The Diversity Crisis in Software Development

Khaled Albusays, Rochester Institute of Technology

Pernille Bjørn, University of Copenhagen

Laura Dabbish, Carnegie Mellon University

Denae Ford, Microsoft Research

Emerson Murphy-Hill, Google

Alexander Serebrenik, Eindhoven University of Technology

Margaret-Anne Storey, University of Victoria

Digital Object Identifier 10.1109/MS.2020.3045817
Date of current version: 11 February 2021

THIS SPECIAL ISSUE of *IEEE Software* focuses on diversity and inclusion in software development, presenting research results and best practices for making the field equitable for all. It is well documented that the industry does not provide evenhanded participation conditions. Research has shown that implicit gender biases significantly impact hiring decisions,[1] women disengage faster than men,[2] Palestinian tech entrepreneurs do not have access to Internet-based distribution and payment platforms,[3] software developers with a visual impairment lack tools to navigate code editors,[4,5] and women are sometimes less likely to get their code accepted.[6] Tools, processes, products, and education are not inclusive. Dimensions such as geography, gender, socioeconomic politics, age, ethnicity, and disability shape who can participate in creating technology.

Since information technology has a direct impact on people's lives, work, and leisure, a lack of diversity among developers might unintentionally constrain people rather than support them in achieving their desired goals. Technologies have politics,[7–9] and if software developers do not intentionally strive to be inclusive, there is a high risk that their designs will unintentionally exclude populations.[10] We need a diverse and inclusive software development workforce, and to achieve this goal, we need tools, processes, and education that enable everybody to have access to successful careers.

We need a diverse and inclusive software development workforce, and to achieve this goal, we need tools, processes, and education.

The software development profession does not reflect the people who use technology. Unfortunately, the literature concerning the diversity crisis in software development is fragmented and scarce. Although it is well accepted that software development is exclusionary, there is a lack of agreement about the underlying causes, the critical barriers faced by potential future developers, and the interventions and practices that may help. Moreover, researchers and practitioners focusing on one diversity dimension rarely consider it in combination with other aspects.

The goal of this special issue is to raise awareness of the diversity crisis in software development and improve equity, inclusion, and diversity in the field. This issue describes critical challenges faced by industry, nonprofit organizations, academic institutions, and society at large. We present a set of articles that practitioners, educators, and researchers can refer to so that they may gain a deeper understanding of the difficulties and barriers that hinder efforts to achieve equity. We also refer to recommendations that may be considered in the context of this struggle. Before discussing the articles we selected, we review some pertinent diversity classifications.

Diversity Classifications

Diversity refers to the variety of representations that exist within a group, based on a large range of facets and characteristics. Diversity is thus multidimensional, and its relevant parameters depend on the particular and situated practice under scrutiny. In software development, diversity is relevant to the groups of people involved in the field's practices and to the technical designs emerging from those practices. Hence, facilitating diversity involves studies of software development approaches and investigations of the technologies that result. The question then becomes, "What are the relevant diversity parameters we should consider when exploring software development practices and technologies?" Given the multifaceted complexion of diversity and the general-purpose nature of software development, it is important to acknowledge that we cannot come up with a one-size-fits-all classification. Instead, we discuss the core categories that often emerge in such research, in particular, gender, race/ethnicity, and disabilities.

Gender is probably the most applied diversity dimension within software development,[11] particularly since computer science and technology professions are known to lack women despite the fact that programming was originally seen as a female occupation, while hardware design was viewed as a realm for males.[12–15] Although we have seen increasing amounts of research investigating the experiences of women in software engineering,[6,16] we want to make it clear that gender is not a binary classification. Thus, we encourage studies that explore experiences along the gender identity spectrum, including trans[17] and gender-nonconforming developers. We urge the research community to consider the multiple sets of gender classifications when performing empirical studies. For example, when conducting surveys, we draw on the recommendation by Scheuerman et al.[18] to make gender a multiple-selection question, including the classifications of *woman*, *man*, *nonbinary*, and *prefer to self-describe*. This best practice, as well as others, can help our community carry out inclusive research and log gender diversity in our studies.

Race and ethnicity, in particular, reentered the political arena in 2020, with the Black Lives Matter movement intensifying in the United States. In computing research, studies involving racial and ethnic minorities have included the impact of mentorship, the experiences of computer science students,[19–21] and advanced academic careers in computer science.[22] As impactful as that work has been, there is a lack of knowledge about racial and ethnic minority experiences among full-time software developers. Because race and ethnicity are not globally applied uniform concepts,[23] we face a question: how can we collect data about race and ethnicity within global software development in our attempt to increase the percentage of people who are not white and who thus do not have a privileged position? We propose that, when conducting global studies, we extend the parameters and dimensions of the race and ethnicity classification scheme so that we can consider a category in which subjects self-describe in relation to their country of residence. Similarly, since the race and ethnicity classification makes sense only within geographical and societal boundaries, we cautiously encourage properly prepared studies within those bounds. Further, it is critical that we consider people's varying experiences of what it means to be a software developer, how these experiences are shaped by where in the world people engage in the field, and how we as researchers conduct inquiries into those experiences, for instance, by considering the power dynamics that arise when outsourcing occurs between the global North and the global South.[24,25]

Disability research in the software engineering arena has mostly focused on the experiences of developers who are blind or have visual impairments. Investigations have examined work experiences,[26] block-based teaching challenges,[27] and interventions to improve software tools.[5] In terms of tooling, prior research found that programmers who are blind have access to fewer features when compared to their sighted peers.[28] This is because current integrated development environments (IDEs) rely heavily on visual abstraction, which is difficult to convey via assistive technologies, indicating that blind programmers lack equivalent access.[29] Such limitations force visually impaired users to navigate code in a linear fashion, one line at a time, and to jump between code blocks.[30] Although screen readers were found to be useful in various applications, software development still has limitations, given the complex visual information in many IDEs, e.g., color coding. Since much of this research investigated methods for enabling visually impaired developers to comfortably write code, we hope to see the emergence of studies centering on a range of other disabilities, such as motor and hearing impairments.

Besides the preceding categories, relevant diversity dimensions include age, sexual orientation, experience, professional disciplines, and caregiving responsibilities. There are many aspects of diversity, and it is important that we invest the effort to identify the relevant ones. But it is essential to acknowledge that even if we have the best intentions, collecting diversity data risks surveilling and marginalizing minority groups, which can become harmful and run counter to our goals. Categories have politics,[7] and by producing categorizations, we are also creating politics concerning potentially vulnerable groups.[31] Further, it is vitally important that we consider diversity classifications as socially and historically embedded. Diversity classifications, such as gender and ethnicity, have changed through time, and we must acknowledge that those transformations were not organic. Instead, they resulted from struggles rooted in historical eras and societal circumstances (e.g., Stonewall considering LGBTQ+ rights and the Civil Rights Movement in the United States), and they have not been the same across the globe. Thus, when we apply diversity classification schemes with the intention of exposing challenges, we must consider societies, geography, and historical contexts so that we do not accidentally commit violence against susceptible populations. No catch-all scheme can be created; instead, careful consideration is required. If in doubt, we ought to ask a group which categories it prefers and always make it possible for people to self-describe.

Cultural Taxation, Intersectionality, and Imposter Syndrome

While diversity describes the variety of representations that exist within a group, *inclusion* refers to whether or not participants with different backgrounds are invited into a group, given full membership, and have a positive experience. Thus, the question is, "What are some of the core barriers for increasing diversity within software development?" While individuals might experience multiple impediments to software development education and professions, we would like to point to three important challenges to inclusion, namely, cultural taxation, intersectionality, and imposter syndrome.

Cultural taxation was introduced by Amado Padilla,[32] in 1994, and refers to the "extra work and effort" forced upon ethnically diverse academics; however, the term is applicable to all types of diversity characteristics. *Cultural taxation* is defined by Padilla as "the obligation to show good citizenship toward the institution by serving its needs for ethnic representation on committees, or to demonstrate knowledge and commitment to a cultural group, which may even bring accolades to the institution but which is not usually rewarded by the institution on whose behalf the service was performed" (see Padilla,[32] p. 26). What is important here is that if we want to promote inclusion in software development, we must acknowledge the cost that minority practitioners pay in time and effort and remember to value their work by using more than mainstream metrics for promotion.

However, it is not just developers who may experience this challenge. Scholars who conduct research in a "diversity area" related to their personal experiences (e.g., gender research in computer science) and who are dedicated to fostering change often end up having dual research careers. As formulated by Padilla, the choice between conducting "ethnic research" or "mainstream research" frequently includes concerns about how "ethnic research" might be viewed on a tenure application. Cultural taxation—sometimes called the *minority tax*—exists for all types of diversity dimensions that are relevant to software development (e.g., gender, race/ethnicity, and disabilities).

Intersectionality explicitly demonstrates how diversity dimensions, such as gender and race, are not mutually exclusive categories but intersecting ones.[33] Kimberlé Crenshaw, who introduced the concept, showed that because antidiscrimination laws are built along single axes, discrimination becomes a multidimensional experience. This means that people who embody multiple diversity dimensions (e.g., Black women) are lost at the intersections between characteristics and theoretically erased (see Crenshaw,[33] p. 139). In this regard, the huge challenge for software development is that a diversity classification scheme risks blinding us to specific challenges faced by people who embody multiple traits. We might create incentives to increase gender diversity in computing but fail to consider ethnicity[20] and disability.[34] Or we might recruit people with diverse backgrounds for user studies but forget to consider candidates who are blind or who have a skin color other than white. Without explicitly considering diversity from an intersectional perspective, both when recruiting software development teams and when recruiting participants for user studies, we risk doubly burdening underprivileged groups.

Imposter syndrome refers to a person's experience of feeling like an intellectual phony despite having outstanding academic and professional credentials.[35] Imposter syndrome is most often experienced by women, but it has been found to affect other minorities, and it is currently part of higher-education introduction programs.[36] With respect to inclusion in software development, it is important that we consider how people who embody one or more diversity characteristics measure their own qualifications and competence. Inclusion is not only about letting people in; people must believe they can be successful and have corroborating experiences. We must consider retention as part of the inclusion strategy. People who are not white, cisgender, and male need to experience a feeling of belonging—that they are equally important members of the profession. They cannot become token members of committees, teams, and groups but must experience true inclusion as we move toward diversity and equity.

Diversity and Inclusion in Software Development

This special issue focuses on these issues of diversity and inclusion in software engineering. We were fortunate to receive 25 submissions to this special issue. After rounds of reviews and discussions, we selected seven. We briefly describe them in this section and hope they help achieve our goal of spreading awareness of the diversity challenges we face while also equipping readers with guidelines and best practices.

The first article, "How Trans-Inclusive Are Hackathons?" by Prado et al., describes a study where the authors surveyed 44 participants and interviewed seven participants to understand the needs and challenges of the transgender community and other minorities in hackathons. The article explains that hackathons are an effective place for people to learn, work on collaborative projects, and meet new colleagues. Although most people enjoy participating, some transgender people decide not to join in because they feel uncomfortable. Based on their findings, the authors suggest guidelines for making hackathons more inclusive for trans developers.

In "Impact of Affirmative Action on Female Computer Science/Software Engineering Undergraduate Enrollment," Simmonds et al. focus on diversity in the next generation of software engineers. They observe that affirmative action programs to boost female enrollment can have positive effects for science, technology, engineering, and mathematics programs; however, the initiatives yield weaker results for computer science/software engineering majors. This means that to encourage women to consider computer science/software engineering, these programs should complement enrollment quotas with other interventions. "Gendered Experiences of Software Engineers During the COVID-19 Crisis" was a timely submission about technology professionals pivoting to working from home during the pandemic. This article explores how gender impacts their experiences and how work–family arrangements affect women and men differently. The authors make suggestions for better supporting gender diversity in this context.

The article "Gender Differences in Public Code Contributions: A 50-Year Perspective," by Zacchiroli, details a longitudinal study to examine women's and men's contributions to open source code. These projects were classified based on the gender of the developer commits with the goal to examine the evolution of the committers to the projects since 1970. Using a gender inference tool, the author showed that female developers' contributions remain low when compared to those of males. Although the author found that men have always authored more open source code, the gap has begun to narrow.

"Insights Into Nonmerged Pull Requests in GitHub: Is There Evidence of Bias Based on Perceptible Race?" by Nadri et al. presents a novel qualitative analysis of nonmerged pull requests and classifies the reasons why the pull requests were not merged based on four perceived racial identities: Asian/Pacific Islander, Black, Hispanic, and White. The authors found that perceptibly Black and Asian/Pacific developers had pull requests that were rejected at higher rates than those of perceptibly White developers. The authors present findings with respect to additional racial identities. Overall, they found that perceptibly non-White developers were more frequently nonmerged without explanation than perceptibly White developers were.

"How Successful Are Open Source Contributions From Countries With Different Levels of Human Development?" by Furtado et al. examines how geographic location associates with pull request submission and acceptance rates in open source projects. The authors analyzed 44,630 pull requests performed by 14,133 contributors to 20 well-known and well-studied open source software projects on GitHub. They associated pull request submission and acceptance rates with the Human Development Index (HDI), which measures three dimensions: health, education, and per capita income. In the analysis, the authors found that a majority of closed pull requests came from developers in the United States, the United Kingdom, and Germany, countries with a high HDI. They call for more research that considers how geographic differences affect participation in open source that fosters an understanding of the observed differences in pull request submission and acceptance rates for various countries.

Finally, in "Combined Intuition and Rationality Increases Software Feature Novelty for Female Software Designers," Pretorius et al. describe an experiment where 80 developers were asked to perform a software design task. The authors examined how design novelty correlated with a developer's gender and preferred cognitive style, which they defined as intuitive, rational, and a mix of the two. They found that women produced more innovative designs and that women who used an intuitive or a mixed cognitive style were more prolific than those who employed a rational approach.

We hope that some of the submissions we could not accept due to limited space will appear in future issues of *IEEE Software*. As the articles in this special issue show, challenges related to diversity and inclusion are at the forefront of modern software practice. We look forward to research that tackles this important subject.

References

1. Y. Wang and D. F. Redmiles, "Implicit gender biases in professional software development: An empirical study," in *Proc. ICSE-SEIS*, 2019, pp. 1–10.
2. H. S. Qiu, A. Nolte, A Brown, A. Serebrenik, and B. Vasilescu, "Going farther together: The impact of social capital on sustained participation in open source," in *Proc. ICSE*, 2019, pp. 688–699.

ABOUT THE AUTHORS

KHALED ALBUSAYS is a graduate research assistant with the Rochester Institute of Technology, Rochester, New York, 14623, USA. Contact him at khaled@mail.rit.edu.

PERNILLE BJØRN is a full professor and the deputy head of the Department for Research, Department of Computer Science, University of Copenhagen, Copenhagen, 2200, Denmark. Further information about her can be found at http://www.circonflexe.dk/pernillebjorn/. Contact her at pernille.bjorn@di.ku.dk.

LAURA DABBISH is an associate professor with the Human-Computer Interaction Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, 15213, USA. Further information about her can be found at http://www.lauradabbish.com/. Contact her at dabbish@cs.cmu.edu.

DENAE FORD is a senior researcher with Microsoft Research, Redmond, Washington, 98052, USA and an affiliate assistant professor at the University of Washington. Further information about her can be found at http://denaeford.me. Contact her at denae@microsoft.com.

EMERSON MURPHY-HILL is a staff research scientist with Google, Mountain View, California, 94043, USA. Further information about him can be found at https://research.google/people/EmersonMurphyHill/. Contact him at emersonm@google.com.

ALEXANDER SEREBRENIK is a full professor with Eindhoven University of Technology, Eindhoven, The Netherlands. Further information about him can be found at https://www.win.tue.nl/~aserebre/. Contact him at a.serebrenik@tue.nl.

MARGARET-ANNE STOREY is a professor with the University of Victoria, Victoria, British Columbia, V8W 3P6, Canada. Further information about her can be found at http://margaretstorey.com. Contact her at mstorey@uvic.ca.

3. P. Bjørn and N. Boulus-Rødje, "Infrastructural inaccessibility: Tech entrepreneurs in occupied Palestine," *ACM Trans. Comput.-Hum. Interact.*, vol. 25, no. 5, pp. 26:1–26:31, 2018. doi: 10.1145/3219777.
4. A. Armaly, P. Rodeghero, and C. McMillan: "AudioHighlight: Code skimming for blind programmers," in *Proc. ICSME*, 2018, pp. 206–216.
5. V. Potluri, P. Vaithilingam, S. Iyengar, Y. Vidya, M. Swaminathan, and G. Srinivasa, "CodeTalk: Improving programming environment accessibility for visually impaired developers," in *Proc. CHI*, 2018, p. 618. doi: 10.1145/3173574.3174192.
6. J. Terrell et al., "Gender differences and bias in open source: Pull request acceptance of women versus men," *PeerJ Comput. Sci.*, vol. 3, no. 3, 2017. doi: 10.7717/peerj-cs.111.
7. L. Winner, "Do artifacts have politics?" *Daedalus*, vol. 109, no. 1, pp. 121–136, Winter 1980. [Online]. Available: https://www.jstor.org/stable/20024652?origin=JSTOR-pdf&seq=1
8. P. Bjørn and E. Balka, "Health care categories have politics too: unpacking the managerial agendas of electronic triage systems," in *ECSCW 2007: Proceedings of the 10th European Conference on Computer Supported Cooperative Work*. Limerick, Ireland: Springer-Verlag, 2007.
9. L. Suchman, "Do categories have politics? The language/action perspective reconsidered," *Comput. Support. Cooper. Work (CSCW)*, vol. 2, no. 3, pp. 177–190, 1994. doi: 10.1007/BF00749015.
10. Z. Steine-Hanson et al., "Fixing inclusivity bugs for information processing styles and learning styles," 2019, arXiv:1905.02813.
11. K. Kohl Silveira and R. Prikladnicki, "A systematic mapping study of diversity in software engineering: a A

perspective from the agile methodologies," in *Proc. CHASE@ICSE*, 2019, pp. 7–10.
12. M. Hicks, *Programmed Inequality: How Britain Discarded Women Technologists and Lost Its Edge in Computing.* Cambridge, MA: MIT Press, 2017.
13. N. L. Ensmenger, *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise.* Cambridge, MA: MIT Press, 2012.
14. P. Bjørn and M. Menendez-Blanco. "FemTech: Broadening participation to digital technology development," in *Proc. 27th ACM Int. Conf. Multimedia*, 2019, pp. 510–511.
15. P. Bjørn and D. Rosner, "Intertextual design: The hidden stories of Atari women," *Human–Comput. Interact*, to be published. doi: 10.1080/07370024.2020.1861947.
16. D. Ford, J. Smith, P. J. Guo, and C. Parnin, "Paradise unplugged: Identifying barriers for female participation on stack overflow," in *Proc. 24th ACM SIGSOFT Int. Symp. Found. Softw. Eng.*, 2016, pp. 846–857. doi: 10.1145/2950290.2950331.
17. D. Ford, R. Milewicz, and A. Serebrenik, "How remote work can foster a more inclusive environment for transgender developers," in *Proc. IEEE/ACM 2nd Int. Workshop on Gender Equality Softw. Eng. (GE)*, 2019, pp. 9–12. doi: 10.1109/GE.2019.00011.
18. M. K. Scheuerman, K. Spiel, O. L. Haimson, F. Hamidi, and S. M. Branham, "HCI guidelines for gender equity and inclusivity," 2019. [Online]. Available: https://www.morgan-klaus.com/gender-guidelines.html
19. R. T. Cummings et al., "An exploration of black students interacting with computing college and career readiness vlog commentary social media influencers," *Comput. Sci. Eng.*, vol. 22, no. 5, pp. 29–40, 2020. doi: 10.1109/MCSE.2020.3005635.
20. Y. A. Rankin and J. O. Thomas, "The intersectional experiences of black women in computing," in *Proc.* 51st *ACM Tech. Symp. Comput. Sci. Educ.*, 2020, pp. 199–205.
21. K. D.Ortiz-Lopez, M. Holanda, R. Furuta, and D. Da Silva, "Educational Initiatives to Retain Hispanic/Latinx students in computing: A systematic literature mapping," in *Proc. Res. Equity and Sustained Participation Eng., Comput. Technol. (RESPECT)*, 2020, vol. 1, pp. 1–2. doi: 10.1109/RESPECT49803.2020.9272481.
22. I. F. Ogbonnaya-Ogburu, A. D. Smith, A. To, and K. Toyama, "Critical race theory for HCI," in *Proc. CHI Conf. Human Factors in Computing Systems*, Apr. 2020, pp. 1–16.
23. W. Darity Jr., D. Hamilton, and J. Dietrich, "Passing on blackness: Latinos, race, and earnings in the USA," *Appl. Econ. Lett.*, vol. 9, no. 13, pp. 847–853, 2002. doi: 10.1080/13504850210149133.
24. S. Matthiesen, P. Bjørn, and C. Trillingsgaard. "Attending to implicit bias as a way to move beyond negative stereotyping in GSE," in *Proc. 15th Int. Conf. Global Softw. Eng.*, 2020, pp. 22–32. doi: 10.1145/3372787.3390432.
25. D. Ford, "Approaches to understanding a socio-cultural experience of global knowledge workers," in *Proc. Engaging Race HCI Workshop, CHI 2020*, Apr. 2020.
26. E. W. Huff, K. Boateng, M. Moster, P. Rodeghero, and J. Brinkley, "Examining the work experience of programmers with visual impairments," in *Proc. IEEE Int. Conf. Softw. Maintenance and Evolution (ICSME)*, 2020, pp. 707–711. doi: 10.1109/ICSME46990.2020.00077.
27. A. Mountapmbeme and S. Ludi, "Investigating challenges faced by learners with visual impairments using block-based programming/hybrid environments," in *The 22nd Int. ACM SIGACCESS Conf. Comput. Accessibility*, 2020, pp. 1–4. doi: 10.1145/3373625.3417998.
28. C. M. Baker, C. L. Bennett, and R. E. Ladner, "Educational experiences of blind programmers," in *Proc. 50th ACM Tech. Symp. Comput. Sci. Educ. (SIGCSE '19).* New York: Association for Computing Machinery, 2019, pp. 759–765.
29. S. Mealin and E. Murphy-Hill, "An exploratory study of blind software developers," in *Proc. IEEE Symp. Visual Languages and Human-Centric Computing (VL/HCC)*, 2012, pp. 71–74. doi: 10.1109/vlhcc.2012.6344485.
30. O O. Falase, A. F. Siu, and S. Follmer, 2019. "Tactile code skimmer: A tool to help blind programmers feel the structure of code," in *Proc. The 21st Int. ACM SIGACCESS Conf. Comput. Accessibility (ASSETS '19).* J. P. Bigham, S. Azenkot, and S. K. Kane, Eds. New York: Association for Computing Machinery, pp. 536–538.
31. G. C. Bowker and S. L. Star, *Sorting Things Out: Classification and Its Consequences.* Cambridge, MA: MIT Press, 2000.
32. A. M. Padilla, "Research news and comment: Ethnic minority scholars; research, and mentoring: Current and future issues," *Educ. Res.*, vol. 23, no. 4, pp. 24–27, 1994. doi: 10.3102/0013189X023004024.
33. K. W. Crenshaw, "Race, reform, and retrenchment: Transformation and legitimation in antidiscrimination law," *Harvard Law Rev.*, vol. 101, no. 7, pp. 1331–1387, 1988. doi: 10.2307/1341398.
34. B. Blaser, C. Bennett, R. E. Ladner, S. E. Burgstahler, and J. Mankoff, "Perspectives of women with disabilities in computing," in *Cracking Digital Ceiling: Women Computing Around World.* C. Frieze and J. L. Quesenberry, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2019, p. 159.
35. P. R. Clance and S. A. Imes, "The imposter phenomenon in high achieving women: Dynamics and therapeutic intervention," *Psychother, Theory, Res. Pract.*, vol. 15, no. 3, p. 241, 1978.
36. A. Parkman, "The imposter phenomenon in higher education: Incidence and impact," *J. Higher Educ. Theory Practice*, vol. 16, no. 1, pp. 51–60, 2016.
37. D. Thakkar, N. Sambasivan, P. Kulkarni, P. Kalenahalli Sudarshan, and K. Toyama, "The unexpected entry and exodus of women in computing and HCI in India," in *Proc. CHI Conf. Human Factors in Comput. Syst.*, 2018, pp. 1–12. doi: 10.1145/3173574.3173926.