A Guide to Ethical Collaboration in Open Source Communities

Written by: Tianna Mañón
Edited by: Stephanie Morillo

Open source communities and technologies bring tremendous potential for transforming our world for the better. But with their use in racially-biased policing and sentencing, surveillance capitalism, unethical data collection, and protester identification and suppression, we're seeing that open source can also deepen divides and compound existing inequities. And, unfortunately, it's often the most vulnerable and marginalized communities that are impacted.

Much of this harm is rooted in the widespread belief that technology is inherently "neutral". As more and more developers become aware of the real-world impact of supposedly"neutral technology, many are looking for ways to mitigate these harms or eliminate them altogether.

The [Ethical Source Principles](#) provide a foundation for creating open source communities that are more welcoming, respectful of our differences, and accountable — an important first step toward creating better outcomes.

These are the seven Ethical Source Principles:

1. Our work benefits the commons.
2. Our work is done in the open.
3. Our community strives to be welcoming and just.
4. Our work respects accessibility as a human right.
5. Our work prioritizes the safety of people.
6. Our work protects privacy.
7. Our work deserves support.

As the world continues to depend on open source technologies, it's critical that communities developing these tools be explicit about their values and find ways to center them in the work they do. While nobody from the outside can say "these are your community values," the Ethical Source Principles can serve as a helpful starting point, providing a foundation for individual communities to build on and then adapt to their particular challenges, priorities, and needs.

Putting shared values into practice requires intentional, collaborative, and ongoing effort; it doesn't happen on its own. If you're not clear on how to start, don't worry; this guide will help you get there. You'll learn how open source maintainers and contributors can work together more effectively to build tools that truly benefit society, starting with making their own communities more safe, welcoming, and equitable.

Why this work is necessary

Open source technologies are increasingly a key component of platform and infrastructure stacks across our industry; in fact, the internet itself is largely built on open source. The majority of these open source technologies are built and maintained by independent, volunteer-led communities of developers, designers, and other contributors.

As stewards of open source, it is our responsibility to find ways to center our values of justice and equity in everything we do. This requires not only being explicit about our values, but also using them as guideposts as we make decisions about why, what, and how we do what we do. And, most importantly, understanding how our work can be used or abused to cause harm. This differs from how open source has historically been practiced.

For decades, the focus has centered on the "open" in open source. It was assumed that simply providing open access to source code would be enough and that if more voices and more people could get involved, it would result in better outcomes for everyone. However, in reality, this hasn't happened. Despite the U.S. Bureau of Labor showing that almost 20% of software developers are women, 95% of respondents in a 2017 GitHub open source survey were white men.

Many technologists are starting to realize simply being "open" isn't enough. As Don Goodman-Wilson wrote in a 2019 blog post, "There is nothing absolute about openness, openness is not an end in itself, but a tool that we can use for good or ill. It turns out to be a pretty good tool when used to build and empower communities. But it can also be an effective tool for exacerbating existing injustices, and creating new ones."

This exclusion and coded access is even evident in some of the language and terminology we commonly use. Take, for example, the "master/slave" terminology which has been used for decades to describe a model of communication between systems. The term is purported to have originated in this context in the early 20th century but as the Black Lives Matter movement of the past several years has called renewed attention to the history of chattel slavery in the United States — and the resulting impact of racism against Black Americans — this term has been called into question as being harmful. In fact, in 2020, there was a huge push to remove these terms altogether in the wake of protests following George Floyd's murder. Companies like GitHub, Twitter, and Apple and projects like Python and Linux have since removed master/slave terms from their source code.

Dr. Safiya Umoja Noble, associate professor at UCLA's Department of Information Studies, demonstrates other ways that technology shapes how we see and interact with each other. In her book *Algorithms of Oppression: How Search Engines Reinforce Racism*, Noble describes how Google algorithms prompted harmful "autocompletes" for years when users would search for anything that included the term "black girls":

"I again retrieved a Google Search results page filled with porn when I looked for 'Black girls.' By then I would have expected that my own search history and engagement with a lot of Black feminist text, videos and books on my laptop would have shifted the kinds of results I would get. It had not."[1]

As another example, YouTube algorithms, which like many are designed to optimize for "engagement" have, in many cases, contributed to the right-wing radicalization of viewers by funneling them into misinformation "black holes." Instagram's engagement algorithms have contributed to young women struggling with body dysmorphia and mental health and Facebook's algorithm deliberately prioritizes content and posts that will trigger a strong response and get you to engage, like or comment– or at the very least, keep you scrolling.

Ultimately, we have learned that when we don't acknowledge and mitigate systemic biases, they are actually given more power through the technologies we create:

"We must look at ourselves (computer people) as being in control of a tremendously powerful tool. ... Computers are becoming an essential part of the social organism itself, particularly its communication and control system. ... When one reflects upon the great forces that we computer people are associated with, it is no longer difficult to grasp, and perhaps to accept, our heavier-than-average share of responsibility," said Edmund Berkeley in The Social Responsibility of Computer People. Berkeley is renowned for his contributions to computer science, including founding the Association for Computing Machinery in 1947. Even at that early stage in our industry, he recognized the tremendous impact of technology on society.

Addressing problems like these requires us to think beyond the code. Open source developers can't continue to focus solely on the technical aspects of our work, while ignoring the very real consequences of its use and abuse downstream.

Ethical collaboration calls on us to ensure that contributors are protected, end users have their needs met, and the potential impact on broader society is fully understood. It's critical to understand that our intentions do not absolve us of responsibility for our actual impact.

How to make immediate change

The above examples highlight how crucial constant self-reflection is as we create new tools and products. It's important to consider what Berkeley and Goodman-Wilson said and ask yourself the following questions to see how you can improve the projects you're responsible for to make them more accessible and less harmful.

---

[1]Noble, S. U. (2018). In Algorithms of oppression: how search engines reinforce racism (p. 18). essay, New York University Press.

Here are some questions to get you started:

- What problem are we trying to solve?
- Who are we building this for?
- How are they served well with this code?
- Beyond our adopters and users, what is the impact on other groups of people who are *not* direct users but can be affected by these technologies? (For example: the use of facial recognition in the criminal justice system for "identification" of suspects)
- What are the ways that what we're creating could be used or abused to cause harm to others, and how can we mitigate that harm?
- Does the impact of its potential abuse and misuse outweigh its positive value?

What goes into ethical collaboration?

Julia Nguyen, founder of the open source mental health collective if-me.org, has spent most of her life online. She was an adolescent in the 2000s, navigating emerging digital communities like Neopets and MySpace. These websites allowed users to design and customize their own profile pages. As a result, she quickly learned how to code different elements, webpages and more, joining groups of similar designers who focused on attribution and recognizing and respecting each other's work.

However, when she went to college, she began to see another side of the technology world. For example, when she would look for help, people would often tell her to just "read the directions" (a call to the widespread use of "RTFM", or "read the fucking manual"). She saw that open source communities could be hostile even to people like her who had deep technical backgrounds.

Julia's experience demonstrates another obstacle present in technology spaces: outright exclusion. Experiences like hers are common, and are responsible for driving many already marginalized people out of this industry, underlining the growing concern that simply getting more people of diverse backgrounds into these roles isn't enough. Instead, we need real safe spaces so that the tools and resources we develop are reflective of that commitment to inclusion and access.

Here are some strategies to ensure your projects are built with inclusivity and accessibility in mind.

Creating a welcoming environment

Codes of conduct are an important tool for creating and sustaining safe, working spaces. Adopting (and adapting) a code of conduct also provides an opportunity to reflect on your responsibilities as an open source maintainer and community leader. There are many open source codes of conduct to choose from, including Contributor Covenant. You should make your

code of conduct easy to find in your code repository, and describe how it will be enforced, to create an environment of safety and accountability.

Your code of conduct should be referenced in your README and contributor guidelines, and can also be surfaced in Slack and other collaborative spaces. Be sure to revisit it regularly and adapt it to reflect any new needs, challenges, or issues that come up. Whenever there's an issue, the code of conduct can be used as a framework for expectation and accountability.

Adopting inclusive communication styles

One of the biggest challenges in open source collaboration is communication, which is largely asynchronous and involves navigating individual communication styles and even cultural or language barriers. Open collaboration can pose real difficulties and challenges, especially when we don't recognize our own biases, or have strong emotional responses to conflict. And it's been shown that the common call to "assume best intentions," actually hinders the communication process.

Creating opportunities for safe and open communication and feedback is essential to ensuring that everyone feels safe in sharing their concerns. This requires that leaders be thoughtful and deliberate in accommodating different styles of communication,

Checking in regularly to get ahead of any issues

It's important to regularly check in with your leaders and contributors because open source communities are constantly growing and changing. Frequent check-ins allow for inclusive communication and are also a critical part of values alignment. Julia initially spent an hour meeting every person who contributed to if-me.org. To save time for both her team and any prospective contributors, she also posted clear language about each contributor role, helping people to make their own decisions about what works for them and how much they can give to the project as well. Detailed contributor guidelines allow everyone to start the conversation on the same page.

You should also encourage contributors to share feedback and suggestions for improvement, for example by hosting office hours, creating safe channels for anonymous feedback, performing regular check-ins, organizing group brainstorming sessions, and even conducting contributor surveys.

Adopting a style guide

Style guides provide guidelines for how we use language and what words we use to describe something. Leverage style guides such as Self-Defined and AlexJS in your open source projects and in your writing. Make these resources accessible in the contributor guidelines and templates to familiarize contributors with them.

Using chatbots

Bots are a useful tool for surfacing information and pointing out things that contributors may miss or forget. You can reference the GitHub Probot site to find apps for [adding status checks for inclusive language](#), [creating automatic replies](#), [receiving notifications when contributors who have been harmful try to open up a PR](#), [surfacing your code of conduct](#), and more.

Creating templates

Tell your contributors exactly what they need to provide in order to open up an issue or pull request. Include clear issue and PR templates that contributors can reference to understand the level of detail needed. Feel free to refer to or adopt documents like the Ethical Source Principles and your code of conduct to define policies that set clear expectations and are flexible to the needs of your community.

Reflections

As we deal with the needs of everyday life and pressures in the workplace, it's tempting to put these concerns on the backburner. We may envision starting new procedures or adding to our code of conduct *after* we meet that next deadline. However, it's crucial that we make this work part of the very structure of creation and constantly reflect on how we can be more inclusive?

- Have you ever had to change how you approach a project because your colleague had a different communication style? What did you do? Were you able to take that change and apply it elsewhere?
- Have you ever received negative feedback from the public on a product or tool you created? What systems and procedures did your workplace use to follow up on that criticism?
- Reflect on a time you didn't feel heard at work? What do you wish your leadership had changed so that you and nobody else feels this way again?
- Do you feel safe challenging your team or contributing your opinion to a project you're currently working on? Why or why not?
- If you're in a leadership role, do you offer several different ways to receive feedback? What is your process for responding to feedback and do you circle around to check in with that person again to ensure it's resolved?
- What tools do you use in the workspace to facilitate conversation?

Leadership is critical when it comes to ethical collaboration

Incorporating ethical practices in open source communities starts with leaders modeling the standards of responsibility and accountability they expect from other participants. By creating systems that allow for healthier interactions, collaboration and products/tools, leaders are able

to get ahead of many of the issues that within workplaces can cause division and exclusion and, downstream, cause harm to the public.

"Our contributors to if-me.org are very enthusiastic, but we don't want them to get overwhelmed, so we instituted a policy that you can only take one case, or ticket, at a time," said Julia, explaining how she checks for the common issue of burnout. Since work culture in most technological spaces can be demanding, she uses this policy to encourage volunteers and users to take on what they can reasonably manage. Instituting policies like these can help prevent burnout, which is more likely to affect women, Black people and Latinxs, but is a critical concern for all contributors. Burnout can have a major impact on both mental and physical health and, long-term, can even contribute to many major diseases like hypertension.

Julia's policy also helps the end user. By reducing the likelihood of burnout, contributors are less likely to inadvertently cause harm to themselves and other members of the community.

[According to the University of California,](#) "[stress and] unconscious bias is far more prevalent than conscious prejudice and often incompatible with one's conscious values. Certain scenarios can activate unconscious attitudes and beliefs. For example, biases may be more prevalent when multi-tasking or working under time pressure."

Providing contributors with the time and resources they need has been shown again and again to result in better, more inclusive and safer projects.

André Arko, founder of [Ruby Together](#) (now part of Ruby Central) and lead developer of [Bundler](#), the dependency manager for the Ruby programming language, is another leader who prioritizes creating welcoming spaces. He knows firsthand how individual and even systemic biases that shape our general culture can be exaggerated in open source projects because manydepend on work from volunteers, who put in significant time implementing new features, answering user tickets, and fixing bugs without any promise of compensation.

André said that armed with this knowledge, he knew it was critical to pay core contributors and, as a result, he saw the biggest difference in participation, retention, output and overall satisfaction:

"That really seems like the only thing that works in 10 years of running this project. People who are getting paid can stick around instead of joining our project and then leaving when the excitement wears off."

André's stance, reflected in the seventh Ethical Source Principle ("Our work deserves support"), is considered radical in a world where most contributors are never compensated, even as [large players in the space benefit from their work](#). As a result, he said, this clearly limits who can participate in these conversations, because for the most part, young, cisgender, white, male developers are more likely to be able to volunteer their free time due to a number of factors, including being more likely to earn higher wages at their primary place of employment and less

likely to take on "second shift" work, a term for the daily needs of the private sphere, like cleaning, cooking, and childcare. Because women are more likely to shoulder this extra work at home, they have less time and energy they can devote to unpaid projects.

André pointed out that this happens in open source communities and projects but also extends to other tech spaces, including conferences, meetups, academic settings, college courses and essentially anywhere people come together to discuss or create technology. "That leads to an entire culture that only prioritizes young-ish, white dude perspectives. In the absence of an equitable society, paying people with money is what works."

Justin Flory, the Open Source Technical Advisor for the United Nations, attends the "Flock to Fedora" conference every year. While most of the attendees consider themselves progressive and believe that they are creating non-harmful tools, Flory said the organizers worked to make the focus on anti-racism and reducing harm integral to the conference itself:

"I remember I attended the conference and instead of having a separate D&I workshop during the weekend like we often see at other conferences, the organizers brought it up in the welcome session," said Justin. "Afterwards a friend came up to me, someone who already cares about these issues, and [said] even he learned something…so, I think by integrating [these critical D&I principles] into the very work we're doing, you get a better result. You can't think about these issues afterward. You have to start from the beginning and, essentially, never stop."

This kind of leadership can guide your project or community towards a kinder, more welcoming environment, instead of depending on contributors to do the additional work, which may not even be reasonable or feasible. As Julia and André prove, contributor-focused policies and decisions can make a major difference. It's also clear that these results don't just happen on their own, and strong leadership is key to long-lasting results.

Reflections

Leadership is one of the most crucial components to building a safe environment. The right leader is able to listen to their team while also modelling compassion, transparency and openness at the top level. Here are some key questions to ask yourself as you do this important work:

- How are you compensating your contributors?
  - If your project is too small to do so financially, what can you provide instead? How can you ensure you're not accidentally contributing to the wage/access gap André mentions?
- How can you work from the top to encourage and sustain contributors, reduce burnout, and promote workplace satisfaction?
- What issues arose in your last project and how have you planned for these this time? Is there space for discussion before major decisions and if not, can you bring in more voices before communicating any major changes?

- Is your team able to safely tell you when they're under-supported or need more resources?

Beyond "good intentions"

As more and more of our modern society comes to rely on open source technologies, we can no longer afford to ignore the outsized impact of our work and the responsibilities that come with it. Our good intentions are simply not enough.

The Ethical Source Principles provide a foundation for creating and sustaining equitable and accessible open-source communities, and adopting the practices of ethical collaboration is an important first step. Centering our shared values in the work that we do brings us closer to unlocking the incredible potential for open technologies to promote broad social good.

And it's critical to understand that this work is never done. The world is constantly changing, and the technologies we create — and how we go about creating them — must evolve too.

"If we want to build a better world, we will have to move beyond quick fixes and silver bullets. Free and open source software platforms can get us part of the way there, but without some big changes, it won't be enough. We need to build ethical structures for the creation of ethical software."
 -- Deb Nicholson at FOSDEM 2020

This paper was published by the Organization for Ethical Source. Learn more about our work at ethicalsource.dev.

Appendix

Resources:
- https://ethicalsource.dev/principles/
- https://diversity.ucsf.edu/resources/unconscious-bias
- https://alexjs.com/
- https://www.selfdefined.app/
- https://probot.github.io/apps/sentiment-bot/
- https://probot.github.io/apps/background-check/
- https://probot.github.io/apps/customizable-comments/
- https://probot.github.io/apps/in-solidarity/
- https://www.contributor-covenant.org/

References:
- https://www.wired.com/story/developer-deletes-code-protest-ice/
- https://dev.to/degoodmanwilson/open-source-is-broken-g60

- https://archive.fosdem.org/2020/schedule/event/capitalismethicaloss/
- https://stackoverflow.blog/2021/01/07/open-source-has-a-funding-problem/
- https://bundler.io/
- https://rubytogether.org/
- https://en.wikipedia.org/wiki/RTFM
- https://thebias.com/2017/09/26/how-good-intent-undermines-diversity-and-inclusion/
- https://www.if-me.org/
- https://firstmonday.org/ojs/index.php/fm/article/view/10419/9404
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5810159/
- https://www.wired.com/story/tech-confronts-use-labels-master-slave/
- https://www.bbc.com/news/business-53273923
- https://www.pcmag.com/news/apple-to-remove-masterslave-and-blacklist-terms-from-coding-platforms
- http://wg20.criticalcodestudies.com/index.php?p=/discussion/59/master-slave-replication-in-mysql

About the Organization for Ethical Source

The Organization for Ethical Source (OES) is a diverse, multidisciplinary, and global community that is revolutionizing how tech culture works. We are investing in tools like Contributor Covenant as part of our commitment to creating a better future for open source communities around the world. If you'd like to help us shape that future, consider becoming an OES member.