The Long Road Ahead: Ongoing Challenges in Contributing to Large OSS Organizations and What to Do

MARIAM GUIZANI, Oregon State University, USA
AMREETA CHATTERJEE, Oregon State University, USA
BIANCA TRINKENREICH, Northern Arizona University, USA
MARY EVELYN MAY, Oregon State University, USA
GERALDINE J. NOA-GUEVARA, Corvallis High School, USA
LIAM JAMES RUSSELL, Corvallis High School, USA
GRISELDA G. CUEVAS ZAMBRANO, Apache Software Foundation, USA
DANIEL IZQUIERDO-CORTAZAR, Bitergia, Spain
IGOR STEINMACHER, Univ. Tecnológica Federal do Paraná, Brazil & Northern Arizona University, USA
MARCO A. GEROSA, Northern Arizona University, USA
ANITA SARMA, Oregon State University, USA

Open source communities hosted in large foundations operate in a complex socio-technical ecosystem, which includes a heterogeneous mix of projects and stakeholders. Previous work has thus far investigated the challenges faced in OSS communities from the point of view of specific stakeholders, primarily at the level of individual projects. None have yet studied the challenges faced within a large, federated open source organization. In this paper, we aim to bridge this gap to identify ongoing challenges contributors face in a mature OSS organization. To do so, we surveyed 624 contributors at the Apache Software Foundation (ASF) and ran 11 semi-structured follow up interviews. We validated our findings through member checking with the interviewees as well as the ASF Diversity and Inclusion (D&I) committee. The contributions of this paper include: (1) an empirically-evidenced conceptual model of the 88 challenges that contributors face in a mature OSS foundation and (2) a set of 48 community-recommended strategies for alleviating these challenges. Our results show that even well-established and mature organizations still face a variety of individual and project-specific challenges and that it is difficult to design a comprehensive set of processes and guidelines to match the needs and expectations of a diverse and large federated community. Our conceptual challenges model and associated strategies to mitigate them can provide guidance to other OSS foundations and projects helping them in building better support processes and tools to create a successful, thriving community of contributors.

Authors' addresses: Mariam Guizani, Oregon State University, Corvallis, OR, USA, guizanim@oregonstate.edu; Amreeta Chatterjee, Oregon State University, Corvallis, OR, USA, chattera@oregonstate.edu; Bianca Trinkenreich, Northern Arizona University, Flagstaff, AZ, USA, bt473@nau.edu; Mary Evelyn May, Oregon State University, Corvallis, OR, USA, mayma@oregonstate.edu; Geraldine J. Noa-Guevara, Corvallis High School, Corvallis, OR, USA, jimenanoag@gmail.com; Liam James Russell, Corvallis High School, Corvallis, OR, USA, liam.russell@student.csd509j.net; Griselda G. Cuevas Zambrano, Apache Software Foundation, Wakefield, MA, USA, gris@apache.org; Daniel Izquierdo-Cortazar, Bitergia, Madrid, Spain, dizquierdo@bitergia.com; Igor Steinmacher, Univ. Tecnológica Federal do Paraná, Campo Mourão, PR, Brazil, & Northern Arizona University, Flagstaff, AZ, USA, igorfs@utfpr.edu.br; Marco A. Gerosa, Northern Arizona University, Flagstaff, AZ, USA, marco.gerosa@nau.edu; Anita Sarma, Oregon State University, Corvallis, OR, USA, anita.sarma@oregonstate.edu.

2573-0142/2021/10-ART407 $15.00
https://doi.org/10.1145/3479551

CCS Concepts: • **Software and its engineering → Open source model**; • **Human-centered computing → Open source software**; • **Information Systems** → *Open Source software*.

Additional Key Words and Phrases: open source foundation; challenges; collaborative development

ACM Reference Format:
Mariam Guizani, Amreeta Chatterjee, Bianca Trinkenreich, Mary Evelyn May, Geraldine J. Noa-Guevara, Liam James Russell, Griselda G. Cuevas Zambrano, Daniel Izquierdo-Cortazar, Igor Steinmacher, Marco A. Gerosa, and Anita Sarma. 2021. The Long Road Ahead: Ongoing Challenges in Contributing to Large OSS Organizations and What to Do. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW2, Article 407 (October 2021), 30 pages. https://doi.org/10.1145/3479551

1 INTRODUCTION

Since the 2000's, we have seen the rise and wide adoption of open collaboration communities [29] with a peer-production model [4, 22]. In this model, large communities, distributed around the world, collaborate to create knowledge-intensive goods mediated by collaborative platforms [29, 81]. Examples of such communities include Wikipedia, some crowdsourcing initiatives, and Open Source Software (OSS) projects.

Open source communities are an exemplar of this peer-production model, where individuals asynchronously collaborate in a complex socio-technical ecosystem, which comprises interdependent OSS projects that themselves exist in the context of larger organizations such as, the Apache Software Foundation (ASF), the Cloud Native Computing Foundation (CNCF), and the Linux Foundation, to name a few. Such foundations are federated organizations including a mix of individual projects and a variety of stakeholders. Some of these projects produce cutting-edge technology that compete with tech startups, while others develop technology that has gone through decades of evolution with legacy code and mature infrastructures. OSS projects in such foundations also include a mix of paid and volunteer contributors working either in standalone OSS projects or open source arms of commercial companies [67, 94], resulting in a variety of motivations and pathways for contributing [38, 58, 115, 118]. Therefore, OSS becomes a unique environment which mixes volunteer and paid contributors from a variety of locations and cultures [88, 130], adopts a meritocratic-intensive structure [89], and extreme transparency [15], making it a different environment than those corporate settings studied by management literature [36, 80, 84]. Such heterogeneity in contributor goals, project ideals, and underlying infrastructure creates an ecosystem of projects where contributors sometimes have to reconcile contrasting legal mandates and contribution philosophies [23], as well as manage outdated documentation and project-specific cultural issues [79].

The challenges arising from such patchwork of differing contribution philosophies, project visions, and infrastructure in a foundation exacerbates the already difficult task of contributing to OSS and can drain contributors enthusiasm and motivation [93]. This can then in turn affect the health and success of the community. Given that the health of the community is what defines a successful open source ecosystem [11, 13, 37], it is important to *understand the challenges contributors face in large, heterogeneous OSS organizations.*

While past works have identified challenges that contributors (e.g., newcomers [48, 71, 103], one-time contributors [58], mentors[2]) face in individual projects, a comprehensive understanding of the ongoing challenges in large and well-established organizations is under-explored. Closing this gap in our understanding is important to help organizations design strategies, processes, and tools, anticipate potential challenges, and uncover hidden problems. This understanding will not only help these large OSS foundations, but also individual projects that aim to grow, mature, or

assemble into large ecosystems. Therefore, in this paper we investigate the following research questions:

RQ1. What are the ongoing challenges in contributing to a mature OSS organization?

Although research works have put a spotlight on the myriad challenges that OSS faces, there is a lack of a comprehensive conceptual model that compiles the challenges faced in a large OSS organization and it is not clear what strategies and resources are available to the community to mitigate these challenges, mainly those that occur at a Foundation level. Therefore, it is important to hear from the community to identify strategies that they believe will help mitigate these challenges. To do so, we pose our second research question:

RQ2. What strategies can alleviate the challenges to contribute to a mature OSS organization?

To answer our research questions, we worked together with the Apache Software Foundation (ASF) to conduct surveys and interviews in their community. ASF is an appropriate case study since it is the "world's largest open source Foundation" [31], which in 2020 celebrated its 20th year anniversary and boasts 227M+ lines of code in stewardship across 350+ projects and initiatives. Moreover, the ASF is committed to better support their community by identifying and mitigating the challenges their contributors face.

In this study, we partnered with the ASF Diversity and inclusion (D&I) committee [34]. We started our research by designing and deploying an online survey with 600+ respondents. The survey was designed in collaboration with the D&I committee and the ASF community at large. We followed up the survey results with 11 interviews to get a deeper understanding of the challenges and the strategies they recommended to overcome the challenges. Using the tenets of grounded theory, we qualitatively analyzed the data from 223 contributors who reported facing ongoing challenges. We then validated our results via member checking with our interview participants and the ASF D&I committee.

Our analysis provided an empirically-evidenced conceptual model of the challenges faced by contributors across three dimensions (Process, Technical, and Social), as well as across the three layers of the Groupware model proposed by Grudin [43]: Individual, Project, and Organization levels. Our results uncovered a set of 88 challenges and a set of 48 community recommended strategies to mitigate these challenges.

The key contributions of our work include:

- An empirically-evidenced conceptual model of 88 ongoing challenges that OSS contributors face categorized into three levels: Individual, Project, and Foundation.
- The first study that investigates the challenges contributors face at the level of a large, federated OSS organization, such as the ASF.
- A set of 48 community-recommended strategies to mitigate ongoing challenges that contributors face, that other peer-production communities and OSS projects can adopt.

2 RELATED WORK

Open collaboration communities collectively produce artifacts in online environments and need a low barrier to entry to be successful [29]. With the goal of mitigating barriers, researchers have been identifying challenges and proposing strategies to facilitate participation in these communities. For example, Wikipedia has been the subject of several studies. In particular, Halfaker et al. [46] and Suh et al. [110] investigated the negative effects of the number of edit reverts and Zhu et al. [131] investigated the effects of negative feedback. Halfaker and Geiger [45] proposed an approach to mitigate some of these challenges. Challenges have also been investigated in the context of other online communities [54, 56, 122].

Despite sharing similarities with these open collaboration communities, OSS communities have their unique challenges and characteristics. For example, the artifacts in OSS are technically complex and interdependent and require complex technical infrastructure to compile and run. Further, the development communities are often part of a larger environment that involves companies, foundations, and a mix of paid contributors and volunteers. OSS projects also considerably differ from traditional software development organizations in terms of incentives, control, and coordination mechanisms [118]. Traditional organizations rely on pay and career incentives, and other benefits stipulated as part of employment contracts [84]. In OSS, contributors participate due to a variety of motivations [38, 118] in a community with distinct governance models, with different levels of control and coordination structures [89]. Additionally, social practice of OSS development is permeated with ethical and ideological aspects [118]. Finally, participation in OSS includes a high degree of transparency in the form of visibility of actions on public artifacts and involves a community of geographically dispersed contributors [15]. Therefore, specific work is necessary to investigate the challenges to contribute in this complex and unique environment.

Indeed, previous work has investigated OSS-specific contribution challenges [49, 52, 98, 99]. Steinmacher and colleagues [97, 99] conducted a mixed-method study and identified 58 barriers faced by newcomers. They relied on data collected from newcomers, core members, and the literature [104] to categorize the challenges. Lee et al. [58] and Pinto et al. [85] investigated the challenges faced by one-time contributors while Balali et al. [2] focused on OSS mentors.
Researchers have also investigated specific types of challenges. For example, toxic environment have been studied in literature [7, 68, 83, 86], where OSS project members have been found to be unfriendly, unhelpful, or behave as elitist [108]. Jensen et al. [52] analyzed the speed at which emails sent by newcomers are answered as well as the role played by gender or nationality in the kind of answers newcomers receive and the reception they face. Similarly, previous work by Steinmacher et al. [106] analyzed how the answers to newcomers' first emails influenced their retention. Gousios et al. [40] reported challenges related to politics or how a project is governed. Finally, Yu et al. [128] found that shortcomings in project regulation and administration are key reasons for volunteers' dissatisfaction and dropouts.

Previous work has also investigated the effects of such challenges on turnover. Past works have found that a majority of newcomers (as high as 80% in some projects) do not become long-term contributors [106]. Pinto et al. [85] found that almost half of the contributors of a project submit a single contribution and do not go back [85], and many participants never have a pull request accepted [101]. Researchers have also reported on the negative influence of this high turnover on team cognition and performance [62, 63], and its costs on the project and society [35, 41]. Software engineering-specific literature has also shown that developers' turnover harms software development projects [3, 47, 80]. The literature has also shown that projects with high turnover are less successful [47] and projects that retain developers for longer periods increase the knowledge base and expertise [94, 129]. Researchers have also shown that developers' turnover impacts software quality [30, 75, 76]. To better understand how to retain contributors, Lin et al. [65] investigated which development activities influence continued OSS contributions and Lee [60] showed that job satisfaction strongly influences turnover intentions.

Our work is complementary to many of these works. The goal here is to create a comprehensive conceptual model of the challenges that contributors face even after overcoming the initial hurdles of being a newcomer, and focuses on a case study of a mature, long living community with well defined contribution processes. Our comprehensive model confirms the existence of challenges that past works have identified, but also finds new ones and places these in the context of three lenses: challenges faced at the individual, project, and foundation levels.

Many works have raised the issue of low diversity in OSS when considering gender [7, 113, 116], language [108], as well as location [108]. Past work has shown that diverse teams are more productive [116], but currently there are several inherent biases in technical peer-production sites, like Stack Overflow [28], including biases in the tools [71, 82], how discussions are handled, and the kinds of contributions that are well recognized [115]. Diversity is also the focus of research in Wikipedia [64, 91, 112, 120]. A survey conducted by the Wikimedia foundation in 2020 [74] showed that only 12% of their contributors are women. This lack of diversity brings issues with, for example, the way women are portrayed in their biographies reported in the encyclopedia [119]. As already mentioned, the goal of this paper to create an overview of challenges faced by contributors across all demographics in a large organization.

Many works have sought out strategies to help with turnover, with a majority of them focusing on newcomers [14, 95, 100, 105, 121, 124]. Mentoring is one such strategy to onboard newcomers to online communities [51, 78, 107]. Balali et al. [2] also identified strategies that mentors use to alleviate challenges that affect mentors and/or newcomers. In a more recent work [1], these authors identified challenges and strategies specific to recommending tasks for newcomers. Although the literature provides an overview of challenges for specific stakeholders and go deep on specific challenges, the current body of knowledge focuses on a project-level perspective. As observed by Lee and Paine [59] and Grudin [42], there is an emphasis in CSCW literature on the individual and project levels, with the organization view remaining under-investigated. To the best of our knowledge there is no systematic identification of challenges and strategies from an organizational perspective in OSS. The present work adds to the literature by investigating a large and mature OSS foundation—the Apache Software Foundation.

3 METHODOLOGY

The data collection and validation used in the study included three phases: online survey (phase 1), follow-up Interviews (phase 2), and validation via member checking (phase 3).In each of these phases we collaborated closely with the ASF.

We used the Apache Software Foundation (ASF) as our case study as it is the world's largest OSS Foundation with more than 460k people and more than 350 projects and initiatives [31]. The ASF also presents an ecosystem that captures challenges across all the three levels of Grudin's groupware model [43]: individual, project, and organization. Finally, the D&I committee at the ASF seeks to understand the state of diversity at the ASF and the challenges faced by its community, and closely collaborated with us in the study design and execution.

3.1 Phase 1: Online Survey

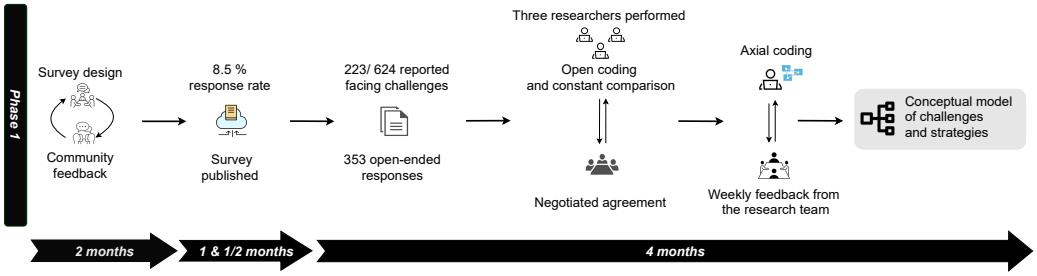

Fig. 1. Overview of the survey methodology (phase 1).

Phase 1 describes the survey design, data collection and analysis (see Fig. 1).

Survey design: We engaged with the ASF D&I committee —which is composed of 18 experienced contributors who have different roles within the foundation including committers, Project Management Committee (PMC) and board members—and the community at large in designing the survey questions. We drafted the initial set of questions by leveraging questions from past surveys in OSS such as, the 2016 ASF Committer Diversity Survey [33], Open Demographics Survey [17], OpenStack Gender Diversity Report [5], Stackoverflow Members Survey [96], and best practices recommended by the Linux Foundation's Community Health Analytics Open Source Software (CHAOSS) D&I Working Group [87]. We then collected feedback from the ASF in a two-month window in the following manner. We shared the survey questions as a Google document with the ASF community, anyone who had an 'apache.org' email could access and comment on the survey questions and collected feedback as comments. We used different channels to reach out to the community for feedback (e.g, JIRA ticket, developers mailing list, ASF slack/D&I channel). After multiple iterations of resolving the comments provided by the community and editing the survey, we scheduled a two-hour listening session with the Apache community to collect additional feedback on the survey questions. We edited the consent form to include the statement that data would be handled as per Apache Privacy Policy and updated the survey questions based on the feedback we received. We reworded some questions and options to make them more inclusive to an international audience and the different genders (e.g., changed 'non binary' to 'gender variant, non conforming/ non binary'). Additionally, the compensation question was edited to include all the different compensation options possible in the ASF projects. We also took out the two 'socio-economic' status questions as these were perceived to be sensitive. Finally, we removed a raffle option (for improving participation) because of the legality of being able to provide raffle options equally across all countries. (See the supplementary material [44] for the list of survey questions, feedback summary, and corresponding changes).

Survey Data Collection: We used LimeSurvey, licensed GPL version 2 or later, as a distribution platform as it is the world's leading open source survey software. We invited ASF committers to participate by sending emails to every 'apache.org' email address and shared a link through the ASF developer mailing lists. When starting the survey, participants were presented with a consent page that described the goal of the survey, the data collection and handling policy, and who to contact. The survey followed an opt-in strategy where participants started the survey if they agreed to voluntarily participate after reading the consent form (see supplementary material [44]). The survey was open for 45 days.

We received 624 responses to our questionnaire, resulting in a response rate of 8.5% based on a considered total community size of 7500 contributors. We maintained the data confidentiality as per Apache Privacy Policies. No identifiable information or IP addresses were collected. In cases where participants provided their email address for follow-up interviews, these were stored separately from their responses.

Out of the 624 respondents, 223 reported that they faced challenges. From these 223 people, 221 openly reported what challenges they faced and 132 mentioned recommended strategies to overcome the challenges. We present the demographics of these 223 participants next.

As Fig. 2 shows, the majority of our 223 survey respondents are volunteers who identify as men (86.5%) and are senior contributors with three or more years of experience at ASF (77.1%). They reside in 35 different countries located in six continents with the majority based in the US. Most of our respondents reported some level of higher education (master's degree 46.2%, undergraduate 32.7%) and are "very confident" about their English proficiency skills (55.7%).

Survey analysis: We followed the principles of grounded theory [109] to qualitatively code and analyse the open responses in the survey. Three researchers performed the analysis, by first independently coding 353 responses to the open-ended questions (221 challenge and 132 suggestion

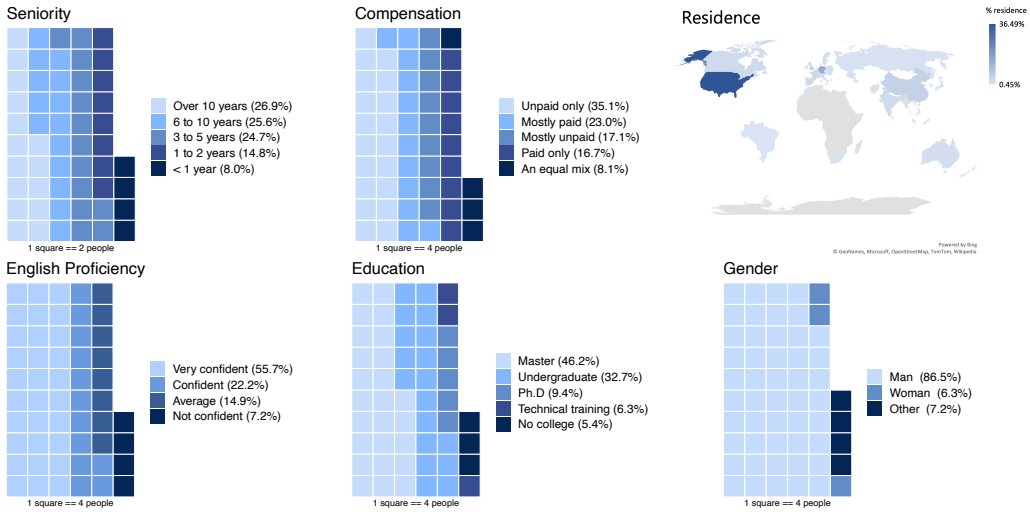

Fig. 2. The background of the 223 survey participants who reported facing challenges.

responses) via open coding with constant comparison method [36, 39], where emerging codes are compared with existing codes, and then met to discuss their codes and reach negotiated agreement. This process consisted of daily iterations in a three month period. Each researcher performed an iterative process of inductively coding the open-ended questions for one participant at a time and built post-formed codes as the analysis progressed and associated them to respective parts of the open-answer text. At this stage our aim was to code the challenges and strategies according to the participants' discourse, and not according to any preconceived data. When the codes were considered ready by each independent researcher, the three researchers met to discuss their codes and reach a negotiated agreement (these meetings were held at least three times a week). During the negotiated agreement meetings, the three researchers compared and discussed their codes for each participant's open-ended questions. For example, if the researchers identified the same challenge, but called it differently then we refined the code nomenclature. And if the disagreement was about the code that was used then it was resolved by refining the code set in the following ways: Merging two codes together, identifying a different granularity level for the codes in disagreement, or checking if the newly emerged code identified by a researcher fits under an existing code.

After completing the open coding, the first author performed daily iterations of axial coding [6, 12] during a one month period to organize the post-formed codes and draw connections between the different codes and categories of the challenges and associate the strategies to the corresponding challenge categories. The axial coding iterations were paired with weekly feedback meetings with the whole research team.

3.2 Phase 2: Follow up Interviews

Fig. 3, depicts the interview design, data collection and analysis steps. After building the conceptual model of challenges and strategies, we wanted to corroborate our findings. To do this, we designed a semi-structured interview script and sandboxed it with the research team. For the interviews we first identified survey respondents who had agreed for follow up conversations. From this set of 69 respondents, we randomly selected a set of 20 interview participants to balance the demographics distribution, as listed in Table 1. We reached out to the participants via email and

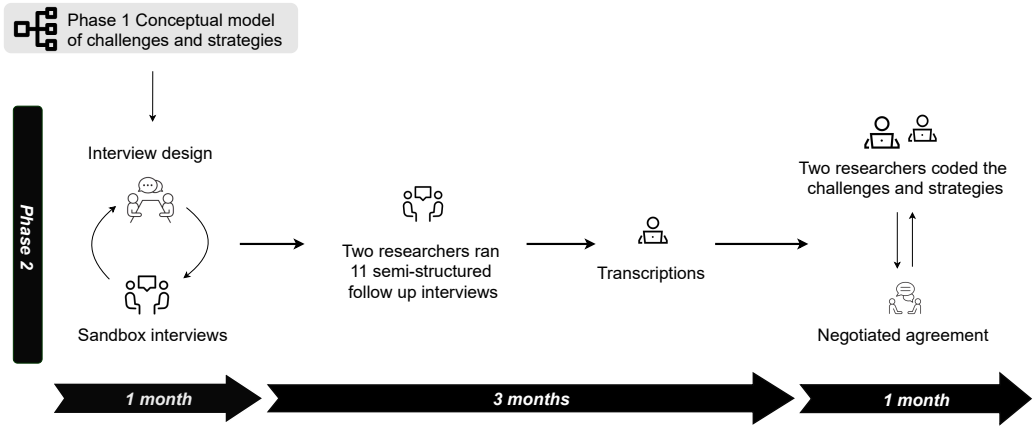

Fig. 3. Overview of the interviews methodology (phase 2).

were able to schedule 11 participants for online interviews. Two researchers ran the semi-structured interviews: one researcher led the interview while the other observed and took notes. Before each interview we obtained their consent to being audio recorded. During the interview, we asked the participants three main questions regarding their current work at ASF, the challenges they face and the strategies they suggested (see supplemental material [44]). Depending on the participants response we would follow up with specific questions; for instance, prompting the participants to provide more information about the challenge they mentioned. The interview lasted between 30 minutes to one hour, after which, we thanked our participants and compensated them with a $50 gift card or its equivalent in donation to the OSS project/organization of their choice.

Table 1 displays the diverse set of demographic attributes of our interview participants. We interviewed four women contributors, five men and two contributors whose gender we cannot disclose to maintain their anonymity. Our interview participants had different education levels with five interviewees having an undergraduate degree, five with master's degree and one who had a Ph.D. The interviewees were from the United States (four participants) and Europe (seven participants from Italy, Russia, Germany) with different English proficiency levels.

Interview Analysis: During a one month period, after transcribing the interviews, the first and second authors independently coded the challenges and strategies for each interviewee. The interview analysis process was similar to the survey data analysis, except that the code set from the survey was used as a starter code set. For each interviewee, the two researchers identified and coded each excerpt that described a challenge or a strategy. This process consisted of associating an excerpt with post-formed code from the survey when applicable and adding new codes when new challenges or strategies emerged.

Twice a week, the two researchers met to discuss their coding of each interviewee to reach a negotiated agreement. We reached saturation after the fourth interview where no new challenges or strategies were identified. Finally, the research team held three one hour meetings to discuss and get the conceptual model ready for the validation phase.

3.3 Phase 3: Member Checking

To validate our findings, we performed member checking in two steps (see Fig. 4). First, with our interviewees. We contacted each participant via email and shared the interview findings in a Google document, which was personalized per interviewee with detail about the challenges and strategies

Table 1. Interview participants' demographics

ID	Gender	Education	English Proficiency	Compensation type	Residence
I-1	Man	Undergraduate	Average	A mix but mostly paid	Russia
I-2	Man	PhD	Very confident	A mix, but mostly unpaid	US
I-3	Woman	Undergraduate	Confident	Unpaid only	Germany
I-4	Woman	Undergraduate	Very confident	An equal mix of paid and unpaid	Ireland
I-5	can not disclose	Undergraduate	Very confident	a mix but mostly paid	US
I-6	can not disclose	Undergraduate	Very confident	Unpaid only	US
I-11	Man	Masters	Not confident	A mix, but mostly unpaid	Italy
I-14	Man	Masters	Very confident	Unpaid only	Japan
I-16	Woman	Masters	Confident	Unpaid only	Germany
I-18	Woman	Masters	Very confident	An equal mix of paid and unpaid	US
I-20	Man	Masters	Very confident	An equal mix of paid and unpaid	UK

they discussed. 10 of the 11 interview participants gave us their feedback by directly commenting on the shared document (four interviewees), responding via email (two interviewees), or scheduling a 10 minute online meeting with the first author (four interviewees). Second, we validated our

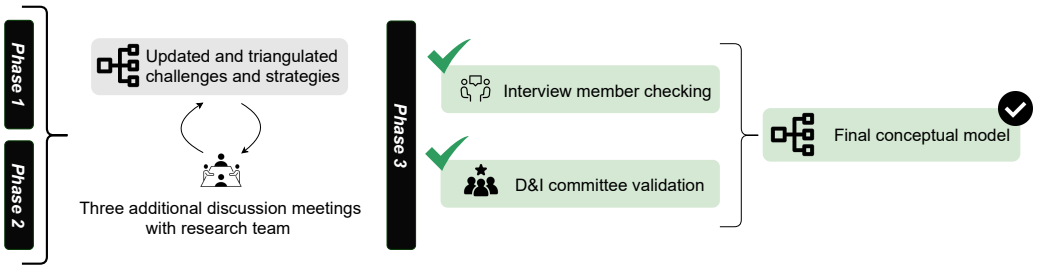

Fig. 4. Overview of two steps member checking methodology (phase 3).

findings with the D&I committee at ASF. We shared a Google document, which was open for 14 days, detailing the conceptual challenge model and the community driven strategies. We used both the mailing list and the slack channel to broadcast the document and received feedback. No one added any additional challenge or strategy or made any suggestion for changes. An additional read out of the results to the ASF community at large is planned in the future.

4 OVERVIEW OF THE CHALLENGES

Our results from the survey and interviews showed that contributors in the ASF face a variety of challenges (88) and recommend a number of strategies (48) to overcome the challenges. We categorized the challenges according to the *level* at which they occur, as described by Grudin [43]: Foundation, Project, and Individual (see rows in Fig. 5). We categorized a challenge at a specific level if the participant explicitly mentioned the level. For example, [S-550] explicitly mentioned that the challenge they faced is related to the ASF: *"ASF policies for contributing, hosting, backwards compatibility were too restrictive"*. In cases where a level was not explicitly mentioned, we determined the level based on who has the agency to resolve the challenge: *"It's also not super clear how the idea of 'rough consensu[s]' works...and how to proceed if 'rough consensus' cannot be reached"*[S-404]. In this instance, we labeled the challenge at the Foundation level because the voting process is a fundamental aspects of the Apache Way and any changes to it has to be Foundation-wide. The supplementary document [44] provides the full list of challenges, with challenges at different levels

differentiated via colors. The challenges at each level were further divided into three dimensions—Process, Technical, and Social—based on where the challenges arose from (see columns in Fig. 5).

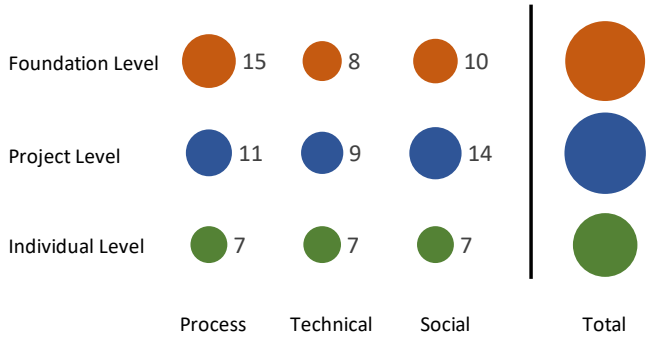

Fig. 5. Distribution of the challenges across the three dimensions (Process: 33, Technical: 24, Social: 31) and levels (Foundation: 33, Project: 34, Individual: 21). Bubble sizes represent the number of challenges found per dimension-lens (See supplemental material [44] for more details of each challenge).

As Fig. 5 shows the majority of challenges reported by participants occur at the Foundation (33) and Project (34) levels. Thus, investigation of organization-level challenges, an under-explored research area, and mechanisms to overcome them is important for effective functioning of large peer-production sites. The majority of challenges at the Foundation level were related to processes, followed by social interactions, and then technical. This distribution of the type of challenge varies at the different levels. At the Project level, Social challenges were the most reported followed by Process and Technical. At the Individual level, these three were equally represented.

In the following sections, we discuss our results. For each category of challenges, we present some examples of the challenges, the levels at which they occur, and the strategies recommended by the community to mitigate them. For the full list of the challenges and community-driven strategies see supplemental material [44].

5 PROCESS CHALLENGES AND STRATEGIES

Processes are crucial in making any organization successful [18]. This is especially true for OSS because of its globally distributed, (usually) voluntary peer-production nature. Specifically, contribution processes in OSS are a key factor in helping projects to attract and retain contributors [126]. Effective processes are needed not only for new contributors joining projects [21], but also for the teams to be productive [127].

Participants reported five types of process-related challenges: "Navigating the contribution process", "Reception Issues", "Getting started in a new project" and "Licenses". In the following section, we discuss the challenges in "Navigating the contribution process" as they account for 70% of the challenges.

5.1 The Apache Way

The Foundation provides a set of governing principles that drive the contribution philosophy and processes of its projects, which are known as the Apache Way. The Apache Way has been refined over the past 20 years and affect some of the largest and longest-lived Open Source Software projects that have revolutionized the industry. A core principle of the Apache Way is: *"There is no 'one way' to the Apache Way. The ASF is not dictatorial and will never compel a rigid path to*

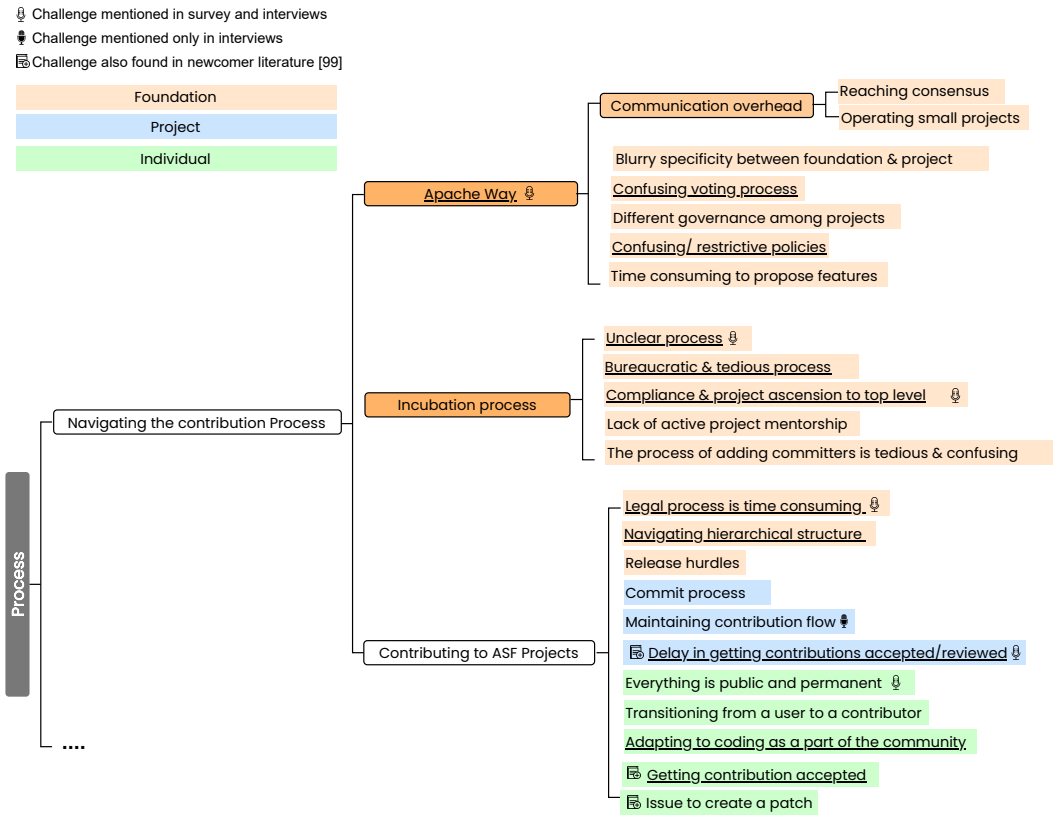

Fig. 6. Challenges related to Navigating the Contribution Process at three levels: Foundation (orange), Project (blue) and Individual (green). The underlined challenges are discussed in the text.

implement our process" [32]. While this allows projects the flexibility to interpret the Apache Way philosophy to best fit their vision, it is at the same time challenging for projects and contributors to figure out how to adopt it. [S-337] *"each time I start to contribute, I have to spend nontrivial time to learn a *new* policy".*

Fig. 6 shows seven challenges associated with the Apache Way, one of which relates to the Voting process, especially for reaching consensus. As [S-404] mentioned: *"It's also not super clear how the idea of 'rough consensu[s]' works...and how to proceed if 'rough consensus' cannot be reached".* This is especially frustrating since conflicting views appear based on the interpretation of the Apache philosophy. As [S-186] says: *"we sometimes make a decision, then sometimes some ASF member shows up, saying this is not possible because of rules. Reading the rules is not working for us. We then try to guess what the spirit and the intend of the rule is, and try to live up to this spirit"* .

5.2 The Incubation Process

The Apache Incubator [19] aims to ease the transition of standalone OSS projects or commercial projects into the ASF, by guiding new "podling" projects on how to adopt the Apache Way and graduate to top-level ASF projects. However, the incubation process is challenging for the onboarding projects despite the ASF providing services and mentors to help with it. Fig. 6 reveals five challenges (**Foundation**). One of them arises because of the flexibility afforded by the Apache Way

and a lack of clear guidelines on how to implement it, which makes it difficult for new projects to understand how to convert their processes to match the Apache Philosophy [32]. As [I-2] explains *"when somebody comes in and says, Well, what does it take to become an Apache project? The only actual true gospel answer is well, it really depends. And there's no single answer."*

Even when projects managed to navigate the unclear incubation process, it can be bureaucratic and tedious making it inconvenient. As [S-615] explains: *"You have to label your project as 'not ready yet' literally everywhere (incubating in releases version, repositories, websites, etc) independently of the prior maturity of your project outside the incubator [despite] inconvenience to the project users and contributors"*. Finally, to become a top-level ASF project, it needs to prove its compliance to the ASF policies and licensing conditions, which is difficult.

5.3 Contributing to ASF projects

Aside from foundation guidelines, joining an ASF project as a contributor presents its own set of specific challenges, as presented in Fig. 6. These 11 challenges span all three levels: Foundation (three), Project (three), and Individual (five).

One of these challenges is the lack of clearly defined legal processes in getting a commercial project inducted into thr ASF and the delays associated with it, which can be especially tedious when reconciling licenses between commercial companies and OSS (**Foundation**). As [I-6] explained: *"in [project name], the challenge [is] bringing the [code] from the [company name] into ASF, there's a lot of legality involved. And that requires a wait period...So that require quite a lot of hand holding"*. Delays can also occur in an already inducted project because of the way project roles are hierarchically structured within the ASF, making it difficult to understand the requirement of transitioning from one role to another. As [S-404] reported: *"It was also not completely clear what extra tasks I had to do to complete my transition to becoming a committer as well as becoming a PMC member later down the line"*. Code contributions too can face delays at the **Project level**. For example, there is delay in getting contributions accepted, requiring contributors having to follow up multiple times, sometimes to no avail. As [S-480] and [S-339] reported: *"project was inactive-bug report/patch was ignored until I've pinged the PMCs"* and *"it was sometimes a struggle to find a person that is available"*.

Challenges in contributing to ASF projects do not stop at the Foundation and Project levels, but also continue at the **Individual Level**. Adapting to coding as part of a community takes time and sometimes a paradigm shift. As [S-655] reported: *"switching from one-man coding to a community approach is sometimes hard"*. Even after adapting to coding as a community, getting contributions accepted can be non trivial. As [S-326] said: *"It was hard to convince the project committers in my decisions and why they should have accepted my code"*.

5.4 Strategies

Table 2 presents strategies to tackle the above challenges. At the Foundation level, flexible guidelines (e.g., the Apache Way) are beneficial as they allow projects freedom, but such flexibility can lead to differing interpretations and cause confusion. Participants recommended supplementing the "Apache Way" flexibility with a clear and understandable introduction to the Foundation's philosophy and principles. A strategy is to provide an easy-to-consume, modern introduction to the ASF (see Table 2, row 1). As [S-147] explained: *"We need a video guideline on Youtube or somewhere else to introduce the process of the basic principle of ASF"*.

In addition to clarifying the Foundation's philosophy, participants also felt the need for clear guidance on specific governance processes, such as the voting mechanism (see Table 2, row 3). As [S-404] recommended: *"More guidance on the voting process and how the process of rough consensus is supposed to be reached and what to do in cases where it cannot be reached"*.

Table 2. Strategies for Navigating the Contribution Process. The *starred strategies are the ones discussed in the text. The source is indicated by: (S&I) for survey and interviews, (S) for surveys, and (I) for interviews.

Challenges for Navigating the contribution process	Strategy	Description of strategy
Apache Way	*Modernize introduction to Apache (S)	Provide a welcome pack, ASF movie, videos, and presentations
	Provide regular training on ASF (S)	Provide training and workshops about ASF principles, community building, and project funding
	*Provide clear guidance on the governance process (S&I)	Provide information about the voting and about how consensus is reached, publish the hierarchical structure
	Give projects more agency (S)	Standardize ASF general policies, but give projects more agency to deviate from the ASF guidelines – provide the flexibility to accommodate different ways of working
	Make becoming a reviewer equitable (S)	Ease the process of becoming a reviewer and provide automatic setup system for new contributors
Incubation Process	*Provide sample success paths to guide through incubation (S&I)	Provide prescriptive guidelines for new projects to follow
	*Remove negative stigma involved with "incubation" status (S)	Recognize the project merit during the incubation process and have mentors take on the guiding hat instead of the assessment hat
Contributing to ASF projects	Provide training on contributing to ASF projects (S&I)	Hand over contribution process and guidelines about both GitHub and non-GitHub projects through videos, provide templates and instructions to fill issues and pull-requests and publish troubleshooting pages
	Encourage knowledge transfer (S&I)	Enable between projects knowledge transfer
	*Democratizing open source licensing (S)	Making OSS licensing more accessible to everyone and provide guidance on when to use the different licenses
	Improve the code reviews process (S&I)	Ease the review process, provide a template walkthrough script that prompts questions
	Provide training and best practices tools for reviewers (S&I)	Provide training/ best practices on how to provide welcoming actionable reviews

The compliance to the Foundation policies is not only important for existing projects, but also for incoming projects. One way to help these projects graduate to top level is to clarify the incubation process by providing examples of successful projects. As [I-2] recommended: *"[I] made the incubator be a little bit more prescriptive... prescriptive in the sense of here's one successful path"*. Given that the incubation process is lengthy, participants felt that the community should recognize the effort during the incubation process and avoid the stigma associated with being a "podling" project. As [S-615] suggested *"Avoiding using the label '-incubating' as synonymous to 'not worthy yet'"*.

Finally, challenges to contributing exist even for projects that are already part of the ASF. For instance, the lack of clearly defined legal processes can cause delays. The community recommended democratizing open source licensing to help with this. For example, [S-404] suggested providing *"more guidance on how to figure out if a particular project's license means it can be legally used... and the difference between licenses that can be vendored and used in a project"*.

6 TECHNICAL: THE CHALLENGES AND THE STRATEGIES

Contributing to open source, as in any large production effort, can be challenging where contributors have to understand the code base and its architecture, as well as the tools and resources used in the project [1, 82, 99]. In a large, mature organization, such as the ASF, where multiple projects co-exist and are interdependent, individuals are faced with legacy code, disparate tools and infrastructure.

Seamlessly managing all these factors can be challenging. These challenges were under two subcategories, Technical hurdles and Documentation problems (see Fig. 7). Here, we detail the challenges and strategies regarding Technical hurdles (accounting for 50% of Technical challenges).

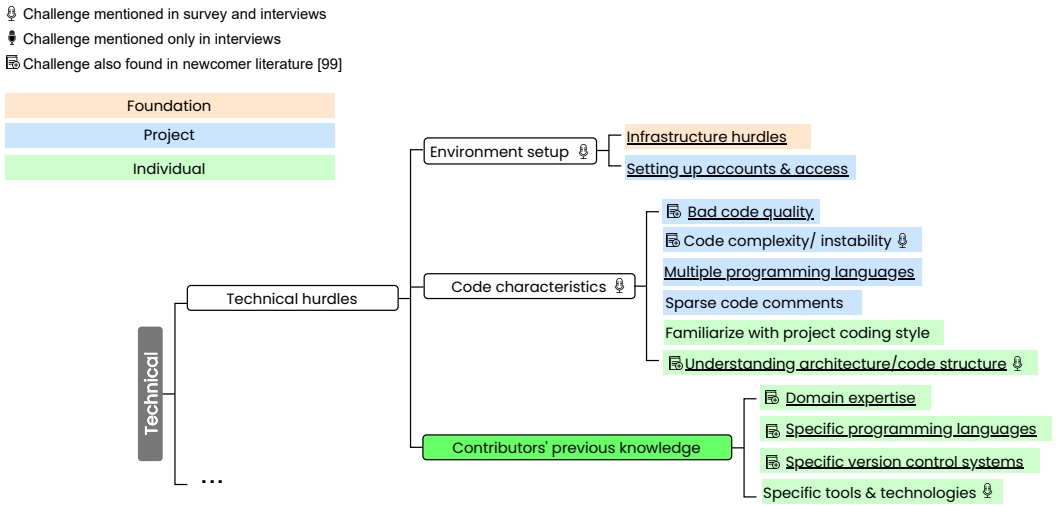

Fig. 7. Challenges related to Technical Hurdles at three levels: Foundation (orange), Project (blue) and Individual (green). The underlined challenges are discussed in the text.

6.1 Technical hurdles

Technical hurdles can create barriers that not only impede contributors joining a project [99], but also in continued participation [130]. Participants reported challenges spanning all three levels, with one challenge at the Foundation and the rest at the Project (5) and Individual (7) levels.

6.1.1 Environment setup. The first steps to begin contributing to a project comprises setting up the environment and access to the project. Doing so can be challenging if older infrastructure, with more complex procedures, are still in use (**Foundation**). For instance, contributor [S-753] felt: *"ASF infrastructure (e.g., for web hosting) can be more limiting and presents more challenges to work with than public cloud offerings"*. These challenges can prompt contributors to create separate lines of development using more advanced infrastructure: [S-550] *"forking and continuing development with several contributors outside of ASF"*. Similarly, setting up accounts and getting access to projects can be challenging in a large organization with many projects (**Project**)— [S-494] *"I have had recurring problems getting my login credentials to work. This slowed almost to a halt my contributions"* .

6.1.2 Code characteristics. Large projects can pose several challenges such as, the use of multiple programming languages, complex code architecture, adopting to project-specific coding style etc. For instance, managing good code quality can be challenging in projects that use multiple programming languages (**Project**). As [S-80] said: *"commit patches that had serious flaws...this is a hard problem in [project name] due to the wide range of languages supported (>15)"*. Writing good quality code can also be challenging if the code architecture is too complex to understand (**Individual**). As [I-18] corroborated that *"understanding what the architecture of the project is what dependencies you need to be aware of"*.

6.1.3 Contributors Previous knowledge. Previous knowledge about the specific tools, technology and programming languages used in the project is an important precursor to contributing, but can be challenging (**Individual**) [100]. As [I-18] mentions *"not familiar with a lot of the tools and systems that Apache projects use..I wasn't able to do it just because I couldn't configure like Jenkins or other systems in my computer"*. Even when contributors have prior experience with the languages and technology, having a deep understanding of the application domain can still be a challenge. As [S-48] says, *"Everybody spoke about applications of programs or websites...I felt like I was illiterate"*.

6.1.4 Strategies. Table 3 shows a set of **strategies** to ease the Technical challenges. One way is the use of automation and popular upcoming technology; *"we should rely more on tools on tooling, which is already known to developers"*–[I-1]. Another was for individuals to be proactive and start communicating about problems they are facing. As [I-20] suggests: *"We are encouraging them to ping us [committers], nag us, multiple times...if the [Pull Request] is not reviewed"*.

Table 3. Strategies to mitigate Technical Hurdles. The *starred strategies are the ones discussed in the text. The source is indicated by: (S&I) for survey and interviews, (S) for surveys, and (I) for interviews.

Strategy to mitigate Technical Hurdles	Description of strategy
*Use automated tools, widely used tools and leverage upcoming technology (S&I)	Use tools that are already familiar to contributors and provide guidance on incorporating upcoming technologies. Provide an automated credential resetting system, leverage automatic code formatting, use single sign-on systems, provide integrated, secure, up to date, and easy to use tools.
*Be proactive and start the communication (I)	Don't hesitate to reach out to committers about the yet to be reviewed PRs

7 SOCIAL: THE CHALLENGES AND THE STRATEGIES

In a traditional workplace, the social atmosphere and environment is one of the aspects of well-being that can impact employees' productivity and retention [10, 123]. In fact, when recruiting new employees many companies look not only at the technical skills but also whether a potential employee fits well within the company culture and vision. In open source, the social environment is even more critical. First, a vast majority of OSS contributors are volunteers which rules out the monetary compensation as a retention factor. Second, OSS is a highly distributed environment where the majority of interaction happens online. This makes it harder to rely on social cues and makes hostility and misunderstanding more likely to rise. Thus, it is crucial to understand the ongoing social challenges that contributors are facing in order to address them and promote a healthy communication and social environment.

The social challenges reported by participants encompass five sub-categories: "Contributors' cultural differences", "Toxic/ unwelcoming environment", "Communication hurdles", "Lack of recognition" and "Geolocation". The "Toxic/ unwelcoming environment" and "Communication" challenges were a majority, accounting for 74% of the challenges. Here, we present these challenges from the Individual, Project and Foundation lenses and examples of strategies to ease these challenges.

7.1 Toxic/ Unwelcoming Environment

Fig. 8 shows the 10 challenges that arise because of a toxic or unwelcoming environment either at the Foundation or the Project level. Seven challenges were reported at the **Foundation level**, one of which includes the organizational structure of the ASF. Its hierarchical structure while beneficial in a large organization, can also be seen as discriminating and unwelcoming to those

lower in the organization. As [I-18] reported: *"no one with less than 10 years of experience is well regarded to lead anything at the Foundation"*. The voting system at the ASF is meant to overcome such (class) discrimination by giving everyone a voice, but the requirement of consensus building and veto power makes the decision making process difficult and unproductive. Such ASF wide discussions could be particularly unwelcoming, which affects overall participation, especially by under-represented groups as [S-62], a woman, said: *"I was happy being a committer but once I saw all the members discussion in the list I was overwhelmed...I have never commented on the members list for this reason"*.

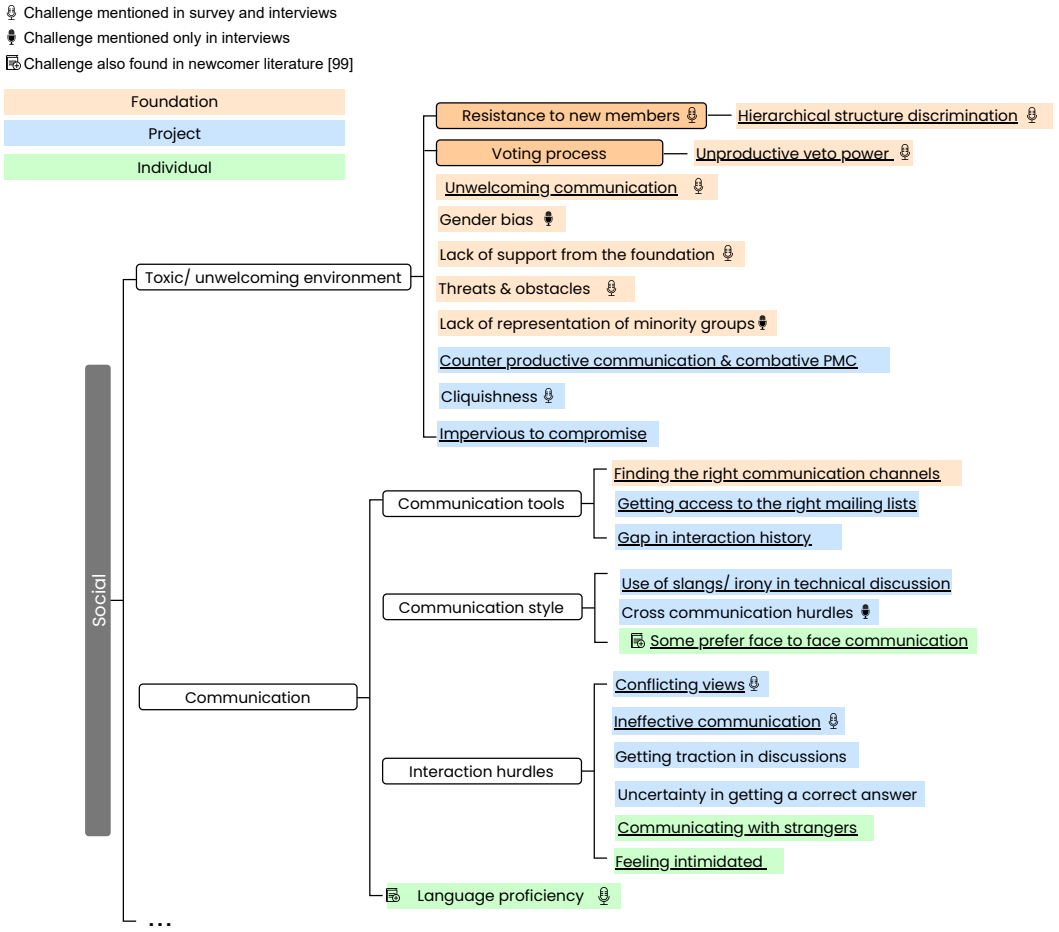

Fig. 8. Challenges related to the Communication and the Toxic/ Unwelcoming Environment subcategories at three levels: Foundation (orange), Project (blue) and Individual (green). The underlined challenges are discussed in the text.

The Toxic/unwelcoming environment is a problem at the **Project** too, with respondents reporting counter-productive communication, combative PMC, leading to an environment that is impervious to compromise. As [S-398] explained, *"There are some very stubborn people who can delay decision-making for too long... who 'can't be wrong'...They make assumptions about other people and lack the ability to see the world from other people's perspective"*.

7.1.1 Strategies. Table 4 presents five **strategies** to combat these challenges. One is to rethink the voting system especially by doing away with the veto power as [I-5] suggested: *"to not require consensus voting for adding committers...because I think consensus voting can sort of reinforce our biases. And by virtue of making it so that any one person's objection, means the candidate can't move forward"*. Another is to promote minority focused meetings so that these contributors can have a safe space to communicate as well as attract more diverse groups. Finally, having some mechanism for calling out misbehavior can help create a healthier community, as [I-3] suggested: *"there needs to be more consequences and there needs to be smaller consequences, and not just the big ones"*.

Table 4. Strategies to mitigate a Toxic/Unwelcoming Environment. The *starred strategies are the ones discussed in the text. The source is indicated by: (S&I) for survey and interviews, (S) for surveys, and (I) for interviews.

Strategies to mitigate Toxic/ Unwelcoming Environment	Description of strategy
Provide guidelines and active support toward an inclusive environment (S&I)	Provide guidelines for a safe and inclusive communication space and provide active support especially to those contributing to difficult circumstances
Include minority groups in the D&I discussion (S&I)	Involve the minority groups in the D&I discussion and decision, provide a trusting environment for them to voice their opinions
*Provide means to keep the community active and healthy (S&I)	Rethinking the unconditional lifetime participation by providing a retiring policy for inactive/ misbehaving contributors at any level and a resumption policy for returning contributors. Implement a way to deal with the misbehaving individual in a gradual manner
*Rethink the voting system (S)	*Remove veto power and find a better conflict management system
*Promote minority focused online meetings (S)	Encourage women involvement by starting a women in apache online meetings

7.2 Communication

Keeping the lines of communication open and inviting is critical in large, distributed environments, where it is difficult to evaluate the tone of conversation via facial expressions or social cues. Communication hurdles can not only make discussions lengthy and tedious, but can also create an unwelcoming environment. In the following section we detail the challenges related to the communication tools, communication styles and online interactions (Fig. 8), as well as the strategies to mitigate these hurdles (Table 5).

7.2.1 Communication tools: The first step towards productive communication is to find the right communication tool. But, doing so can be challenging especially when faced with multiple options in a large organization. Fig. 8 depicts the challenges related to communication tools.

One challenge is that the large number of communication channels makes it difficult to pick the right channel for the specific situation (**Foundation**). For example, different topics could be ear-marked for different mailing lists or even different communication media. As [S-407] mentioned *"Find the right communications channels, which depends not only on hard facts, but also on people."* Finding and getting access to the right communication channels is even trickier when there is no systematic transition to a communication channel after acquiring a new role. For example, some contributors were not even aware of the existence of some channels as they progressed through their roles in the project (**Project**). As [S-468] reported *"not automatically invited to the private@ list...never knew about it until a long time after I became part of the PMC!"*

Even with the right communication tools at hand, gaps in interaction history can cause communication hurdles. Such gaps can arise when someone joins an already ongoing discussion. As [S-317] said *"Communication mediums were hard to follow, especially IRC... If you get offline and come back few hours later, there is no history of what happened"*. This is particularly challenging for those contributors who are volunteers with only a few hours to spare.

7.2.2 Communication style: Another step towards productive communication is to foster an inclusive communication style (**Project**). One way to foster inclusive communication is to avoid unnecessary jargon, which makes discussions difficult to follow and understand, especially for non-native English speakers. As [S-259] explains, *"it is still hard to understand phrases, slangs or irony from native speakers on operational lists"*. It is also important to be aware of the communication preferences of different contributors. For example, the absence of in-person communications (**Individual**) can adversely impact some contributors: *"Getting up to speed with the technical aspects of the particular project I was working on was more difficult due to the lack of face to face contact with other project participants"* [S-629].

7.2.3 Interaction hurdles: As with communication in any team, interactions in OSS can have its own hurdles. For example, conflicting views (**Project**) over the different aspects of a project can result in disagreements such as *"Technical dispute over architectural direction of a project"* [S-431]. When this is paired with ineffective communication resulting from an overwhelming amount of communication or mismatch in communication styles, conflicting views can escalate making it hard to reach an agreement. From an individual's perspective (**Individual**), interacting with community members can be intimidating, especially when there are disagreements. This is made worse when interactions are with strangers. As [S-382] said *"Sending an email (on the mailing lists) to unknown people is nerve-raking"*.

7.2.4 Strategies. Table 5 present eight **strategies** to combat communication challenges. One is to create communication best practices to foster meaningful and inclusive conversations that allow discussions to stay focused on the topic. Similarly, creating a list of helpful, knowledgeable contacts can guide contributors to reach productive and safe communication. As [S-362] suggests *"kind of 'help hot line'?... so that they can get help in communications and avoiding misunderstandings"*. Finally, it is useful to leverage both public and private channels depending on the situation. As [I-2] recommends *"if you have a dispute with somebody, I think it's often much easier to resolve that in private... and it's easier to speak to one person than speak to 100"*.

8 DISCUSSION

Social interactions are challenging in a large, federated organization. In an extremely large organization, such as the ASF, which is mainly powered by a large transient volunteer contributor base, creating effective communication structures can be especially challenging. In fact, our respondents reported more social than technical challenges when contributing to the ASF. These challenges likely arise because the ASF includes contributors from different backgrounds, subgroups, and philosophies who discuss technical as well as governing aspects of the Foundation. Indeed, participants reported finding the right communication channels, getting access to the right mailing lists, and ineffective communication to be challenging. Communication has also been found to be challenging in other open collaboration communities [24, 25, 61, 77, 90]. Many of these challenges may arise because contributors in large peer-production sites have to coordinate and communicate with unfamiliar teams [55, 66]. Further, as discussed by Brooks Jr [8], the amount of communication (and communication overhead) increases combinatorially based on the size of the team.

Table 5. Strategies to minimize Communication Hurdles. The *starred strategies are the ones discussed in the text. The source is indicated by: (S&I) for survey and interviews, (S) for surveys, and (I) for interviews.

Strategies for effective communication	Description of strategy
*Create communication best practices with suggestions (S&I)	Create communication guidelines with suggestion and encourage summarizing and avoiding the use of jargon and complicated English
Transition from tacit to explicit (S&I)	Document best practices, tools for communication and provide a process for conflict resolution
*Leverage both public and private channels and disclose their visibility (S&I)	Praise in public, coach and critique in private and disclose the visibility of each communication mailing list
Automate a role-based sign up to mailing list (S)	Provide a better alternative to signing up on a mailing list where people with different roles are automatically signed up to the right mailing lists
*Create a list of helpful contacts (S)	Provide a list of helpful contacts, a communication group and a help hotline for contributors who face communication difficulties
Leverage real time communication (I)	Incorporate real time communication tools to foster connection and a sense of belonging and create guideline on how and when to use real time communication
Provide the option of offline training when possible (I)	Organize offline training to help those who prefer face-to-face interactions
Use productivity management strategies (S&I)	Give yourself time to adapt, set a goal for yourself, seek help when needed and leverage grammar tool to help with English

Further, the hierarchical leadership structure adopted by the ASF was found to be challenging by some, especially those who were joining the ASF from commercial companies. Marlow et al. [70] argue that a team's underlying leadership structure moderates the relationship between the team's communication and its performance, such that the relationship is stronger for teams with a shared leadership configuration as compared to those with a hierarchical leadership structure. Hoch and Kozlowski [50] reinforces this idea, claiming that supplementing hierarchical leadership with shared leadership and structural support is more important when teams are more virtual in nature. The strategies recommended by participants point to the need for shared leadership–"give projects more agency...and accommodate different ways of working" as well as the need for clearer guidance on the governance structure, including publishing the hierarchical structure in the ASF.

Toxic environment remains an issue in open source. Open source communities, including the ASF, are highly dependent on volunteer contributions, but toxic work environments with unfriendly or unhelpful contributors [86, 108] can lead to high turnover. In OSS, where all communications are public, communicating with contributors with no shared familiarity and in perceived hostile environments can be intimidating, as in the words of [S-382]: *"Sending an email to unknown people is nerve racking"*. Our participants identified 10 challenges (and five strategies to mitigate them) related to a toxic environment. Gender bias and lack of representation of minority groups were among those. Indeed, women are a minority group in OSS [7, 69, 117, 125], and have been reported to get delayed feedback during code reviews [7, 20], are frequent recipients of negative reviews with words that are demeaning to women [83], and even have lower code acceptance rates [113] when they are newcomers. Women are usually more restrained in the discussion forums, regardless of how senior or productive they are [9] which can drown out voices in a "pushyocracy" system [79]. Given such existing biases, women often use pseudonyms in their profiles to avoid being judged as females [27, 57].

Task interdependence, contrasting contribution ideology, and geographical distribution can also increase overall conflict levels and toxicity in OSS teams [25]. For example, some participants found it challenging that the ASF solely depends on online communications, where differences in communication styles can be amplified leading to some PMC members appearing combative. These factors can create interaction hurdles as [S-557] found that *"interacting with community members can be intimidating"*. According to Filippova and Cho [25], participatory decision-making and a transformational leadership style can help to reduce this challenge. In fact, participants

recommended creating such a participatory environment by including minority groups in decision-making such that their voices are heard.

OSS is mature, but contribution and decision making processes remain challenging. The contribution model of OSS has evolved significantly from the early 2000's, when groups of talented software hackers volunteered to produce software, into a more mainstream and commercially viable model [26, 92, 102]. In this changed landscape, well-known companies and foundations are not only using OSS, but also open-sourcing their products and joining and managing OSS communities [92]. However, even after 20 years of open source production, the very processes in creating such peer-produced products still remain a challenge. Actually, the very popularity of the OSS model, which has created large foundations with hybrid-contribution models that also serve as career pathways [115], may be creating additional process-related challenges. Challenges in navigating the contribution process was heavily cited by our participants and amounted to 70% of all process-related challenges in our study. These challenges included the very basics of the contribution processes, such as creating a patch to submit code [2, 58]). This suggests that communities need to monitor the current state of practice and identify bottlenecks and points for improvement [111] to their processes.

There is no one-size-fits-all. Over the years, the Apache Software Foundation has collectively organized best-practices and principles designed to enable disparate communities of volunteers to collaborate and produce high-quality software and documentation. Although a common set of foundation-wide rules brings uniformity to the projects and facilitates migration across projects and interoperability [53], the ensuing need for compliance to complex "regulations" can be challenging. Such type of dissatisfaction with regulations and administration are a key reason for volunteers quitting OSS projects [128]. The challenges reported by our participants reflect this tension with the foundation regulations, ranging from the vagueness of regulations to the restrictiveness of legal matters, such as the need for approved CLAs (Contributor License Agreement) before starting contributions. Some contributors found the rules to be restrictive or confusing leading to communication overheads and delays, whereas others felt there was too much flexibility with no clear rules (see Fig. 6). Creating the right balance in guidelines is challenging for large, federated foundations such as the ASF, which prides itself on its flexibility and aggregates projects of different sizes, ages, and domains. Industry involvement further complicates the legal matters, governance, and contribution culture. As Zhou et al. [130] and Daniel et al. [16] point out, commercial involvement can generate conflicts of interest, differing contribution ideology, and vision of future project directions, which can in turn negatively affect coordination and work-related behavior. Setting up expectations is thus important in this context; which our participants found to be poorly documented both at the project and foundation levels.

Comparison with previous literature. We found several challenges (68% of the challenges we identified) that were not reported in the previous work on challenges faced by OSS contributors [40, 49, 52, 58, 65, 85, 98, 99, 101, 106, 108, 113, 114, 116, 124][1], as presented in Table 6 (see supplemental [44] for detailed view of this comparison). The majority of the newly identified challenges relate to processes (79%) and social aspects (74%), probably due to our foundation perspective, which is the origin of most of the newly identified challenges (88%).

As noted in Table 6, there is a considerable intersection (54%) between technical factors covered by previous work and the ones that we identified (see Table 7). A majority of these were reported in the literature on newcomers' barriers [99, 106]. For example, previous literature corroborates

[1]We selected these previous works by using a snowballing sampling technique using the work from Steinmacher et al. [99] as a starting point. We analyzed papers that cited this work (forward snowballing) or had been cited by it (backward snowballing). We repeated this process for all relevant papers, until we found no new papers that reported challenges for contributing to OSS.

Table 6. The number and percentage of challenges not found in related work about newcomer barriers and OSS challenges.

	Foundation	Project	Individual	**Total**
Technical	6 (75%)	4 (44%)	1 (14%)	**11 (46%)**
Social	8 (80%)	12 (86%)	3 (43%)	**23 (74%)**
Process	15 (100%)	8 (73%)	3 (43%)	**26 (79%)**
Total	**29 (88%)**	**24 (71%)**	**7 (33%)**	**60 (68%)**

that, at the individual level, the lack of technical background and domain expertise can hinder the contribution process [99, 101, 108]. Our results indicate that in a large OSS foundation such as the ASF, even experienced contributors face barriers previously associated with newcomers. This might be because ASF contributors often participate in multiple projects that have different characteristics and use different tools and technologies, which contributors need to learn when they join a different project. In our work, we also identified other project-specific technical challenges, such as multiple programming languages within and between projects that could be challenging even for experienced contributors (Fig. 7 in Section 6). Some of the technical challenges that our work reported at the foundation level was also reported by prior work, but in the context of a single project. This is probably due to the lack of research on challenges at the organization level [43].

Table 7. The list of challenges found in literature. The subcategory (number) represents the total number of challenges found in that category. For each evidenced challenge the (letter) describes the level at which the challenge appeared in our finding: (F)- foundation level, (P)- project level, and (I)- individual level. The *starred subcategories are the ones that are discussed in details in the results section (see sections 5, 6, and 7 and the supplemental document [44] for more details).

Technical subcategories	**Challenges evidenced by literature**	**Literature source**
*Technical hurdles (12)	Bad code quality (P)	[99]
	Code complexity/ instability (P)	[99] [58]
	Sparse code comments (P)	[58]
	Familiarize with project coding styles (I)	[40]
	Understanding architecture/code structure (I)	[58] [99]
	Infrastructure hurdles (F)	[49] [40] [124]
	Domain expertise (I)	[99]
	Specific programming languages (I)	[58] [99] [101]
	Specific version control systems (I)	[99]
	Specific tools & technologies (I)	[108]
Documentation problems (12)	Fragmented documentation (F)	[99]
	Unwieldy documentation (F)	[58]
	Outdated documentation (P)	[99]
Social subcategories	**Challenges evidenced by literature**	**Literature source**
*Toxic/ unwelcoming environment (10)	Hierarchical structure discrimination (F)	[40]
	Gender bias (F)	[114]
*Communication (13)	Ineffective communication (P)	[108] [40]
	Feeling intimidated (I)	[108]
	Some prefer face to face communication (I)	[99]
	Language proficiency (I)	[108] [99]
Geolocation (3)	Remote/ asynchronous communication (P)	[108]
	Imposter syndrome/ fear of making mistakes / self doubt (I)	[40][124] [101]
Process subcategories	**Challenges evidenced by literature**	**Literature source**
*Navigating the contribution process (23)	Delay in getting contributions accepted/ reviewed (P)	[99] [40]
	Getting contribution accepted (I)	[99]
	Issue to create a patch (I)	[99] [58]
Reception issues (4)	Not receiving an answer (P)	[99] [106]
	Delayed answers (P)	[99]
Getting started in a new project (3)	Selecting a project/ task (I)	[99]
	Familiarize with project members (I)	[49]

Social and process related challenges are less evidenced by prior work (26% and 21% respectively). Similarly to the technical challenges, most of the previously identified process and social challenges appear at the individual or project level (see Table 7).

While we found that a variety of toxic/unwelcoming environment challenges—especially ones that appear at the foundation level—can worsen and deter social interaction (see Fig. 8, Section 7), two of these challenges have also been found at the project level. For instance, Gousios et al. [40] found that hierarchical structure discrimination can be problematic at the project level, especially when project owners do not want contributions and are resistant to new members' participation. Finally, most of the process-related challenges unveiled here could not be found in the examined literature (79%, as shown in Table 6). Our findings show that the process challenges identified by prior literature exist in the ASF, but a variety of new challenges arises especially at the foundation level. In fact, a large OSS foundation usually has its own processes and policies such as the voting and legal process which can be challenging not only to already established ASF projects but also to incubating projects. For instance, the ASF contributors found the incubation process to be bureaucratic and tedious which can make a project's compliance with the foundation processes and ascension to top-level a nontrivial endeavor (see Fig. 6, section 5).

Implications for researchers. Although the ASF community provided many ideas for strategies, there remain gaps in how to operationalize these strategies to mitigate the extensive list of challenges that still plague OSS. Our findings serve as a call for future research in understanding which strategies can be applied in what context. For example, social barriers are challenging and sparsely covered by the reported strategies. Researchers can use our findings to plan further qualitative and quantitative studies to investigate specific challenges. In particular, some of the solutions and mitigation strategies used in other CSCW domains and in the CMC (Computer Mediated Communication) field could serve as models for supporting communication and coordination.

On the other hand, identifying and organizing the challenges faced by contributors in a large OSS organization can benefit the larger CSCW community as large-scale, open collaboration models are still under-explored in the literature. Our results can serve as a reference to guide further research in different kinds of peer-production communities. For example, which of our results can be transferred and adapted? Which of the challenges are common to other communities? How these challenges or strategies need to be adopted given differences in the context of production? Such broader investigations can help build more general models and theories about contribution challenges in open collaboration communities working as a federation.

Implications for practitioners. Finally, our study results provide insights for communities and large organizations who want to reduce the challenges that contributors face, laying a foundation for designing better support tools and strategies. Additionally, the empirical evidences of the different challenges catalogued here is important as many studies and practical strategies are motivated by or deal with anecdotal evidence.

9 THREATS TO VALIDITY

This section presents the limitations and the reliability of our results from the perspective proposed by Merriam [72] for qualitative work.

Construct validity in qualitative research is related to the definition of the constructs. The first issue can arise if we ask incorrect questions in our survey or interviews. We sought to mitigate this challenge by working closely with the D&I committee at the ASF to craft the survey and interview questions such that it was accessible to the community. The second validity can arise in the qualitative analysis process. To avoid misinterpretation in the qualitative coding of the data, we used the constant comparison method. As new codes emerged, the first author compared it with

the existing code set and met frequently with the research team to discuss and clarify the codes. The code set generated from the survey results were then compared with the interview data.

Internal validity is related to the credibility that researchers were able to capture the reality as close as possible, which in our case is accurately capturing the challenges and strategies. The first challenge can arise from a biased sampling of the ASF contributors. We believe this limitation was low as we deployed the survey widely receiving 600+ respondents who represented a wide set of demographics and projects. We then followed up the survey with interviews to get a deeper understanding of the challenges. We reached saturation after the fourth interview where no new challenges or strategies were identified. Finally, we validated our findings with the community via a two-step member checking process. First, we received feedback from the interviewees and then from the broader D&I committee at the ASF. Additionally, we compared our findings with the literature on software engineering and OSS to sharpen construct definitions and increase internal validity.

Reliability refers to the extent that the results can be replicated. In short, it is difficult to replicate qualitative research since human behaviors, feelings, and perceptions change over time. As Merriam [73] mentions: there can be numerous interpretations of the same data for qualitative research. However, the more important question is whether the results are consistent with the data collected. To maintain consistency, three researchers independently coded the survey answers via open coding, which were constantly compared with already existent codes. Additionally, two researchers independently coded the interview transcripts using the code set from the survey. All researchers met multiple times to reach a negotiated agreement. To allow replication of our study, we describe our method in detail in the methodology section above.

10 CONCLUSION

Large organizations such as, the Apache, CNCF, Linux, Eclipse, Gnome, and Mozilla foundations, provide operational, legal, and financial support for a broad range of open source software projects, as well as establish best practices, contribution processes, and governance models. However, the heterogeneity of the individual projects and the large number of stakeholders create a complex ecosystem that involves individuals and companies from different cultures and backgrounds with different goals and expectations. This work is the first to investigate the comprehensive set of challenges of a large and mature OSS foundation, extending the current literature that focuses primarily on challenges faced by individuals and projects.

We empirically identified *88 ongoing challenges that contributors face and 48 strategies to mitigate such challenges in the Apache Software Foundation.* We analyzed these challenges from two different perspectives: (a) Process, Technical, and Social dimensions; and (b) Foundation, Project, and Individual levels. Not surprisingly, in an organization with a large number of mature and technically experienced contributors, individual and technical challenges are less common and the majority of challenges relate to the Process (33) and Social (31) dimensions and occur at the Foundation (33) and Project (34) levels.

Contributors reported several process-related challenges, many of which occurred at the Foundation level and arose from difficulty in implementation of and compliance with the foundation rules and governance. Our results show that despite decades of evolution, mature OSS foundations still face challenges in designing a comprehensive set of guidelines that can cater to the needs and expectations of a diverse set of projects and contributors.

Social interactions are also particularly challenging in the context of a large, federated organization. Participants reported limitations in the communication tools in use, communication styles, and interaction hurdles arising from conflicting views, an inability to get traction in discussions, or feeling intimidated. Participants also reported challenges related to a toxic or unwelcoming

environment, which can be especially detrimental to organizations that are highly dependent on volunteer contributions. Our findings show that the communication culture and discussion tone need to be set at the Foundation level itself to promote a welcoming environment for projects and individuals.

In summary, our results provide insights for current foundations and for projects that want to aggregate under larger organizations, laying a foundation for building better support structures for contributors. With our findings, large organizations can design strategies, processes, and tools to overcome anticipated challenges at the Foundation, Project, and Individual levels. In addition, researchers can use our model to plan further qualitative and quantitative studies to investigate specific challenges, their interplay, and their in-practice implications. Special attention is needed for those challenges not covered by previous works and not linked to specific strategies.

To conclude, through this work we have taken the first steps in identifying ongoing challenges that contributors face in a large heterogeneous organization and also collected a set of strategies that participants recommended. In the future, we plan to perform subgroup analyses to investigate which of these challenges impact specific demographics (e.g., experts vs. novices, paid vs. unpaid, gender, communication skills, place of residence). We also plan to work with the ASF in implementing some of these strategies to help it attract and retain a diverse set of contributors and help it reach its guiding principle of "community over code". We also hope that other organizations also recognize their Foundation-level challenges and implement strategies to create a diverse, successful ecosystem that not only creates successful products, but also creates healthy and thriving communities.

ACKNOWLEDGEMENTS

We thank the ASF board of directors for their help in promoting D&I and all the survey respondents and interviewees for their time and the insights. We thank Zixuan Feng for his help in data visualization. This work is partially supported by the National Science Foundation (grants 1815486, 1815503, 1900903, and 1901031), and CNPq (grant #313067/2020-1).

REFERENCES

[1] Sogol Balali, Umayal Annamalai, Hema Susmita Padala, Bianca Trinkenreich, Marco A Gerosa, Igor Steinmacher, and Anita Sarma. 2020. Recommending Tasks to Newcomers in OSS Projects: How Do Mentors Handle It?. In *Proceedings of the 16th International Symposium on Open Collaboration*. 1–14.

[2] Sogol Balali, Igor Steinmacher, Umayal Annamalai, Anita Sarma, and Marco Aurelio Gerosa. 2018. Newcomers' barriers... is that all? an analysis of mentors' and newcomers' barriers in OSS projects. *Computer Supported Cooperative Work (CSCW)* 27, 3-6 (2018), 679–714.

[3] Julian M Bass, Beecham Sarah, Mohammed Abdur Razzak, and John Noll. 2018. Employee retention and turnover in global software development: comparing in-house offshoring and offshore outsourcing. In *2018 IEEE/ACM 13th Int'l Conf. on Global Software Engineering (ICGSE)*. IEEE, 77–86.

[4] Yochai Benkler and Helen Nissenbaum. 2006. Commons-based peer production and virtue. *Journal of political philosophy* 14, 4 (2006), 394–419.

[5] Bitergia. 2018. Gender Diversity Analysis in the OpenStack Community. https://superuser.openstack.org/wp-content/uploads/2018/06/Gender-Diversity-Analysis-in-the-OpenStack-Community-2018.pdf. [Online; accessed 2020-12-30].

[6] Andreas Böhm. 2004. Theoretical Coding: Text Analysis in. *A companion to qualitative research* 1 (2004).

[7] Amiangshu Bosu and Kazi Zakia Sultana. 2019. Diversity and inclusion in open source software (OSS) projects: Where do we stand?. In *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. IEEE, 1–11.

[8] Frederick P Brooks Jr. 1995. The mythical man-month (anniversary ed.). (1995).

[9] Gemma Catolino, Fabio Palomba, Damian A Tamburri, Alexander Serebrenik, and Filomena Ferrucci. 2019. Gender diversity and women in software teams: How do they affect community smells?. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. IEEE, 11–20.

[10] Derek Clements-Croome. 2006. *Creating the productive workplace*. Taylor & Francis.

[11] Jailton Coelho and Marco Tulio Valente. 2017. Why modern open source projects fail. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*. 186–196.
[12] Juliet M Corbin. 1990. *Basics of qualitative research: Grounded theory procedures and techniques*. Sage.
[13] Kevin Crowston, James Howison, and Hala Annabi. 2006. Information systems success in free and open source software development: Theory and measures. *Software Process: Improvement and Practice* 11, 2 (2006), 123–148.
[14] Davor Cubranic, Gail C Murphy, Janice Singer, and Kellogg S Booth. 2005. Hipikat: A project memory for software development. *IEEE Transactions on Software Engineering* 31, 6 (2005), 446–465.
[15] Laura Dabbish, Colleen Stuart, Jason Tsay, and Jim Herbsleb. 2012. Social coding in GitHub: transparency and collaboration in an open software repository. In *Proceedings of the ACM 2012 conference on computer supported cooperative work*. 1277–1286.
[16] Sherae Daniel, Likoebe Maruping, Marcelo Cataldo, and James Herbsleb. 2011. When cultures clash: Participation in open source communities and its implications for organizational commitment. (2011).
[17] Open demographics. 2021. Open demographics documentation. http://nikkistevens.com/open-demographics/. [Online; accessed 2020-12-30].
[18] Nicolas Ducheneaut. 2005. Socialization in an open source software community: A socio-technical analysis. *Computer Supported Cooperative Work (CSCW)* 14, 4 (2005), 323–368.
[19] J. C. Duenas, H. A. Parada G., F. Cuadrado, M. Santillán, and J. L. Ruiz. 2007. Apache and Eclipse: Comparing Open Source Project Incubators. *IEEE Software* 24, 6 (2007), 90–98. https://doi.org/10.1109/MS.2007.157
[20] Carolyn D Egelman, Emerson Murphy-Hill, Elizabeth Kammer, Maggie Morrow Hodges, Collin Green, Ciera Jaspan, and James Lin. 2020. Pushback: Characterizing and Detecting Negative Interpersonal Interactions in Code Review.
[21] Omar Elazhary, Margaret-Anne Storey, Neil Ernst, and Andy Zaidman. 2019. Do as I Do, Not as I Say: Do Contribution Guidelines Match the GitHub Contribution Process?. In *2019 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 286–290.
[22] Joseph Feller, Patrick Finnegan, Björn Lundell, and Olof Nilsson. 2008. The servicitization of peer production: Reflections on the open source software experience. In *Information technology in the service economy: Challenges and possibilities for the 21st century*. Springer, 353–355.
[23] Chaim Fershtman and Neil Gandal. 2007. Open source software: Motivation and restrictive licensing. *International Economics and Economic Policy* 4, 2 (2007), 209–225.
[24] Anna Filippova and Hichang Cho. 2015. Mudslinging and manners: Unpacking conflict in free and open source software. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. 1393–1403.
[25] Anna Filippova and Hichang Cho. 2016. The effects and antecedents of conflict in free and open source software development. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. 705–716.
[26] Brian Fitzgerald. 2006. The transformation of open source software. *MIS quarterly* 30, 3 (2006), 587–598.
[27] Denae Ford, Mahnaz Behroozi, Alexander Serebrenik, and Chris Parnin. 2019. Beyond the code itself: how programmers really look at pull requests. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. IEEE, 51–60.
[28] Denae Ford, Justin Smith, Philip J Guo, and Chris Parnin. 2016. Paradise unplugged: Identifying barriers for female participation on stack overflow. In *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*. 846–857.
[29] Andrea Forte and Cliff Lampe. 2013. Defining, understanding, and supporting open collaboration: Lessons from the literature. *American Behavioral Scientist* 57, 5 (2013), 535–547.
[30] Matthieu Foucault, Marc Palyart, Xavier Blanc, Gail C Murphy, and Jean-Rémy Falleri. 2015. Impact of developer turnover on quality in open-source software. In *Proc. of the 2015 10th Joint Meeting on Foundations of Software Engineering*. ACM, 829–841.
[31] Apache Software Foundation. 1999. Apache Software Foundation. https://www.apache.org/. [Online; accessed 2019-05-17].
[32] Apache Software Foundation. 1999. Apache Software Foundation. https://www.apache.org/theapacheway/. [Online; accessed 2019-06-15].
[33] Apache Software Foundation. 2016. ASF Committer Diversity Survey. https://cwiki.apache.org/confluence/display/COMDEV/ASF+Committer+Diversity+Survey+-+2016. [Online; accessed 2020-12-30].
[34] Apache Software Foundation. 2019. Promoting and Studying Diversity and Inclusion in Open Source. http://diversity.apache.org/. [Online; accessed 2019-06-10].
[35] Andrew N. Garman, Josie Corbett, Jane Grady, and Julie Benesh. 2005. Ready-to-use-simulation: THE HIDDEN COSTS OF EMPLOYEE TURNOVER. *Simulation & Gaming* 36, 2 (2005), 274–281. https://doi.org/10.1177/1046878104273254

[36] Svetla Georgieva and George Allan. 2008. Best Practices in Project Management Through a Grounded Theory Lens. *Electronic Journal of Business Research Methods* 6, 1 (2008).

[37] Matt Germonprez, Georg JP Link, Kevin Lumbard, and Sean Goggins. 2018. Eight observations and 24 research questions about open source projects: illuminating new realities. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–22.

[38] Marco Gerosa, Igor Wiese, Bianca Trinkenreich, Georg Link, Gregorio Robles, Christoph Treude, Igor Steinmacher, and Anita Sarma. 2021. The Shifting Sands of Motivation: Revisiting What Drives Contributors in Open Source. In *Proceedings of the 43rd International Conference on Software Engineering (ICSE)*.

[39] BG Glaser. [n.d.]. Theoretical Sensitivity: Advances in the Methodology of Grounded Theory. 1978, Mill Valley, Calif.

[40] Georgios Gousios, Margaret-Anne Storey, and Alberto Bacchelli. 2016. Work practices and challenges in pull-based development: the contributor's perspective. In *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE)*. IEEE, 285–296.

[41] Michelle I Graef and Erick L Hill. 2000. Costing Child Protective Services Staff Turnover. *Child Welfare* 79, 5 (2000).

[42] Jonathan Grudin. 1994. Computer-supported cooperative work: history and focus. *Computer* 27, 5 (1994), 19–26.

[43] Jonathan Grudin. 1994. Groupware and social dynamics: Eight challenges for developers. *Commun. ACM* 37, 1 (1994), 92–105.

[44] Mariam Guizani, Amreeta Chatterjee, Bianca Trinkenreich, May Mary Evelyn, Geraldine Jimena Noa, Liam James Russell, Griselda Cuevas Zambrano, Daniel Izquierdo-Cortazar, Igor Steinmacher, Marco Aurelio Gerosa, and Anita Sarma. 2021. Supplemental Document. https://figshare.com/s/629364cce93f6e9774c3.

[45] Aaron Halfaker and R Stuart Geiger. 2020. Ores: Lowering barriers with participatory machine learning in wikipedia. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW2 (2020), 1–37.

[46] Aaron Halfaker, Aniket Kittur, and John Riedl. 2011. Don't bite the newbies: how reverts affect the quantity and quality of Wikipedia work. In *Proceedings of the 7th international symposium on wikis and open collaboration*. 163–172.

[47] Tracy Hall, Sarah Beecham, June Verner, and David Wilson. 2008. The impact of staff turnover on software projects: the importance of understanding what makes software practitioners tick. In *Proc. of the 2008 ACM SIGMIS CPR Conf. on Computer personnel doctoral consortium and research*. ACM, 30–39.

[48] Christoph Hannebauer, Matthias Book, and Volker Gruhn. 2014. An exploratory study of contribution barriers experienced by newcomers to open source software projects. In *Proceedings of the 1st International Workshop on CrowdSourcing in Software Engineering*. 11–14.

[49] Christoph Hannebauer and Volker Gruhn. 2017. On the relationship between newcomer motivations and contribution barriers in open source projects. In *Proceedings of the 13th International Symposium on Open Collaboration*. 1–10.

[50] Julia E Hoch and Steve WJ Kozlowski. 2014. Leading virtual teams: Hierarchical leadership, structural supports, and shared team leadership. *Journal of applied psychology* 99, 3 (2014), 390.

[51] Gary Hsieh, Youyang Hou, Ian Chen, and Khai N. Truong. 2013. "Welcome!": Social and Psychological Predictors of Volunteer Socializers in Online Communities. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work (CSCW '13)*. ACM, New York, NY, USA, 827–838. https://doi.org/10.1145/2441776.2441870

[52] Carlos Jensen, Scott King, and Victor Kuechler. 2011. Joining Free/Open Source Software Communities: An Analysis of Newbies' First Interactions on Project Mailing Lists. In *Proceedings of the 2011 44th Hawaii International Conference on System Sciences (HICSS '11)*. IEEE Computer Society, Washington, DC, USA, 1–10. https://doi.org/10.1109/HICSS.2011.264

[53] Corey Jergensen, Anita Sarma, and Patrick Wagstrom. 2011. The onion patch: migration in open source ecosystems. In *Proceedings of the 19th ACM SIGSOFT symposium and the 13th European conference on Foundations of software engineering*. ACM, 70–80.

[54] Elisabeth Joyce and Robert E Kraut. 2006. Predicting continued participation in newsgroups. *Journal of Computer-Mediated Communication* 11, 3 (2006), 723–747.

[55] Ralph Katz. 1982. The effects of group longevity on project communication and performance. *Administrative science quarterly* (1982), 81–104.

[56] Cliff Lampe and Erik Johnston. 2005. Follow the (slash) dot: effects of feedback on new members in an online community. In *Proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work*. 11–20.

[57] Amanda Lee and Jeffrey Carver. 2019. FLOSS Participants' Perceptions about Gender and Inclusiveness: A Survey. In *41st International Conference on Software Engineering*.

[58] Amanda Lee, Jeffrey C Carver, and Amiangshu Bosu. 2017. Understanding the impressions, motivations, and barriers of one time code contributors to FLOSS projects: a survey. In *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*. IEEE, 187–197.

[59] Charlotte P Lee and Drew Paine. 2015. From The Matrix to a Model of Coordinated Action (MoCA) A Conceptual Framework of and for CSCW. In *Proceedings of the 18th ACM conference on computer supported cooperative work & social computing*. 179–194.

[60] Patrick Chang Boon Lee. 2002. The social context of turnover among information technology professionals. In *Proc. of the 2002 ACM SIGCPR Conf. on Computer personnel research*. ACM, 145–153.
[61] Saerom Lee, Hyunmi Baek, and Jungjoo Jahng. 2017. Governance strategies for open collaboration: Focusing on resource allocation in open source software development organizations. *International Journal of Information Management* 37, 5 (2017), 431–437.
[62] John M Levine and Hoon-Seok Choi. 2004. Impact of personnel turnover on team performance and cognition. In *Team cognition: Understanding the factors that drive process and performance*, E. Salas and S.M. Fioren (Eds.). American Psychological Association, 153—-176.
[63] John M Levine, Richard L Moreland, Linda Argote, and Kathleen M Carley. 2005. *Personnel turnover and team performance*. Technical Report. PITTSBURGH UNIV PA.
[64] Sheen S Levine and Michael J Prietula. 2014. Open collaboration for innovation: Principles and performance. *Organization Science* 25, 5 (2014), 1414–1433.
[65] Bin Lin, Gregorio Robles, and Alexander Serebrenik. 2017. Developer turnover in global, industrial open source projects: Insights from applying survival analysis. In *2017 IEEE 12th International Conference on Global Software Engineering (ICGSE)*. IEEE, 66–75.
[66] Glenn Littlepage, William Robison, and Kelly Reddington. 1997. Effects of Task Experience and Group Experience on Group Performance, Member Ability, and Recognition of Expertise. *Organizational Behavior and Human Decision Processes* 69, 2 (1997), 133 – 147. https://doi.org/10.1006/obhd.1997.2677
[67] Hanna Mäenpää, Simo Mäkinen, Terhi Kilamo, Tommi Mikkonen, Tomi Männistö, and Paavo Ritala. 2018. Organizing for openness: six models for developer involvement in hybrid OSS projects. *Journal of Internet Services and Applications* 9, 1 (2018), 17.
[68] Raghvendra Mall, Mridul Nagpal, Joni Salminen, Hind Almerekhi, Soon-Gyo Jung, and Bernard J Jansen. 2020. Four Types of Toxic People: Characterizing Online Users' Toxicity over Time. In *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society*. 1–11.
[69] A Mani and Rebeka Mukherjee. 2016. A study of FOSS 2013 survey data using clustering techniques. In *2016 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*. IEEE, 118–121.
[70] Shannon L Marlow, Christina N Lacerenza, Jensine Paoletti, C Shawn Burke, and Eduardo Salas. 2018. Does team communication represent a one-size-fits-all approach?: A meta-analysis of team communication and performance. *Organizational Behavior and Human Decision Processes* 144 (2018), 145–170.
[71] Christopher Mendez, Hema Susmita Padala, Zoe Steine-Hanson, Claudia Hilderbrand, Amber Horvath, Charles Hill, Logan Simpson, Nupoor Patil, Anita Sarma, and Margaret Burnett. 2018. Open source barriers to entry, revisited: A sociotechnical perspective. In *Proceedings of the 40th International Conference on Software Engineering*. 1004–1015.
[72] Sharan B Merriam and Elizabeth J Tisdell. 2015. *Qualitative research: A guide to design and implementation*. John Wiley & Sons.
[73] Sharan B Merriam and Elizabeth J Tisdell. 2015. *Qualitative research: A guide to design and implementation*. John Wiley & Sons.
[74] Meta. 2020. Community Insights/Community Insights 2020 Report — Meta, discussion about Wikimedia projects. https://meta.wikimedia.org/w/index.php?title=Community_Insights/Community_Insights_2020_Report&oldid=20459027 [Online; accessed 29-March-2021].
[75] Audris Mockus. 2009. Succession: Measuring transfer of code and developer productivity. In *Proc. of the 31st Int'l Conf. on Software Engineering*. IEEE Computer Society, 67–77.
[76] Audris Mockus. 2010. Organizational volatility and its effects on software defects. In *Proc. of the eighteenth ACM SIGSOFT Int'l symposium on Foundations of software engineering*. ACM, 117–126.
[77] Mark Mortensen and Pamela J Hinds. 2001. CONFLICT & SHARED IDENTITY IN GEOGRAPHICALLY DISTRIBUTED TEAMS.. In *Academy of Management Proceedings*, Vol. 2001. Academy of Management Briarcliff Manor, NY 10510, B1–B6.
[78] David R. Musicant, Yuqing Ren, James A. Johnson, and John Riedl. 2011. Mentoring in Wikipedia: A Clash of Cultures. In *Proceedings of the 7th International Symposium on Wikis and Open Collaboration (WikiSym '11)*. ACM, New York, NY, USA, 173–182. https://doi.org/10.1145/2038558.2038586
[79] Dawn Nafus. 2012. 'Patches don't have gender': What is not open in open source software. *New Media & Society* 14, 4 (2012), 669–683.
[80] Robbie T Nakatsu and Charalambos L Iacovou. 2009. A comparative study of important risk factors involved in offshore and domestic outsourcing of software development projects: A two-panel Delphi study. *Information & Management* 46, 1 (2009), 57–68.
[81] Shaul Oreg and Oded Nov. 2008. Exploring motivations for contributing to open source initiatives: The roles of contribution context and personal values. *Computers in human behavior* 24, 5 (2008), 2055–2073.

[82] Susmita Hema Padala, Christopher John Mendez, Luiz Felipe Dias, Igor Steinmacher, Zoe Steine Hanson, Claudia Hilderbrand, Amber Horvath, Charles Hill, Logan Dale Simpson, Margaret Burnett, et al. 2020. How Gender-biased Tools Shape Newcomer Experiences in OSS Projects. *IEEE Transactions on Software Engineering* (2020).

[83] Rajshakhar Paul, Amiangshu Bosu, and Kazi Zakia Sultana. 2019. Expressions of sentiments during code reviews: Male vs. female. In *2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. IEEE, 26–37.

[84] Lawrence Peters. 2003. Managing software professionals. In *IEMC'03 Proceedings. Managing Technologically Driven Organizations: The Human Side of Innovation and Change*. IEEE, 61–66.

[85] Gustavo Pinto, Igor Steinmacher, and Marco Aurélio Gerosa. 2016. More common than you think: An in-depth study of casual contributors. In *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, Vol. 1. IEEE, 112–123.

[86] Gede Artha Azriadi Prana, Denae Ford, Ayushi Rastogi, David Lo, Rahul Purandare, and Nachiappan Nagappan. 2020. Including Everyone, Everywhere: Understanding Opportunities and Challenges of Geographic Gender-Inclusion in OSS. *ACM Transactions on Software Engineering and Methodology (TOSEM)* (2020).

[87] CHAOSS Project. 2017. Diversity and inclusion metrics. https://github.com/chaoss/wg-diversity-inclusion/tree/master/focus-areas. [Online; accessed 2020-12-30].

[88] Ayushi Rastogi, Nachiappan Nagappan, Georgios Gousios, and André van der Hoek. 2018. Relationship between geographical location and evaluation of developer contributions in github. In *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*. 1–8.

[89] Eric Raymond. 1999. The cathedral and the bazaar. *Knowledge, Technology & Policy* 12, 3 (1999), 23–49.

[90] Ruqin Ren and Bei Yan. 2017. Crowd diversity and performance in Wikipedia: The mediating effects of task conflict and communication. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 6342–6351.

[91] Yuqing Ren, Jilin Chen, and John Riedl. 2016. The impact and evolution of group diversity in online open collaboration. *Management Science* 62, 6 (2016), 1668–1686.

[92] Gregorio Robles, Igor Steinmacher, Paul Adams, and Christoph Treude. 2019. Twenty Years of Open Source Software: From Skepticism to Mainstream. *IEEE Software* 36, 6 (2019), 12–15.

[93] Rien Sach, Helen Sharp, and Marian Petre. 2011. Software engineers' perceptions of factors in motivation: The work, people, obstacles. In *2011 International Symposium on Empirical Software Engineering and Measurement*. IEEE, 368–371.

[94] Sonali K Shah. 2006. Motivation, governance, and the viability of hybrid forms in open source software development. *Management science* 52, 7 (2006), 1000–1014.

[95] Jefferson Silva, Igor Wiese, Daniel German, Christoph Treude, Marco Gerosa, and Igor Steinmacher. 2020. A Theory of the Engagement in Open Source Projects via Summer of Code Programs. In *Proceedings of the ACM Symposium on the Foundations of Software Engineering (FSE 2020) (FSE '20)*.

[96] stackoverflow. 2018. StackOverflow: developer survey results. http://bit.ly/35D9ycQ. [Online; accessed 2020-12-30].

[97] Igor Steinmacher, Ana Paula Chaves, Tayana Conte, and Marco Aurélio Gerosa. 2014. Preliminary empirical identification of barriers faced by newcomers to Open Source Software projects.. In *Proceedings of the 28th Brazilian Symposium on Software Engineering (SBES '14)*. IEEE Computer Society, 51–60.

[98] Igor Steinmacher, Tayana Conte, and Marco Aurélio Gerosa. 2015. Understanding and Supporting the Choice of an Appropriate Task to Start With In Open Source Software Communities. In *2015 48th Hawaii International Conference on System Sciences (HICSS '15)*. IEEE, 5299–5308.

[99] Igor Steinmacher, Tayana Conte, Marco Aurélio Gerosa, and David Redmiles. 2015. Social barriers faced by newcomers placing their first contribution in open source software projects. In *Proceedings of the 18th ACM conference on Computer supported cooperative work & social computing*. 1379–1392.

[100] Igor Steinmacher, Tayana Uchoa Conte, Christoph Treude, and Marco Aurélio Gerosa. 2016. Overcoming open source project entry barriers with a portal for newcomers. In *Proceedings of the 38th International Conference on Software Engineering*. 273–284.

[101] Igor Steinmacher, Gustavo Pinto, Igor Scaliante Wiese, and Marco Aurélio Gerosa. 2018. Almost there: A study on quasi-contributors in open-source software projects. In *2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE)*. IEEE, 256–266.

[102] Igor Steinmacher, Gregorio Robles, Brian Fitzgerald, and Anthony Wasserman. 2017. Free and open source software development: the end of the teenage years. *Journal of Internet Services and Applications* 8, 1 (05 Dec 2017), 17. https://doi.org/10.1186/s13174-017-0069-9

[103] Igor Steinmacher, Marco Aurelio Graciotto Silva, Marco Aurelio Gerosa, and David F Redmiles. 2015. A systematic literature review on the barriers faced by newcomers to open source software projects. *Information and Software Technology* 59 (2015), 67–85.

[104] Igor Steinmacher, Marco Aurélio Graciotto Silva, Marco Aurélio Gerosa, and David F. Redmiles. 2015. A systematic literature review on the barriers faced by newcomers to open source software projects. *Information and Software Technology* 59 (March 2015), 67–85. https://doi.org/10.1016/j.infsof.2014.11.001

[105] Igor Steinmacher, Christoph Treude, and Marco Aurelio Gerosa. 2018. Let me in: Guidelines for the successful onboarding of newcomers to open source projects. *IEEE Software* 36, 4 (2018), 41–49.

[106] Igor Steinmacher, Igor Wiese, Ana Paula Chaves, and Marco Aurélio Gerosa. 2013. Why do newcomers abandon open source software projects?. In *2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*. IEEE, 25–32.

[107] Igor Steinmacher, Igor Scaliante Wiese, and Marco Aurélio Gerosa. 2012. Recommending mentors to software project newcomers. In *Proceedings of the Third International Workshop on Recommendation Systems for Software Engineering (RSSE '12)*. IEEE Computer Society, Washington, DC, USA, 63–67. https://doi.org/10.1109/RSSE.2012.6233413

[108] Margaret-Anne Storey, Alexey Zagalsky, Fernando Figueira Filho, Leif Singer, and Daniel M German. 2016. How social and communication channels shape and challenge a participatory culture in software development. *IEEE Transactions on Software Engineering* 43, 2 (2016), 185–204.

[109] Anselm Strauss and Juliet M Corbin. 1997. *Grounded theory in practice*. Sage.

[110] Bongwon Suh, Gregorio Convertino, Ed H Chi, and Peter Pirolli. 2009. The singularity is not near: slowing growth of Wikipedia. In *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*. 1–10.

[111] Wikan Sunindyo, Thomas Moser, Dietmar Winkler, and Deepak Dhungana. 2012. Improving open source software process quality based on defect data mining. In *International Conference on Software Quality*. Springer, 84–102.

[112] Marcin Sydow, Katarzyna Baraniak, and Paweł Teisseyre. 2017. Diversity of editors and teams versus quality of cooperative work: experiments on Wikipedia. *Journal of Intelligent Information Systems* 48, 3 (2017), 601–632.

[113] J Terrell, A Kofink, J Middleton, C Rainear, E Murphy-Hill, and C Parnin. 2016. Gender bias in open source: Pull request acceptance of women versus men.(Jan 2016). *PeerJ Computer Science* (2016).

[114] Josh Terrell, Andrew Kofink, Justin Middleton, Clarissa Rainear, Emerson Murphy-Hill, Chris Parnin, and Jon Stallings. 2017. Gender differences and bias in open source: Pull request acceptance of women versus men. *PeerJ Computer Science* 3 (2017), e111.

[115] Bianca Trinkenreich, Mariam Guizani, Igor Wiese, Anita Sarma, and Igor Steinmacher. 2020. Hidden Figures: Roles and Pathways of Successful OSS Contributors. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW2 (2020), 1–22.

[116] Bogdan Vasilescu, Daryl Posnett, Baishakhi Ray, Mark GJ van den Brand, Alexander Serebrenik, Premkumar Devanbu, and Vladimir Filkov. 2015. Gender and tenure diversity in GitHub teams. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*. 3789–3798.

[117] Bogdan Vasilescu, Alexander Serebrenik, and Vladimir Filkov. 2015. A data set for social diversity studies of GitHub teams. In *2015 IEEE/ACM 12th working conference on mining software repositories*. IEEE, 514–517.

[118] Georg Von Krogh, Stefan Haefliger, Sebastian Spaeth, and Martin W Wallin. 2012. Carrots and rainbows: Motivation and social practice in open source software development. *MIS quarterly* (2012), 649–676.

[119] Claudia Wagner, David Garcia, Mohsen Jadidi, and Markus Strohmaier. 2015. It's a man's Wikipedia? Assessing gender inequality in an online encyclopedia. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 9.

[120] Claudia Wagner, Eduardo Graells-Garrido, David Garcia, and Filippo Menczer. 2016. Women through the glass ceiling: gender asymmetries in Wikipedia. *EPJ Data Science* 5 (2016), 1–24.

[121] Jianguo Wang and Anita Sarma. 2011. Which bug should I fix: helping new developers onboard a new project. In *Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering*. 76–79.

[122] Yi-Chia Wang, Robert Kraut, and John M Levine. 2012. To stay or leave? The relationship of emotional and informational support to commitment in online health support groups. In *Proceedings of the ACM 2012 conference on computer supported cooperative work*. 833–842.

[123] P Warr. 1998. What is our current understanding of the relationships between well-being and work. *Journal of Occupational Psychology* 63 (1998), 193–210.

[124] Vincent Wolff-Marting, Christoph Hannebauer, and Volker Gruhn. 2013. Patterns for tearing down contribution barriers to FLOSS projects. In *2013 IEEE 12th International Conference on Intelligent Software Methodologies, Tools and Techniques (SoMeT)*. IEEE, 9–14.

[125] Pavlina Wurzelová, Fabio Palomba, and Alberto Bacchelli. 2019. Characterizing women (not) contributing to open-source. In *2019 IEEE/ACM 2nd International Workshop on Gender Equality in Software Engineering (GE)*. IEEE, 5–8.

[126] Kazuhiro Yamashita, Yasutaka Kamei, Shane McIntosh, Ahmed E Hassan, and Naoyasu Ubayashi. 2016. Magnet or sticky? Measuring project characteristics from the perspective of developer attraction and retention. *Journal of Information Processing* 24, 2 (2016), 339–348.

[127] Murat Yilmaz and Rory V O'Connor. 2011. A software process engineering approach to improving software team productivity using socioeconomic mechanism design. *ACM SIGSOFT Software Engineering Notes* 36, 5 (2011), 1–5.
[128] Yiqing Yu, Alexander Benlian, and Thomas Hess. 2012. An empirical study of volunteer members' perceived turnover in open source software projects. In *2012 45th Hawaii International Conference on System Sciences*. IEEE, 3396–3405.
[129] Minghui Zhou and Audris Mockus. 2010. Developer fluency: Achieving true mastery in software projects. In *Proceedings of the eighteenth ACM SIGSOFT international symposium on Foundations of software engineering*. 137–146.
[130] Minghui Zhou, Audris Mockus, Xiujuan Ma, Lu Zhang, and Hong Mei. 2016. Inflow and retention in oss communities with commercial involvement: A case study of three hybrid projects. *ACM Transactions on Software Engineering and Methodology (TOSEM)* 25, 2 (2016), 1–29.
[131] Haiyi Zhu, Amy Zhang, Jiping He, Robert E Kraut, and Aniket Kittur. 2013. Effects of peer feedback on contribution: a field experiment in Wikipedia. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2253–2262.

Received January 2021; revised April 2021; accepted July 2021