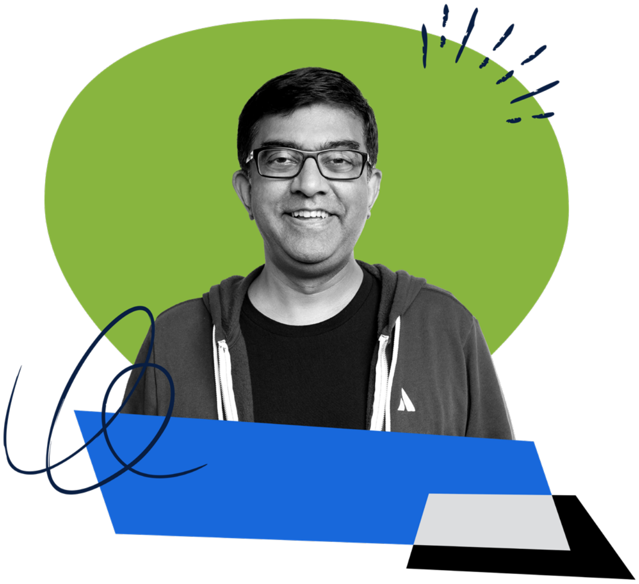
ATLASSIAN

The State of Developer Experience in 2025

Table of contents



RAJEEV RAJAN
Chief Technology Officer at Atlassian

A note from Atlassian's CTO

In a year of economic turbulence, rapid AI breakthroughs, and growing expectations to ship faster, developer experience (DevEx) continues to take center stage in leadership conversations. As systems grow increasingly complex, teams are weighed down by escalating technical debt, mental fatigue, fragmented tools, and clunky workflows. Generative AI is supercharging parts of software development, while raising new challenges around security, cost efficiency, and system reliability.

This pressure-cooked mix of innovation and strain demands a closer look at how AI is reshaping the developer experience – and what that means for the future of software development across the industry.

The State of Developer Experience (DevEx) Report is our annual opportunity to check in with developers and leaders from around the world to uncover these challenges and identify what's working and what's not. We kicked off this report with a goal to quantify this ubiquitous development practice that's so difficult to measure. We wanted to capture a more concrete picture of the developer experience of real teams across the industry. Each year, we look into how developers are connecting emotionally with their jobs, assess friction points, and identify emerging trends.

This year in particular, we sought to understand more about how development teams use AI. We revisited teams' feelings towards their own productivity, what's working well for them, and which roadblocks are continuing to challenge their workflows.

As we like to say here at Atlassian, countless parts of our industry have changed over the decades, but a basic truth has held steady: you can't have productivity without joy. This belief drives not only how we build our own products and support our engineers, but also how we strive to improve the developer experience for teams everywhere.

There's much to take away from this year's report. Hopefully, this compendium of insights can help with decision-making for your teams, too.

RAJEEV RAJAN
Chief Technology Officer at Atlassian

The State of DevEx results: a snapshot

AI adoption jumped from last year:

- **AI tools are producing more value:** 68% of developers reported a sizeable time saving of more than 10 hours a week from using AI. This is a significant jump from last year, where 54% of developers said they were yet to experience significant productivity benefits by using AI.

A widening gap between leaders and developers:

- **An increasing number of developers are reporting that leaders don't understand their pain points:** A strong majority (63%) feel top leaders at their organization don't understand the challenges developers face in their roles, up from 44% reported in last year's survey.

AI adoption is rising, but friction persists across the software development lifecycle:

- Developers are still losing valuable time to non-coding tasks: **50% report losing 10+ hours per week**, and **90% lose 6+ hours or more**, largely due to organizational inefficiencies.
- The biggest culprits? Not just one thing – **friction is everywhere**. Developers and managers alike cite a **fragmented knowledge landscape** as a key challenge: searching for services, documents, APIs, or onboarding new tech eats up time and slows teams down.

PART 1

What AI really means for developer workflows

After a year of limited adoption and mixed results for developers in 2024, it's clear AI is now saving developers time. As the space matures, a developer-centric approach is key to ensuring AI enhances productivity. In this section, we'll dive into survey insights and explore what's really working, and where there's room for improvement.

The State of AI: a snapshot

- **Generative AI (GenAI) is a workhorse:** 68% of developers report a time savings of more than 10 hours a week using GenAI, with significant gains for non-coding tasks.
- **Managers and developers are closely aligned on AI gains:** 70% of managers saved more than a quarter of their time.
- **AI is reshaping a developer's workweek:** A majority of developers are using their AI time savings to focus on improving code quality.

AI is gaining its footing across most dev teams

In our 2024 State of DevEx report, we saw a disconnect between leadership's expectations of AI and the everyday reality for developers. While a majority of managers ranked AI as the number one solution to improving developer productivity and satisfaction, surprisingly, only 38% of developers reported any time saving at all.

This year, we wanted to circle back on the state of developer sentiment around AI, tools being used, and areas where AI is adding value for teams. Things have changed significantly.

A majority of developers are now relying on AI for a lion's share of their tasks, saving over a quarter of their workweek. Managers have also made improvements in understanding their team's usage. When asked again this year about how AI is improving team productivity, managers were aligned with their developers, reporting time savings on par with what their teams were sharing.

68%

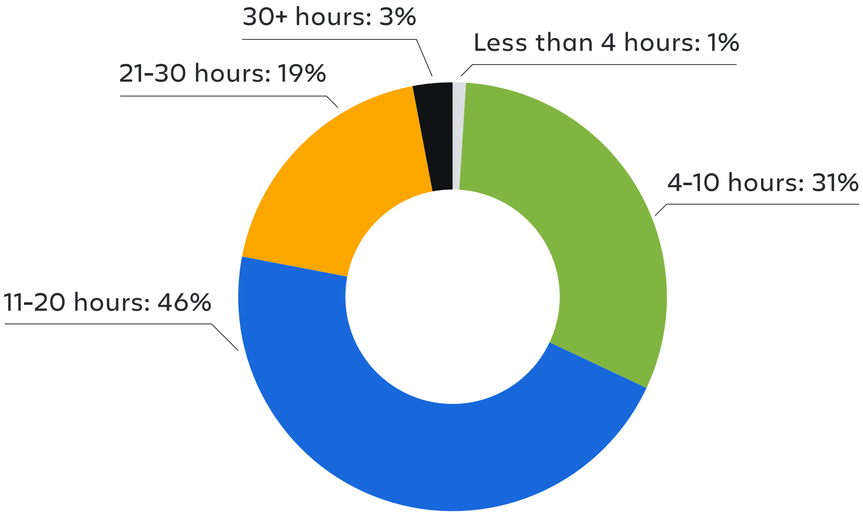
of developers are saving 10+ hours per week using AI

Takeaway #1: Generative AI is a workhorse

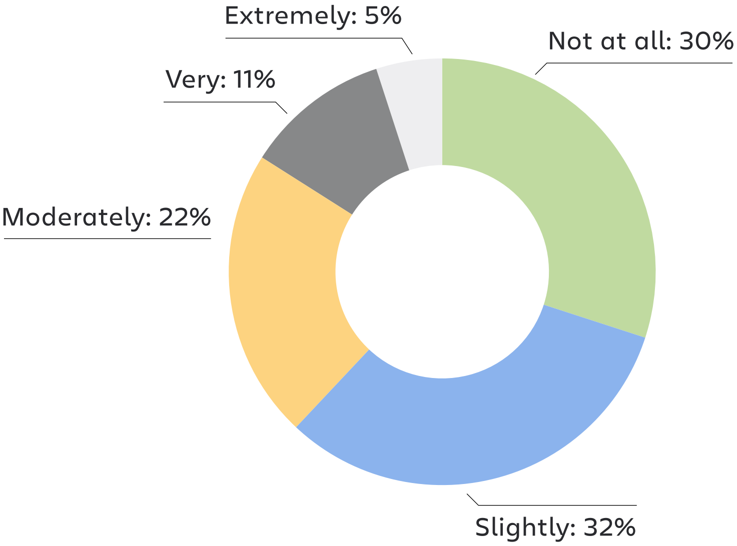
Most developers shared that they are using Generative AI (GenAI) tools to save time, with 68% reporting saving more than 10 hours a week across all of their workflows. While numerous reports study time savings on coding tasks, this survey focused on the holistic impact of GenAI across the entire working week.*

This is a sizeable jump from last year's survey, where 54% of developers reported only slight or moderate improvements to productivity* from leveraging AI tools.

Hours a week developers save on manual tasks using GenAI tools, from this year's State of Developer Experience 2025



Responses in last year's State of Developer Experience 2024 survey on how helpful AI tools were in improving developer productivity



*For this survey, we defined developer productivity as "delivering high-quality software at speed."

Takeaway #2: Leaders and developers are finding common ground on GenAI

70%

of managers reported that their team, as a whole, saves 10+ hours a week with GenAI

Interestingly, managers and developers are closely aligned on how much time GenAI is saving their teams. When asked how much time GenAI tools save their developers on tasks they would otherwise perform manually, 70% of managers said more than a quarter of their time.

Takeaway #3: AI is reshaping a developer's workweek

So, how are developers spending that extra time? The top priority: **improving code quality**. Close behind were **building new features**, **enhancing engineering culture**, and **developing documentation** – all sharing nearly equal attention.

This is great news for companies – developers are completing tasks more quickly and using the time they save to improve quality and develop new features. As companies start leveraging GenAI for more tasks outside code assistance, these gains will compound.

Top 4 ways developers are reallocating their time

1. Improve code quality
2. Develop new features
3. Improve engineering culture
4. Developing documentation

Takeaway #4: GenAI is versatile beyond the code

When asked what tasks they use GenAI for beyond coding, developers shared a closely distributed breakdown between five tasks: search and finding information, testing, writing and improving documentation, automating workflows, and chat, or sparring with AI.

Non-coding tasks developers report using GenAI for

1. Search and finding information
2. Testing
3. Writing and improving documentation
4. Automating workflows
5. Chat, or sparring with AI

PART 2

Putting developers at the center of AI adoption

The state of developer productivity: a snapshot

- **Growing empathy gap between leaders and developers:** A strong majority of developers (63%) feel top leaders at their organization don’t understand the challenges and pain points developers face in their roles, up by 19% from last year’s survey.
- **AI adoption is rising, but friction persists across the software development lifecycle:** Developers are losing valuable time to non-coding tasks: **50% report losing 10+ hours per week**, and **90% lose 6+ hours or more**, largely due to organizational inefficiencies
- **Alignment = increased productivity:** When managers work to understand developer pain points we see higher levels of productivity. Less time lost to obstacles and inefficiencies correlates directly with higher satisfaction with developer experience investment.

Organizational inefficiencies continue to disrupt developer workflows

Even with the rise in AI adoption and a growing ecosystem of developer tools, developers are still experiencing disruptions that take them out of flow. **50% of developers now report losing more than 10 hours** of their working week due to inefficiencies. From fragmented workflows to time lost searching for information, these persistent friction points continue to limit the full potential of AI gains.

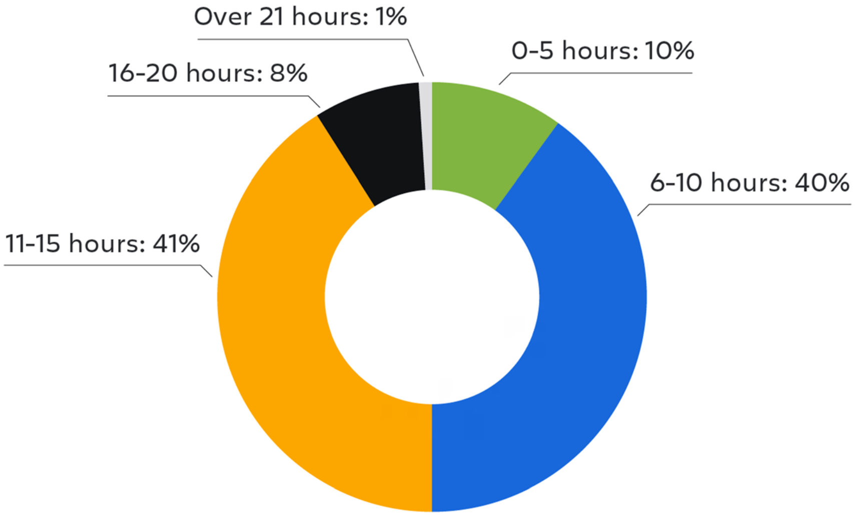
2025

50%

of developers lose

10+ hours of their week

due to inefficiencies



Further exacerbating this challenge, developers feel that leaders aren't aware of the blockers that are getting in their way. 63% of developers report that they feel that their leaders fall short in understanding the friction points they face, also up from 44% last year.

2025

63%

of developers feel leaders don't understand their pain points

2024

44%

of developers feel leaders don't understand their pain points

For a company with 500 developers, losing 10 hours per week per engineer adds up to approximately $7.9 million annually*.

> **10 hours a week of inefficiencies = a loss of $7,917,000 annually for a company with 500 developers**

*According to Stack Overflow's 2024 Developer Survey, the average salary for developers worldwide in 2024 was $63,333 USD (this is the median salary calculated off Stack Overflow's developer salary range of $60-75K USD, as reported in the "Salary and experience by language" section of their report). If a developer loses 10 hours weekly on average to inefficiencies, those inefficiencies come at an annual cost of $1,583,400 per 100 engineers per year. The larger the organization, the higher the cost of inefficiency.

Given the scale of these losses, it's crucial to identify and address the key friction points in a developer's day-to-day. To close this disconnect with their developers, managers should consider a developer-first, data-informed, and empathy-driven approach.

Use data to reveal, not to micromanage

Managers often fall into the trap of measuring developer productivity solely by output, such as story points completed or the number of tickets closed. In truth, developer productivity itself is hard to measure – but what you can measure are the obstacles and friction points that get in the way of it.

Listen in with regular feedback loops

To truly understand developers' pain points, it's crucial to listen actively, not just analyze data. Leaders should create space for developers to regularly share concerns about roadblocks, unclear requirements, or heavy workloads.

By consistently gathering feedback, you foster a culture where developers feel safe to voice challenges, leading to continuous improvement.

A closer look at developer friction points

> Several friction points from last year; finding information, lack of clear direction, and collaboration with other teams are ranking higher this year as causes of time waste.

When asked about the top areas causing the most friction and time waste to their role, both developers and managers reported an overwhelming number of factors.

Comparing the results to last year, we saw three areas move up in terms of causing time waste: finding information*, lack of clear direction, and collaboration with other teams. There were also two new causes among the top 5, both related to increasing complexity and cognitive load.

2025: Top areas causing friction and time waste

1. Finding information
2. New technology
3. Switching context between tools
4. Lack of clear direction
5. Collaboration with other teams
6. Tech debt

2024: Top areas causing friction and time waste

1. Tech debt
2. Insufficient documentation
3. Build processes
4. Lack of time for deep work
5. Lack of clear direction
6. Local environment setup
7. Collaboration with other teams

*In the context of this comparison, we have treated "Finding Information", the top reported area of friction in 2025 and "Insufficient documentation," the second reported area of friction in 2024, as synonymous. "Finding information" and "Insufficient documentation," are closely related and often have overlapping pain points.

Breaking down the bottlenecks

Our survey findings highlight some common challenges worth looking into:

- **Finding information:** When teams are able to self-serve information, they are 4.9 times more effective, 4.4 times more productive, and 4.4 times more adaptable.
- **Cognitive load:** The need to constantly search for information is a major contributor to cognitive load. According to the 2024 Stack Overflow Developer Survey, 61% of developers spend more than 30 minutes a day searching for answers to problems.
- **Tech debt:** Technical debt slows down development and increases costs. It can lead to more bugs, limit scalability, and lower team morale. If left unaddressed, tech debt can hurt the user experience and make it harder to adapt or grow the product.

'Finding information': AI's core advantage still stumping devs

'Finding information' emerged at the top of the leading causes of time waste for development teams. What is interesting is that 'finding information,' along with the second and third friction points (new technology and switching context between tools) are AI mainstays. With teams reporting increased adoption and higher time savings, you would expect to see a decrease in these pain points that AI is designed to address.

Here are some actionable steps your team can take to tackle information overload:

Improve documentation and review permissions

One of the reasons developers could be struggling to find information, even with AI, is because work isn't well documented, or is blocked by permissions.

Our State of Teams 2024 report which surveyed 5,000 knowledge workers further validates this common challenge. 55% of surveyed workers reported that they struggle to locate necessary data. Additionally, 50% also shared that they have found themselves duplicating work already being done by another team.

55%

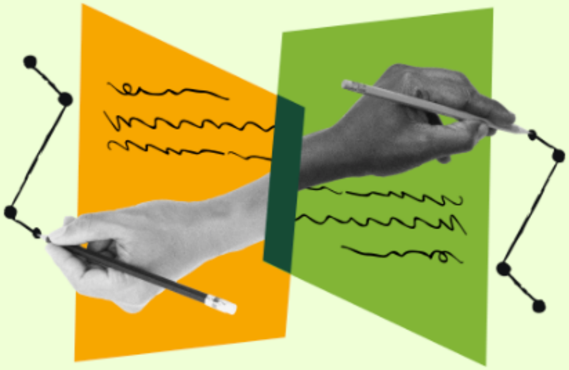
of knowledge workers **find it hard to track down information** despite knowing a lot of people at their job

50%

of knowledge workers have worked on a project and only **later found out that another team was working on the same thing**

56%

of knowledge workers say that teams at their company **plan and track work in different ways, which makes it hard to collaborate**



The report serves as a reminder that empowering developers to find the information they need quickly and independently can go a long way.

Obsess over documentation quality

Great documentation lays the groundwork for teams to get aligned faster, saving time so that they can focus more on designing and innovating. AI simplifies this process by organizing content, improving tone, and adding context for better clarity. AI can also spot key action items and summarize lengthy documents while fitting seamlessly into your workflow. Paired with a quick review, AI content tools can help teams create more consistent and accessible documentation.

Learn more about how we do this at Atlassian.

Build and maintain self-serve documentation

The report also found that teams with easy access to self-serve information are 4.9 times more effective, 4.4 times more productive, and 4.4 times more adaptable.

When teams make information self-serve, they are:

4.9x
more likely to be
effective

4.4x
more likely to be
productive

4.4x
more likely to be
adaptable

Teams can also use AI integrations in the IDE to simplify self-service documentation, improve discoverability, and strengthen collaboration.

Atlassian's Rovo CLI is one example of an agentic CLI experience that brings intelligent development assistance directly to the terminal. It enables developers to interact with an AI agent using natural language for tasks like asking for documentation, summarizing code changes, or publishing updates as part of their workflow.

By connecting Rovo CLI to the MCP (Model Context Protocol) server and IDE extensions, developers can generate, update, and retrieve documentation directly from the editor, reducing context switching and manual search. The MCP server becomes a central hub that integrates knowledge management and project management tools like Confluence and Jira.

> After deploying Rovo Search, 71% of Atlassian developers reported they were able to find documentation for developer tools and tech stacks that they need to do their job, up from 63% last year.

Learn more about trends and challenges around team collaboration in the full State of Teams 2024 report.

Tackling the 84% of the work week spent outside the IDE

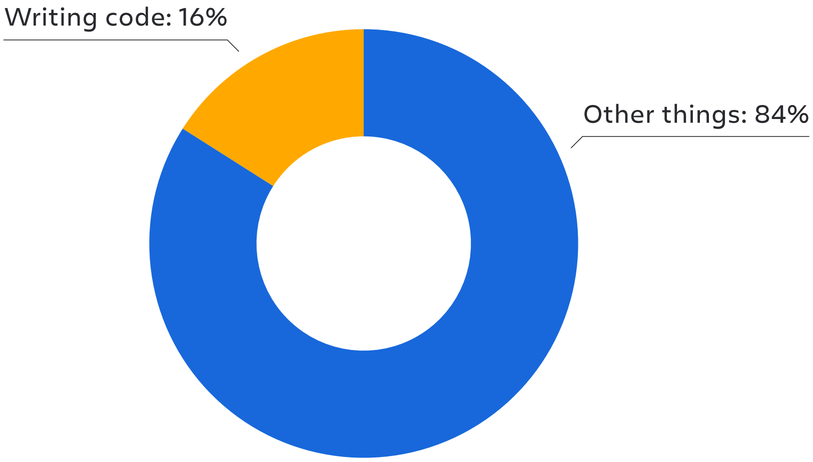
If you recall from the beginning of the report, 68% of developers are saving 10+ hours a week using AI. That's a significant time-saving! And there's still room for improvement, when we consider that a greater percentage of developers are dealing with 10+ hours a week of friction due to organizational inefficiencies.

When we scan the landscape of developer AI solutions today, most capabilities focus on coding. It's easier – and more tempting than ever – to automate developers' favorite part of the job.

The Developer AI Landscape

Category	Capabilities
Code generation	Using AI to write or help write code
Documentation generation	Using AI to write code or procedural documentation
Pull request summaries	AI creates a summary of the changes contained within a Pull Request
Unit test generation	AI automatically suggests or creates unit tests for changes in a Pull Request

Most Developer AI tools today can be categorized into these four buckets. These features span a wide portion of the software developer lifecycle and pack quite a punch in terms of impact and utility. They also have one thing in common: they're centered around coding.



+IDC Survey Spotlight, Feb 2025. "How do you developers spend their time?"

This imbalance slows delivery, reduces value, and chips away at morale. Instead of tapping into their creative potential, developers are bogged down by documentation, meetings, and maintenance work.

While optimizing coding time matters, **the bigger opportunity lies in reducing friction across everything else**. That's where AI can make a transformative impact – streamlining non-coding tasks and improving the overall developer experience.

Teams that collaborate with AI see exponential gains in performance and output, giving their organizations a competitive edge. Leaders can accelerate this shift by spotlighting internal success stories – showcasing how developers are using AI to elevate their work provides both inspiration and a blueprint for others to follow.

IDC Survey Spotlight, Feb 2025, "How do developer spend their time?"

Bridging the developer-leadership gap

For AI to meaningfully improve the developer experience, it must solve real pain points – **without adding new friction**. That starts by listening.

And developers are feeling that disconnect: **63% say leadership doesn't understand their pain points** – a steep rise from 44% last year.*

Developers are closest to the work and often see problems first. But those insights only help if they're clearly communicated – with specific examples, measurable impact, and business context. Leaders need to invite and act on this input through retrospectives, surveys, and skip-level meetings that foster open dialogue.

Many organizations now have dedicated Developer Experience (DevEx) or Developer Enablement teams focused on reducing friction and improving workflow efficiency. These teams are uniquely positioned to bridge the gap between leadership and engineering, helping surface real pain points and prioritize solutions – whether AI-driven or not. Empowering DevEx teams to lead these conversations ensures that improvements are grounded in the developer reality, not just leadership intent.

When communication flows both ways – with clarity, empathy, and intent – teams build trust, surface issues early, and stay aligned on what matters most.

Once friction points are clearly understood, the next step is to select solutions that directly address them. The goal isn't AI adoption for its own sake, but empowering developers to do their best work, faster and with fewer blockers.

Developers who aren’t bogged down by broken tooling, unclear documentation, or inefficient processes are able to spend more time doing what they do best – building great software. We’ve seen this firsthand at Atlassian. That’s why we believe leadership must focus on streamlining workflows, reducing cognitive load, and tackling the technical and operational debt that slows teams down. This belief led us to launch our internal initiative, Developer Joy.

> “Paying down technical debt and automating manual processes is an ongoing investment. Like joining a gym, it’s a lifestyle change that starts with an initial ‘get back in shape’ push.”
>
> **Rajeev Rajan, Chief Technology Officer, Atlassian**

Through the Developer Joy program, we automated repetitive tasks like pull request management, modernized outdated test libraries, and addressed long-standing technical debt. We also tracked developer joy as a company-level OKR, ensuring visibility across all organizations, so everyone could contribute to this goal – not just the engineering team.

Over two years, this focused effort resulted in a 50% increase in developer satisfaction.

Our takeaway is simple: when we listen to our engineers and act on their feedback, we build better, faster, more resilient engineering teams.

PART 3

Measuring developer activity != productivity

Frameworks can be useful starting points, but ultimately no single framework can fully capture what matters most in your team's unique environment.

The SPACE Framework is favored by a majority of dev teams

The SPACE framework ranks as the most used measure of developer productivity this year, with more than 50% of developers using it to measure how their teams are delivering high-quality software at speed. While this was a new form of measurement introduced into our survey this year, it quickly rose to the top, with last year's number one, "Amount of code written, removed, or updated," bumped down to third place.

The SPACE framework and deployment frequency's growing mindshare might indicate an increasingly balanced approach to measuring developer productivity.

2025: Top 5

1. The SPACE Framework
2. Development frequency
3. Amount of code written, removed, or updated
4. Number of story points a team completes per sprint
5. Hours worked

2024: Top 5

1. Amount of code written, removed, or updated
2. Deployment frequency
3. Number of story points a team completes per sprint
4. Hours worked
5. Change failure rate

Rethinking the way your team measures DevEx

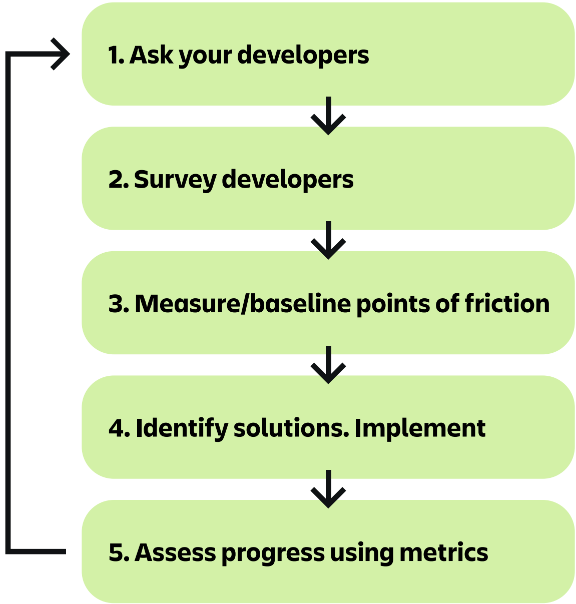
There are a lot of frameworks out there – like SPACE and DORA metrics – that aim to make it easier for organizations to measure engineering performance. These can be useful starting points, offering structured ways to think about things like satisfaction, efficiency, and delivery performance. But ultimately no single framework can fully capture what matters most in your team's unique environment.

The real value comes from not just adopting a framework wholesale, but from thoughtfully adapting it to reflect your own unique circumstances.

Developer experience and engineering culture are unique to each organization. Each team has their own identity which is shaped over time, and impossible to replicate.

In order to capture an accurate view of how your teams are performing and what's working for them, it's important to consider your organization's context, values, and ways of working.

The path to improving Developer Experience



1. Ask your developers
2. Survey developers
3. Measure/baseline points of friction
4. Identify solutions. Implement
5. Assess progress using metrics

That doesn't mean you need to start from scratch though. There are companies that have done this well, and their journeys can offer valuable inspiration. But it's crucial to focus on *how* they found the right measures for *their* environment. Not just want they ended up measuring. Trying to apply someone else's metrics directly to your own teams can backfire and lead to misleading signals.

PART 4

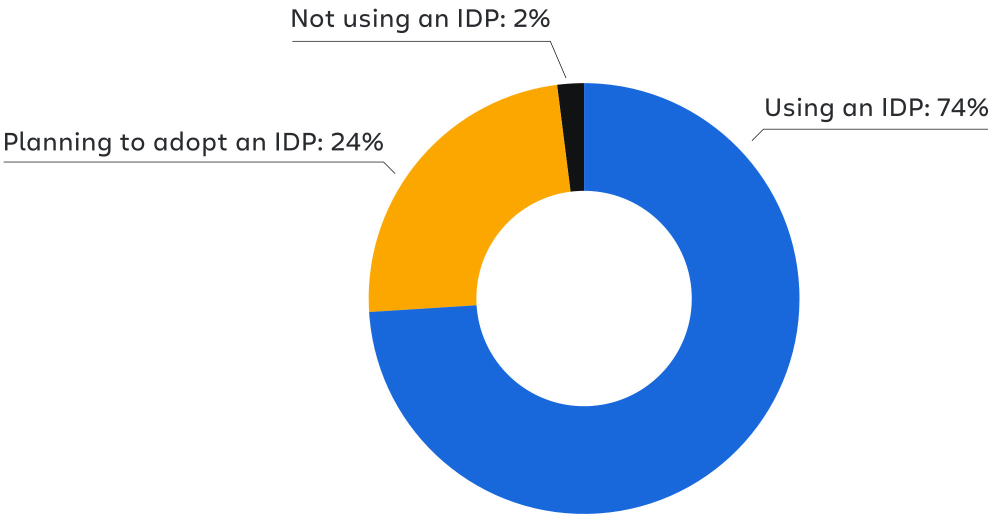
Engineering orgs continue to see benefits from IDP adoption

Most development teams are either actively using or are planning on using Internal Developer Platforms.

Companies are increasingly turning to Internal Developer Platforms (IDPs) to improve developer experience, increase the pace of innovation, and drive standardization and compliance. Our survey results show that 98% of developers are working in organizations that are either actively using or are planning on using IDPs.

IDPs enable developers to spend more time on building software and reduce the amount of time spent on toilsome activities unrelated to writing code. Instead of searching for documentation, figuring out service ownership, or stitching together fragmented tools, developers get a single UI that provides all of the information they need to move faster and collaborate across teams - from service health metrics to deployment pipelines and integration guides.

As the adoption of IDPs grows, their benefits are becoming clearer, with organizations realizing that these platforms can significantly accelerate development cycles and improve overall operational efficiency.



The top four reported benefits of using IDPs were improved scalability and reliability, increased developer experience and productivity, faster time to market, and reduced operational costs.

Top reported benefits of IDPs

Improved application scalability and reliability	IDPs improve scalability and reliability by automating resource allocation, ensuring consistent environments, and enabling quick issue detection.
Enhanced Developer Experience and increased productivity	By automating mundane tasks and simplifying workflows, IDPs improve the developer experience and enable teams to focus on high-priority projects.
Faster time to market	IDPs accelerate time to market by offering developers self-service tools, automated workflows, and real-time feedback for faster development and deployment.
Reduced operational costs	IDPs reduce operational costs by centralizing event tracking and dependencies, enabling faster issue resolution and decision-making.

Taking it further with IDPs

To get the most value out of your Internal Developer Platform, focus on improving developer experience with easy onboarding, low cognitive load, and self-service workflows. Treat the platform as you would a product – gather feedback, track usage, and prioritize what developers actually need. Drive consistency with reusable components and templates, and scale safely using guardrails and automation that support speed without sacrificing security.

ATLASSIAN | Domino's

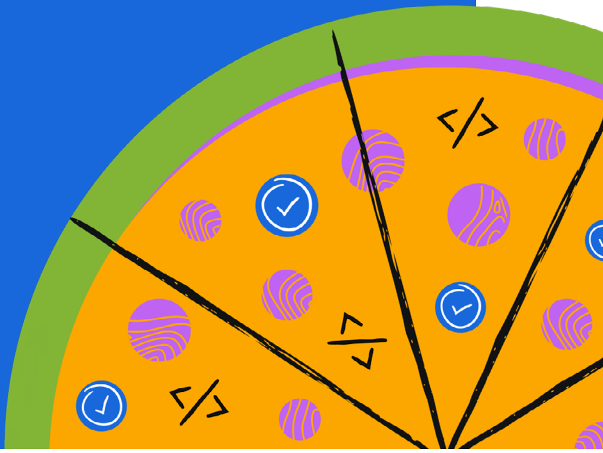
How Domino's Pizza delivers engineering excellence and system reliability with Compass*

Challenge: After shifting to a product delivery model in 2022, Domino's Pizza Enterprises Ltd realized the need for an internal developer portal. Documentation became scattered across project spaces, leading to team and information silos. Lacking shared components and best practices, their IT environment quickly became a state of software sprawl. The shift in responsibilities also meant teams inherited supportive services for many components they had never worked on before. Teams also lacked critical insight into the health and performance of their systems.

Solution: Domino's uses standardized components in Compass to drive system uniformity, which allows engineers to onboard quickly and maintain best practices with minimal disruption – even with annual team changes. Additionally, the ability to configure Jira projects to use Compass components affords the ability to track issues across payment APIs, teams, and projects. The Compass software catalog's unified source of truth lets teams quickly locate component documentation and ownership information, driving faster incident response and improving system uptime and reliability.

*Content sourced from **'From Dough to Deployment: Dominos' Recipe for Success with Atlassian Compass'** at Team '25 in Anaheim, CA

> In the heat of the moment, particularly in a commerce environment where every minute of degradation impacts sales, being able to find the information that you need through Compass has helped us find the information faster and save the business money.
>
> **ANDREW FRASER**
>
> Software Engineering Manager at Domino's Pizza Enterprises



PART 5

Final thoughts

While AI is proving to be a powerful lever in improving productivity, the developer experience still faces notable hurdles.

Adoption of AI tools is on the rise, with both developers and managers now reporting meaningful time savings – often 10+ hours a week across both coding and non-coding tasks. Scaling these productivity gains across engineering orgs will come from solving the deeper workflow inefficiencies that plague teams every day.

The widening empathy gap between developers and leadership is a growing concern. Over 60% of developers feel their leaders don't understand the challenges they face, and more than half report losing over 10 hours each week due to organizational inefficiencies. These aren't just productivity issues – they're culture issues.

Frameworks like SPACE and tools like Internal Developer Platforms (IDPs), especially when paired with AI, are helping organizations take a more holistic approach to measuring and improving the developer experience. But data alone isn't enough. Conversations, empathy, and a focus on real-world friction points are what truly move the needle.

Ultimately, improving developer experience isn't about chasing the newest tools or obsessing over output metrics. It's about understanding the human side of software development – removing barriers, listening deeply, and creating an environment where developers can do their best work.

Closing thoughts:

- **AI gives teams a real advantage, and there is still room to grow.** Developers and managers are seeing real-time savings from AI – often over 10 hours a week – but most tools focus on coding, which is only a small part of a developer's overall workflow.
- **Organizational inefficiencies are a major drain on developers' time.** Over 50% of developers lose 10+ hours weekly due to inefficiencies like tool sprawl, poor information access, and context switching – highlighting the need for systemic changes.
- **There's a widening empathy gap between developers and leadership.** 63% of developers feel their leaders don't understand their challenges, indicating a disconnect that could impact morale, retention, and productivity.
- **Holistic frameworks Like SPACE are replacing output-only metrics.** The SPACE framework is now the top choice for measuring developer productivity, signaling a shift away from outdated metrics like lines of code toward more human-centered, outcome-based assessments.
- **Real developer experience gains come from addressing friction points.** To truly improve DevEx, organizations should go beyond automating enjoyable tasks like coding and focus on solving the frustrating parts of the job – information silos, unclear direction, and tech debt.

PART 6

Methodology

We worked with Wakefield Research to run an online survey surveying a total of 3,500 developers across the US, Australia, France, Germany, United Kingdom, and India.

- The survey was fielded from March 13, 2025 to March 23, 2025
- 1,750 respondents reported a title with a minimum seniority of software developer manager
- 1,750 respondents reported a title with a seniority below that of software developer manager
- Respondents were recruited through an email invitation

Statistical significance:
Results of any sample are subject to sampling variation. The extent of this variation depends on the number of people surveyed and the percentages reported. In this study, there's a 95% chance that the reported results are within ±1.66 percentage points for the Global Sample, ±3.1 points for the U.S., and ±4.38 points for Australia, France, the U.K., and India, compared to what would be found if the entire population had been surveyed.

PART 7

Recommended reading

Want to learn more?

About Atlassian: Atlassian unleashes the potential of every team. Our AI-powered software development platform helps engineering teams deliver high-quality software faster. More than 300,000 organizations – from startups to global enterprises – trust Atlassian to accelerate every stage of the software development lifecycle, from ideation to impact.

Learn more about Atlassian and get more of our insights on developer experience:

| in

Domino's

From dough to deployment: Dominos' recipe for success with Atlassian Compass

Read now

ATLASSIAN

How We Build Engineering Blog

Read now

Bitbucket

Code and CI/CD, powered by the Atlassian platform

Learn more

Learn more about how we help improve our own developer experience within Atlassian.

ATLASSIAN