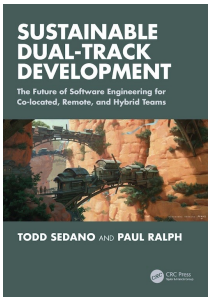
Chapter 12

Towards Organizational Justice

People [seem] unaware of the differences in their behavior towards different peers, and [underestimate] the impact of their small acts of interpersonal exclusion on their peers, both in terms of job performance as well as their own team's effectiveness.
– Juliet Bourke

Key Takeaways

- Creating a just organization with a diverse workforce, "you belong here" culture, and tailored support requires intentional work; it won't just happen.
- Organizational injustice is rooted in both conscious and unconscious bias.
- Conscious biases can (sometimes) be corrected with education, but employees who refuse to reform should be terminated.
- Unconscious biases affect everyone and are difficult to eliminate; instead, focus on redesigning tasks, policies, and systems to prevent biases or mitigate their effects.
- To *recruit* diverse talent: hire paid interns; keep your application process simple, easy, and transparent; and don't use AI for screening.
- To *retain* diverse talent: share power, improve your onboarding process, offer flexible working conditions, and develop allyship and mentorship skills.
- Reform your promotion process to enhance fairness.



Dealing with Bias

Sare—a talented engineer, author, and all-around amazing person—shared her perspective on being a woman in software engineering. Here's what she had to say:

"Women are constantly asked to prove that they are 'technical enough.' This is my life. This is why I've quit the industry twice. It's hard to find a software company where my coworkers generally treat me like I'm a person, not an aberration. Because I'm a consultant, working around great engineers doesn't solve the problem. I have this challenge with every client in every external meeting. It takes me a few weeks for clients to accept that I can not only lead a team because I can coordinate people but also actually know how to code. There's always that moment of shock when I pair with someone because they didn't expect I actually knew what I was talking about. Being constantly asked, 'So are you really a programmer?' takes a toll.

"Because of this constant bias, all women in 'male' jobs and all minorities have to develop better communication skills, better emotional intelligence, better emotional regulation, and have a huge reservoir of patience just to come to work. Because of this constant bias, those immeasurable skills aren't often recognized or valued. Promotion guidelines should include 'responding to systemic marginalization with grace.'

"As we think about ways to quantify advancement, we also need to think about how to make it reasonable for people who don't share the same privileges.

"If underrepresented minorities are doing all this extra work every day just to be on the same playing field, that means they have fewer resources to do their regular work, let alone 'work harder' for promotion. It's easy to say, 'Yeah, take a few hours and learn some language,' or, 'Spend some time skilling up in some technology,' when you don't have to face the fatigue and burnout from justifying yourself all day on top of your regular programming work.

"I've struggled for advancement. I've seen others promoted who appear to be doing the same things I'm doing. I've asked for feedback about what I can do to gain that promotion and received only vague or unhelpful advice. When I ask my colleagues for feedback, they say, 'You're doing great. I really like working with you. Clients love you.' But then, when I ask what I need to improve to be promoted, again, it's silence."

Introduction

This book is about optimizing the sociotechnical process of software development. A crucial but oft-overlooked part of this optimization is making the professional environment fair and just because people do their best work when they feel that they are treated fairly.

In Sare's story, we hear about the cumulative effects of injustice, belittling, and demeaning behavior. Common themes in stories of discrimination also include self-doubt and confusion. If you have experienced injustice in the workplace, we assure you that you are not alone; stories like Sare's (and worse) are common throughout the tech industry. They range from the seemingly minor to illegal, heinous behavior. If this seems far-fetched and

unlike your organization, remember that people are more likely to share negative experiences with trusted community members to whom the story pertains. For example, women are more likely to discuss sexist behavior with other women. Therefore, those least likely to experience injustice are also least likely to hear about it.

Organizational justice is neither a liberal nor conservative cause. Everyone cares about justice; they just don't always agree on what is just in a specific situation. We (Paul and Todd) have very different political views, yet we reached agreement on virtually all of the content below, and we both feel that it's relevant regardless of the latest political discourse or your personal ideology. Indeed, it's not easy for people like us—middle-aged, straight, white men from rich, western countries who have little-lived experience of discrimination—to discuss these issues without seeming tone-deaf and part of the problem. However, expecting those most often victimized to do all the work of fighting for a better world is part of the injustice, so with humility, we have to try.

We shared this chapter with several beta readers from diverse social groups and made many updates and alterations based on their feedback.

How Do I Know if There's a Problem? Assume There's a Problem.

Many injustices are difficult to address for several reasons:

- The people experiencing the issue don't report it. They may believe that they will be ignored or face retaliation even if the company has a no-retaliation policy. Many high-profile professionals have been fired for complaining about injustices.
- The person receiving the report doesn't understand it or take it seriously. Because we all have different life experiences, managers often misunderstand the emotional impact of events on their staff or dismiss legitimate concerns as people being oversensitive. When problems are reported as items in retros or a concerned message instead of a formal complaint, the medium doesn't always convey how serious the problem seems to the person reporting it.
- The team (or manager, HR professional, etc.) understands what the report says but doesn't recognize it as a problem. Professionals increasingly recognize the ethical problems with discriminating based on race, gender, sexual orientation, and religion, yet continue to rationalize and defend discrimination based on nationality, disability, age, and socioeconomic class.
- The team (or manager, HR professional, etc.) recognizes that there is an issue but doesn't deal with it effectively. Justice issues often require complex, challenging, permanent changes to systems and processes. Many people just want an easy solution (i.e., lip service).

All of these reasons are compounded by the fact that injustice often comes in the form of death-by-a-thousand-cuts rather than a singular event. Injustice that stems from many small infractions instead of one big infraction is harder to report, understand, and address.

Unfortunately, injustice is ubiquitous to the point that you should simply assume your team has serious problems. When people take umbrage at the suggestion that such problems might exist, ask, "How would we know there are problems if no one has reported them?"

Measuring Psychological Safety

Psychological safety is the degree to which a specific person in a specific situation feels that they can take interpersonal risks without negative consequences. If you're confident that you won't face any repercussions from criticizing your boss in a retro, then your psychological safety in that situation is high. High psychological safety is important because "organizations in which managers value the employee who speaks up, questions existing practices, and suggests new ideas are better able to improve and learn" [144].

A good way to measure psychological safety is to use Edmondson's Psychological Safety Scale [145], shown below. To calculate the score, sum the points for each answer. For example, if someone answered 'disagree, agree, neutral, neutral, agree, strongly disagree, neutral,' then their score would be $4+4+3+3+2+1+3=20$. Higher scores indicate greater psychological safety.

	strongly disagree	**disagree**	**neutral**	**agree**	**strongly agree**
If you make a mistake on this team, it is often held against you	5	4	3	2	1
Members of this team are able to bring up problems and tough issues	1	2	3	4	5
People on this team sometimes reject others for being different	5	4	3	2	1
It is safe to take a risk on this team	1	2	3	4	5
It is difficult to ask other members of this team for help	5	4	3	2	1
No one on this team would deliberately act in a way that undermines my efforts	1	2	3	4	5
Working with members of this team, my unique skills and talents are valued and utilized	1	2	3	4	5

(In our experience, more umbrage indicates more serious issues.) The question isn't *is there a problem?* The questions you should be asking are, "What are the most serious problems, and how can we address them?" Some simple ways to raise these questions include:

- Run an annual anonymous survey including a psychological safety scale (see the Measuring Psychological Safety sidebar).

- Schedule a quarterly team health assessment facilitated by someone from outside your management reporting structure—ideally outside the company.
- Look for missing feedback loops that would help identify problems in your team or organization.
- Encourage engineers, managers, and leaders to ask their peers (and direct reports) for feedback routinely.

Creating a Healthy Work Environment

Tech companies *should* create healthy work environments where people are given opportunities to grow and feel respected, heard, and both physically and psychologically safe, not only because it's the right thing to do but also because it improves morale, job performance, retention, and profitability. Software is created by people, and people do their best work when they feel appreciated, included, and respected. Therefore, our goal is to provide concepts that will help you create a welcoming team and a welcoming company.

We will divide our discussion into four sections. Justice is built upon a foundation of increasing perspectives, fostering a "you belong here" culture, providing tailored support, and sharing power. These elements are interconnected; each must be addressed for optimal results. Before tackling each of these foundations of justice, we next introduce an important cross-cutting concern: the issue of conscious and unconscious bias.

Conscious vs. Unconscious Bias

> *I have a really hard time with the idea that we as a society should accommodate badly behaved people because of the great stuff they create.*
> *– Sarah Mei*

Conscious Bias

Conscious bias is when people have prejudices that they know about. Conscious bias leads to overt, explicit discrimination. *Unconscious bias* is when people have prejudices that they *don't* know about. Unconscious bias leads to accidental, unintended discrimination. There's a lot of talk in management and government circles about unconscious bias and unconscious bias training (more on that below), yet many people remain overtly and proudly prejudiced.

When an employee espouses conspiracy theories about how straight, white men can no longer find work because employers only hire women and non-whites, that's *conscious* bias.[1] When a professor says something like, "women just don't get computers," that is conscious bias.[2] When someone refuses to work with a trans person, objects to a co-worker wearing

[1] According to the US Bureau of Labor Statistics, the 2024 unemployment rate among those aged 20+ years was 2.9% for white men, 3.4% for white women, 4.3% for Hispanic men, 4.8% for Hispanic women, 5.3% for black women, and 5.6% for black men https://www.bls.gov/web/empsit/cpsee_e16.htm)

[2] Female computer science majors have higher grades, on average, than male students [146].

a religious symbol, or complains that hiring racial minorities will interfere with the team's dynamics, that's usually *conscious* bias.

Such people are typically aware of their negative attitudes toward one or more minority groups. They don't think of these negative attitudes as biases; they think they're "the truth." Media and peer groups can reinforce these beliefs, but the popular belief in "media bubbles" or "echo chambers" is misleading. Rather, wealthy individuals, organizations, and nations "us[e] social media to spread computational propaganda and disinformation" to manipulate public opinion [147]. Many people deriding minority groups are being manipulated by these powerful interests.

Regardless, we can confront *conscious* bias through a three-step process:

1. **Isolate.** People who hold expressly bigoted beliefs are toxic to your organizational culture and should be rapidly isolated not only from people they're biased against but also from decision-making or other social processes that affect the people they're biased against. Think of isolation as a temporary measure to give the employee an opportunity to reform.

2. **Educate.** Conscious biases are typically rooted in incorrect beliefs. Whether these incorrect beliefs can be changed depends on the individual's *belief perseverance*: the degree to which a person resists changing their mind based on new evidence [148]. People with low belief perseverance can be rehabilitated by exposure to facts (e.g., scientific literature about sexual orientation) or personal experience (e.g., working with a woman who is an excellent coder). The latter is tricky because most people don't want to work with someone who's prejudiced against them.

3. **Terminate.** People with high belief perseverance tend to retain their biases no matter what. When employees won't reform their prejudicial behavior, eject them from your organization. Their technical contributions are not worth the negative impacts on their co-workers and organizational culture.

Unconscious Bias and Treating Others Differently

Despite the prevalence of conscious bias, unconscious bias is *much* more common. Unconscious bias is part of being human. We're *all* biased in ways we're not aware of. That's why they call it "unconscious." Unconscious biases lead us to treat people differently depending on the groups we think they belong to. This differential treatment can be insulting and upsetting, but many people just don't seem to notice because people have wildly different life experiences.

For example, I (Paul) was once on a shuttle bus at Frankfurt airport. I asked the person sitting next to me if she spoke English (because I don't speak German). She, visibly annoyed, responded "Sprechen Sie Deutsch?" (do you speak German?). I said, sorry, no, at which point she visibly relaxed and began chatting to me in English. It turned out that she was an anthropologist, originally from India, now living in Frankfurt. Unfortunately, despite her fluency in German, many Germans (especially men) insisted on speaking English with her, in other words, treating her as an outsider. She found this repeated othering deeply hurtful. She relaxed when she realized that I wasn't othering her; I spoke to her in English because

I didn't speak German. Rather than treating her as an outsider, I was an outsider myself. This interaction illustrates two important points about unconscious bias and inclusion:

1. Sometimes, statements or actions that seem innocuous can be genuinely hurtful to others. To maintain an inclusive environment, we therefore need to listen to and empathize with people with different life experiences.

2. Sometimes, people upset us or make us feel excluded completely by accident or in ways they couldn't have anticipated, so we should try to avoid being overly sensitive or jumping to conclusions.

People tend to react negatively to one or the other of these two points depending on their political leanings. On one side, some people feel like, 'Why should I work so hard making others comfortable? No one seems to care if I'm comfortable.' On the other side, some people feel like suggesting that someone is too sensitive belittles their feelings because repeated, seemingly-innocuous behaviors can accumulate to seriously harm a person's psychosocial well-being.

Consider two extreme examples. In one organization we observed, there was a trans person who changed their name to reflect their new gender. Their line manager continued to call them by their deadname (pre-tansition name), and the manager felt that expecting him to remember a new name was unreasonable. That's failing to empathize. Contrastingly, in May 2020, a black male bird-watcher in New York's Central Park asked a white woman to leash her dog (in an on-leash area). The woman called 911 and claimed he was threatening her [149]. That's being too sensitive.

Both failing to empathize and oversensitivity create toxic workplaces. Failing to empathize erodes individuals' self-esteem, sense of belonging, etc. Being too sensitive makes forthright conversations and conflict resolution impossible because everyone's so afraid of causing offense that they hide their true feelings. Complicating matters, people often dismiss toxic behavior by claiming the victims are being too sensitive *and* dismiss oversensitivity by claiming prejudice. So, how can we differentiate reasonable from unreasonable behavior? There's no universal answer, but we can suggest the following considerations.

- Consider the implicit social contract governing the situation. What would most people view as normal? For example, if Maggie is upset at Prem for eating his lunch at lunchtime in the break room because Maggie dislikes hearing people eat, Maggie is being too sensitive. But if Maggie is upset at Prem for watching pornography in the break room, Maggie is justified because Prem's behavior is outside the bounds of the implicit social contract governing the workplace.

- Consider if the behavior is repeated. For example, if Prem comments that Maggie looks good in that outfit once, and Maggie immediately submits a formal complaint, Maggie might be overreacting. But if Prem comments on Maggie's looks every day, Maggie is more justified in complaining.

- Consider whether the problematic behavior has been discouraged. For example, if Prem comments that Maggie looks good every day, and Maggie smiles and thanks Prem, there's no signal that the behavior is hurtful. In contrast, if Maggie tells Prem

that Prem's comments make her uncomfortable, *and Prem keeps doing it anyway,* Prem is more clearly in the wrong.

- Consider intentions.[3] Suppose Prem is reviewing Maggie's code, and gives some blunt criticism. Maggie is more justified in complaining if Prem is intentionally trying to undermine Maggie's credibility and contributions than if he just isn't very good at phrasing critiques.

None of these considerations are absolute. People have different perceptions of the implicit social contracts governing our workplaces. Some behaviors are so unreasonable that it doesn't matter that it was only one time, the behavior hadn't been previously discouraged, or the intentions were good. But creating a healthy, just work environment requires a dual willingness to (1) empathize with others and try to understand their diverse perspectives and (2) avoid oversensitivity and defensiveness.

Unconscious Bias and Judgment

Unconscious biases also lead us to *judge* people differently depending on the groups to which we think they belong. For example, suppose we asked 100 managers at 100 different companies to each rate the job performance of their software developers. We might find that people who drink kombucha get better ratings. Is job performance actually correlated with kombucha consumption? No. Is this a huge problem? Probably not. But across many industries and job types, subjective assessments of job performance are correlated with gender, race, sexual orientation, disability, age, physical attractiveness, socioeconomic class, and many other factors in ways that do not reflect real differences in job performance.

Unfortunately, people tend to be less concerned about unconscious bias, less prone to hold others accountable for them, and less supportive of efforts to combat them [150]. Mitigating unconscious bias isn't easy. Fischoff [151, p. 426] identified five approaches to addressing unconscious bias:

1. "warning about the possibility of bias without specifying its nature";
2. "describing the direction (and perhaps extent) of the bias that is typically observed";
3. "providing a dose of feedback, personalizing the implications of the warning";
4. "offering an extended program of training with feedback, coaching, and whatever else it takes" to overcome the bias; and
5. redesigning the person-task system to mitigate the bias.[4]

[3] At this point, some readers familiar with the contemporary discourse around workplace behavior are cracking their knuckles before embarking on an epic complaint letter explaining how intentions don't matter as much as impact. Of course good intentions do not always excuse hurtful behavior but ignoring intentions is bad moral philosophy. No reasonable person thinks that accidentally bumping into someone and knocking them over is morally equivalent to tackling somebody to the ground intentionally. Intentions do not excuse us from responsibility for our actions, but hurting people on purpose is obviously worse than hurting people by accident, especially when the accidental harm is difficult to predict.

[4] This one isn't a direct quote because Fischoff didn't put it quite so concisely. Give him a break. Four out of five ain't bad!

The problem with options one, two, and three is that they don't work [151]. The problem with option four is that it only sort of works. Many organizations offer some form of *unconscious bias training*. "Unconscious bias training can be effective for reducing implicit *bias*, but it is unlikely to eliminate it ... [or] change *behavior*" [152, p. 6; italics ours]. Moreover, "unconscious bias training interventions are not generally designed to reduce explicit bias and those that [are] have yielded mixed results" [152, p. 6]. Furthermore, unconscious bias training is expensive, time-consuming, and breaks down every time someone new joins the team. To be clear, we don't think unconscious bias training does any harm; it's just not a silver bullet.

Therefore, our focus for the remainder of this chapter is on option five: accept that people have unconscious biases and redesign systems and processes to suppress or mitigate these biases. Structural debiasing works. For example, when hiring, unconscious biases make us more likely to shortlist candidates who are more like us. Removing certain identifying information like names and pictures from résumés mitigates these biases.

Armed with this background on human biases, we can now move on to our first foundation of justice: increasing the number of perspectives.

Increase Perspectives

The first foundation of organizational justice is increasing the number of perspectives. Diverse software teams include people with different genders, ages, sexual orientations, abilities, experience levels, and ethnic, cultural, and economic backgrounds. While there is no such thing as a perfectly diverse team, some teams are clearly more diverse than others.

More diverse teams and organizations have more varied perspectives, which makes them more resilient [153] and better at sensemaking, critical thinking, decision-making, and innovating [154]. However, these varied perspectives also increase intragroup conflict, which damages trust, team cohesion, communication, cooperation, and job satisfaction [155].

This mix of benefits and drawbacks creates the so-called *diversity paradox*: "If [firms] embrace diversity, they risk workplace conflict, and if they avoid diversity, they risk loss of competitiveness" [156].

The diversity paradox and the purported drawbacks of diversity are bogus. What's actually happening is organizations hire people who don't fit their (typically young, straight, white, male) employee archetype without any attempt to be inclusive, tailor support, or share power. In the USA, for example, organizations frequently:

- hire Muslims, Hindus, Sikhs, Buddhists, etc. and expect them to work through their religious holidays and festivals;
- hire women but provide no maternity benefits; and
- hire people with disabilities without accommodating their disabilities.

If you join a team that systematically excludes you, treats you differently from everyone else, and doesn't understand or accommodate your needs, then you may struggle to trust them, identify with them, or enjoy your job. It's not because the team is more diverse; it's because the organization mistreats everyone who doesn't fit its employee archetype.

Recruit a Diverse Workforce

Organizations can hire staff in many different ways. They can post job ads online and see who applies. They can contact people who've expressed interest in working for them in the past. They can ask current employees for recommendations. They can hire recruiting agencies to find candidates for them. They can directly approach individuals (typically with big reputations in a certain area). They can poach employees from competitors.

Organizations can increase diversity by hiring interns and addressing issues throughout the hiring process, from job descriptions and advertisements to résumé screening, interviewing, and making offers.

Hire Interns

The best way to hire early-career professionals is through a paid internship program. From the perspective of a software company, internships are not about supporting the local university or getting cheap workers; the primary purpose of an internship is to evaluate potential hires. Working with someone for a few months is vastly more effective for assessing their skills and fit for your team than any interview. When the intern graduates, you will know whether you want to hire them.

From a diversity perspective, paid internships help attract candidates from lower socioeconomic backgrounds. Unpaid internships exacerbate discrimination and should be avoided. Organizations can also grow their personnel funnel by supporting code boot camps, apprenticeship programs, and diversity-themed networking groups. The only real drawback of the internship program is that it mainly applies to early-career hires.

Internships are relatively inexpensive and low-risk. If you get a terrible intern, just limit their activities so they don't break anything and they'll be gone in a few months with no hard feelings. Compare this to hiring a professional developer who costs 5–10 times as much and could sabotage the company, be difficult to fire, and sue for wrongful dismissal.

Some tips for good intern experiences include the following [157].

- Recruit interns with good academic records. Academically well-prepared interns tend to have better internship experiences and contribute more to their host companies.
- Pay interns. Expecting interns to work for you for free is discriminatory (and illegal in many places). Reputable companies do not engage in volunteer internship scams. You are not doing the intern a favor by providing a volunteer internship. Yes, they are learning from you, but they are also working for you, and they have to eat and pay rent like everyone else. Pay interns at least a little more than minimum wage.
- Clarify your expectations; namely, that interns treat their internship as a real job, proactively find suitable tasks, ask questions, treat every assigned task as a learning experience, and ask for help when they need it.
- Thoroughly on-board your interns. Introduce them to their teams. Start them with simple, well-defined tasks and allow them to advance to more complex tasks as they are able. Get them into the regular pair programming rotation early, even if they can't contribute much yet, so they start to feel like part of the team. Avoid giving interns busywork.

- Treat interns with respect. Interns may lack experience, but they are adult human beings due the basic respect and dignity afforded any other adult human being. Call them by name; do not refer to anyone as "the intern" or by some stupid nickname they might secretly hate.

Of course, most organizations won't be able to fulfill all their staffing needs through an internship program, so we still need to improve conventional hiring practices.

Write Diversity-Friendly Job Ads

From a diversity perspective, publicly posting job openings is important. Hiring people known to or recommended by current employees typically exacerbates homogeneity. Directly contacting qualified individuals *can* be used to recruit more diverse staff, but again, the people known to the organization often resemble the people who already work there. Recruiting agencies, meanwhile, "could promote workforce diversity, while in reality they often inhibit it" [158, p. 195]. In most cases, therefore, we recommend handling recruitment in-house and advertising all openings.

The next step is writing job descriptions and advertisements that will attract a diverse pool of candidates. Job ads should include a *diversity statement.* Here is an example from a recent position at the university where Paul works:

> *We encourage applications from people of culturally diverse backgrounds; equity target groups include women, people with disabilities, people who identify as LGBTQ+, and people of Aboriginal or African descent. Applications from all qualified candidates who would contribute to the diversity of our community are particularly welcome.*

Diversity statements help to convey your organization's commitment to justice. Alone, however, they're not sufficient.

Job ads tend to contain a lot of coded language: statements, phrasing, or omissions that subtly imply the kind of candidates desired. For example, phrases like "fast-paced and competitive environment," "strong work ethic," and "commitment to organizational goals" are often code for a stressful job with unpaid overtime and no work-life balance.

Coded language can discourage certain candidates. For example, "Job advertisements for male-dominated areas [employ] greater masculine wording" (e.g., competitive, dominant) and this wording often makes the job less attractive to female applicants [159]. Indeed, the job ad may be less attractive to any applicant who values collaboration over individualism. Racially charged language discourages racialized persons from applying. For example, after Seth Godin's book *Tribes* was published in 2014, North American businesses rapidly adopted the term *tribe* for marketing and recruiting despite it carrying negative connotations for black and indigenous peoples. Since remembering all the problematic words and phrases is difficult, we recommend using an automated text analyzer tool to detect and remove coded language.

Moreover, what *isn't* said can be just as important as what is said. For example, omitting descriptions of benefits like flex work and remote work may turn off candidates with caregiving responsibilities, while omitting financial benefits may discourage applicants from lower socio-economic backgrounds. Current text analyzers don't help with these kinds of omissions, so it's up to you to include all relevant benefits.

One last thing: don't expect candidates to jump through stupid hoops to apply for your positions. People should apply for (professional) jobs by submitting a résumé and possibly a cover letter. Don't ask people to copy-paste information from their résumés into some convoluted form, convert their résumé into a specific format (e.g., a spreadsheet), provide data that isn't really pertinent to the job, take tests (e.g., a personality test), or complete time-consuming tasks (e.g., programming puzzles) at the application stage.

Some people think a difficult application process will discourage weaker applicants, but they've got it backward. Difficult applications turn away the *strongest* applicants because they always have other opportunities. Keep your application process simple, easy, and transparent. Be clear about what the application process is. Unqualified applications can be weeded out. Qualified people who don't apply cannot be weeded in.

One thing that can help without unduly irritating your applicants is asking them to explain, in a paragraph or three, why they believe they're the right person for the job. Letting people explain their qualifications and worth in their own words—letting applicants tell their own story—tells you much more about them than most typical résumés.

Screen Candidates without Biasing Information

Screening—that is, selecting or *shortlisting* the most qualified candidates—is particularly fraught with unconscious bias [160]. Prejudice in résumé selection is often triggered when the screener (correctly or incorrectly) infers personal characteristics of the applicant from the information provided. For example, if Mohammad Arif graduated from the Islamic University of Madinah in 2002, we might guess that Mohammad is a Muslim man in his late forties from Saudi Arabia. These guesses can then activate group stereotypes and trigger discriminatory decision-making.

To de-bias the screening process, try the following.

1. Make a list of information that is useful for screening but does not readily support inferring personal characteristics (e.g., level of education, whether qualifications are relevant to the job role).

2. Have someone not directly involved in hiring extract the desired information from the applications / résumés into a spreadsheet. Do not include candidate names; instead, give each candidate a unique identifier (e.g., "A01") and make a separate *key* linking identifiers to names and contact information.

3. Have the people who *are* directly involved in hiring use the anonymized spreadsheet for shortlisting.

Do *NOT* use AI to screen candidates. AI screeners have baked-in biases that usually can't be explained or detected. Indeed, some less reputable companies use the AI to launder discrimination like criminals use a casino to launder money.

Adopt Working Interviews

Conventional job interviews (in which applicants answer a series of questions) don't work. Candidates may lie. Interviews select for charisma, charm, extraversion, and physical appearance regardless of whether these attributes matter for the job in question. Conventional

job interviews don't adequately assess technical skills. Google-style mind traps ("suppose someone magically shrinks you down to 3 cm tall and drops you in a blender; you have five seconds before they turn it on, what do you do?") tell you how good people are at silly thought puzzles, not how good they are at unit testing or user-centered design. Some companies ask obscure technical questions that don't reflect the job, expect interviewees to have memorized vast amounts of syntax in a specific language, or place artificial burdens on interviewees to see how they handle working under pressure.

Interviews also discriminate against neurodivergent candidates, particularly candidates who are prone to anxiety, even if a little anxiety wouldn't impair their job performance. Interviews trigger stereotyping and prejudice because the interviewer can't help but see and hear the interviewee.

The best way to hire early-career professionals is through a paid internship program, but when internships are not feasible, we recommend *working interviews* in which the candidate performs tasks similar to those found in the job.[5] However, giving an engineer an unfamiliar task to complete in an unfamiliar integrated development environment (IDE), programming language, etc. tells you how the candidate will perform on their first day rather than in the long term. You also can't give every candidate different tasks, or you won't be able to compare their performances. We recommend the following two-phase process, which was successfully demonstrated by Rob Mee, founder of Pivotal Labs.

Working Interviews Phase One

Phase one consists of the following steps.

1. Before any of the interviewees arrive, the interviewer prepares a programming task and a script comprising a task description and a series of guided, open-ended questions.
2. When an interviewee arrives, the interviewer greets the interviewee.
3. The interviewer begins the script. They describe the task. They remind the candidate that it doesn't matter if they know the programming language since their job is to talk in pseudocode and let the interviewer transcribe their discussion into the programming language's syntax.
4. The interviewer shares their computer screen and does all the typing.
5. The interviewer begins iterating between asking questions (e.g., "What could we test now?" or "How would we refactor this method to remove duplication?") and implementing the interviewee's ideas.
6. When the candidate answers the question well, the interviewer types the code and moves on.
7. When the interviewee answers poorly, they lose one point, and the interviewer asks a follow-up coaching question to help steer the candidate in the right direction. If the candidate answers well, the interviewer types the code and moves on.

[5] Check with your legal team before implementing working interviews. Depending on your jurisdiction, you may need to pay applicants a certain amount or have a certain kind of contract.

8. If the interviewee still can't complete the task, they lose a second point, but then the interviewer completes the task and moves on anyway. The interviewer doesn't tell the candidate that they lost a point; they just subtly make a tick mark on a piece of paper or make a mental note.

Moving on helps maintain the candidate's sense of progress and accomplishment from building code together, often in a programming language they didn't know. (You want applicants who *aren't* hired to tell their friends how pleasant the interview process was, not badmouth your hiring practices.) Moving on also limits stress by implicitly communicating that individual mistakes are not deal-breakers—'the interview is still going, so I must still have a chance.'

This process produces a score for each applicant. To proceed, you can select the top n applicants (for some value of n) or all applicants who exceed some threshold, t (for some value of t). Over time, it's easy to develop a reasonable threshold—too high and you won't fill your vacancies; too low and you'll spend too much time on phase two.

While the score is the main driver of selection, other observable behavior matters. If the interviewer felt that the interviewee was worth a risk (because they displayed amicability, empathy, fast learning, etc.), then the interviewer could recommend moving them forward despite a weaker score. Conversely, if the interviewer felt that the candidate displayed some red flags (e.g., inability to accept criticism, interrupting, unwillingness to collaborate, complaining about the exercise), the interviewer could recommend not moving them forward despite a stronger score. This subjective evaluation may allow some prejudice to creep back into the process but it's unavoidable because technical skill is not the only thing interviewers need to assess and there simply is no objective way to assess temperament.

The purpose of phase one is to score candidates' coding skills, reliably, in no more than one hour per candidate. The sessions improve comparability across applicants by being highly scripted and similar in length. They negate familiarity issues by using routine tasks, having interviewees pair program with interviewers, and beginning with an empty test and an empty class.[6]

Working Interviews Phase Two

Selected applicants move on to the second phase: two half-day coding sessions. Schedule these sessions on the same day if possible.

You want to give candidates a sense of working in your codebase and following your practices. Choose a story that: (1) you've implemented recently, (2) took 2–4 hours, (3) involved some development and some refactoring, and (4) doesn't require too much domain context or explanation. (At Pivotal, if a good story could not be found, the company had several canned projects to use.)

The interviewer reverts the codebase to before the story was implemented. When the candidate arrives, the interviewer reminds them that since these are real stories, there is no expectation for how long the work should take, and finishing the story in the allotted time may be unrealistic. The interviewer then shows the candidate the story with its attachments (e.g., mock-ups, acceptance criteria), verifies that the candidate understands what needs to

[6] Pivotal used object-oriented languages, but the principle of beginning with a clean slate applies more broadly.

be done, and discusses how the candidate might approach the story. The interviewer briefly overviews the codebase, indicating which files might need modifying. Like in phase one, the interviewer and candidate pair program, beginning with the interviewer doing all the typing. Once the candidate seems comfortable, the interviewer offers the candidate a chance to type. If the candidate is comfortable typing, the interviewer shifts to coaching; otherwise, they maintain the same roles. At the end of the session, the interviewer and candidate reflect on what they accomplished. Later, the code is discarded.

By repeating the same story with each candidate, the (same) interviewer can compare candidates' performances. While not as comprehensive as a multi-month internship, a full-day coding interview is about as much as one can expect for a mid-career hire.

Alternatives to Working Interviews

Working interviews may work for non-programmer roles; however, we have not observed any or found any scientific studies about working interviews with product designers or product managers. Therefore, we have no basis to predict their effectiveness. Constructing appropriate simulated tasks for these roles may be more difficult.

One common alternatives for designers is to submit a portfolio of their prior work. A portfolio-based approach is also useful for hiring developers active in open-source communities. However, some work is difficult to represent as a portfolio, and some applicants are forbidden from sharing their prior work. Moreover, you need some way to verify that the candidate actually did the work in the portfolio, and a portfolio doesn't tell you much about communication and teamwork skills.

Other options for interviewing product designers and product managers include critiquing an existing product or role-playing common tasks like interviewing a user, prioritizing a backlog, or drawing a mock-up. Again, there isn't good evidence for which approach works best, but we do know that conventional question-answer job interviews don't work very well, so we recommend experimenting with alternatives.

Hiring

Whatever your approach, assess lots of different skills associated with the role and avoid focusing on attributes that aren't actually necessary for the role (e.g., physical attractiveness). Avoid potentially prejudicing information before the decision. Even knowing that a candidate attended a top-tier school can bias scores. Stick with the anonymized spreadsheet. Do not look through the original résumés until preliminary decisions have been made. Do not ask candidates to share potentially compromising information like private social media accounts.

Keep the task consistent across applicants and use a consistent scoring system. We can't just make up the scoring system as we go because we'll invent different systems for different candidates, and these differences inflate prejudices. One specific problem we've observed is interviewers perceiving someone as more technical, then asking more technical questions, which reinforces the assumption that the candidate is more technical.

Moreover, to achieve greater diversity in your organization, keep positions open until at least one candidate who would add to the organization's diversity makes it to the last round of evaluation. Focus on the diversity of your hires, not your candidates. If you

don't get (m)any applicants from underrepresented groups, focus on filling your funnel (e.g., by advertising to more diverse audiences, sponsoring diversity-focused events, and including diversity statements in job ads). But don't hire unqualified people simply to increase diversity; that's unfair to both the applicants and your team.

All that said, hiring a diverse workforce is insufficient for organizational change. *Retaining* a diverse workforce and *benefiting* from their varied perspectives requires changing the organization, as we'll discuss next.

You Belong Here

Inclusion is the degree to which a group, community, organization, or society shares power, respect, responsibility, and resources with *all* of its members. Inclusion is not only about people feeling like they belong; it's about how the group behaves toward individual members.

The benefits of a diverse workforce will only materialize if employees feel included. For example, suppose a French software company is modifying a legacy product for a new market: Egypt. Realizing they lack local knowledge, they try to hire an Egyptian product designer. This is expensive, time-consuming, and frustrating. They have to find a qualified Egyptian who speaks French, jump through hoops for l'Office français de l'immigration et de l'intégration, cover moving expenses, and offer a relocation bonus.

Eventually they find a highly-qualified candidate: Eman. Now, suppose the localization team ignores Eman's suggestions. When Eman suggests that a certain user experience structure will work better for Egyptians, they say, "That's not how we do things around here." After a while, Eman stops trying to bring their own perspective and replicates the company's ways of working. Now, the company's efforts at diversity are wasted because, while the team is more diverse, their *perspective* is not.

Recruiting people for their varied perspectives and then tacitly insisting they abandon those perspectives and toe the company line seems pretty daft, but organizations do it constantly. No one says, "Abandon your foreign perspective." They say, "You have to learn our company culture" and "That's not going to work here."

To benefit from a diverse workforce, organizations and teams must embrace varied perspectives from onboarding onward.

Fix Onboarding and Create a Welcoming Environment

Onboarding is the process of orienting and training new hires (or sometimes internal transfers). At some organizations, onboarding is limited to showing new employees to their cubicles and perhaps wishing them good luck—more like nonboarding. Failing to socialize new employees implicitly communicates that the employee (a) is not really valued and (b) is on their own.

Even when organizations have a more substantial process for socializing new employees, they tend to underestimate the magnitude of their internal jargon, norms, and assumptions. People often throw around obscure acronyms or terms that have different meanings to different people.

New hires need more extensive onboarding, including things like:

- an office tour;
- icebreakers, Q&A sessions, social events;
- documentation of team norms and practices;
- codebase walkthroughs;
- technical training related to the team's tooling;
- welcome emails, calls, kits (including, e.g., a glossary of buzzwords and jargon or lists of contacts), or meetings (with a representative from human resources or upper management);
- a buddy program and / or a mentor program;
- announcements of new hires; and
- a swag bag.

Some of these onboarding practices are simply intended to make new people feel welcome. However, as Cable et al. [161, p. 24] explain:

> *The traditional methods of onboarding ... assume that organizational values are something to be taught to and adopted by newcomers. This creates a tension: when newcomers are 'processed' to accept an organization's identity, they are expected to downplay their own identities, at least while they are at work. However, subordinating one's identity and unique perspectives may not be optimal in the long run for either the organization or the individual employee because suppressing one's identity is upsetting and psychologically depleting. Moreover, newcomers may not internalize the organizational values even if they appear to comply through external behaviors; over and above compliance, leaders need employee engagement if they want employees to contribute on their own and in ways that are not programmed. Socialization practices that get newcomers to behave inauthentically might not be sustainable because they do not fully engage the employee and they do not address broader issues concerning emotional exhaustion and work dissatisfaction.*

Instead of indoctrinating new hires, use the onboarding process to embrace the new hire's individuality and create an environment in which they can flourish by doing the following.

1. Encourage the new hire to "just be yourself" during onboarding events. Frame the workplace as a somewhere employees can be authentic. Accept myriad personal styles as long as people aren't hateful, toxic, or harmful to others.
2. Ask the new hire to identify their strengths and how their team can best take advantage of these strengths. Frame the job as endless opportunities to be your best self, to excel, and to grow as a person.

3. Ask the new hire to identify the conditions under which they flourish, and then manifest those conditions. For example, if the new hire does their best work at a standing desk, get them a standing desk.

4. During introduction sessions, specifically ask new hires to introduce themselves by talking about what they're good at, what they're like at their best, and their personal blockers (e.g., some people excel in hectic, high-pressure environments; others do their best work alone in quiet, low-stress environments). Try introducing someone from an often marginalized group "based on what they lead, and not what they do" (see Figure 12–1).

5. Provide a checklist for the new hire to chart their self-paced onboarding. The checklist is reviewed with their manager at expected intervals of day one, week one, month one, and month three. A written checklist (instead of verbal to-dos) facilitates self-assessment of onboarding progress.

6. Remind new hires that you expect them to be learning sponges for their first three months and not necessarily deliver work immediately. Acknowledge that they may still feel the need to deliver tons of value—that's normal—but you don't have that expectation.

This is not an exercise in lip service. If you tell people to be themselves and then expect everyone to dress / act / speak / be the same, or ask people their strengths and then assign them tasks unrelated to their strengths, or ask people what they need to succeed and then don't get them any of it, then they will rightly conclude that you're disingenuous. You have to follow through.

Embracing diversity doesn't mean abandoning all organizational norms. Teams can have coding standards. Organizations can have policies. Places can have ways of doing things. It just means respecting people as individuals and being open to new ideas.

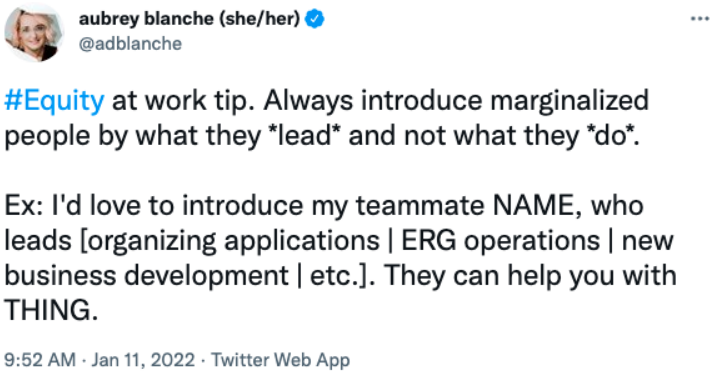


Figure 12–1: Aubrey Blanche's Introductions Technique [162]

Practically speaking, whether your onboarding program includes icebreakers, a buddy program, technical training, etc., isn't as crucial as conveying that you value the new hire and their unique perspective.

Retain Diverse Talent

All this work recruiting and onboarding a diverse workforce will only pay off if organizations *retain* their workers. Significantly elevated turnover rates happen for people with underrepresented identities [163], [164], [165] because experienced or perceived discrimination reduces their organizational commitment, limits the ability to advance in their careers, and makes them want to seek other opportunities [166]. Employees from underrepresented groups have more negative experiences at work. From outright discrimination to lack of mentoring, these employees face more barriers and are therefore less included. Preliminary research in Paul's lab suggests that perceived organizational justice is one of the strongest predictors of employee retention. The most effective tactic for addressing this issue is sharing power.

Create an Inclusive Climate by Sharing Power

Creating an inclusive organization is about power. Who has the authority to make decisions? How are decisions made? Whose concerns are elevated, and whose are marginalized? Does everyone in the organization understand how decisions are made and who has what authority? How is the power wielded (justly or unjustly?), and for whose benefit?

To move beyond lip service, organizations must replace theatrics (creating the illusion of caring about inclusion rather than actually fostering an inclusive culture) with a focus on power-sharing and transparency. Write down who has what authority, discuss who *should* have what authority, and *include* everyone whose lives are affected by the outcome of a decision in the decision-making process. Consider the problems and fixes suggested by Okun and Jones [167] summarized in Table 12–1.

Be a Good Mentor

Employees at each stage of their career need mentoring to reach their full potential. Mentoring has the following three dimensions.

1. **Role-modeling**: exhibiting the attributes and actions you believe your mentees should exhibit; being an effective professional and a good person.

2. **Advising:** listening to the mentee's struggles, providing recommendations (especially around professional development and career progression), sharing insider knowledge about how to advance in the organization, helping the mentee strategize for advancement, and perhaps even teaching some new skills directly.

3. **Brokering:** taking an active role in driving the mentee's career forward by facilitating professional networking and actively creating opportunities for the mentee.

What differentiates mentoring from merely advising is a focus on career advancement (i.e., *brokering*). Brokering must be tailored to the mentee's goals. For example, when Paul has a graduate student who wants to be a professor, brokering means bringing the student

Table 12–1: Aspects of Organizational Culture That Prevent Inclusion (adapted from [167]*)

Problem	Symptoms	Fixes
Paternalism and Power Hoarding	Authority is centralized in people who believe they should make decisions for those without power.	Include in decision-making everyone affected by decisions; explain that effective leaders share power; demand economic democracy.
Opacity	Those with power don't explain why, how, or by whom decisions are made; authority is undocumented.	Demand transparency; document authority and decision rationale; reject organizational propaganda.
Defensiveness, Right to Comfort, and Fear of Open Conflict	The organization prioritizes protecting existing power structures, suppresses criticism, and maintains the status quo; making those with power feel uncomfortable is seen as deviant. Leaders suppress conflict, conflate raising difficult issues with rudeness and demand politeness.	Name defensiveness as a problem and explain how it sabotages the organization's mission; understand that discomfort is essential for growth and learning. Do not expect people facing serious problems to raise them in acceptable ways.
Only the Right Way	Our way is the only right way; anyone who can't see that is broken.	Accept that goals can be achieved in many ways, but once a group has chosen a way, honor it; call out people acting like their way is the only way.
Metrics Fixation	Only measurable outcomes matter; discomfort with emotion.	Find ways to measure important things that are being ignored; create your own scales; argue that qualitative accounts of the right things are better than quantitative measures of the wrong things.
Perfectionism	Focusing on what's wrong; failing to learn from mistakes because making mistakes is conflated with being a mistake.	Give balanced feedback; focus on learning from mistakes instead of assigning blame.
Sense of Urgency and Either-Or Thinking	Illusion of urgency leads to constant rushing; no time to be inclusive. Complex situations are reduced to false dichotomies: right vs. wrong; with us or against us.	Make realistic plans; include inclusivity goals and time to work on them. When stakes are high, reject pressure to make important decisions immediately, slow down, think more deeply, and *generate more than two alternatives.*
Objectivity	Believing that it is possible to be objective and that emotions are bad; mistaking one's own perspective for "facts" while other perspectives are "opinions."	Understand that all human phenomena, from music to financial accounting, is subjective; push yourself to understand and validate others' perspectives.
Inclusivity Theater	The organization prioritizes appearing to care about inclusion over actually being inclusive.	Focus on the problems listed in this table; dispute the usefulness of common inclusivity theater techniques (e.g., wellness committees, diversity-inclusion training); call out people paying lip service to inclusion while acting exclusionary.

*Adapted to apply more specifically to tech companies; *inclusivity theater* is our addition.

to conferences, introducing them to other influential professors, involving them in academic societies, getting them peer review work for scholarly publications, writing them reference letters, helping them apply for scholarships and awards, and most of all, involving them in many research projects and publications to expand their résumé.

Analogously, brokering for an ambitious young developer means actively driving their career forward by introducing them to influential people within the organization, getting them involved in the kinds of teams and projects that are likely to get noticed and lead to advancement, writing reference letters, publicly lauding their contributions, nominating them for internal awards, and putting them up for promotion. Brokering means taking an active role in advancing the mentee's career. A broker puts their own reputation at stake to advance the mentee.

Be a Good Ally

Brokering is related to allyship. An *ally* is like a broker for a person or group who faces some discrimination or oppression, for example, due to their gender, ethnicity, religion, sexual orientation, age, or disability. You don't have to be someone's mentor to be their ally. Allyship isn't usually a formal arrangement.

Juliet Bourke [168] recommends the following three simple allyship techniques.

1. **Helping each other out** by making introductions, sharing needed information, and amplifying others: "During my observations, I often saw peers subtly endorse and amplify each other (e.g., 'As Pedro said. . . '), thus helping to underscore a peer's point and increasing their potential influence over proceedings."

2. **Taking emotional care of others**, including showing honest interest in their personal lives, creating safe spaces for venting, or simply checking in on a peer.

3. **Making physical connections** with one's body and sharing space by "walking together to meetings, deliberately sitting next to each other, or if a meeting was virtual, sharing their personal backgrounds rather than using an impersonal corporate photo, and exaggerating positive non-verbal cues."

Try Reverse Mentoring

One effective technique for training upper management in inclusiveness is *reverse mentoring.* "Reverse mentoring is defined as the pairing of a younger, junior employee acting as a mentor to share expertise with an older, senior colleague as the mentee" [169]. While initially conceived as a means of sharing technical expertise, reverse mentoring facilitates the sharing of cultural perspectives and promotes diversity and inclusion. When a senior professional, manager, or executive from a dominant group takes on reverse mentors from frequently disadvantaged groups, the mentors can help the mentee get more in tune with the concerns of these groups. This isn't just about helping the mentee be more politically correct, it's about helping the mentee understand more diverse perspectives. Meanwhile, the mentors benefit from access to senior management, social capital, and leadership development [169].

Critical success factors for reverse mentoring include:

- "emphasize diversity;"
- "consult mentees before making the pairing formal;"
- address mentees' fear of revealing their lack of knowledge;
- "ensure strong commitment from mentees;" and
- train the mentors to structure their sessions with the mentees effectively [170].

In addition, reverse mentoring is *work* that must be *compensated*. Mentors must be released from regular duties to make time for mentoring, and mentoring must count for performance appraisal and promotion.

Avoid Fake Consultation

Many corporations, non-profits, and government agencies prepare for significant decisions through consultation (e.g., focus groups, town halls, questionnaires) to understand stakeholders' preferences and priorities. In our experience, consultations are often fake: they backfill justifications for already-made decisions, or gauge upcoming pushback and determine how to minimize it. Don't do fake consultations. Including people in decision-making means actually incorporating everyone's perspectives. Genuine consultation can be challenging. People have inconsistent—often incompatible—priorities. We can't make everyone happy, but people are much more accepting of decisions they disagree with when you're transparent about conflicting priorities and explain why you believe your solution is the best compromise.

Run Inclusive Meetings

Much of our advice in this chapter is directed to those in positions of authority, but one place where everyone can help foster a more inclusive culture is in meetings.

The most important inclusivity practice in meetings is to ensure that everyone who has a stake in the outcome of the meeting is invited to the meeting. This sounds obvious but is tricky in practice. Team members dynamically self-organize to solve problems. For example, an engineer facing a technical challenge might go directly to the team member they believe is most likely to be able to help and hash out the challenge one-on-one. For important engineering challenges, though, including the whole team is essential. This often takes longer, but if you ensure that the whole team understands the problem, the solution, and why this is the best solution, you'll have a more inclusive culture *and* the whole team will better understand the codebase.

The second most important inclusivity practice is good conversational turn-taking, meaning that:

1. only one person speaks at a time;
2. either no one interrupts, or there are specific rules about when and how someone is allowed to interrupt (e.g., a moderator is allowed to interrupt an off-topic discussion);

3. everyone in the meeting has a chance to speak; and
4. speakers are concise.

Good turn-taking does not mean putting people on the spot. If someone hasn't contributed to a discussion, you don't have to point at them and demand they add something. You can ask if anyone, especially anyone who hasn't already spoken, has anything to add.

Additional techniques to improve inclusivity in meetings include: speaking in reverse order of seniority (newest or least-experienced member first); repeating others' ideas to show that you're listening and confirm your understanding before offering your opinion; and amplifying ideas you agree with.

When amplifying an idea, explicitly credit its source (e.g., "I like Maryam's idea to refactor the service layer component..."). In our research, women often complained that their ideas were ignored until a man said the same thing and took credit.

Lastly, take care to avoid explaining things to people who already understand them and to avoid condescension.

Avoid Extra Work for Underrepresented Minorities

One pitfall to avoid is the tendency to volunteer team members from underrepresented minorities for extra work. Underrepresented minorities often take on additional work—not only mundane clerical work like note-taking but also education and advocacy work related to diversity and inclusion—which is often unrecognized, unrewarded, and unhelpful for career advancement.

When I (Todd) ran *Balancing Power Imbalances* workshops at Pivotal, I used to ask managers to recommend local co-facilitators. One recommended engineer explained that she'd prefer not to have been asked because she was tired of needing to say "no" to organizational justice efforts. Now I ask broader groups if anyone wants to volunteer so no one has to refuse explicitly.

Similarly, in Gusto's postmortem meeting template, the request for a volunteer notetaker is immediately followed by a reminder that underrepresented minorities disproportionately volunteer for the role, and maybe someone else could volunteer instead. Nudges like this can encourage better dynamics.

Confront Injustice in Real Time

> *Equality is a practice. It is how employees behave, leaders lead, and workplaces work.*
> *– Michelle P. King*

The old management adage "praise in public, criticize in private" is good advice unless a behavior causes significant, acute harm to others. Consider the following examples.

- Being late for meetings that require promptness (neither significant nor acute harm)—address in private. Sure, a consistently late teammate can be annoying and undermine a culture of punctuality over time, but that's not significant or acute enough to warrant calling someone out publicly.

- Misusing the travel budget (significant but not acute)—address in private. Despite being serious, nothing is gained by making a show about it in front of the whole team.
- Breaking the build (acute but not significant)—address in private, if at all. We've all broken the build. It's not a big deal. Publicly shaming someone for breaking the build will just make the team afraid to take risks.
- Calling someone a racist slur (acute and significant)—address immediately and publicly. In this example, calling out the offender publicly is not just about correcting the behavior, it's about ensuring everyone present sees that the behavior is unacceptable. Silence would imply that the behavior was OK.

Correcting behavior does not mean being a jerk. You don't have to yell, insult, or demean. In fact, calmly but firmly stating that some behavior will not be tolerated is often more powerful than losing your cool.

In practice, calling out misbehavior is intrinsically challenging because disturbing the interpersonal peace by "being rude" or "making a scene" is often seen as worse than the misbehavior itself. However, this attitude inhibits the organizational shift toward justice. Failing to confront injustice allows it to continue unimpeded and makes you complicit.

When you have to correct someone publicly, doing so calmly helps prevent others from protecting offenders by refocusing the situation on you "being rude." And yet, correcting people *too* kindly can also send the wrong message. If someone is abusing their co-workers, being too nice makes it look like you care more about the offender's feelings than the victim's. Calm but firm is usually best.

Confront Acute, Significant Injustice Immediately

When you must address injustice publicly, we suggest the following four-step process. Here, we are writing from the perspective of a concerned colleague who has no authority over the perpetrator or victim.

1. **Interject.** First, you must interrupt the situation by inserting yourself into it. You might say, "Wait a minute" or "Hold on" or "Hold up."
2. **Settle.** Take a deep breath. Give everyone in the situation a moment to settle down. If you are emotional, give yourself a moment. But don't wait long enough for the interruption to pass and the negative interaction to resume.
3. **Understand.** Ask the person(s) who engaged in the inappropriate behavior if you correctly perceived what just happened. You might say, "Did you just..." followed by whatever you think you saw or heard. Give them a chance to back down. Explain how the behavior could be harmful and specifically ask if that's what they meant. If they say, "Oh, that's not what I meant ..." (to say or do), play along.
4. **Educate.** If the perpetrator backs down, you might say, "In the future, let's try to..." and explain how the situation could be handled better. If the perpetrator doubles down, you might explain that the behavior makes you uncomfortable or isn't acceptable. Focus on how things should be done in the future. If the situation escalates,

suggest everybody take a break to cool down or include your manager or HR representative to mediate. Even if you can't de-escalate the situation, you still did your part. You voiced opposition to injustice. You showed the victim(s) of the injustice that you are on their side and that someone does care.

Here's an example of how this could go down. If you find intervening like this intimidating, really imagining specific co-workers in these roles and reading your lines aloud may help you feel more able to intervene.

```
Man [to woman]: You really should smile more.
You: Hold on, [deep breath] did you just tell her to smile more?
Man: Yeah. What's it to you?
You: There's this whole thing about not telling women to smile more.
It's like you're saying she's got to prioritize her appearance to you
over how she actually feels. Is that what you meant?
Man: Of course not. But everyone looks better when they smile. That's
not sexism, it's just human nature.
You: Sure, and you'd look better in a three-piece suit, but we aren't
the fashion police, so let's agree not to offer any unsolicited advice
about each other's appearances and get back to work.
```

In the fictional conversation, the "you" role used humor to take the edge off a difficult conversation. That's just one style, and other styles might work better for you. If you miss your opportunity to address an incident in real time, Michelle Mijung Kim [171] suggests the following prompts.

1. "Hey, can we check in about something that happened last week?"
2. "I'm sorry that I didn't address this earlier, but I've given it some thought and wanted to share this."
3. "I've not been able to get this off my mind. Can we have a quick chat?"

If you have more organizational power than the person, be more specific about the purpose of the meeting so that the person doesn't unnecessarily panic about a last-minute meeting with their boss.

Acknowledge Your Mistakes

Mistakes are inevitable, so we need to learn to acknowledge them, learn from them, and react gracefully when we are called out for making them. When you get called out, resist your natural impulse to be overly sensitive or defensive. You do not have to agree with what is being said to hear it, acknowledge that someone is upset, and consider their perspective. You don't need to decide whether you agree or not, or have a solution, immediately. You should probably take time to reflect on how to make it right. Calling someone out often takes significant emotional labor, and you can acknowledge their courage.

Finch [172] suggests the following phrases for acknowledging that you have room to grow:

- "I recognize that I have work to do."
- "I'm going to take some time to reflect on this."
- "I appreciate the labor you've put in."
- "I apologize; I'm going to do better."
- "I want to make this right."
- "What I'm gathering is [insert what you've learned]."
- "I'm doing some research now, but if anyone has a favorite resource on this topic that they'd like to share, I'm totally open to it!"
- "Thank you."
- "I believe you."

Re-envision the Tech Culture

We have to change our unconscious definition of what a software developer looks like, so we start recognizing talent that looks a little different than what we are used to.
– Ainsley Robertson [173]

Improving inclusion is all about culture change. One way to change an organizational or industry culture is through what Michelle P. King calls a "success prototype," the pattern or role model that others in an industry emulate to achieve success [174]. King argues that companies promote people who match their success prototypes.

Who are your engineering role models? Many tech professionals admire a brilliant engineer (who has spent years abusing colleagues who wrote code he didn't like) or an eccentric billionaire (who made some of that money through securities fraud). Is every role model you can think of a tall, attractive, wealthy, English-speaking, middle-aged, straight white man? Who does your organization put on stage? Who represents your team to outsiders? Who is in leadership?

You can help improve your organization's and industry's success prototypes by:

- identifying and praising people who don't fit existing success prototypes but have qualities you admire;
- getting your team to share opportunities for exposure (e.g., face time with upper management or clients, leading demos;
- choosing good role models for yourself and telling others about them;
- reflecting on and challenging who your organization and industry present as role models; and
- encouraging your organization to elevate more diverse success prototypes.

A good role model is someone who: you can relate to because they share some aspects of your background, challenges, or experiences; exemplifies the qualities you admire; and doesn't have a bunch of other qualities that are incompatible with your values. A role model can be a celebrity or someone you know personally.

At the start of my career, I (Todd) thought excellent engineering was a solo genius finding the best solution to a complex problem. In time, I realized that the best ideas usually emerge from collaboration. Being effective is not really about having the best idea; it's about collaborating with others to see ideas implemented. Unfortunately, many companies value engineers based solely on their technical skills, ignoring toxic behavior. They might reward individuals who "solve" complex problems over the weekend, despite negative impacts on the rest of the team who spend the next week fixing the "solution" so it actually works.

Many tech companies continue to devalue collaboration and communication. Consider Akila, a highly collaborative team lead who forged high-performing teams and routinely received excellent performance reviews. When her employer was acquired by a Fortune Future 50 and Fortune 100 Best Companies to Work For company, her new management devalued her collaboration skills and complained that she needed to be seen "doing more" by contributing to multiple projects simultaneously and reporting on the status of more work items—never mind that multitasking is inefficient [51]. She quickly moved to a new company that values agile principles and explicitly rewards and promotes engineers for improving team collaboration.

Another problematic trend is creating a two-tier system that disempowers young engineers by overemphasizing seniority. Does your organization make a big deal about who is a "senior" engineer and who is a "junior" engineer? Why? Years of experience is at best a weak predictor of developer job performance. Indeed, we routinely learn from our early-career colleagues despite our decades of experience. De-emphasizing hierarchy, especially within a team, usually produces better interpersonal dynamics. De-emphasizing hierarchy also helps organizations create more inclusive cultures because people from underrepresented groups are more likely to be perceived as "junior" and consequently disempowered.

Inclusive Software Process

Much has been written about optimizing software processes for efficiency, effectiveness, and agility, but what about inclusivity? Some argue that agile methods are more inclusive because they are more people-focused, which is true in principle but not always in practice. Agile was created by a relatively privileged, homogeneous group who were accustomed to having a lot of power within their organizations.

Consider the move from waterfall-like phased development to two-week sprints. A typical mid-career engineer in a medium-sized Silicon Valley company might find two-week sprints less stressful because they can manage their workload by controlling the sprint's scope. In contrast, a typical mid-career engineer in a large Indian tech firm might have no control over their bi-weekly deliverables. For them, two-week sprints create incessant crunch time. Agile proponents might despair that "you're not supposed to do it that way!" But that *is* how it's done in many companies [175].

Similarly, many members of underrepresented groups have had negative experiences in pair programming and code reviews (see Chapter 10). These negative experiences don't

mean that sprints, pair programming, and code reviews are inherently exclusionary. An inclusive software process isn't really about selecting inclusive practices instead of exclusionary ones. *Inclusion is all about power: who selects the practices and determines how they're implemented.* Exclusion happens when practices are corrupted by power imbalances or imposed on people for whom those practices are problematic.

This is trickier than just letting a team determine its own software development process. The organization may have legitimate reasons for constraining the software development process in certain ways or standardizing the process across work groups. Even if teams set their own processes, the team as a whole could impose exclusionary practices on individual team members. And even if the team agrees on the practices, individual team members may still enact practices in an exclusionary way. So, how does one devise an inclusive software process?

Our advice is to apply the suggestions summarized in Table 12–1. To devise an inclusive software process, try the following.

- Include in designing the process everyone affected by the process.
- Transparently document both the process and its honest rationale. Consider co-authoring a norms document that makes the implicit, explicit.
- Leverage retrospective meetings to identify inclusivity problems.[7]
- Start from the assumption that there will always be inclusivity problems.
- Prioritize addressing inclusivity problems in the software process, *regardless of how those concerns are raised* (e.g., informally, impolitely).
- Be vigilant of people acting like there's only one right way to build software. You don't have to adopt a practice just because it's recommended by any particular book, community, or software engineering micro-celebrity. Finding practices that work for everyone on your team is more important than adhering to any particular guidelines.

Inclusive Product Design

Many tech companies seem pathologically unable to imagine any user who isn't like their employees. Too often, companies release software that doesn't work well for users who aren't tech-savvy, upper-class, young, English-speaking, disability-free, and dependent-free. Let's consider some examples.

Many software products (e.g., social media platforms) are used extensively by children and teenagers. Instead of providing an appropriate user experience, they simply state that minors aren't allowed to use the system and include some easily bypassed age verification questions in the account setup. Kids just lie about their age and carry on using the system. Meanwhile, the company sidesteps its responsibility to provide an appropriate experience for a potentially lucrative target market while sometimes harming its users.

[7] If the retro itself is identified as problematic for inclusion, adapt it as needed (e.g., conceal who proposed each retro topic, use a round-robin format where each person takes a turn speaking, in order, without interruption). For retros to surface inclusivity problems, the retro itself must be inclusive.

Every time a company changes the icon of a consumer-facing mobile application, many people *permanently* lose the ability to use that app. If Grandpa has moderate dementia, he might remember how to use a tablet to call his grandchildren. He might be able to figure out incremental interface changes. But if the *icon* changes significantly, it's game over. Grandpa can't find it, and there's a point at which many people with dementia just don't learn new things.

Meanwhile, much of the web is catastrophic for visually impaired users even though HTML that complies with the W3C Accessibility Guidelines[8] is among the best technologies for the visually impaired. Mobile apps frequently include infinitesimal fonts that screen readers cannot enlarge or read. Some products cannot be used without providing a home address, effectively excluding users who are unhoused, even when there is no reason to do so. And to this day, many products continue to feature binary "male or female" gender questions. It's not hard to include "other—please describe" with a free text response. Many of these apps don't really need to know the user's gender anyway.

Moreover, the hype around generative AI is palpable as we write this. Tech companies are scrambling to embrace the automated generation of text, audio, still images, and video in thousands of ways despite the demonstrable algorithmic bias.

While it may not be possible to design every product such that it works well for every possible user, greater awareness of user diversity and their involvement in product design is crucial for acting ethically and can provide a competitive advantage. If your product works well for the visually impaired and none of your competitors' products do, visually impaired users will be very loyal—they don't have a choice. Indeed, if you can foster a reputation of being friendly to any frequently victimized or disadvantaged group, you'll experience customer loyalty like never before.

Three keys to building more inclusive products are:

1. recruit a diverse team of professionals to build the product;
2. recruit a diverse group of users and other stakeholders for research; and
3. generate more diverse personas (see Chapter 7).

A more diverse team is more likely to recognize common falsehoods that programmers believe. For example, many programmers seem to believe that everyone has one first, one last, and maybe one or two middle names, all of which can be represented in ASCII and fit in, say, 30 characters. In contrast, the famous painter Picasso's full name was "Pablo Diego José Francisco de Paula Juan Nepomuceno María de los Remedios Cipriano de la Santísima Trinidad Ruiz y Picasso." Some people have names that mix character sets (e.g., including both Japanese kanji and Latin letters) that don't work in Unicode or include numbers or punctuation or words that are offensive in English. And that's just names. Programmers commonly hold similar incorrect beliefs about email addresses, dates and times, economics, geography, computer networks, measurement, and software engineering.[9]

More generally, designing more inclusive, accessible products is another area where your local university or nonprofit organization can be a fantastic resource. Want to know if your

[8] https://www.w3.org/WAI/standards-guidelines/wcag/

[9] For an extensive list, see https://github.com/kdeldycke/awesome-falsehood

system is hostile to minorities? Some scientists specialize in that. Is your system accessible to the local unhoused population? Go to a homeless shelter and pay people to try it. You'll get answers *and* feel like a good person.

Tailor Support

We can't tackle inequality if we are in denial it even exists.
– Michelle P. King

Equality refers to treating everyone the same regardless of individual differences. *Equity* refers to giving people what they need to succeed by considering their differences. For example, giving every employee the exact same chair aims for equality, while giving every employee a chair that fits their shape and needs aims for equity.

Equity means embracing individual differences and tailoring organizational support for each individual so that they can do their best work and be their best selves. For many developers, collaborating with people who act like they don't really belong is exhausting, demoralizing, and far harder than the technical challenges they face.

"Although all employees may behave more professionally at work compared to more casual settings ... conceal[ing] significant cultural aspects of themselves to minimize stereotyping ... is a necessary behavior for Black employees to be perceived as professionals" in many organizations [176]. Many women must routinely prove their technical abilities to their peers to be taken seriously. People transitioning between genders wonder how much of their authentic selves to bring to work for fear of discrimination. People hide disabilities to avoid being stigmatized.

We can create more equitable organizations by reducing these impediments in three key areas: organizational support, promotions, and feedback.

Support Diverse Employees

The more diverse your employees are, the more diverse their needs will be. The first step to supporting employees better is to *ask them what they need.*

To collect good data on employee needs, use *open-ended questions.* Simply choosing or ranking options generated by management produces junk data. Open-ended questions (e.g., "how can we best support you?") work much better.

You should also use *anonymous feedback mechanisms.* One-on-one meetings or focus groups may help in organizations that have high psychological safety. However, management tends to overestimate psychological safety, and the employees who need the most support are probably the ones who feel the least safe.

Some often-recommended ways of supporting employees include the following.

- Avoid hot-desking and open floor plans. Not having a space of their own makes already marginalized employees feel more marginalized and less like part of a team [177]. Encourage employees to personalize their spaces (e.g., with family pictures).

- Embrace flexible work locations. Having everyone in the office together is great, but some employees may sometimes need to work elsewhere for many reasons including caregiving responsibilities and personal safety.
- Embrace flexible hours. People from different cultural or religious backgrounds observe different holidays. People with complex family lives may need to work at unusual times. As discussed in Chapter 5, we recommend establishing core work hours and being flexible otherwise. Let each team set their own core work hours based on their unique constraints.
- Avoid overtime. Overtime, whether officially mandatory or an implied expectation, harms employees and usually backfires on organizations. The more hours people work, the *slower* they work, the more mistakes they make, and the less motivation they have.
- Allow employees to work part-time. For some employees, working part-time at a proportionately lower salary is a good trade. It's better to keep good people part-time than lose them entirely.
- Encourage breaks while pairing. While we are big believers in pair programming, it's OK to continue solo while your pair attends a medical appointment, picks up their kids, or shows a plumber the water heater that's just exploded in their basement.
- Consider paid time off, minimum vacation policies, and mandatory leave (i.e., employees are required to take all of their entitled vacation, parental leave, etc.).

We can look for ways to normalize different work hours. For example, Marchese et al. [178] suggest adding this phrase to communications for distributed teams: "My working hours may be different than yours. Please do not feel that you need to reply outside your normal working hours."

Fix Your Promotions Process

Creating a fair promotion process is exceptionally difficult. Promotion decisions tend to be intuitive and subjective or based on misleading indicators of performance (see the *Cognitive Biases Affect Promotion* sidebar on page 229).

Intuitive approaches to promotion suffer from significant biases because all sufficiently large organizations develop in-group / out-group dynamics. People naturally tend to divide the world into "us versus them." We divide ourselves based on nationality, race, gender, sexual orientation, religion, political ideology, etc. but don't always identify with others who are like us on any particular dimension.

We tend to perceive members of our in-group more favorably, pay more attention to them, and reward them. We are more likely to ascribe in-group members' successes to hard work and more likely to ascribe out-group members' successes to luck. Members of the in-group are, therefore, more likely to be promoted. This leads to the common pattern of organizations having homogeneous senior leadership despite efforts to hire more diverse employees.

The solution seems obvious: define a career ladder with clear performance standards for each step, and promote employees who exceed the performance expected at their step. Except *there aren't good, objective measures of job performance for software professionals.*

Hence a paradox: bias manifests due to subjective judgment, but it's not possible to remove the human from the equation. So how can promotion decisions be as equitable as possible? There's no simple answer, but here are some suggestions:

1. Promotion decisions should be made by a committee. The more diverse the committee, the more it resists in-group bias. In this case, "diverse" means including people at different career levels and roles (not just senior managers) as well as diverse demographics (gender, race, etc.). Structure the committee so that all members have equal say and feel safe giving their honest opinion. Failing that, vote by secret ballot.[10]

2. The committee should promote those who excel in the tasks *they are given.* Many organizations value some types of work (e.g., shipping new features; cross-team collaboration) more than others (e.g., crucial maintenance, unblocking team members) such that promotion is mostly determined by tasks assignment rather than performance. This leads employees to deprioritize their assigned tasks in favor of other work. More privileged employees and in-group members are more likely to engage in this kind of deviant behavior. If you're going to reward people for ignoring what you've asked them to do, you may as well forget promotions and just play musical chairs: when the music stops, everyone who has a seat gets a raise; everyone else, better luck next year!

3. The committee should review a list of common biases (as mentioned above) prior to the start of each promotion session.

4. The promotion process should be transparent. The promotion committee's deliberations should be observable, either live or recorded, to everyone in the company. Promotion statistics should be shared throughout the organization.

5. Promotion cases and reasons for decisions should be written down and provided to applicants *in writing.* Writing things down hinders murky, biased thinking.

6. Employee pay should be fully transparent. Ensuring everyone is being paid equitably for each career level is much easier when everyone can see what everyone else is getting paid.[11] Organizations should have an anomalies fund to enable imbalance corrections.

7. The organization should hold the promotions committee responsible for following its rules fairly. Employees should be able to appeal a promotion decision to someone outside their direct line of reporting, or (for small organizations) directly to the CEO. The appeals process must be genuine: if no appeal is ever upheld, the appeals process is fake. Promotion committees (or individuals) who display a pattern of unreasonable or biased decision-making should be replaced.

[10] Voting by secret ballot protects committee members, but those seeking promotion won't know who opposed their application.

[11] Management resistance to pay transparency has nothing to do with claims that it increases animosity within teams and everything to do with how it empowers workers to negotiate fairer pay.

Cognitive Biases Affect Promotion

Many biases that can affect promotions, including the following

1. Affinity bias—favoring people with similar backgrounds.
2. Recency bias—over-weighting recent accomplishments and experiences.
3. Halo and Horns bias—over-attending to first impressions, so an early stellar accomplishment can raise an overall mediocre performance, while a single dramatic failure may taint evaluations of an otherwise reliable performer.
4. Distance bias—over-weighting accomplishments from people who work near us compared to people who work in other offices or remotely.
5. Conformity bias—changing our beliefs to fit in with those around us.

As with other unconscious biases, redesigning the person-task system to prevent the bias is the most effective way of mitigating it. However, if you can't figure out how to redesign the system, reading the definitions of some common biases aloud at the beginning of a promotions committee meeting is better than nothing.

8. Promotion should be based on all of an employee's work at their current level, not just whatever they've done most recently. Employees should be encouraged to maintain an ongoing self-reflection document, including their accomplishments, professional development, and attempts at self-improvement. This shouldn't be onerous: a few minutes every few days should be sufficient. A good time would be immediately after the retrospective meeting. This self-reflection document will then be a strong resource for informing promotion decisions. Employees should be able to ask others for support compiling their promotion package.
9. Large organizations with multiple promotion committees should run an annual calibration process where each committee reads other committees' application packages and decisions and unifies job levels and pay organization-wide.
10. Promotion committees should not, under any circumstances, use AI for promotion decisions. AI systems simply conceal rather than prevent bias.
11. The promotion system should not assume everyone has a supportive manager (see next). In addition to the applicant's manager's appraisal, the promotions process should take into account feedback from applicant's peers and subordinates.

In many tech companies, employees are "put up for promotion" by their manager. The problem with this is that some employees and their managers just don't get along. Sometimes it's because the manager is a jerk or the employee is incompetent, but other times

they're just incompatible. From the individual employee's perspective, if a manager's feedback seems unfairly negative, the employee should document their achievements and their objections to the manager's points in more detail.

In other organizations, the individual employee applies for promotion. The problem with this model is that in-group employees tend to be more aggressive in seeking promotions; out-group employees tend to wait longer.

The solution is threefold: (1) every year, all employees should submit a performance review and be considered for promotion; (2) promotion decisions should be based on the employees' self-appraisal as well as their manager(s) appraisal of them; (3) promotion committees should investigate significant discrepancies between manager appraisals and self-appraisals, and not automatically side with the manager. Promotion must not be based solely on the manager's evaluation.

If the manager is supportive, the manager and employee get along, and the manager has sufficient technical expertise in the employee's area, they can co-edit the employee's self-reflection document, and the performance appraisal can be based on this single, unified account.

Give Better Feedback

> *Every portrait that is painted with feeling is a portrait of the artist, not of the sitter.*
> – *Oscar Wilde*

Some feedback is *intentionally* destructive, such as when a supervisor encourages a subordinate to quit, a subordinate sabotages a manager's career, or a developer writes a nasty code review to get back at a rival. As explained earlier in this chapter, employees who continue to express conscious biases have no place in a just organization.

One extreme form of consciously biased feedback is gaslighting. The term *gaslighting* was popularized by *Gaslight*, the 1944 movie adaptation of the 1938 play *Gas Light*, in which the protagonist is manipulated into believing she is losing her sanity. Gaslighting thus refers to "psychological manipulation of a person, usually over an extended period of time, that causes the victim to question the validity of their own thoughts, perception of reality, or memories and typically leads to confusion, loss of confidence and self-esteem, uncertainty of one's emotional or mental stability, and a dependency on the perpetrator" [179].

For example, one woman we know was offended that her co-worker said "oh, just have your boyfriend help you" (thus assuming that she is heterosexual, in a dating relationship, and needs a man's help). When her managers minimized the experience—"oh, that's not so bad," "boys will be boys," etc.—that was gaslighting. Examples are easy to find on social media where people don't believe someone's experience, minimize it, or argue that their perception is compromised.

Gaslighting is closely related to *victim blaming*; that is, when the victim of wrongdoing, rather than the perpetrator, is held responsible. In a software development context, victim blaming could include blaming an individual software developer for failing to meet impossible goals or expecting a technical support rep to maintain perfect professionalism while being screamed at by an irate customer.

Well-intentioned feedback can also be biased and harmful as in the following examples.

- Urging a black person to soften their tone in meetings reinforces anger stereotypes.
- Discouraging a woman from ending statements with a question (as it makes her seem unconfident) ignores the fact that many women have found that seeming too assertive will hurt their careers.
- Telling an LGBTQ2S+ person, 'conceal your gender identity because it will help you advance' prioritizes the employers' bigotry over the employee's human rights.
- Encouraging a person to conceal a disability to appear more competent implies that their disability makes them incompetent.
- Suggesting that an indigenous employee stop wearing symbols of their indigenous identity (e.g., jewelry) to appear more professional reinforces the prejudice that indigenous people are somehow less professional or desirable in the workforce.

Dealing with unconsciously biased or hurtful feedback is difficult precisely because the giver doesn't perceive the harm and may be genuinely trying to help. When it's something specific, like a manager always telling their female subordinates to smile, the perpetrator can simply be informed of the problem. If they stop the behavior, the problem is solved. If not, it might be conscious bias after all and we're back to "isolate-educate-terminate." However, there are infinitely many ways of giving biased feedback, and no one can memorize all of the things that someone else finds hurtful. So we need more general advice such as the following.

- When receiving feedback that you feel is biased or otherwise problematic, have the courage to say why. If you don't correct problematic behavior, it may continue.
- Avoid tone-policing. Don't expect people who feel abused to be polite about it. Focus on the substance of complaints, not the way in which they're delivered. This is difficult because you need enough self-control to avoid defensiveness.
- When you witness people giving problematic feedback, speak up.
- When giving feedback, consider: (1) if your feedback could reinforce stereotypes; and (2) whether your feedback is phrased optimally to help the recipient. For example, sometimes people need to be told their work isn't good enough, but you don't have to be a jerk about it.
- Foster psychological safety within teams, especially with managers, so everyone feels secure enough to discuss problematic feedback (see sidebar on page 200).
- Encourage everyone in the organization to practice *reflexivity*; that is, reflecting on our own biases. Knowing oneself deeply helps us avoid problematic behaviors.

Share Power

Organizations blame injustice on individuals to avoid reforming their systems. To improve justice, dismantle the organizational structures that enable biased outcomes.

When a small group of people are permitted to hoard power, make decisions for others, act in secret, suppress dissent, and use junk data dressed up as "objective metrics" to obscure injustice, organizations cannot become just. When meaningful action is replaced with theatrics—mindfulness training, well-being apps, sensitivity workshops, wellness committees, adding a token minority on an advisory board, etc.—organizations will remain unjust [180].

"Inclusion" doesn't mean "included in the organization", it means "included in determining how things are done." The more power is decentralized and distributed to individual employees, the less individual prejudice can manifest as injustice.

Evolving your Organization toward Justice

To grow the organization toward justice, we must dismantle and reinvent the structures that enable injustice in the first place.

One woman we interviewed told us how she surfaced discriminatory comments to her manager. The manager intervened, and while the office culture initially improved, it quickly devolved back to old habits. She was frustrated. Her initial report was stressful and carried real risk to her career. Management thought they had "solved the problem." Injustice is not a singular problem that can be solved. Justice is an ongoing effort to improve our situations.

Resist any kind of one-and-done thinking. When you are the person receiving reports and intervening, set a reminder to follow up in a few weeks (or months) to see if initial improvements endure.

What Could Future Organizations Be Like?

If the reader will humor a little naked idealism, we'd like to explore what future organizations could be like if organizational justice was genuinely taken seriously.

Imagine working not for a publicly or privately owned company but for an organization that is effectively owned by its employees. Imagine if all your c-suite and director-level positions were elected by employees as well as investors, such that executives must balance employee and investor interests to maintain their positions. Imagine being absolved of the endless quest for unsustainable growth because the company is not driven by greed.

Imagine an organization where management and leadership were just different kinds of work, no more or less important than product design and development. Imagine if your pay was actually based on how well you do your job, not on which job you are assigned.

Imagine that you didn't have to worry about:

- getting laid off because the company wasn't allowed to lay off employees unless it was facing bankruptcy,
- being punished for taking parental leave because parental leave was *mandatory*,

- being punished for refusing overtime because overtime was *banned*, and
- being discriminated against because no one had the power to discriminate against you.

Imagine an organization structured around actively resisting prejudice instead of faking the appearance of social responsibility. Imagine selecting new hires based on their qualifications and experience without knowing anything about their demographics.

While this all may seem far-fetched, several organizations are making progress in this general direction—see Laloux and Appert [181] for further advice.

Many of these visions of a better future are based on economic democracy, a socioeconomic philosophy including the idea of broader participation in economic policy [182]. Economic democracy is not exactly capitalism, socialism, or communism. It is a roadmap for how economic activity could be organized to make the world better, and if you care about the Earth being a place where humans can still thrive in a century or two, economic democracy makes a lot more sense than how we're running things right now.

Summary

Most people think about organizational justice as a series of steps: first, you increase perspectives, then improve belonging, then eventually tailor support, then (one day maybe but probably never) share power. As usual, reality is less linear.

Increasing perspectives without tailored support is just public relations. Organizations can neither benefit from, nor retain, a diverse workforce without inclusion and tailored support. Furthermore, recruiting more diverse employees won't make an organization more just if that organization is saturated with structures (power hoarding, paternalism, opacity, inclusion theater, etc.) that reinforce and propagate injustice.

The main message of this chapter is that, while individuals can be problematic, focus on problematic structures. Focus on power: who has it, how do they use it, how do they abuse it, and how can it be better shared.

Changing the tech industry is a *generational* project. Maybe you think there's nothing you can do. Maybe you're not in a position of power. Maybe you are not a member of any particular minority group and don't see this as your fight.

But here's the thing: human history is not full of people in power having epiphanies and spontaneously giving away their power. Human history is full of people who are being abused, *and their allies*, demanding change. Human history is full of people forcing the changes they want to see in the world.

Expecting the most victimized people to peacefully and politely change the structures that are oppressing them, without any help, is totally unrealistic. Change depends on action from people who already have power. For most North American and European software companies, that's middle-aged white men. That's why the longest and most detailed chapter in this book—written by a couple of middle-aged white men—is about organizational justice.

Focus on what you *can* do. When opportunities arise, be supportive, call out injustice, and ask for change. Even if it doesn't seem to be working, just keep at it. Every time you bring up these issues, every time you voice support for a more just world, you shift

perceptions of what is possible and necessary a teeny-tiny bit. If all you can do is tell someone who's having a bad day that you appreciate them and that, yes, they do belong here, you're still helping.

One of (the German philosopher) Hegel's central ideas is that we are all trapped in the zeitgeist of our time. Ideas around organizational justice are changing so fast that by the time you read this, some of our text may be cringey—we'll appear to have used the wrong words and failed to engage with key issues of our future / your present. But we have to try. So we did our best.

Remember: never mistake an imperfect ally for a deadly enemy.