blau index values and Gini coefficient range from 0 (homogenous/equal) to 1 (heterogenous/unequal).

		Diversity (Blau Index)		***Disparity (Gini)***
ID	**Contributors** (% Women)	**Gender**	**Country**	**Tenure** (Median Quarters)
8	213 (5%)	0.10	0.89	0.41 (13)
11	255 (7%)	0.14	0.80	0.32 (22)
13	152 (5%)	0.11	0.85	0.33 (16.5)
17	4,753 (7%)	0.16	0.87	0.62 (1)
14	225 (4%)	0.11	0.91	0.39 (22)
12	623 (9%)	0.18	0.92	0.47 (11)
15	294 (10%)	0.20	0.83	0.48 (17.5)
16	2,279 (8%)	0.17	0.87	0.50 (10)
18	1,328 (7%)	0.16	0.85	0.45 (16)

ER Q2.1 Which projects have the highest and lowest levels of social diversity in each ecosystem?

All ecosystems and the projects within them have low levels of gender diversity (Table 31). The highest gender diversity is observed in project 15 (0.29), a project with no corporate support. It is the highest compared to all other projects in its ecosystem and in the sample. Three projects are tied for the lowest gender diversity (0.00) in the sample: two are company owned and one has no corporate influence. The project types with the lowest and highest gender diversity are provided in Table 33. Projects with the highest gender diversity were nearly evenly split across no corporate influence and company owned. The results of the statistical test described in the previous section, and the results presented in this section do not provide evidence that there is a meaningful difference in gender diversity on the basis of project types. In other words, neither companies nor communities are cultivating higher levels of diversity in FLOSS ecosystems when compared to each other.

All ecosystems and most of the projects within them have high levels of country diversity. The project types with the lowest and highest country diversity are provided in Table 32. A project with no clear corporate influence in ecosystem 12 has the highest level of country diversity (0.92) and a project with no clear corporate influence in ecosystem 15 has the lowest level of country diversity (0.00). Six out of nine (67%) projects with the highest country diversity in their ecosystems had no corporate influence. While this does not provide strong evidence that projects with no corporate influence tend to be more globally open, it does suggest that there may be differences in country diversity between project types that should be examined in future research. In sum, no project types are more or less open to different gender groups and

projects with no corporate influence may be more open to globally distributed groups. Next, I describe quantitative characteristics of project ecosystems.

Table 31 Project types with the lowest and highest gender diversity.

Ecosystem	No. Projects	Lowest Gender Diversity	Highest Gender Diversity
8	2	Owned by a company	Receive corporate support
11	2	No clear corporate influence	Owned by a company
12	2	No clear corporate influence	No clear corporate influence
13	2	Owned by a company	Owned by a company
14	2	Receive corporate support	Receive corporate support
15	3	No clear corporate influence	No clear corporate influence
16	4	Receive corporate support	No clear corporate influence
17	11	Owned by a company	Owned by a company
18	10	No clear corporate influence	Tie: No corporations and Company owned

Table 32 Project types with the lowest and highest gender diversity.

Ecosystem	No. Projects	Lowest Country Diversity	Highest Country Diversity
8	2	Owned by a company	No clear corporate influence
11	2	Owned by a company	No clear corporate influence
12	2	No clear corporate influence	No clear corporate influence
13	2	Owned by a company	Owned by a company
14	2	Receive corporate support	Receive corporate support
15	3	No clear corporate influence	No clear corporate influence
16	4	Receive corporate support	Tie: No corporations and Corporate support
17	11	Owned by a company	No clear corporate influence
18	10	No clear corporate influence	No clear corporate influence

Quantitative Differences

ER Q3. What are the quantitative characteristics of projects that lack corporate involvement?

A set of multilayered network graphs were generated to examine quantitative characteristics of project ecosystems. Values for two metrics were extracted and evaluated for this: average degree and edge density. First, average degree, or the average number of connections nodes have in a network, provides information about how contributors are distributed across an ecosystem. If most contributors participate in most or all of the projects in an ecosystem, then we can generally expect the network to have larger average degree. Second, network density, or the ratio between edges in a graph and the maximum number of possible edges in a graph, communicates information about connectivity. Edge density has implications for network effects with more powerful effects occurring in highly dense networks and reflects the value of the network (e.g., for the transmission of information, social contagion, etc.).

Unlike network density, average degree varies greatly on the basis of network size and its range corresponds to the number of nodes in the graph. For example, although ecosystem 12 is only made up of 2 projects, its contributor base is larger than other ecosystems containing the same number of projects. Its average degree is therefore much larger than other 2-project ecosystems. Relevant to this study's research questions, this small ecosystem has no corporate influence and has high average degree and edge density compared to small ecosystems that are company owned, receive corporate support, or have a blend of project types. For small ecosystems, the lowest edge density is observed in those that have company-owned or corporate-supported projects while he highest edge density is observed in ecosystems with no corporate influence (Table 33). For larger ecosystems, the lowest edge density is observed in a mixed

project ecosystem with a no corporate influence majority and the highest edge density is observed in ecosystems with corporate support and/or company ownership.

Table 33 Metrics extracted from multilayered network. Corporate involvement is abbreviated to CI. CI values: 3 company/corporate owned; 2 corporate support; 1 no corporate influence.

ID	CI	Average Degree	Density
8	Mixed: 2, 3	121.01	0.14
11	Mixed: 1, 3	192.65	0.19
12	1	610.85	0.25
13	3	105.92	0.17
14	2	162.34	0.19
15	3	278.96	0.24
16	Mixed: 1, 2	1173.06	0.13
17	Mixed: majority 2, 3	2454.52	0.13
18	Mixed: majority 1	252.69	0.04

In sum, the results of the ecosystem analysis provide some evidence for differences in participation and diversity based on corporate involvement in projects. First, ER Q1 focused on differences in tenure and sustained participation. The distribution of tenure in projects that are company owned or have some corporate support have greater skew. This means that they have a large number of contributors that have a very short tenure in the project (i.e., more episodic volunteers). This is also reflected in the comparison for survival curves based on project types indicating that short-term contribution patterns are less common in projects with no corporate influence (i.e., sustained participation is more common). Second, ER Q2 was concerned with levels of social diversity. Some differences in social diversity were observed between project types. Projects with no corporate involvement and projects with corporate support had higher levels of country diversity compared to projects owned by a company, but no clear difference emerged when looking at gender diversity. Third, ER Q3 was concerned with differences in

quantitative characteristics based on ecosystem networks. In small ecosystems, greater network density is observed with there is little or no corporate influence. This is reversed in larger ecosystems where the greatest network density is observed with there is some corporate presence or ownership. This research therefore finds evidence of differences between projects that have no or some corporate engagement and projects that are owned by a company and that variations made be associated with the size of the ecosystem.

Discussion

The goal of this research was to characterize the perceptions of contributors on decision making and beliefs associated with diversity and corporations in FLOSS projects and to examine if and how this is reflected in observable behavior in a social coding platform. The results of the survey research highlight the heterogeneity of FLOSS contributors' beliefs, behaviors, and experiences regarding diversity and corporations. The results of the GitHub data analysis indicate that company-owned FLOSS projects are in general neither significantly better nor worse than community-maintained projects in terms of overall participation trends but may be less open to a diverse set of potential contributors.

A key takeaway for this research is that FLOSS contributors experience important differences between projects based on the presence of corporations in them. However, when looking at behavioral data extracted from FLOSS projects, we see that the presence of a corporation or company is not associated with greater levels of diversity, even though some contributors perceive corporations as having a positive influence on this dimension of group composition. Organizations interested in affecting change in FLOSS projects, and in particular those projects in which they are not primary owners and maintainers, must reflect on why their own internal practices, norms, and/or culture have not already produced diverse collaborations.

Meaningful action and policy relevant to internal dynamics is paramount before any attempt to impose top-down measures on FLOSS communities, which display sensitivity to control, is made in projects. This is essential for not only influencing those communities, but also ensuring the integrity and longevity of diversity as a commitment. Communities of contributors who are especially oriented towards software freedom rather than personal benefits and profit are unlikely to respond positively when they perceive organizational actions as undue or unjustified control. Surface-level actions without meaningful change to organizational policy are seen as empty gestures centered on presenting an image of inclusivity and concern for equity rather than an embodiment of such values. As one survey participant expressed, "*I'm displeased by GitHub's move to rename the "master" branch, I think it's a bullshit statement and it doesn't solve any of our modern-day problems*" [GH11]. Research on diversity and anti-discrimination programs in the workplace have shown that such initiatives, motivated by a desire to protect public image and reduce legal risks, do not only fail to increase diversity, they effectively reduce it further (Dobbin & Kalev, 2022).

The results of the GitHub data analysis suggest that company-owned projects tend to have lower levels of country diversity in their contributor base. Even though country diversity was generally high across all project types, it was limited in certain continents and geographical regions—Africa, Oceania, West Asia, and South America had lower levels of representation compared to North America, Europe, and East Asia. Organizations interested in extending use of FLOSS and participation in projects to a broader group may then want to consider ways that they can lower barriers to contribution for underrepresented populations. A survey participant provided an example of a relevant barrier that limits the achievement of diversity:

> "*While open source is used worldwide, not everyone has the technical ability or time to contribute. Being able to measure diversity in projects, and work to improve it, would be a huge benefit to the broader open source community. For example, much code is written by English-speaking programmers, making it challenging for non-native speakers to follow comments, method / function names, etc."* [SM34].

Addressing such barriers can then help move FLOSS towards greater inclusivity at the global level. In this specific case, organizations can intentionally devote resources to translation, allocating funds to services and paying non-English speakers in underrepresented countries to support the completion of this work. Such an action would not only immediately confer benefits to global labor, it could potentially have cascading effects by providing access to a much broader audience.

Given these findings, I now revisit the notion of ecosystems analyses. My goal is to add to the study of FLOSS development as an important and contemporary form of work. In this way, I can point the way for notional interventions designed to affect phenomena associated with ecosystem (see Table 36). These interventions can target particular needs (e.g., diversity, accessibility, etc.) in the hopes of improving innovation and addressing societal needs. This adaptation of Meadows' leverage points can also be used to guide future research in this area. In short, in order to better understand FLOSS ecosystems, a broader theoretical and methodological integration is needed. This study focused on providing a characterization of system intent, specifically goals and paradigms, that can inform strategies for improving inclusivity and equity in FLOSS projects.

Table 34 Revisiting Meadows' (1999) leverage points, adapted for FLOSS ecosystems.

Level	Places to Intervene	System Characteristics	Places to Intervene in FLOSS
Shallow	Parameters (such as subsidies, taxes, standards)	*Parameters*	Public Policy Publicly Funded Initiatives
	Length of delays, relative to the rate of system change	*Feedbacks*	Products: Accessibility Ease of Use/Modification
	Strength of negative feedback loops		Products: General Specialized
	Gain around driving positive feedback loops		Userbase Expansion: Customers (Profit) Contributors (Code Quality)
Deep	Structure of information flows (access to information)	*Design*	Structure: Platform Transparency/Opaqueness
	Rules of the system		Licensing
	Power to add, change, or self-organize system structure		Platform Design
	Goals of the system	*Intent*	Goals (Agent, Collective): Profit Social Good
	Mindset/paradigm out of which the system arises		Mindset/Paradigms: Freedom Transparency Popularity/Success
	Power to transcend paradigms		Agent (low-medium) Collective (medium-high)

Although the survey samples varied along a number of social and technical dimensions, there was some degree of homogeneity and in particular with regard to gender. Relatedly, the big data sample was similarly dominated by men. This of course means that insights regarding the experiences of underrepresented and marginalized groups in FLOSS are not readily available in the data. However, what this study does afford is an improved understanding of how norms, beliefs, and behaviors of this group of FLOSS contributors maintain the status quo of

participation in tech development. As Hoffmann (2019) notes in their work on bias and fairness in big data: “by centering disadvantage, we fail to question the normative conditions that produce - and promote the qualities or interests of - advantaged subjects” (p. 907). The demographics of the survey samples indicate that some substantial portion of FLOSS contributors hold a social position which is indeed advantaged in Western society. The majority of the sample was White, male, educated, and had income levels at or above the median income level in the USA (Semega & Kollar, 2022). Each of these dimensions is not only relevant to understanding who has greater amounts of time and resources necessary to engage in volunteer labor in FLOSS projects, but also provides a reference to understand the situatedness and positions from which survey participants engaged with study concepts.

The results of the thematic analysis applied to elaborations on the importance or unimportance of diversity can be contextualized from both an understanding of these positions and empirical studies of social factors in FLOSS projects. Some participants expressed the belief that diversity was generally unimportant or particularly unimportant in FLOSS development. Further, participants expressed caring more about (their perceptions of) technical skills, trustworthiness, and social compatibility. Even among participants who viewed diversity as good, the view of code quality, engineering, and/or merit as the primary determining factor persists. However, as noted in earlier chapters, prior research demonstrates that the evaluation of these attributes differs along social dimensions: there is a growing body of evidence that contributors whose gender or race is perceptible and underrepresented tend to have less work accepted in FLOSS projects (Terrell et al., 2017; Nadri et al., 2020). Relating these findings to research on team assembly, beliefs about diversity as unimportant or secondary are reflected in, and reified by, implicit or explicit individual preferences—compositional and relational

mechanisms—which influence the formation of collaborations. These then are constitutive of the normative conditions which continuously produce an advantaged subject in FLOSS development.

The results of the RTA applied to participant responses regarding their experiences and feelings towards corporations in FLOSS development reveal that underlying motivational and reward differences for working in varied contexts. In the first case, a subset of FLOSS contributors delineate the role of their economic needs and management style preferences. These participants note that participation in corporate-involved projects grants them immediate and more distal benefits in the form of compensation and networks, respectively. They additionally note that the management styles employed in projects with corporate involvement ensure that they have access to personnel and resources to enable the completion of their work. In the second case, a subset of FLOSS contributors describe the impediments of management style to their work, making it needlessly laborious through increased bureaucracy and constraining their ability to innovate due to top-down control. Further, they may view the profit-based goals of the corporation to not be in line with their personal values—improving codebases as a greater good––or the broader values of openness and unrestricted use in the free software development paradigm.

This extends prior work in the organizational sciences by elaborating the specific economic, organizational, and work-related factors that drive observed membership change (Mobley, 1982) in technology development, and provides evidence for a set of FLOSS contributor types that can be differentiated along these three dimensions. The importance of these dimensions and contributor types is more salient when connecting them to the development of interventions to improve diversity and inclusion in FLOSS. The former group appears

amenable to top-down interventions under conditions where they are able to continue to derive economic benefits and resources to support this work remain available. The latter group in contrast may reject top-down interventions, particularly when they lack transparency and leadership fails to make explicit how such interventions will produce benefits for code quality. As a number of survey participants described a lack of practical or functional value in diversity, there exists an opportunity for organizations, including GitHub, to develop information resources that bridge this gap in understanding. There is also a need for organizations to develop and amplify awareness around tools and metrics for the evaluation of diversity initiatives. Given the abundance of information in online spaces, the production of these resources and tools is not enough to ensure their efficacy. Additional research is needed to understand how the design of platforms can be modified to structure information flows in favor of increased awareness. As such, the findings from this dissertation provide avenues for further research and/or notional interventions in FLOSS in order to make advances in this important contemporary form of work.

Participant responses to survey questions suggest that the achievement of diversity in FLOSS projects is constrained by two factors: a lack of perceived value and a lack of tools. Although research in other work domains demonstrates the utility of diversity for task outcomes (e.g., innovation), there is a lack of research on FLOSS development which links diversity with team processes and performance. Researchers and organizations alike have an opportunity to investigate this relationship and potentially motivate more apathetic contributors to value diversity in their projects. The rationale provided by participants for the importance of diversity in FLOSS projects provides a starting point from which to design a research program that builds a body of evidence for the social and technical value of compositional differences. For example, researchers can study the discussion of use and edge cases in a project and the emergence of

different types of software bugs later in the project lifetime. The lack of tools for achieving and measuring diversity may appear straightforward but is more complicated, especially when considering how to alter norms around the behavior that makes projects less inclusive and mitigate biases against members of underrepresented groups. To measure diversity, maintainers can make use of existing tools (e.g., analytics dashboards developed by CHAOSS). To achieve diversity though, it is necessary to analyze how existing contributor behavior deters the participation of some groups and identify strategies to reduce the likelihood of such behavior. Such an analysis can inform the development of moderation tools that are well-suited for social coding platforms. Project maintainers may then be better equipped to increase diversity in FLOSS development.

The adoption of FLOSS development practices by organizations and the growing number of collaborative multi-organizational/institutional collaborative initiatives may indeed be useful in encouraging a more inclusive, global workforce. Yet, organizations invested in making change must practice care in pursuing such collaborations. In particular, the responses of survey participants suggest that some corporations lack social capital with respect to FLOSS communities (contributors and users). Therefore, it is important for organizations to be judicious in forming collaborations. Further, they should heed the advice given to those seeking to participate in FLOSS development: thoroughly vet the history of potential collaborators with affected communities. Acquisitions in FLOSS ecosystems can also be quite disruptive depending on the social-normative attitudes held by the FLOSS community and may produce a loss with respect to advances made towards greater inclusivity and diversity. For example, Microsoft has a contentious relationship with FLOSS communities, being an early opponent of free and open source licensing. Following Microsoft's acquisition of GitHub, some survey participants shared

that they migrated away from the platform, while others braced for an inevitable departure, taking a 'wait and see' attitude as to when corporate ownership would impinge on their values.

Limitations

The studies described here had a number of limitations that should be addressed in future research. First, regarding the survey sample size, the low number of respondents limited the interpretability of the findings. Although other studies in this and related literature had similarly sized samples, the nature of the questions would be best addressed with a larger sample. Second, related to the sample, there were differences based upon country of origin. Although this did provide some interesting insights, future research would need to cast a broader net by recruiting from the community more directly and internationally, as well as with a broader and more generic call. Third, relevant to this, this research is limited to an evaluation of US-based organizational decisions and policies and their effects on participation and diversity. As one of the participants in the committers sample noted in a free comment box at the end of the survey, "*As a European, I found the news to be too US-centric*" [GH15]. To take a more global perspective on FLOSS, and account for how corporations and cultures may interact with attitudes in this ecosystem, not only would the sample need to be larger, but the questions in the survey would need to be more representative of international participants. Future research can query FLOSS contributors about programs and initiatives that originate in other regions in addition to collaborative international efforts.

In terms of awareness and information gathering related to group diversity, the survey research is limited in that only provides insight about what sources are used by FLOSS contributors without elaboration on how those sources vary in the degree to which they support awareness. The results of this research could provide a more complete picture of contributor

awareness through the addition of survey questions which specifically focus on how the perceptibility of social information varies between platforms and communication channels. Research on contribution in FLOSS projects finds that participation is distributed across a set of online platforms and communication channels. It is therefore important to determine the particular features of collaborative work tools that afford an assessment of diversity in combination with how different FLOSS contributors use, or do not use, those tools for this purpose.

Finally, an important limitation of this study was a lack of data on feedback loops within the FLOSS ecosystem. This is needed to clarify their relationship with participation and the expansion of a project's contributor base. In FLOSS development, products are a source of feedback in that their accessibility and usability for different groups and applications can expand or constrain the contributor base. By including feedback in analyses like those presented here, we can have outcome indicators more directly mapped to success or failure in meeting user needs. Systematically mapping product uses and applications and their relation differences in user and contributor bases may help reveal key traits of products that lead to increased innovation and elucidate existing gaps in tech that can be targeted to better serve underrepresented and marginalized groups.

CHAPTER 7. CONCLUSION

This dissertation highlights the multilevel nature of participation dynamics in FLOSS and specifically the need to more carefully study the variations between projects that arise along organizational, technical, social, and moral dimensions. The results of Study 1 demonstrated that differences in task complexity and technical aspects of group composition mediate the negative effects of turnover on productivity. Study 2 provided evidence for the association between social aspects of group composition and rates of turnover, and further clarified gender differences in long-term and episodic participation in FLOSS projects. Study 3 examined differences in participation and diversity between projects with different organizational structures and control mechanisms, showing that they vary in terms of both perceived and actual openness and inclusivity.

Study 1 focused on overall developer turnover rates and sought to examine how technical barriers operationalized as task complexity predict participation in addition to how the distribution of experienced contributors in a project is associated with productivity. A series of related results emerged from an analysis of FLOSS project data, both replicating and extending prior research: high turnover was associated with lower levels of productivity due to knowledge loss; inequality in the distribution of experience (few experts and many newcomers) was associated with lower levels of productivity due to the low contribution rates of newcomers; and high task component complexity was associated with lower levels of productivity likely as the result of difficulty associated with high complexity. However, examining the interaction of these factors revealed a more complicated relationship between them and productivity. The negative effect of turnover-induced knowledge loss was reversed when experience was unequally distributed, and component complexity was high. This finding suggests that the presence of more

experienced developers who are able to effectively coordinate work to integrate the contributions of newcomers in existing codebases is associated with high levels of productivity when turnover is high in FLOSS projects.

Study 2 was similarly focused on experience, participation, and turnover with an additional consideration of contributor gender and social aspects of group composition. The results of this study strengthened claims in existing research showing lower levels of participation among women in FLOSS development and contributed to the body of evidence that women typically disengage from the GitHub platform before men. This research extended understanding of gender differences in participation dynamics, finding that both long-term and ephemeral participation is less common and occurs at lower rates among women when compared to men. Lastly, this study showed that the gender makeup of a contributor group is predictive of turnover, with more homogenous groups being associated with higher levels of turnover.

Study 3 moved beyond a project-level analysis of the aforementioned research concepts to account for both project and ecosystem level features associated with participation and social aspects of group composition in FLOSS development. Furthermore, this study included a qualitative component to characterize developer perceptions of study concepts and a quantitative component to model the relationship between project and ecosystem features and observed participation behavior. Through an analysis of corporate presence in FLOSS projects, this study examined differences resulting from top-down control and bottom-up decision making. The qualitative component of this research revealed subgroups among FLOSS contributors in terms of values and goals: study participants differentially value diversity and are primarily guided by a desire to reap personal benefits or maintain codebases as a social good. The results of the quantitative component research suggest that variations in participation occur on the basis of

corporate involvement in projects, with both short-term and long-term participation rates being higher in projects that are not owned by a company. While there was no evidence of differences along the dimension of gender, some evidence was found that projects with no corporate influence exhibit higher levels of openness along the dimension of geographical location as they had among the highest levels of country diversity. This study demonstrates that to differentially influence participation and both encourage and maintain diversity in FLOSS projects, it is important to develop strategies that are appropriate for the sociopolitical makeup of the contributor base.

Taken together, these studies provide a multilevel view of participation of FLOSS projects by modeling the individual, group, and collective factors that influence behavior in online platforms. This research aims to shift away from a completely localized view of participation dynamics in FLOSS development to additionally account for factors that are exogenous to a specific project but also influence participation. Taking a systems perspective of participation in FLOSS development enables an analysis of the interaction between phenomena at different levels to build a more holistic view of tech production and opportunities for effecting change within it. Based on this approach, I have identified critical differences in system intent, both for individua agents and collectives, that can inform future research and the evaluation of interventions for social good in the production of technology.

APPENDIX A: CONTRIBUTOR PERCEPTIONS SURVEY

Perceptions of Diversity

When you participate, or are considering participating, in an open source/FLOSS project, are you aware of the diversity of the group of people who contribute to it?

Likert-type scale: 1 (definitely not) to 5 (definitely yes)

What information sources in online platforms (e.g., user profile, comments, etc.) support your awareness of diversity in FLOSS projects? Enter NA if there are none/you do not attend to this information.

Free response text

In your experience, how easy or difficult has it been for you to assess the diversity of the group of people who contribute to an open source project?

Likert-type scale: 1 (extremely easy) to 7 (extremely difficult)

What types of diversity have you been aware of in open source projects?

Check all that apply: age, gender, nationality, race, ethnicity, education level, expertise level, socioeconomic status, disability, other (please specify)

How important is diversity in FLOSS projects to you?

Likert-type scale: 1 (not at all important) to 5 (extremely important)

Why is/isn't diversity important to you?

Free response

What types of diversity in FLOSS projects are important to you?

Check all that apply: age, gender, nationality, race, ethnicity, education level, expertise level, socioeconomic status, disability, other (please specify)

How much does the level and type of diversity in a FLOSS project influence your decision to contribute to it?

Likert-type scale: 1 (not at all) to 5 (a great deal)

Perceptions of Corporate Involvement

Please specify the percentage of your contributions to different types of open source projects. For example, if you contribute to 3 projects that have corporate involvement and 1 project that does not have corporate involvement, you would enter 75 in the first entry box and 25 in the second entry box.

Projects with corporate involvement:___

Projects that do not have corporate involvement:____

Does your contribution experience differ for open source projects with and without corporate involvement?

Likert-type scale: 1 (never) to 5 (always)

Please use the text box below to write 1-3 sentences explaining your response.

How much does the lack or presence of corporate involvement influence your decision to continue participating in a project after initial contribution?

Likert-type scale: 1 (not at all influential) to 5 (extremely influential)

Please use the text box below to write 1-3 sentences explaining your response.

Do you maintain awareness of the activities of corporations outside of the open source projects they are involved in?

Likert-type scale: 1 (never) to 5 (always)

What types of information associated with a corporation are you aware of?

Check all that apply: product releases; employment opportunities; acquisitions; collaborations with other organizations; collaborations with communities; other (please specify)

Next you will be shown a series of headlines and asked to indicate your awareness of the event.

Likert-type scale: 1 (not at all aware) to 5 (extremely aware)

Stimuli consist of headlines describing current events associated with FLOSS development

Microsoft acquisition of GitHub; DARPA-Linux joint initiative for R&D; GitHub decision to maintain contract with ICE; GitHub fires Jewish employee for using the term Nazi then offers job back; GitHub Co-pilot preview;

Did any of these events or others influence your decision to participate or continue participating in an open source project?

Yes/No

Specify events that influenced decision to begin participating

Specify events that influenced decision to end participation

Did any of these events or others influence your decision to use or continue using an online platform/tool (e.g., GitHub)?

Yes/No

Specify events that influenced decision to begin use

Specify events that influenced decision to end use

Demographics

Age:____

Gender:____

Race/Ethnicity:____

What is your country of residence?____

What is your highest level of education?

Annual income:____

Employment status:____

Primary occupation:____

Involvement in FLOSS development

Expertise

Number of years in FLOSS development:____

Number of open source projects contributed to (overall):____

Number of open source projects currently contributing to:____

Role in projects and contribution types

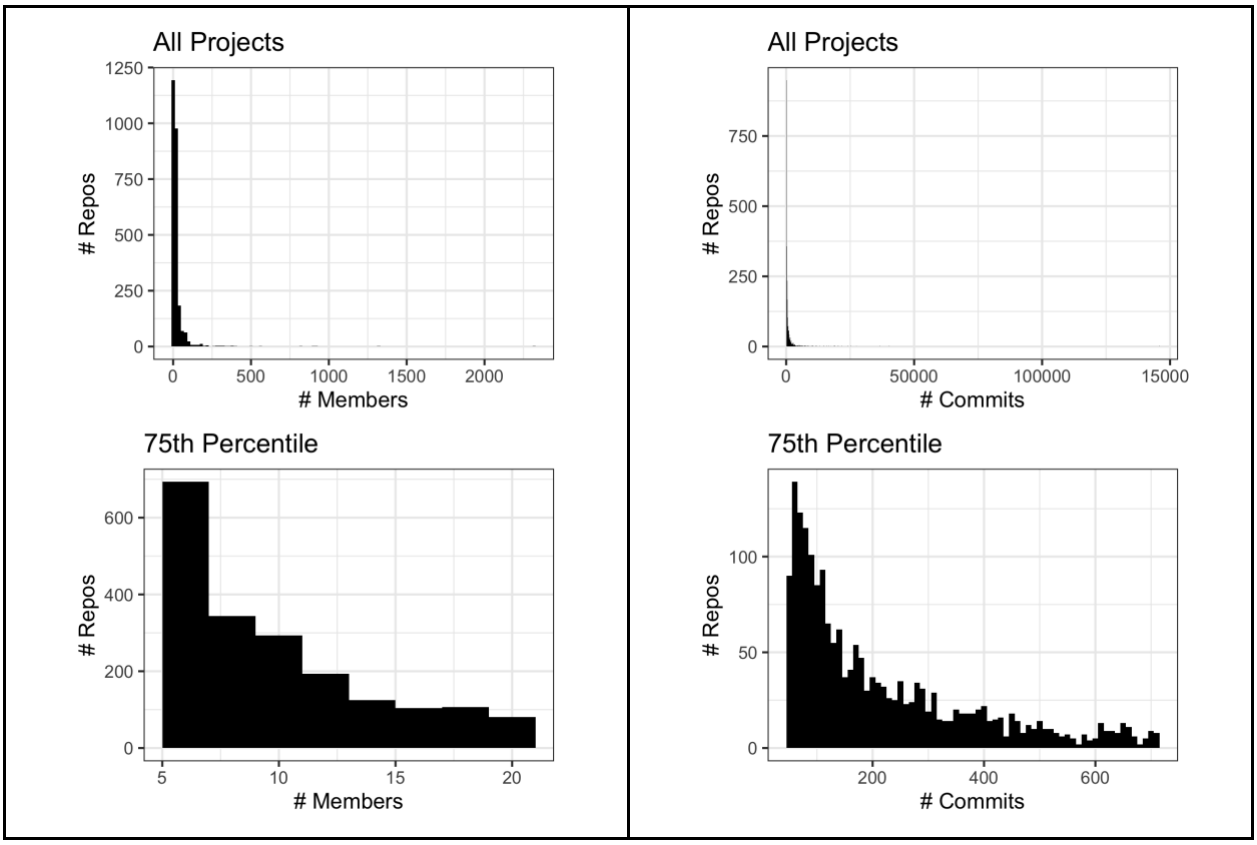
What platforms do you use to participate in FLOSS development:

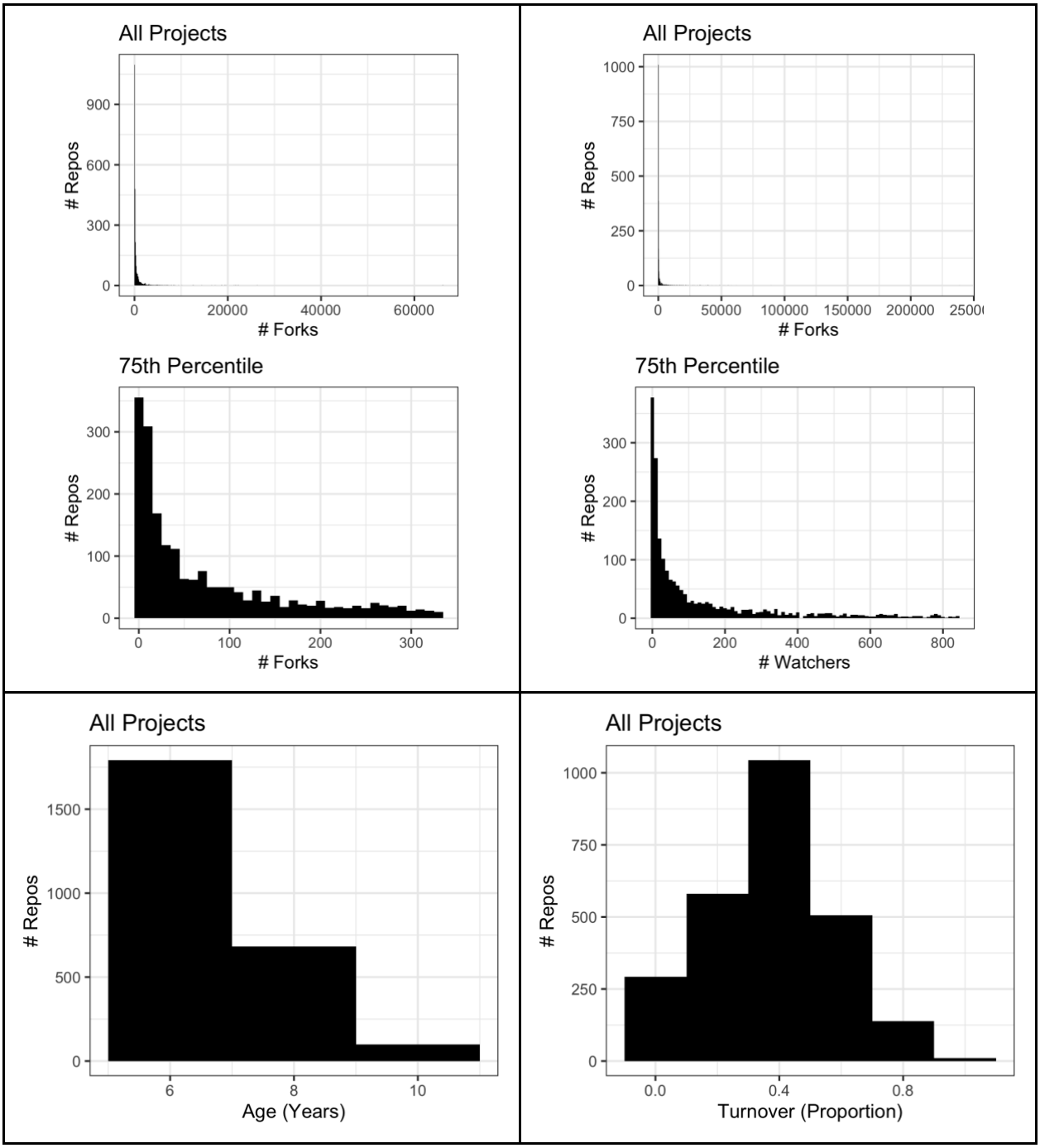
GitHub, GitLab, BitBucket, SourceForge, Other (please specify)

Motivation

Do you receive direct compensation (e.g., salary, contract) for your participation in the project? Yes/No

APPENDIX B: STUDY POPULATION DESCRIPTIVES IN ECOSYSTEM ANALYSIS


All Projects
Repos
Members
75th Percentile
Repos
Members
All Projects
Repos
Commits
75th Percentile
Repos
Commits


All Projects
Repos
Forks
All Projects
Repos
Forks
75th Percentile
Repos
Forks
75th Percentile
Repos
Watchers
All Projects
Repos
Age (Years)
All Projects
Repos
Turnover (Proportion)

APPENDIX C: INFORMATION FOR ECOSYSTEM RESEARCH SAMPLE

List of Ecosystem Projects for Each Project in Sample. NA indicates that no projects were identified using cross-references in repository comment data.

ID	Project Owner	Project Name	Ecosystem
1	spring-cloud	spring-cloud-cloudfoundry	NA
2	t-crest	patmos	NA
3	Wirecloud	wirecloud	NA
4	invisserver	invisAD-setup	NA
5	Midnight Commander	mc	NA
6	seotoaster-team	seotoaster	NA
7	azavea	opendataphilly-ckan	NA
8	wmixvideo	nfe	junit-team/junit4
9	ctrl-alt-d	django-aula	NA
10	Novik	ruTorrent	NA
11	rainforestapp	rainforest-cli	urfave/cli
12	sonata-project	ecommerce	sonata-project/SonataAdminBundle
13	atlasapi	atlas-persistence	google/gson
14	OpenImageIO	oiio	imageworks/OpenShadingLanguage
15	galaxyproject	planemo	galaxyproject/galaxy bgruening/notebooks
16	hyperspy	hyperspy	matplotlib/ipympl matplotlib/matplotlib sympy/sympy
17	projectcalico	calico	go-logr/zapr kubernetes/kubernetes kubernetes/kops, metallb/metallb, projectcalico/bird projectcalico/logrus

			Sirupsen/logrus onsi/ginkgo onsi/gomega tigera/operator
18	cgeo	cgeo	drewnoakes/metadata-extractor jhy/jsoup google/guava mapsforge/vtm mapsforge/mapsforge ReactiveX/RxJava spotbugs/spotbugs square/okhttp square/leakcanary

APPENDIX D: MANN-WHITNEY *U* TESTS FOR TENURE

Tenure (measured in quarters) is greater for projects which are company- or corporate-owned (*Mdn* = 9) than for projects that receive corporate support (*Mdn* = 1), $U = 4{,}897{,}450$, $p < 0.001$. Tenure (measured in quarters) is greater for projects with no corporate influence (*Mdn* = 10) than for projects that receive corporate support (*Mdn* = 1), $U = 1{,}2841{,}758$, $p < 0.001$.

APPENDIX E: IRB EXEMPT STATUS

Olivia Newton

Subject: MOD00001756 has been approved
Date: Friday, April 2, 2021 at 12:52:08 PM Eastern Daylight Time
From: irb@ucf.edu
To: Olivia Newton

Template:IRB_T_Post-Review_Approved

Notification of Approval

To: Olivia Newton

Link: MOD00001756

P.I.: Olivia Newton

Title: Membership Change in Open Source Software Ecosystems

Description: This submission has been approved. You can access the correspondence letter using the following link:

Correspondence_for_MOD00001756.pdf(0.01)

To review additional details, click the link above to access the project workspace.

UNIVERSITY OF CENTRAL FLORIDA

Institutional Review Board
FWA00000351
IRB00001138, IRB00012110
Office of Research
12201 Research Parkway
Orlando, FL 32826-3246

EXEMPTION DETERMINATION

April 2, 2021

Dear Olivia Newton:

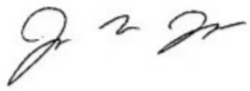
On 4/2/2021, the IRB determined the following submission to be human subjects research that is exempt from regulation:

Type of Review:	Modification / Update
Title:	A Multilevel Model of Membership Change in Open Source Software Ecosystems
Investigator:	Olivia Newton
IRB ID:	MOD00001756
Funding:	Name: GRADUATE STUDIES
Grant ID:	None
Documents Reviewed:	• Explanation of Research, Category: Consent Form; • Invitation to Participate, Category: Recruitment Materials; • Request for Exemption Form, Category: IRB Protocol; • Surveys/Questionnaires, Category: Survey / Questionnaire;

This determination applies only to the activities described in the IRB submission and does not apply should any changes be made. If changes are made, and there are questions about whether these changes affect the exempt status of the human research, please submit a modification request to the IRB. Guidance on submitting Modifications and Administrative Check-in are detailed in the Investigator Manual (HRP-103), which can be found by navigating to the IRB Library within the IRB system. When you have completed your research, please submit a Study Closure request so that IRB records will be accurate.

If you have any questions, please contact the UCF IRB at 407-823-2901 or irb@ucf.edu. Please include your project title and IRB number in all correspondence with this office.

Sincerely,



Racine Jacques, Ph.D.
Designated Reviewer

REFERENCES

Abson, D. J., Fischer, J., Leventon, J., Newig, J., Schomerus, T., Vilsmaier, U., von Wehrden, H., Abernethy, P., Ives, C. D., Jager, N. W., & Lang, D. J. (2017). Leverage points for sustainability transformation. *Ambio*, *46*(1), 30–39. https://doi.org/10.1007/s13280-016-0800-y

Aljemabi, M. A., & Wang, Z. (2018). Empirical study on the evolution of developer social networks. *IEEE Access*, *6*, 51049–51060. https://doi.org/10.1109/ACCESS.2018.2868427

Anklam, P. (2005). Knowledge management: The collaboration thread. *Bulletin of the American Society for Information Science and Technology*, *28*(6), 8–11. https://doi.org/10.1002/bult.254

Argote, L., Aven, B. L., & Kush, J. (2018). The effects of communication networks and turnover on transactive memory and group performance. *Organization Science*, *29*(2), 191–206. https://doi.org/10.1287/orsc.2017.1176

Aue, J., Haisma, M., Tomasdottir, K. F., & Bacchelli, A. (2016). Social diversity and growth levels of open source software projects on GitHub. *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '16*, 1–6. https://doi.org/10.1145/2961111.2962633

Bach, P. M., & Terry, M. (2010). The future of FLOSS in CHI research and practice. *Proceedings of the 28th of the International Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '10*, 4473. https://doi.org/10.1145/1753846.1754177

Balali, S., Steinmacher, I., Annamalai, U., Sarma, A., & Gerosa, M. A. (2018). Newcomers'

barriers. . . is that all? An analysis of mentors' and newcomers' barriers in OSS projects. *Computer Supported Cooperative Work (CSCW)*, *27*(3–6), 679–714. https://doi.org/10.1007/s10606-018-9310-8

Barcomb, A., Kaufmann, A., Riehle, D., Stol, K.-J., & Fitzgerald, B. (2018). Uncovering the periphery: A qualitative survey of episodic volunteering in free/libre and open source software communities. *IEEE Transactions on Software Engineering*, 18. https://doi.org/10.1109/TSE.2018.2872713

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*(1), 1–48. https://doi.org/10.18637/jss.v067.i01

Begel, A., Bosch, J., & Storey, M.-A. (2013). Social networking meets software development: Perspectives from GitHub, msdn, Stack Exchange, and topcoder. *IEEE Software*, *30*(1), 52–66.

Blincoe, K., Harrison, F., & Damian, D. (2015). Ecosystems in GitHub and a method for ecosystem identification using reference coupling. *Proceedings of the 12th Working Conference on Mining Software*, 202–211. https://doi.org/10.1109/MSR.2015.26

Blincoe, K., Springer, O., & Wrobel, M. R. (2019). Perceptions of gender diversity's impact on mood in software development teams. *IEEE Software*, 1–1. https://doi.org/10.1109/MS.2019.2917428

Braun, V., & Clarke, V. (2019). Reflecting on reflexive thematic analysis. *Qualitative Research in Sport, Exercise and Health*, *11*(4), 589–597. https://doi.org/10.1080/2159676X.2019.1628806

Cambon, J., Hernangómez, D., Belanger, C., & Possenriede, D. (2021). tidygeocoder: An R

package for geocoding. *Journal of Open Source Software*, *6*(65), 3544. https://doi.org/10.21105/joss.03544

Canfora, G., Di Penta, M., Oliveto, R., & Panichella, S. (2012). Who is going to mentor newcomers in open source projects? *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering - FSE '12*, 1. https://doi.org/10.1145/2393596.2393647

Casalnuovo, C., Vasilescu, B., Devanbu, P., & Filkov, V. (2015). Developer onboarding in GitHub: The role of prior social links and language experience. *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2015*, 817–828. https://doi.org/10.1145/2786805.2786854

Chou, S.-W., & He, M.-Y. (2011). The factors that affect the performance of open source software development - the perspective of social capital and expertise integration: Performance of OSS development. *Information Systems Journal*, *21*(2), 195–219. https://doi.org/10.1111/j.1365-2575.2009.00347.x

Clark, J. (2016, June 23). Artificial intelligence has a "sea of dudes" problem. *Bloomberg*. https://www.bloomberg.com/news/articles/2016-06-23/artificial-intelligence-has-a-sea-of-dudes-problem

Coleman, E. G. (2013). *Coding freedom: The ethics and aesthetics of hacking*. Princeton University Press.

Constantinou, E., & Mens, T. (2017). Socio-technical evolution of the Ruby ecosystem in GitHub. *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 34–44. https://doi.org/10.1109/SANER.2017.7884607

Crowston, K., & Howison, J. (2005). The social structure of free and open source software

development. *First Monday*, *10*(2). https://doi.org/10.5210/fm.v10i2.1207

Dabbish, L., Stuart, C., Tsay, J., & Herbsleb, J. (2012). Social coding in GitHub: Transparency and collaboration in an open software repository. *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work - CSCW '12*, 1277. https://doi.org/10.1145/2145204.2145396

Dahlander, L., & Magnusson, M. G. (2005). Relationships between open source software companies and communities: Observations from Nordic firms. *Research Policy*, *34*(4), 481–493. https://doi.org/10.1016/j.respol.2005.02.003

Daniel, S., Agarwal, R., & Stewart, K. J. (2013). The effects of diversity in global, distributed collectives: A study of open source project success. *Information Systems Research*, *24*(2), 312–333. https://doi.org/10.1287/isre.1120.0435

Daniel, S. L., Maruping, L. M., Cataldo, M., & Herbsleb, J. (2018). The impact of ideology misfit on open source software communities and companies. *MIS Quarterly*, *42*(4), 1069–1096. https://doi.org/10.25300/MISQ/2018/14242

Dess, G. G., & Shaw, J. D. (2001). Voluntary turnover, social capital, and organizational performance. *Academy of Management Review*, *26*(3), 446–456.

Dobbin, F., & Kalev, A. (2022). *Getting to diversity: What works and what doesn't.* The Belknap Press of Harvard University Press.

Dorfman, R. (1979). A formula for the Gini Coefficient. *The Review of Economics and Statistics*, *61*(1), 146. https://doi.org/10.2307/1924845

Ehmke, C. A. (2014). *Contributor Covenant Code of Conduct*. Contributor Covenant. https://www.contributor-covenant.org/version/1/4/code-of-conduct.html

El Asri, I., & Kerzazi, N. (2019). Where are females in OSS projects? Socio technical

interactions. In L. M. Camarinha-Matos, H. Afsarmanesh, & D. Antonelli (Eds.), *Collaborative Networks and Digital Transformation* (Vol. 568, pp. 308–319). Springer International Publishing. https://doi.org/10.1007/978-3-030-28464-0_27

Elliott, M. S. (2003). The virtual organizational culture of a free software development community. *Proceedings of the Third Workshop on Open Source Software*, 5.

Espinosa, J. A., Slaughter, S. A., Kraut, R. E., & Herbsleb, J. D. (2007a). Team knowledge and coordination in geographically distributed software development. *Journal of Management Information Systems*, *24*(1), 135–169. https://doi.org/10.2753/MIS0742-1222240104

Espinosa, J. A., Slaughter, S. A., Kraut, R. E., & Herbsleb, J. D. (2007b). Familiarity, complexity, and team performance in geographically distributed software development. *Organization Science*, *18*(4), 613–630. https://doi.org/10.1287/orsc.1070.0297

Fiore, S. M. (2008). Interdisciplinarity as teamwork: How the science of teams can inform team science. *Small Group Research*, *39*(3), 251–277. https://doi.org/10.1177/1046496408317797

Fitzgerald, B. (2006). The transformation of open source software. *MIS Quarterly*, *30*(3), 587. https://doi.org/10.2307/25148740

FLOSS and FOSS. (2021, September 11). GNU Project. https://www.gnu.org/philosophy/floss-and-foss.en.html

Foucault, M., Palyart, M., Blanc, X., Murphy, G. C., & Falleri, J.-R. (2015). Impact of developer turnover on quality in open-source software. *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2015*, 829–841. https://doi.org/10.1145/2786805.2786870

Fronchetti, F., Wiese, I., Pinto, G., & Steinmacher, I. (2019). What attracts newcomers to onboard on OSS Projects? TL;DR: Popularity. In F. Bordeleau, A. Sillitti, P. Meirelles, & V. Lenarduzzi (Eds.), *Open Source Systems* (Vol. 556, pp. 91–103). Springer International Publishing. https://doi.org/10.1007/978-3-030-20883-7_9

Fuller, S. (1988). *Social epistemology*. Indiana University Press.

Geertz, C. E. (1973). Thick description: Toward an interpretive theory of culture. In *The Interpretation of Cultures: Selected Essays* (pp. 3–30). Basic Books.

Geiger, R. S., Varoquaux, N., Mazel-Cabasse, C., & Holdgraf, C. (2018). The types, roles, and practices of documentation in data analytics open source software libraries: A collaborative ethnography of documentation work. *Computer Supported Cooperative Work (CSCW)*, *27*(3–6), 767–802. https://doi.org/10.1007/s10606-018-9333-1

Germonprez, M., Allen, J. P., Warner, B., Hill, J., & McClements, G. (2013). Open source communities of competitors. *Interactions*, *20*(6), 54–59. https://doi.org/10.1145/2527191

Germonprez, M., Kendall, J. E., Kendall, K. E., Mathiassen, L., Young, B., & Warner, B. (2017). A theory of responsive design: A field study of corporate engagement with open source communities. *Information Systems Research*, *28*(1), 64–83. https://doi.org/10.1287/isre.2016.0662

Germonprez, M., Lipps, J., & Goggins, S. (2019). The rising tide: Open source's steady transformation. *First Monday*. https://doi.org/10.5210/fm.v24i8.9297

Golzadeh, M., Decan, A., Legay, D., & Mens, T. (2021). A ground-truth dataset and classification model for detecting bots in GitHub issue and PR comments. *Journal of Systems and Software, 175*, 110911. https://doi.org/10.1016/j.jss.2021.110911

Gousios, G., & Spinellis, D. (2012). GHTorrent: Github's data from a firehose. *2012 9th IEEE*

Working Conference on Mining Software Repositories (MSR), 12–21. https://doi.org/10.1109/MSR.2012.6224294

Gousios, G., Storey, M.-A., & Bacchelli, A. (2016). Work practices and challenges in pull-based development: The contributor's perspective. *Proceedings of the 38th International Conference on Software Engineering*, 285–296. https://doi.org/10.1145/2884781.2884826

Gralha, C., Goulao, M., & Araujo, J. (2019). Analysing gender differences in building social goal models: A quasi-experiment. *2019 IEEE 27th International Requirements Engineering Conference (RE)*, 165–176. https://doi.org/10.1109/RE.2019.00027

Haraway, D. (1988). Situated knowledges: The science question in feminism and the privilege of partial perspective. *Feminist Studies*, *14*(3), 575. https://doi.org/10.2307/3178066

Harris, A. M., Gómez-Zará, D., DeChurch, L. A., & Contractor, N. S. (2019). Joining together online: the trajectory of CSCW scholarship on group formation. *Proceedings of the ACM on Human-Computer Interaction*, *3*(CSCW), 1–27. https://doi.org/10.1145/3359250

Harrison, D. A., & Klein, K. J. (2007). What's the difference? Diversity constructs as separation, variety, or disparity in organizations. *Academy of Management Review*, *32*(4), 1199–1228. https://doi.org/10.5465/amr.2007.26586096

Hausknecht, J. P., & Holwerda, J. A. (2013). When does employee turnover matter? Dynamic member configurations, productive capacity, and collective performance. *Organization Science*, *24*(1), 210–225. https://doi.org/10.1287/orsc.1110.0720

Hilderbrand, C., Perdriau, C., Letaw, L., Emard, J., Steine-Hanson, Z., Burnett, M., & Sarma, A. (2020). Engineering gender-inclusivity into software: ten teams' tales from the trenches. *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering,*

433–444. https://doi.org/10.1145/3377811.3380371

Hiller, N. J., DeChurch, L. A., Murase, T., & Doty, D. (2011). Searching for outcomes of leadership: A 25-Year review. *Journal of Management, 37*(4), 1137–1177. https://doi.org/10.1177/0149206310393520

Himmelsbach, J., Schwarz, S., Gerdenitsch, C., Wais-Zechmann, B., Bobeth, J., & Tscheligi, M. (2019). Do we care about diversity in human computer interaction: A comprehensive content analysis on diversity dimensions in research. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, 1–16. https://doi.org/10.1145/3290605.3300720

Hoffmann, A. L. (2019). Where fairness fails: Data, algorithms, and the limits of antidiscrimination discourse. *Information, Communication & Society*, *22*(7), 900–915. https://doi.org/10.1080/1369118X.2019.1573912

Huang, S., & Cummings, J. N. (2011). when critical knowledge is most critical: Centralization in knowledge-intensive teams. *Small Group Research, 42*(6), 669–699. https://doi.org/10.1177/1046496411410073

Imtiaz, N., Middleton, J., Chakraborty, J., Robson, N., Bai, G., & Murphy-Hill, E. (2019). Investigating the effects of gender bias on GitHub. *Proceedings of the 41st International Conference on Software Engineering*, 700–711. https://doi.org/10.1109/ICSE.2019.00079

James, T., Galster, M., Blincoe, K., & Miller, G. (2017). What is the perception of female and male software professionals on performance, team dynamics and job satisfaction? Insights from the trenches. *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*, 13–22. https://doi.org/10.1109/ICSE-SEIP.2017.31

Jarczyk, O., Jaroszewicz, S., Wierzbicki, A., Pawlak, K., & Jankowski-Lorek, M. (2018). Surgical teams on GitHub: Modeling performance of GitHub project development processes. *Information and Software Technology*, *100*, 32–46. https://doi.org/10.1016/j.infsof.2018.03.010

Joblin, M., Apel, S., Hunsen, C., & Mauerer, W. (2017). Classifying developers into core and peripheral: An empirical study on count and network metrics. *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, 164–174. https://doi.org/10.1109/ICSE.2017.23

Joblin, M., Apel, S., & Mauerer, W. (2017). Evolutionary trends of developer coordination: A network approach. *Empirical Software Engineering*, *22*(4), 2050–2094. https://doi.org/10.1007/s10664-016-9478-9

Johnson, I., McMahon, C., Schöning, J., & Hecht, B. (2017). The effect of population and "structural" biases on social media-based algorithms—A case study in geolocation inference across the urban-rural spectrum. *CHI '17 Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 1167–1178. https://doi.org/10.1145/3025453.3026015

Kelty, C. M. (2008). *Two bits: The cultural significance of free software*. Duke University Press. https://books.google.com/books?id=RarZAAAAMAAJ

Keyes, O. (2018). The misgendering machines: Trans/HCI implications of automatic gender recognition. *Proceedings of the ACM on Human-Computer Interaction*, *2*(CSCW), 1–22. https://doi.org/10.1145/3274357

Klein, G., Ross, K. G., Moon, B. M., Klein, D. E., Hoffman, R. R., & Hollnagel, E. (2003). Macrocognition. *IEEE Intelligent Systems*, *18*(3), 81–85.

Kozlowski, S. W., Hully, S. M., Nason, E. R., & Smith, E. M. (1999). Developing adaptive teams: A theory of compilation and performance across levels and time. In D. R. Ilgen & E. D. Pulakos (Eds.), *The Changing Nature of Performance: Implications for Staffing, Motivation, and Development* (p. 452). Jossey-Bass Publishers.

Levine, J. M., & Choi, H.-S. (2004). Impact of personnel turnover on team performance and cognition. In E. Salas & S. M. Fiore (Eds.), *Team Cognition: Understanding the Factors that Drive Process and Performance.* (pp. 153–176). American Psychological Association. https://doi.org/10.1037/10690-008

Levine, S. S., & Prietula, M. J. (2014). Open collaboration for innovation: Principles and performance. *Organization Science*, *25*(5), 1414–1433. https://doi.org/10.1287/orsc.2013.0872

Liu, C., Yang, D., Zhang, X., Ray, B., & Rahman, M. M. (2018). Recommending GitHub projects for developer onboarding. *IEEE Access*, *6*, 52082–52094. https://doi.org/10.1109/ACCESS.2018.2869207

Lungeanu, A., Sullivan, S., Wilensky, U., & Contractor, N. S. (2015). A computational model of team assembly in emerging scientific fields. *Proceedings of the Winter Simulation Conference*, 4057–4068. https://doi.org/10.1109/WSC.2015.7408559

Majumder, A., Datta, S., & Naidu, K. V. M. (2012). Capacitated team formation problem on social networks. *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '12*, 1005. https://doi.org/10.1145/2339530.2339690

Marlow, J., Dabbish, L., & Herbsleb, J. (2013). Impression formation in online peer production: Activity traces and personal profiles in GitHub. *Proceedings of the 2013 Conference on*

Computer Supported Cooperative Work, 117. https://doi.org/10.1145/2441776.2441792

McDonald, N., & Goggins, S. (2013). Performance and participation in open source software on GitHub. *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, 139–144. https://doi.org/10.1145/2468356.2468382

McMahan, P., & Evans, J. (2018). Ambiguity and engagement. *American Journal of Sociology, 124(*3), 860–912. https://doi.org/10.1086/701298

McNicol, A. (2013). None of your business? Analyzing the legitimacy and effects of gendering social spaces through system design. In M. Rasch & G. Lovink (Eds.), *Unlike Us Reader: Social Media Monopolies and Their Alternatives* (pp. 200–219). Institute of Network Cultures.

Meadows, D. H. (1999). *Leverage Points: Places to Intervene in a System*. The Sustainability Institute.

Mesmer-Magnus, J. R., DeChurch, L. A., & Wax, A. (2012). Moving emotional labor beyond surface and deep acting: A discordance–congruence perspective. *Organizational Psychology Review, 2*(1), 6–53. https://doi.org/10.1177/2041386611417746

Mobley, W. H. (1982). *Employee turnover, causes, consequences, and control*. Addison-Wesley.

Mockus, A. (2010). Organizational volatility and its effects on software defects. *Proceedings of the 18th ACM SIGSOFT International Symposium Foundations of Software Engineering - FSE '10*, 117. https://doi.org/10.1145/1882291.1882311

Nafus, D. (2012). 'Patches don't have gender': What is not open in open source software. *New Media & Society*, *14*(4), 669–683. https://doi.org/10.1177/1461444811422887

Nagappan, M., Zimmermann, T., & Bird, C. (2013). Diversity in software engineering research. *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering -*

ESEC/FSE 2013, 466. https://doi.org/10.1145/2491411.2491415

Nassif, M., & Robillard, M. P. (2017). Revisiting turnover-induced knowledge loss in software projects. *2017 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, 261–272. https://doi.org/10.1109/ICSME.2017.64

Newton, O. B. (2020, January). *Defining and promoting societal benefits in open source software development*. GROUP4GOOD Workshop at GROUP'20: The 2020 ACM International Conference on Supporting Group Work, Sanibel Island, Florida, USA. https://group4good.files.wordpress.com/2020/01/position-paper_newton.pdf

Newton, O. B., Fiore, S. M., & Song, J. (2018). Developing theory and methods to understand and improve collaboration in open source software development on GitHub. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *62*, 1118–1122. https://doi.org/10.1177/1541931218621256

Newton, O. B., Fiore, S. M., & Song, J. (2019). Expertise and complexity as mediators of turnover-induced knowledge loss in open source software development. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. 62nd International Annual Meeting of the Human Factors and Ergonomics Society, Seattle, WA, USA.

Newton, O. B., Saadat, S., Song, J., Fiore, S. M., & Sukthankar, G. (2022). EveryBOTy counts: Examining human–machine teams in open source software development. Topics in Cognitive Science, tops.12613. https://doi.org/10.1111/tops.12613

Newton, O. B., & Song, J. (2022). Modeling gender differences in membership change in open source software projects (arXiv:2206.08485). arXiv. http://arxiv.org/abs/2206.08485

Newton, O. B., & Stanfill, M. (2019). My NSFW video has partial occlusion: deepfakes and the technological production of nonconsensual pornography. *Porn Studies*.

https://doi.org/10.1080/23268743.2019.1675091

Nidy, D. R., & Kwok, F. (2005). Community source development: An emerging model with new opportunities. *CHI '05 Extended Abstracts on Human Factors in Computing Systems - CHI '05*, 1697. https://doi.org/10.1145/1056808.1057000

Noble, S. U. (2020, July 1). The loss of public goods to big tech. *Noema*. https://www.noemamag.com/the-loss-of-public-goods-to-big-tech/

Orlowska, A., Chrysoulas, C., Jaroucheh, Z., & Liu, X. (2021). Programming languages: A usage-based statistical analysis and visualization. *2021 The 4th International Conference on Information Science and Systems*, 143–148. https://doi.org/10.1145/3459955.3460614

Ortu, M., Destefanis, G., Counsell, S., Swift, S., Marchesi, M., & Tonelli, R. (2016). How diverse is your team? Investigating gender and nationality diversity in GitHub teams. *PeerJ Preprints*. https://doi.org/10.7287/peerj.preprints.2285v1

Pawade, D., Dave, D. J., & Kamath, A. (2016). Exploring software complexity metric from procedure oriented to object oriented. *2016 6th International Conference - Cloud System and Big Data Engineering (Confluence)*, 630–634. https://doi.org/10.1109/CONFLUENCE.2016.7508195

Prana, G. A. A., Treude, C., Thung, F., Atapattu, T., & Lo, D. (2018). Categorizing the content of GitHub README files. *ArXiv:1802.06997 [Cs]*. http://arxiv.org/abs/1802.06997

Qiu, H. S., Li, Y. L., Padala, S., Sarma, A., & Vasilescu, B. (2019). The signals that potential contributors look for when choosing open-source projects. *Proceedings of the ACM on Human-Computer Interaction*, *3*(CSCW), 1–29. https://doi.org/10.1145/3359224

Qiu, H. S., Nolte, A., Brown, A., Serebrenik, A., & Vasilescu, B. (2019). Going farther together: The impact of social capital on sustained participation in open source. *2019 IEEE/ACM*

41st International Conference on Software Engineering (ICSE), 688–699. https://doi.org/10.1109/ICSE.2019.00078

R Core Team. (2022). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Rajanen, M., & Iivari, N. (2015). Power, empowerment and open source usability. *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*, 3413–3422. https://doi.org/10.1145/2702123.2702441

Rigby, P. C., Zhu, Y. C., Donadelli, S. M., & Mockus, A. (2016). Quantifying and mitigating turnover-induced knowledge loss: Case studies of chrome and a project at avaya. *Proceedings of the 38th International Conference on Software Engineering - ICSE '16*, 1006–1016. https://doi.org/10.1145/2884781.2884851

Robles, G., & Gonzalez-Barahona, J. M. (2006). Contributor turnover in libre software projects. In E. Damiani, B. Fitzgerald, W. Scacchi, M. Scotto, & G. Succi (Eds.), *Open Source Systems* (Vol. 203, pp. 273–286). Springer US. https://doi.org/10.1007/0-387-34226-5_28

Rosen, L. E. (2005). *Open source licensing: Software freedom and intellectual property law*. Prentice Hall PTR.

Ruths, D., & Pfeffer, J. (2014). Social media for large studies of behavior. *Science*, *346*(6213), 1063–1064. https://doi.org/10.1126/science.346.6213.1063

Salas, E., Fiore, S. M., & Letsky, M. P. (Eds.). (2012). *Theories of team cognition: Cross-disciplinary perspectives*. Routledge/Taylor & Francis Group.

Sarma, A., Gerosa, M. A., Steinmacher, I., & Leano, R. (2016). Training the future workforce through task curation in an OSS ecosystem. *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering - FSE 2016*,

932–935. https://doi.org/10.1145/2950290.2983984

Semega, J., & Kollar, M. (2022). Income in the United States: 2021 (No. P60-276; Current Population Reports). U.S. Census Bureau. https://www.census.gov/content/dam/Census/library/publications/2022/demo/p60-276.pdf

Schilt, K., & Westbrook, L. (2009). Doing gender, doing heteronormativity: "Gender normals," transgender people, and the social maintenance of heterosexuality. *Gender & Society*, *23*(4), 440–464. https://doi.org/10.1177/0891243209340034

Schmaus, W. (2015). Determinism: Social and economic. In *International Encyclopedia of the Social & Behavioral Sciences* (pp. 241–246). Elsevier. https://doi.org/10.1016/B978-0-08-097086-8.03126-3

Schoder, D., Putzke, J., Metaxas, P. T., Gloor, P. A., & Fischbach, K. (2014). Information systems for "wicked problems" – Research at the intersection of social media and collective intelligence. *Business & Information Systems Engineering*, *6*(1), 3–10.

Shaw, J. D., Delery, J. E., Jenkins, G. D., & Gupta, N. (1998). An organization-level analysis of voluntary and involuntary turnover. *Academy of Management Journal*, *41*(5), 511–525. https://doi.org/10.2307/256939

Singh, V. (2012). Newcomer integration and learning in technical support communities for open source software. *Proceedings of the 17th ACM International Conference on Supporting Group Work - GROUP '12*, 65. https://doi.org/10.1145/2389176.2389186

Solanas, A., Selvam, R. M., Navarro, J., & Leiva, D. (2012). Some common indices of group diversity: Upper boundaries. *Psychological Reports*, *111*(3), 777–796. https://doi.org/10.2466/01.09.21.PR0.111.6.777-796

Steinhardt, S. B., Menking, A., Erickson, I., Marshall, A., Zelenkauskaite, A., & Rode, J. (2015). Feminism and feminist approaches in social computing. *Proceedings of the 18th ACM Conference Companion on Computer Supported Cooperative Work & Social Computing - CSCW'15 Companion*, 303–308. https://doi.org/10.1145/2685553.2685561

Steinmacher, I., Gerosa, M., Conte, T. U., & Redmiles, D. F. (2019). Overcoming social barriers when contributing to open source software projects. *Computer Supported Cooperative Work (CSCW)*, *28*(1–2), 247–290. https://doi.org/10.1007/s10606-018-9335-z

Stokols, D., Misra, S., Moser, R. P., Hall, K. L., & Taylor, B. K. (2008). The ecology of team science: understanding contextual influences on transdisciplinary collaboration. American Journal of Preventive Medicine, 35(2), S96-S115.

Tennekes, M. (2018). tmap: Thematic maps in R. *Journal of Statistical Software*, 84(6), 1–39. https://doi.org/10.18637/jss

Terrell, J., Kofink, A., Middleton, J., Rainear, C., Murphy-Hill, E., Parnin, C., & Stallings, J. (2017). Gender differences and bias in open source: Pull request acceptance of women versus men. *PeerJ Computer Science*, *3*, e111. https://doi.org/10.7717/peerj-cs.111

Tourani, P., Adams, B., & Serebrenik, A. (2017). Code of conduct in open source projects. *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 24–33. https://doi.org/10.1109/SANER.2017.7884606

Trow, D. B. (1960). Membership succession and team performance. *Human Relations*, *13*(3), 259–269. https://doi.org/10.1177/001872676001300306

Tufekci, Z. (2015). Algorithms in our midst: Information, power and choice when software is everywhere. *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing - CSCW '15*, 1918–1918.

https://doi.org/10.1145/2675133.2697079

Uzzi, B., Mukherjee, S., Stringer, M., & Jones, B. (2013). Atypical combinations and scientific impact. *Science*, *342*(6157), 468–472. https://doi.org/10.1126/science.1240474

van der Vegt, G. S., Bunderson, S., & Kuipers, B. (2010). Why turnover matters in self-managing work teams: Learning, social integration, and task flexibility. *Journal of Management*, *36*(5), 1168–1191. https://doi.org/10.1177/0149206309344117

Vasilescu, B. (2014). Human aspects, gamification, and social media in collaborative software engineering. *Companion Proceedings of the 36th International Conference on Software Engineering - ICSE Companion 2014*, 646–649. https://doi.org/10.1145/2591062.2591091

Vasilescu, B., Posnett, D., Ray, B., van den Brand, M. G. J., Serebrenik, A., Devanbu, P., & Filkov, V. (2015). Gender and tenure diversity in GitHub teams. *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*, 3789–3798. https://doi.org/10.1145/2702123.2702549

Vasilescu, B., Serebrenik, A., & Filkov, V. (2015). A data set for social diversity studies of GitHub teams. *2015 IEEE/ACM 12th Working Conference on Mining Software Repositories*, 514–517. https://doi.org/10.1109/MSR.2015.77

von Krogh, G., Spaeth, S., & Lakhani, K. R. (2003). Community, joining, and specialization in open source software innovation: A case study. *Research Policy*, *32*(7), 1217–1241. https://doi.org/10.1016/S0048-7333(03)00050-7

Waxman, O. B. (2017, August 8). Women in tech and the history behind the controversial Google diversity memo. *Time*. https://time.com/4892094/google-diversity-history-memo/

What is Free Software? (2022, June 25). GNU Project. https://www.gnu.org/philosophy/free-

sw.html#four-freedoms

Whiting, M. E., Blaising, A., Barreau, C., Fiuza, L., Marda, N., Valentine, M., & Bernstein, M. S. (2019). Did it have to end this way? Understanding the consistency of team fracture. *Proceedings of the ACM on Human-Computer Interaction*, *3*(CSCW), 1–23. https://doi.org/10.1145/3359311

Wickham, H. (2016). *ggplot2: elegant graphics for data analysis* (2nd ed.). Springer International Publishing. https://doi.org/10.1007/978-3-319-24277-4

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T., Miller, E., Bache, S., Müller, K., Ooms, J., Robinson, D., Seidel, D., Spinu, V., … Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Why Open Source Misses the Point of Free Software. (2022, February 3). GNU Project. https://www.gnu.org/philosophy/open-source-misses-the-point.html

Xu, Y., Pace, S., Kim, J., Iachini, A., King, L. B., Harrison, T., DeHart, D., Levkoff, S. E., Browne, T. A., Lewis, A. A., Kunz, G. M., Reitmeier, M., Utter, R. K., & Simone, M. (2022). Threats to online surveys: Recognizing, detecting, and preventing survey bots. Social Work Research, svac023. https://doi.org/10.1093/swr/svac023