



Unblocking Workflows

The Guide to Developer
Productivity in 2022



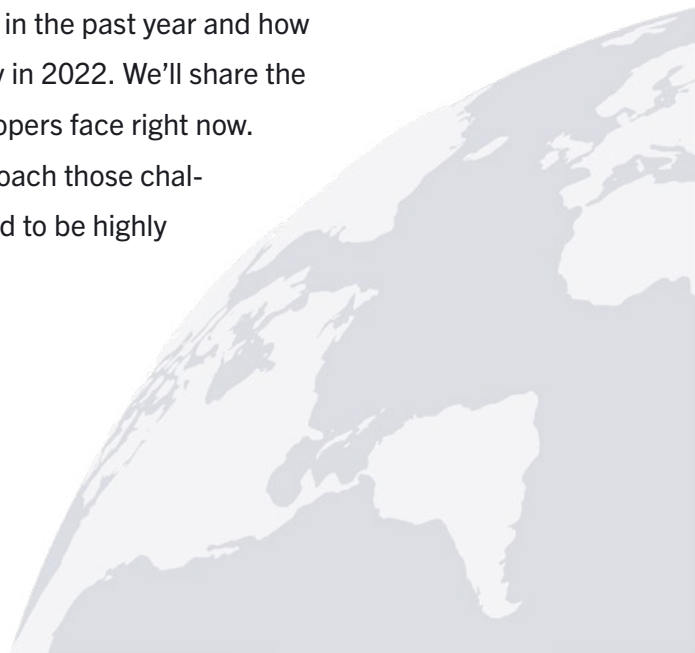
The State of Developer Productivity

The landscape of software development has changed dramatically in the past year. As the world adjusts to new models for work and new standards for security, development teams face more obstacles to agile development than in previous years. Teams must contend with increased demands to produce software faster, more securely, and with fewer resources than ever before.

Shipping high-quality software quickly requires alignment across all phases of the development lifecycle, as well as visibility and control across the teams and tools that power digital workflows. Yet the tools and processes that many teams rely on often fall short of their promises, leaving teams less productive than they need to be. Investing the time to understand and optimize these areas allows R&D teams to be as innovative and agile as the market demands.

In This Guide

In this guide, we'll look at how the industry has changed in the past year and how these changes will impact developer teams' productivity in 2022. We'll share the results of a survey that focused on the challenges developers face right now. Then, we'll dive deeper into how organizations can approach those challenges to ensure that their developer teams are equipped to be highly productive, efficient, and innovative.



What Do We Mean by ‘Developer Productivity’?

The term *developer productivity* comes with some baggage. Some people feel that it’s used as a way to wring every last bit of work out of a team, single out underperformers, or find a way to shoehorn all of the work on which developers spend their time into a single, achievable output.

Productivity is a complex concept, and it doesn’t easily boil down to a single, concise metric. That hasn’t stopped organizations from trying, but those efforts often leave teams with one-dimensional metrics — lines of code written, features shipped, bugs squashed — that don’t adequately reflect the work that development teams are doing. Other metrics like employee Net Promoter Score, product uptime, and customer retention paint a fuller picture of productivity, but can vary wildly from team to team.

Because metrics can be so varied, we’ve decided not to focus on specific ones in this guide. Instead, we’ll be looking at the practices and processes that often feed into those numbers and build the foundation of the real goal behind measuring developer productivity:



THE GOAL

Keeping your development team engaged, innovative, and building great products that they’re proud of.



Table of Contents

I. Software Has Eaten the World: Every Company Is Now a Software Company

A. How Are Development Teams Changing in 2022?

1. Remote Work Is Now the Norm
2. All Industries Are Becoming High-Security Industries
3. Explosive Demand for New Apps Drives Faster Development
4. Retaining Talent Is Essential but Challenging

II. Developer Productivity Survey Results

A. Key Findings

III. How Highly Effective Teams Approach Productivity Challenges

- A. Fragmented Tools
- B. Distributed Team Collaboration Issues
- C. Finding and Retaining Developer Talent
- D. Manual Task Overload
- E. Fragile Workflows

IV. Key Takeaway: To Stay Competitive, Teams Must Invest in Enabling Productive Team Collaboration

V. About Mattermost

Software Has Eaten the World: Every Company Is Now a Software Company

Perhaps the most critical, seismic shift to the global economy is the fulfillment of Marc Andreessen's assertion — which almost feels cliché after a decade — that “[software is eating the world](#).” All companies are now software companies, and developers are now mission-critical to companies of all sizes and industries. Developer tools designed for a niche audience now command serious attention. As Vijay Gurbaxani [wrote about GitLab's \\$11 billion IPO](#) in October 2021:

“The success of GitLab's IPO is evidence that investors are betting on the trend towards building software as a core competence in companies worldwide. It's also another timely reminder that code is a critical asset. To build valuable software better than rivals, **CIOs must accelerate their investments in in-house software capabilities—talent, tools and development practices—while tapping GitLab or other productivity offerings that can supercharge their armies of coders.**”

Simply put, companies now compete on their ability to build and operate software. CIOs must invest in software capabilities to stay competitive — and they are. Gartner predicts that worldwide IT spending will [exceed \\$4.5 trillion in 2022](#), and software engineering roles represent one of the fastest-growing job categories in the world.

What does this mean for developer teams in 2022? Let's look at some of the key trends that will have a massive impact on how developers move forward.

How Are Development Teams Changing in 2022?

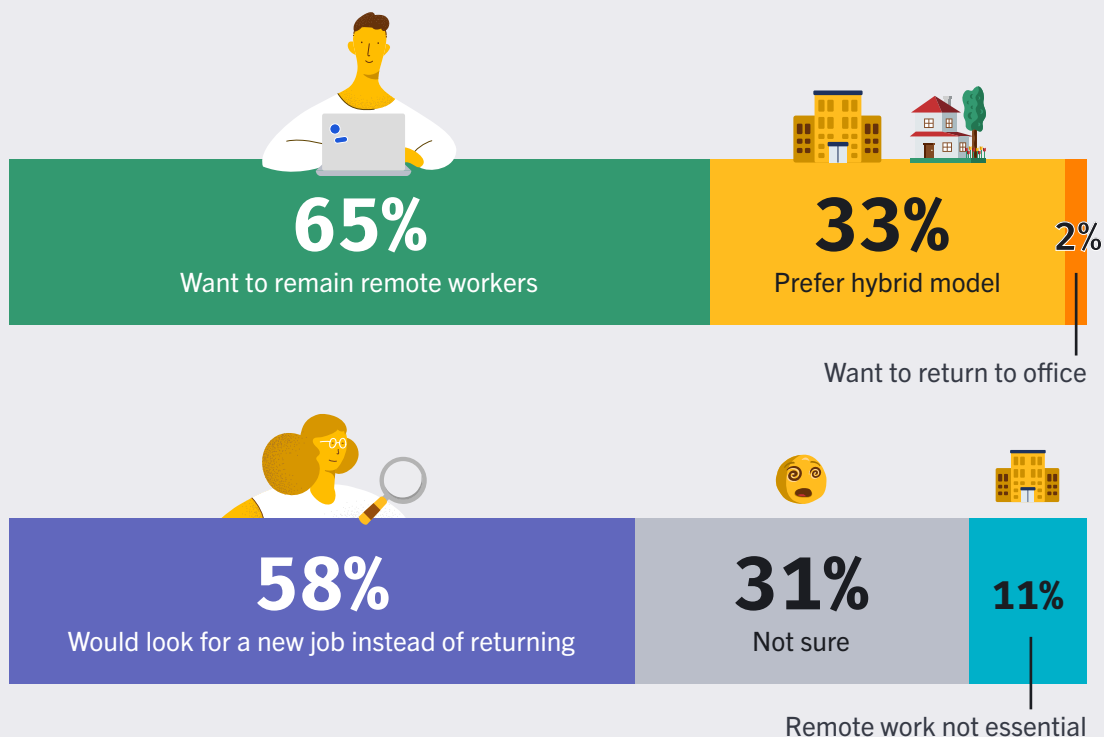
Remote Work Is Now the Norm

In 2021, organizations stopped reeling from the uncertainty of the pandemic work patterns of 2020 and started thinking seriously about how and where their teams would be working in the future. For many teams, that meant settling into distributed or partially remote models that were intentionally designed — not those that emerged as a pandemic stop-gap.

Even companies that were previously undecided about allowing remote work are now forced to consider it as a long-term workforce strategy; **only 2% of workers** who engaged in remote work during the pandemic wanted to return to the office instead of staying remote or moving to a hybrid model. Additionally, more than half of them said that they would look for a new job instead of returning to the office.

Workers Want to Stay Put in Home Office

Survey responses among workers who engaged in remote work during the pandemic (2021)



Companies are now taking a strategic approach to investing in the digital workflow and security technologies they rushed to production to support their remote teams during the pandemic. Developers now expect their employers to be able to support both long-term remote work and more flexible in-office time. As such, organizations must have a game plan for keeping their workforces focused, connected, and content — whether they're returning to the office, committing to fully remote models, or finding themselves somewhere in between.

How Remote Work Impacts Productivity:

In our survey of developers and dev team managers, more than half of respondents said that remote work has a positive impact on their productivity, giving them more flexibility in their workday and more ability to focus. But while remote work can be a boon when it's time for heads-down solo work, it can also create friction for collaborative work. Survey respondents also stated that collaboration and sharing information were key challenges that kept teams from working together as effectively as possible.

All Industries Are Becoming High-Security Industries

In November 2021, the Biden administration announced that civilian federal agencies will be required to fix hundreds of cybersecurity flaws and earmarked [\\$1 billion in state, local, tribal, and territorial cyberdefense grants](#).

In previous years, a heavy emphasis on security practices might have remained the domain of government organizations and high-security industries like finance and healthcare. But [high-profile cybersecurity incidents in 2021](#) led to increasing awareness of IT security vulnerabilities, especially in their reliance on third-party tools; [one study found](#) that supply chain attacks are up 650% in 2021. Developers



have become much more conscious about the code they import into their software relative to the code they write themselves.

In response to these trends, software teams across all industries must make security a top priority. Many teams integrate security into a DevSecOps practice to further streamline the development lifecycle. DevSecOps adoption is set to increase in 2022, as security becomes an increasingly integral part of DevOps processes and developer team cultures. Demand for cybersecurity experts is exploding, and it's predicted to become [one of the fastest-growing roles](#) in the next few years.

How Security Impacts Productivity:

It's no surprise that both preventing and responding to security incidents are huge resource sinks for engineering organizations. Faster-moving release cycles and increasingly distributed teams only compound security concerns.

Resolving security vulnerabilities takes time away from development teams, reducing their ability to focus on shipping new, innovative products and features. Even worse, if incident response workflows aren't well-executed, teams can lose essential time and compound issues when responding to security breaches.



“One of the biggest challenges of remote workforce security is that you just don’t have that same secure perimeter that you used to have. When you’re in a controlled office space, you know exactly what’s going on. You’ve got locking doors, you’ve got cameras, you’ve got access control methods and auditing. You just don’t have that kind of infrastructure in work from home environments.”
—IT Director, Nuclear Technology Firm, [Tackling remote workforce security challenges post-pandemic](#)

Explosive Demand for New Apps Drives Faster Development

According to [GitLab DevSecOps Survey 2021](#), 60% of developers are releasing code twice as fast as before, thanks to DevOps — up 25% from before the pandemic began. As more and more organizations undertake digital transformations in 2022, they'll drive even more demand for software applications to support their digital operations.



[Experts predict](#) that over 500 million digital apps and services will be developed and deployed using cloud-native approaches by 2023 — matching the number of apps that have been built during the last 40 years. These applications will not only have to be delivered quickly but will also have to meet the ever-increasing bar for quality and security; according to [a recent microsurvey by CNCF](#), 85% of respondents indicated that modernizing security is very important to their organization's cloud-native deployment.

How Software Demand Impacts Productivity:

This explosive demand puts more pressure on developers to take on more responsibility and produce more of their best work faster. To rise to the occasion, teams need to have the infrastructure and processes required to enable faster delivery without compromising on quality. It's no surprise that the adoption of DevOps best practices will continue to grow in 2022 to support this situation. However, if DevOps workflows aren't well-implemented within the organization, productivity will suffer due to excessive complexity and siloed knowledge.

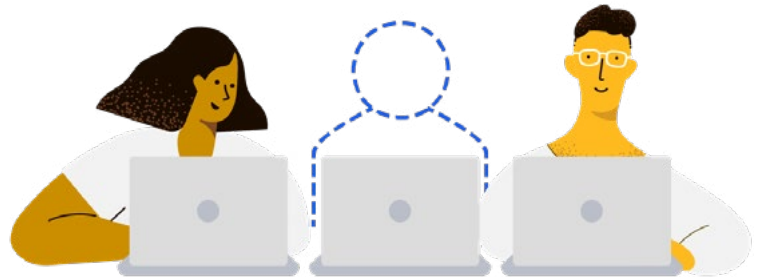
Unfortunately, these efforts to do more work faster are backfiring. Development teams running at full capacity continuously have given rise to increased burnout, especially for remote teams with fuzzier boundaries between working and being offline. One survey found

that [80% of developers are feeling burned out](#) in 2021, with nearly half of those citing heavy workloads as the primary reason.

Retaining Talent Will Become a Greater—and More Important — Challenge

Demand for good development talent is [higher than ever](#); despite fluctuations in the job market in the past couple of years, the unemployment rate for developers is virtually zero, with projected workforce growth of 75% in the next decade. More remote hiring means that developers have more opportunities available than ever before. For employers, this means finding and holding onto talented developers will be exceptionally challenging.

Developers will be less willing to endure on-the-job frustrations, from outdated tooling that slows them down to mountains of technical debt that keeps them from working on the challenges that they're truly interested in. By investing in tools that eliminate manual tasks and mundane work, you can empower developers to focus on high-value work.



How Hiring Impacts Productivity:

Every HR lead will tell you that hiring and onboarding eat up a huge amount of time, money, and other team resources. Dealing with missing critical roles on your team, navigating the onboarding process for new hires, and coping with senior developers leaving can all have a negative impact on your team's productivity. [Experts say](#) that it often takes one or two years for a new employee to reach the productivity of an existing employee.

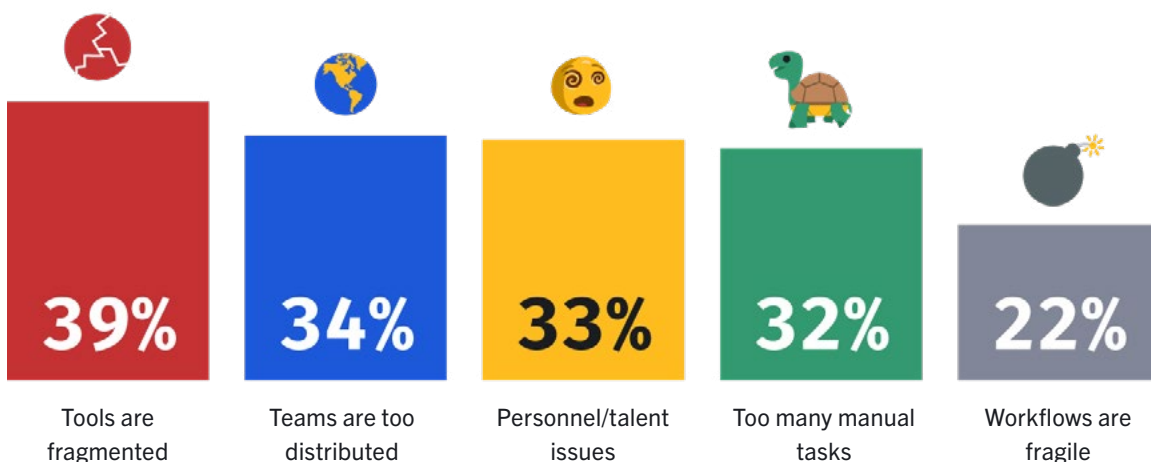
By contrast, retaining members of your team drives faster, more creative problem-solving over time. A team that has worked together for a longer time learns to communicate better and solve problems faster. Building a happy team is an investment in the long-term productivity of your organization.

Developer Productivity Survey Results

To get a better idea of where teams are facing the biggest productivity challenges in 2022, we surveyed over 300 developers and development team managers to learn more about their productivity habits and challenges. The results reflect larger trends across the industry: to be more productive in coming years, teams must focus on improving collaboration and streamlining their workflows.

Key Challenges for Developer Teams in 2021

Every team faced productivity challenges in 2021, but dealing with fragmented toolsets was the biggest blocker to moving as quickly and effectively as they wanted to.



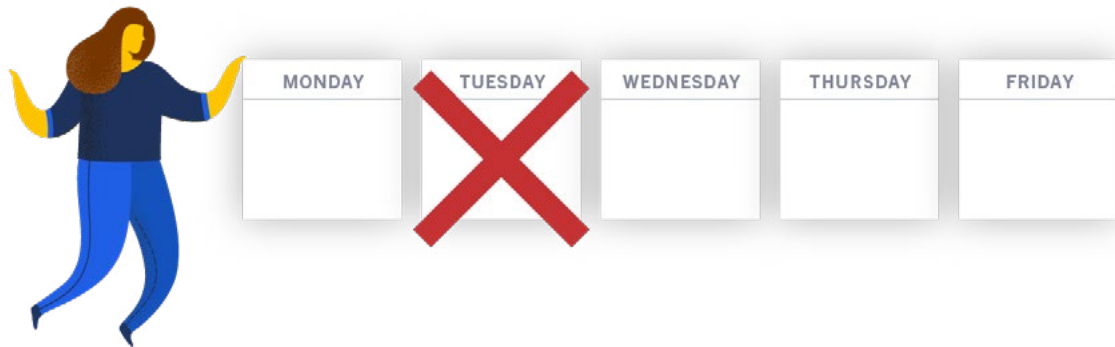
What else did developers struggle with during 2021? Check out some of the most common write-in answers on the survey:

- ✓ Task Planning Prioritization & Time Management
- ✓ Automation
- ✓ QA Testing
- ✓ Better Developer Resources and Tools



Developers Lose a Full Work Day on Productivity Challenges Each Week

Context switching really does kill productivity — 71% of developers waste up to eight hours a week on productivity challenges.



Remote Work Has Big Benefits for Dev Teams

While working with distributed teams is a challenge, overall developers love working from home — over 50% said that their productivity improved as a result of working remotely.

QA Testing and Automation Thrive in Remote Environments

Teams that focused on QA testing and automation were 21.7% more likely to indicate remote collaboration has improved during the pandemic than all developers who responded. In fact, none of these developers indicated that collaboration got worse during the pandemic.

“I’d like a push for automated QA testing to completely replace manual testing. Even if it takes longer to write up front, I think it saves time in the long run.”

—Software Engineer, Mattermost Developer Productivity Survey 2022



Improved Tooling Will Help Unlock More Effective Teams

We asked developers **what one thing would improve software development productivity in 2022**, and many said that tooling improvements are key.

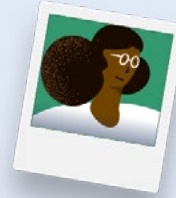
“Give devs the tools to define and create their own infrastructure.”

Infrastructure Engineer



“Add some lightweight ways to share state/purpose among the team.”

Principal Software Engineer



“Reduce the use of disruptive messaging tools.”

Professor of Software Management Practice



“Build common tooling practices across the team.”

Senior Data Scientist



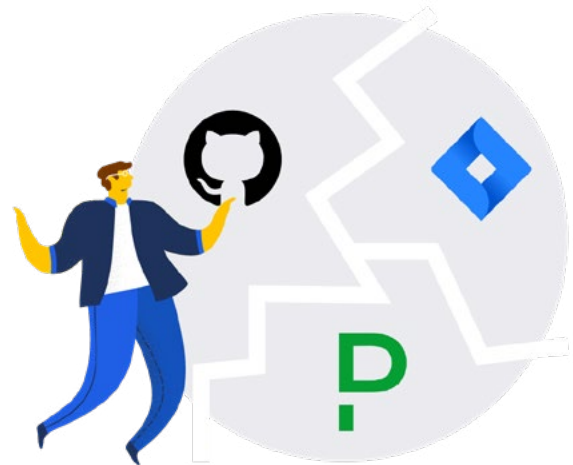
In the next section, we'll dig deeper into the most common productivity roadblocks that developer teams face — and how to bust through them in 2022.

How Highly Effective Teams Approach Common Developer Productivity Challenges

If you found yourself sweating at the mere description of any of those challenges, you're not alone. No matter how effective they are, every team has areas of their workflow that could be improved. So, how can your team tackle the challenges you'll face in the coming year? Let's take a closer look at how high-performing teams approach busting through productivity blockers — and what experts say teams can do to become more productive.

Fragmented Tools

What's the problem? Every development team has its own unique blend of tools that help them ship code. But it's hard to keep all those tools working together seamlessly. Contending with fragmented tools was the most common challenge cited in our survey of developer teams, with 39% of respondents calling it a major challenge.



One of the biggest enemies of productivity is context switching. Context switching between tools creates workflow friction that slows down work and introduces opportunities for essential information, tasks, and mission-critical communications to slip through the cracks. The processes and tools that enable people to eliminate context switching, focus more deeply, and work more effectively will be an essential investment for highly productive developer teams in the next year.

Solution: Build Connections with More Robust Integrations

Developer teams are always going to need a slew of tools to help them build, ship, monitor, and manage software. The key to keeping those tools working seamlessly is to focus on bringing them together and integrating them as much as possible. Building this connective tissue between the critical tools in your workflow helps ensure you spend less time managing tools and more time on task.

Look to ChatOps for Inspiration

Adopting [ChatOps](#) practices can be especially helpful for teams struggling with tool fragmentation. Using slash commands, webhooks, APIs, and other integration methods, a mature ChatOps practice brings all your comments, notifications, and metrics into one place. ChatOps creates a digital command center for your team, providing a single source of truth to reference and making it easier to pull information from (and push information to) other tools in your workflow without context switching.

Leverage Open Source to Build Bespoke Integrations

One of the key challenges that teams face when trying to build a more interconnected toolset is weak integrations. Out-of-the-box integrations might promise seamless workflows but end up lacking functionality that's essential for your team — so you end up having to go back and forth between differ-

“Software is truly eating the world, we can even say that software is the new currency. We are increasingly seeing the impact of this incredible demand on the developers who are responsible for this growth. Developer productivity is a real challenge across industries, especially with an increasing skills gap and explosion of tools. We expect organizations will look to manage this growth by unifying tools and other processes to improve productivity and results.”

—Paul Nashawaty, Senior Analyst, ESG

ent tools anyway. To truly get away from tool fragmentation, you'll have to go beyond basic integrations. Many teams are now turning to open source software to ensure that they can extend and customize tools to fit their workflows more precisely. Open source technologies allow you to build out more robust integrations with the functionality that your team needs.

Team Distribution Collaboration Issues

What's the problem? Teams are more likely than ever to be fully or partially distributed. While over half of teams say that remote collaboration has gotten better in the last year, many still agree that team distribution is a significant blocker to developer productivity.

According to a survey from Atlassian on collaboration, [a whopping 82% of teams struggle](#) with cross-functional collaboration. As the lines between development, operations, security, and support teams become less clear, ensuring that everyone has access to the information they need when they need it is key.



Solution: Improve Communication Culture and Tools to Optimize Collaboration

While the past two years have been a crash course in remote work for many IT workers, truly effective remote work is a learned skill, and not everyone thrives in a remote environment without additional support. Organizational behaviors that worked fine in office environments often fall flat in remote environments, where in-person interactions can't fill gaps in things like process documentation.

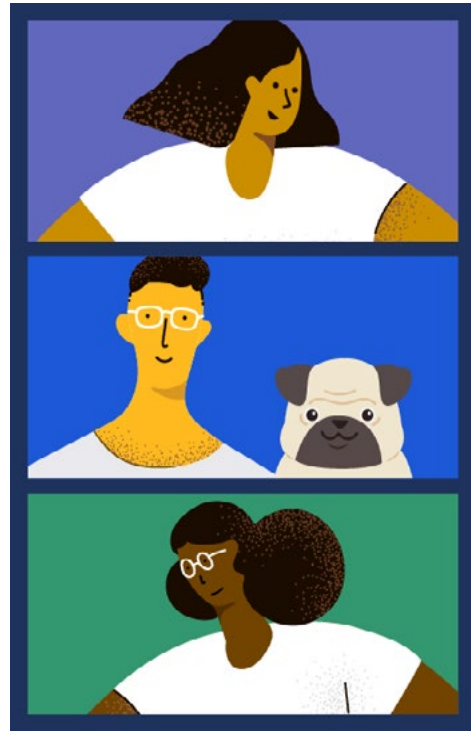
Work on Improving Async Collaboration Practices

It may seem like the best way to improve communication is to put more meetings on the books. But developer teams in particular also cite too many meetings as a major produc-

tivity blocker, and more time on Zoom calls won't solve the real root of your communication woes. In our survey, 62% of developers who said they wanted to focus on improving resources like documentation and training indicated that tool fragmentation is also a significant problem.

Instead of adding more meetings to the calendar, work on shifting team culture towards collaboration — especially asynchronous collaboration, with a heavier emphasis on documentation and the automation of manual processes.

Establishing best practices for everything from code reviews to task descriptions and taking the time to coach team members will go a long way towards enabling everyone for success. Focus on keeping team knowledge in an easily accessible, central location so that team members can answer their own questions or know who to ask. For repeatable processes, create resources such as templates, shared checklists, and automated workflows in project management tools. When everyone is starting from the same place, with the same information, it is significantly easier to move forward together.



Look for Signal Boosters, Not Noise-Makers

You probably have more than a few collaboration tools in your stack — but are they actually enabling collaboration? The right tools can break through information silos and help teams function at their best, whether they're in the same city or spread across time zones. Unfortunately, many teams use collaboration tools that don't serve their needs and leave gaps in their workflows, increasing the amount of context switching they need to do every day.

If your team is struggling both with communication and tool fragmentation, it's time for an audit of the tools your team uses for collaboration and how they're using them. Defining a

single source of truth where your team can find essential documents, stay up-to-date on in-flight projects, and connect with each other is a critical step to enabling your team to regain productive work time.

“Every organization – and its software development teams – is under immense pressure to deliver software faster, better, and more securely. And they must often do it with ever fewer resources, even as organizations are increasingly competing on their ability to deliver software-enabled solutions to the market. All of that leads to a need to improve developer productivity. But more than any simple, one-dimensional metric, what that really means is that they need to nurture a development team that is engaged, excited by innovation, and driven to build products that delight customers.”

–Charlie Araugolo, Analyst, Intellyx

Finding and Retaining Developer Talent

What's the problem? Developers who took our survey agree with the macro trends; one-third said that developer talent concerns impacted their productivity in 2021. But hiring and retaining a talented, high-functioning developer team requires more than just a high salary. To attract and retain top developer talent, organizations must provide a number of cultural, organizational, and technological improvements to enable the brightest minds to do their best work.



Solution: Invest in Creating a Developer-Friendly Environment

In the [How to Attract, Develop and Retain Great Software Engineering Talent Report \(November 11, 2021\)](#), Gartner analysts advise that teams invest in fostering environments for their teams that reduce friction for software development workflows:

“Optimize flow by creating an environment with minimal distractions and impediments so people are in flow more often. Sponsor agile ways of working and break down siloed thinking. Establish continuous integration and continuous delivery practices and tools to minimize delays. People will have more time to spend on value-add work and not on wrestling with the tools, the infrastructure, excessive bureaucracy and outdated management practices.”

Lean into the Technologies That Developers Want to Work With

Developers love to be challenged. Solving the same problems over and over again using the same technology almost begs them to start searching for a new role. Demand for faster innovation has driven teams to embrace new technologies and incorporate them into their tech stack faster than ever. [Developers are favoring organizations](#) that use languages they actually want to work with, such as JavaScript or Rust, or that show reasonable modernity using cloud services on AWS, GCP, or Digital Ocean.

Evaluate whether there are new technologies that would not only solve technical problems you might face but also make the work more engaging and interesting for your team. As reported in Forrester’s report for Atlassian on the value of [Open Source Collaboration](#), having “the right collaboration tools and technology is a floor raiser that provides the foundation for people-driven ceiling raisers like culture and processes.”

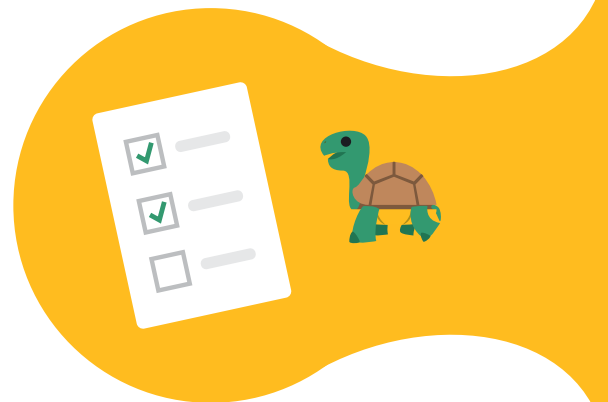
Become a Part of the Open Source Community

Another key way to become a more appealing organization for developers is to get involved in the open source ecosystem. High-performing developers are often contributors to cutting-edge open source tools, so supporting the open source community and encouraging your team to contribute to open source projects is an easy way to keep talent engaged.

To this end, many organizations are participating in the [Open Source Friday initiative](#), carving out dedicated time for their developers to contribute to open source projects that they use, are interested in, or simply want to support. Showing both prospects and your current employees that you care about the projects they care about can go a long way toward attracting and retaining talent.

Manual Task Overload

What's the problem? In our survey, 34% of developers said that their teams have too many manual tasks to manage. A study by Forrester found that [only one-third of IT tasks are currently automated](#), leaving teams overwhelmed by repetitive manual tasks. Unfortunately, grinding away at manual work can prevent developers from solving larger problems and prevent your organization from being innovative.



Solution: Automate Strategically

Automation is the clear answer to the problem of repetitive manual tasks, but there's a reason that more developer teams haven't already automated their pain away. Automation — especially on a large scale — is often challenging to implement and maintain.

Start Small and Work Your Way Up

You don't have to automate your entire workflow to reap the benefits of automation. Challenging your team to find small tasks to automate in 2022 can have a compounding effect

on your overall productivity. Even fairly “trivial” tasks like automatically creating a Jira ticket when a customer reports an issue can make a huge difference in keeping your team focused on what matters most.

Invest in Technologies That Enable Automation

Infrastructure management tools like Kubernetes help automate deployment, scaling, and management processes and streamline workflows for your team. While changes to infrastructure and tooling are certainly a bigger lift than simpler automation, they’ll pay off in the long run. Remember that automation is a complex process and represents an ongoing investment rather than an end-state of “everything is automated.”

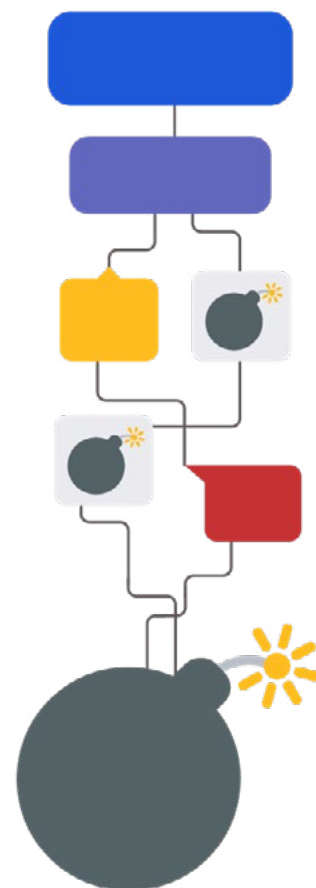
Fragile Workflows

What’s the problem? Nearly a quarter of teams we surveyed said that fragile workflows are holding their teams back from being productive. Survey respondents who indicated they have problems with workflows were 23.5% more likely to indicate they are facing rising expectations due to the pandemic compared to all developers (70% vs. 56.7%).

Solution: Create More Robust Workflows

Build Better Integrations

As with fragmented toolsets, weak integrations are a common culprit. Investing in building more robust tool integrations keeps your team from continually dealing with broken or out-of-date connections that keep them from staying in a state of flow.

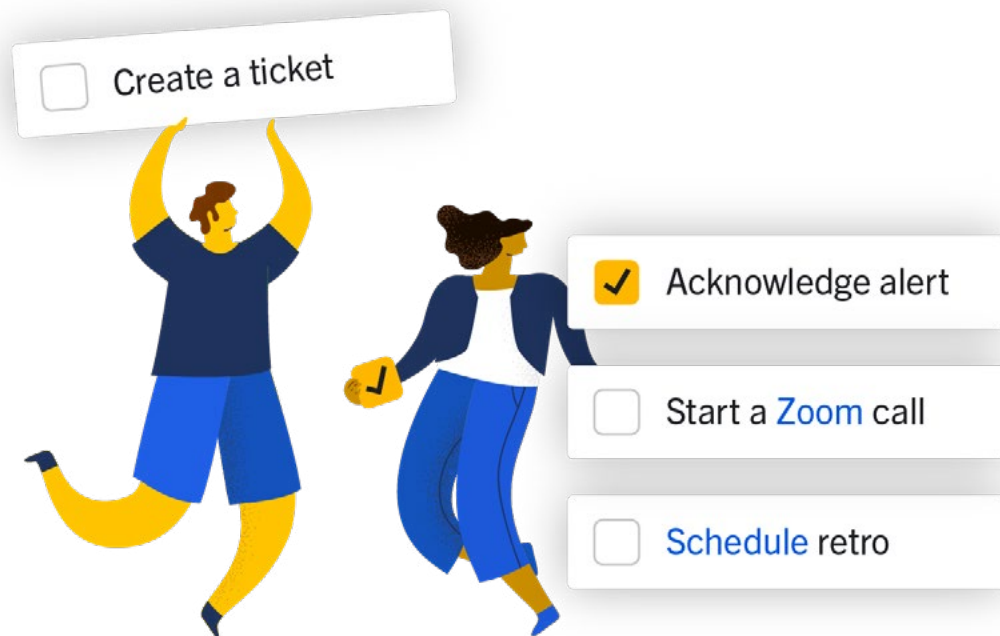


Document and Refine Repeated Processes

Poorly defined processes can lead to fragile workflows and also reduce your team's ability to work asynchronously. Developers can get stuck waiting on others to complete their tasks or lose time searching for the answer to "What's next?" Ensuring that repeated processes are clearly defined and documented, highly visible to all stakeholders, and easy to keep up-to-date when things change helps create more robust workflows for everyone.

Invest in Fixing Broken Systems

Finally, invest in fixing what's broken. Tools that are out of date are a poor fit for your team's needs. Technical debt can bring even the most productive teams to a screeching halt and push them into cycles of working and reworking on the same problems. Have a plan for working down technical debt, and invest in streamlining deployment processes that might be introducing friction into their daily work.



To Stay Competitive, Organizations Must Invest in Enabling Productive Team Collaboration

These trends all point to a common, overarching imperative for developer team leaders: collaboration. **From shipping faster and ensuring organizational security to retaining high-performing team members, organizations must empower their teams not just to perform well as individuals, but to collaborate more effectively.**

Open source is one essential approach to help software organizations resolve the productivity challenges that their teams face. Open source tools are created by engineers solving problems for themselves — and sharing those solutions with the world. The best engineers aspire to be active in open source contributions and communities and they want to work for companies that embrace these ideals.



Moreover, open source powers innovation. With full access to source code, the users of open source software have the freedom to create extensions, customizations, and integrations that can supercharge productivity, or contribute directly to core code. According to a recent RedHat survey, [90% of IT leaders are using enterprise open source software](#) within their organizations. This makes perfect sense:



Open source enables teams to build the perfect tool for the job.

Ultimately, productivity is about working better together. Whether your team's workflow needs a complete overhaul or just a few tweaks to refine processes, finding the right tools, following the right practices, and instilling a culture that is dedicated to fostering collaboration will keep your developers aligned, productive, and innovative in 2022 and beyond.

About Mattermost

Mattermost is an open source platform for secure collaboration across the entire software development lifecycle.

Hundreds of thousands of developers around the globe trust Mattermost to increase their productivity by bringing together team communication, task and project management, and workflow orchestration into a unified platform for agile software development.

Founded in 2016, Mattermost's open source platform powers over 800,000 workspaces worldwide with the support of upwards of 4,000 contributors from across the developer community. The company serves more than 800 customers — including Samsung, Nasdaq, SAP, the European Parliament, and the United States Air Force — and is backed by world-class investors including Redpoint, YC Continuity, Battery Ventures, and S28 Capital.

To learn more, visit www.mattermost.com.



Mattermost

[Mattermost.com](https://mattermost.com)



9