

Artificial intelligence: Adoption and attitudes



Takeaways

The vast majority of organizations across all industries surveyed are shifting their priorities to more deeply incorporate AI into their applications and services. A corresponding majority of development professionals are relying on AI to help them perform their core role responsibilities — and reporting increases in productivity as a result. Development professionals’ perceptions that using AI is necessary for remaining competitive in today’s market is pervasive and appears to be an important driver of AI adoption for both organizations and individual development professionals.

Introduction

It would be difficult to ignore the significant impact that AI has had on the landscape of development work this year, given the proliferation of popular news articles outlining its effects, from good¹ to bad² to ugly.³ So, while AI was only discussed as one of many technical capabilities affecting performance in our 2023 Accelerate State of DevOps Report,⁴ this year we explore this topic more fully.

As the use of AI in professional development work moves rapidly from the fringes to ubiquity, we believe our 2024 Accelerate State of DevOps Report represents an important opportunity to assess the adoption, use, and attitudes of development professionals at a critical inflection point for the industry.

Findings

Adopting Artificial Intelligence

Findings on the adoption of AI suggest a growing awareness that AI is no longer “on the horizon,” but has fully arrived and is, quite likely, here to stay.

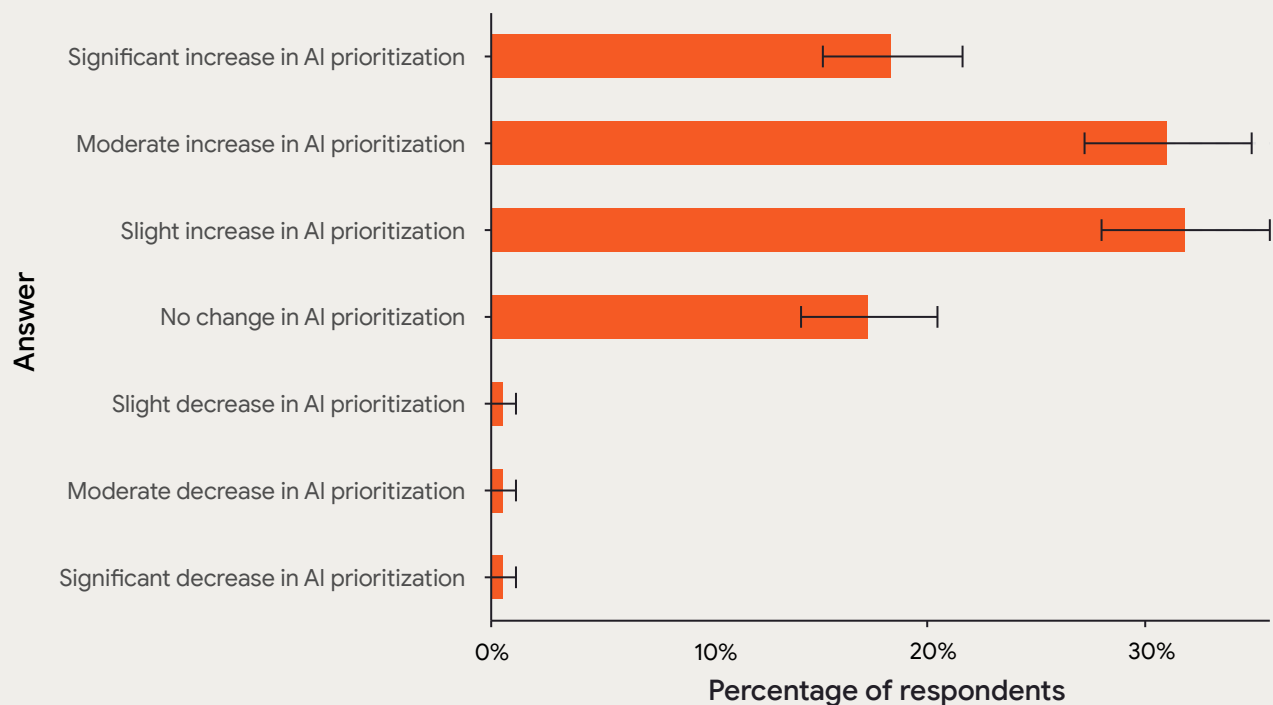
Organizational adoption of Artificial Intelligence

The vast majority of respondents (81%) reported that their organizations have shifted their priorities to increase their incorporation of AI into their applications

and services. 49.2% of respondents even described the magnitude of this shift as being either “moderate” or “significant.”

Notably, 3% of respondents reported that their organizations are decreasing focus on AI — within the margin of error of our survey. 78% of respondents reported that they trusted their organizations to be transparent about how they plan on using AI as a result of these priority shifts. This data is visualized in Figure 2.

Changes in organizational priorities concerning AI



Error bar represents 89% uncertainty interval

Figure 2: Respondents’ perceptions of their organizations’ shifts in priorities toward or away from incorporation of AI into their applications and services.

Participants from all surveyed industries reported statistically identical levels of reliance on AI in their daily work, which suggests that this rapid adoption of AI is unfolding uniformly across all industry sectors. This was somewhat surprising to us. Individual industries can vary widely with respect to their levels of regulatory constraints and historical pace of innovation, each of which can impact rates of technology adoption.

However, we did find that respondents working in larger organizations report less reliance on AI in their daily work than respondents working in smaller organizations, which is consistent with prior literature indicating larger firms more slowly adapt to technological change because of their higher organizational complexities and coordination costs.⁵



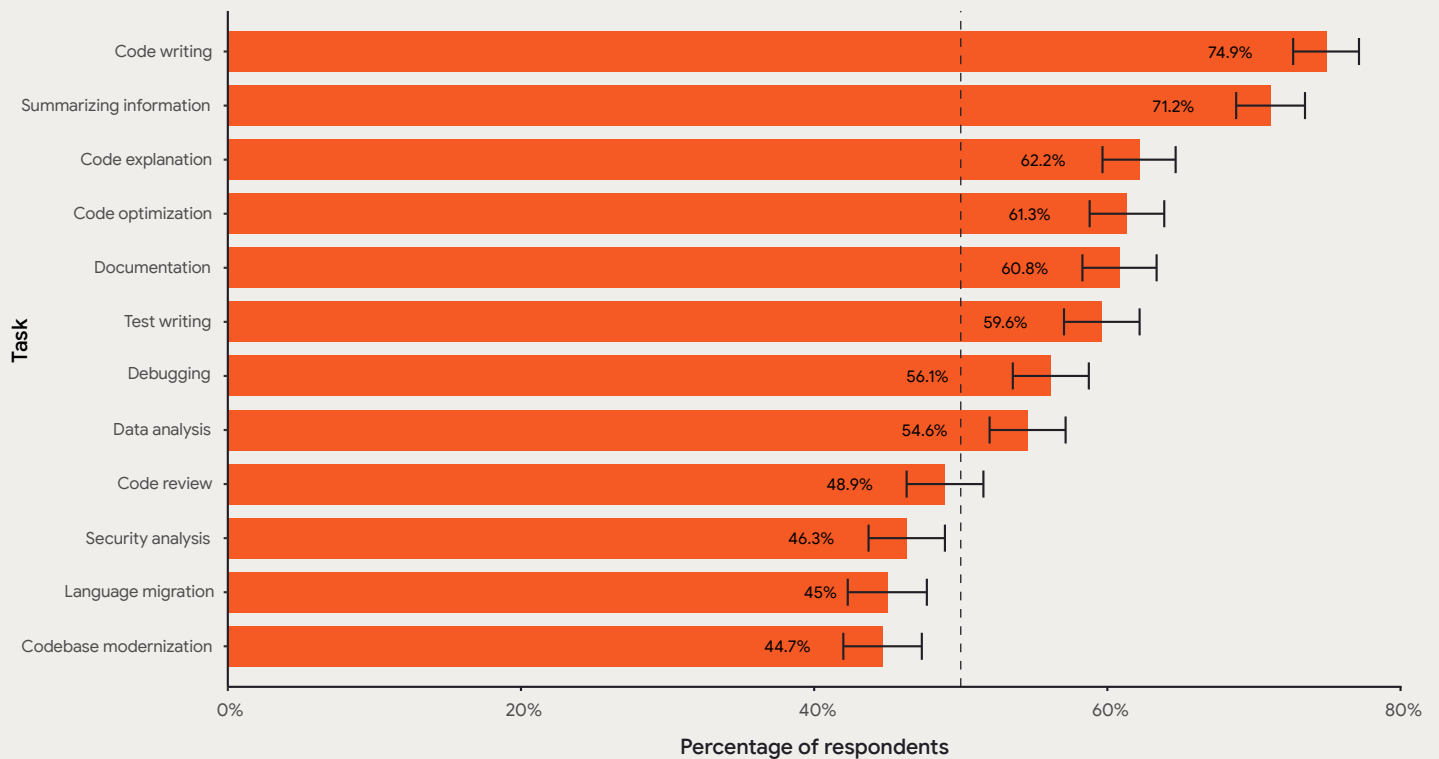
Individual adoption of artificial intelligence

At the individual level, we found that 75.9% of respondents are relying, at least in part, on AI in one or more of their daily professional responsibilities. Among those whose job responsibilities include the following tasks, a majority of respondents relied on AI for:

1. Writing code
2. Summarizing information
3. Explaining unfamiliar code
4. Optimizing code
5. Documenting code
6. Writing tests
7. Debugging code
8. Data analysis

Of all tasks included in our survey responses, the most common use cases for AI in software development work were writing code and summarizing information, with 74.9% and 71.2% of respondents whose job responsibilities include these tasks relying on AI to perform them — at least in part. This data is visualized in Figure 3.

Task reliance on AI



Error bar represents 89% credibility interval

Figure 3: Percentage of respondents relying on AI, at least in part, to perform twelve common development tasks

Chatbots were the most common interface through which respondents interacted with AI in their daily work (78.2%), followed by external web interfaces (73.9%), and AI tools embedded within their IDEs (72.9%). Respondents were less likely to use AI through internal web interfaces (58.1%) and as part of automated CI/CD pipelines (50.2%).

However, we acknowledge that respondents' awareness of AI used in their CI/CD pipelines and internal platforms likely depends on the frequency with which they interface with

those technologies. So, these numbers might be artificially low.

We found that data scientists and machine learning specialists were more likely than respondents holding all other job roles to rely on AI. Conversely, hardware engineers were less likely than respondents holding all other job roles to rely on AI, which might be explained by the responsibilities of hardware engineers differing from the above tasks for which AI is commonly used.



Drivers of adoption of artificial intelligence

Interview participants frequently linked the decision to adopt AI to competitive pressures and a need to keep up with industry standards for both organizations and developers, which are increasingly recognized to include proficiency with AI.

For several participants' organizations, using AI at all was seen as "a big marketing point" (P3)⁶ that could help differentiate their firm from competitors. Awareness that competitors are beginning to adopt AI in their own processes even prompted one firm to forgo the typical "huge bureaucracy" involved in adopting new technology because they felt an urgency to adopt AI, questioning "what if our competitor takes those actions before us?" (P11).

At the individual level, many participants linked their adoption of AI to the sentiment that proficiency with using AI in software development is "kind of, like, the new bar for entry as an engineer" (P9). Several participants suggested fellow developers should rapidly adopt AI in their development workflow, because "there's so much happening in this space, you can barely keep up... I think, if you don't use it, you will be left behind quite soon" (P4).

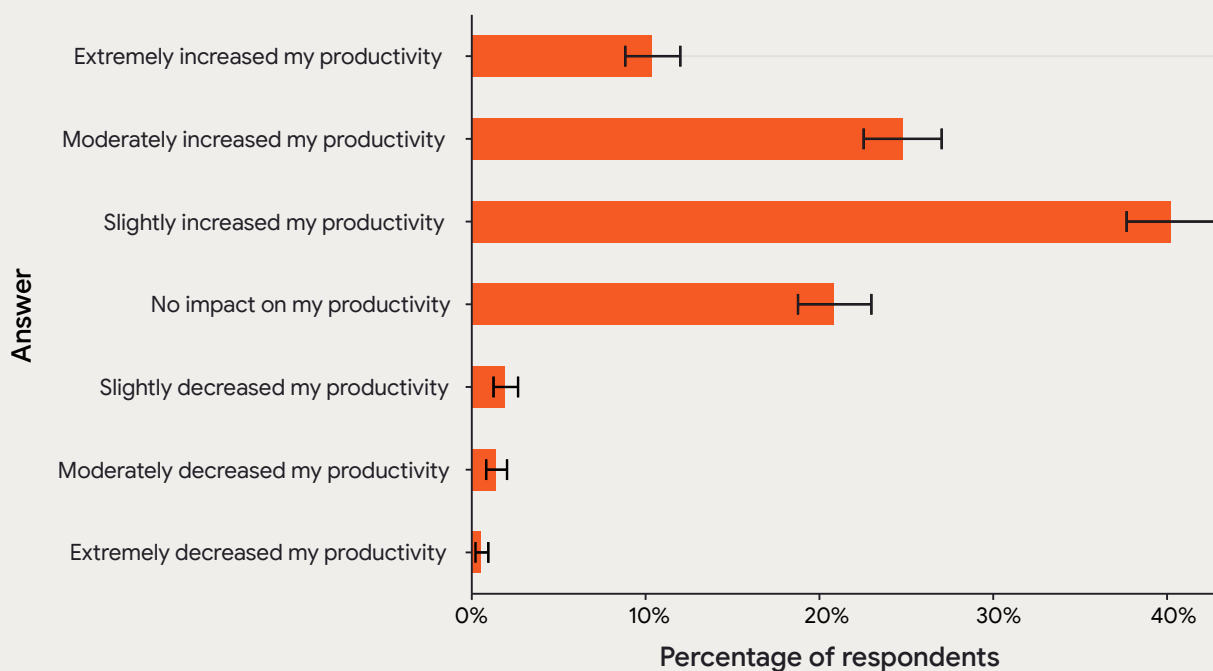
Perceptions of artificial intelligence

Performance improvements from artificial intelligence

For the large number of organizations and developers who are adopting it, the benefits of using AI in development work appear to be quite high. Seventy-five percent of respondents reported positive productivity gains from AI in the three months preceding our survey, which was fielded in early 2024.

Notably, more than one-third of respondents described their observed productivity increases as either moderate (25%) or extreme (10%) in magnitude. Fewer than 10% of respondents reported negative impacts of even a slight degree on their productivity because of AI. This data is visualized in Figure 4.

Perceptions of productivity changes due to AI



Error bar represents 89% uncertainty interval

Figure 4: Respondents' perceptions of AI's impacts on their productivity.

Across roles, respondents who reported the largest productivity improvements from AI were security professionals, system administrators, and full-stack developers. Although they also reported positive productivity improvement, mobile developers, site reliability engineers, and project managers reported lower magnitudes of productivity benefits than all other named roles.

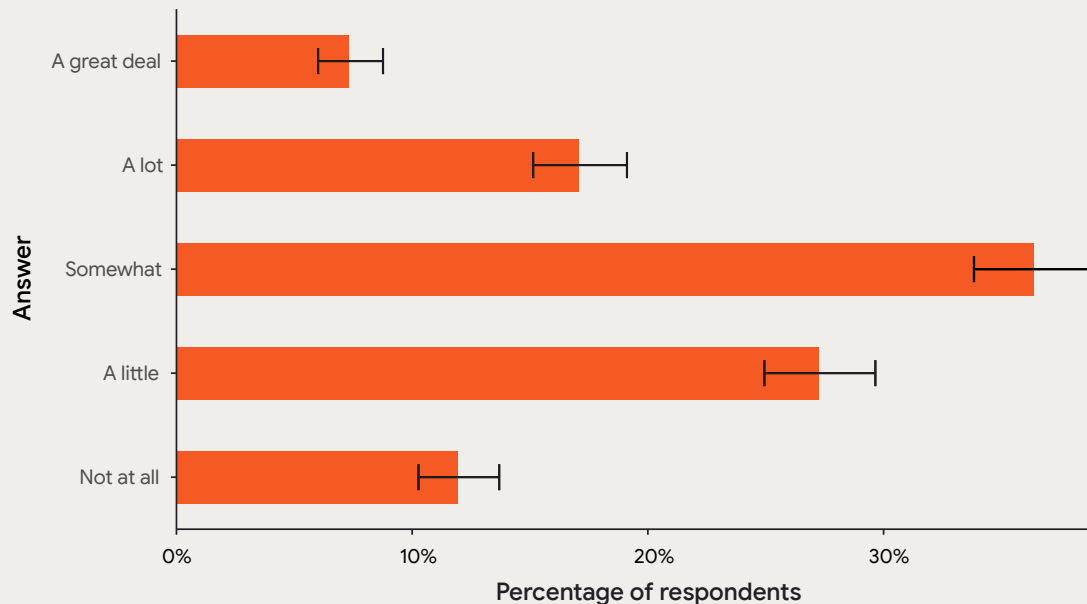
Although we suspected that the novelty of AI in development work, and corresponding learning curve, might inhibit developers' ability to write code, our findings did not support that hypothesis. Only 5% of respondents reported that AI had inhibited their ability to write code to any degree. In fact, 67% of respondents reported at least some improvement to their ability to write code as a result of AI-assisted coding tools, and about 10% have observed "extreme" improvements to their ability to write code because of AI.

Trust in AI-generated code

Participants' perceptions of the trustworthiness of AI-generated code used in development work were complex. While the vast majority of respondents (87.9%) reported some level of trust in the quality of AI-generated code, the degree to which respondents reported trusting the quality of AI-generated code was generally low, with 39.2% reporting little (27.3%) or no trust (11.9%) at all. This data is visualized in Figure 5.



Trust in quality of AI-generated code



Error bar represents 89% uncertainty interval

Figure 5: Respondents' reported trust in the quality of AI-generated code.

Given the evidence from the survey that developers are rapidly adopting AI, relying on it, and perceiving it as a positive performance contributor, we found the overall lack of trust in AI surprising. It's worth noting that during our interviews, many of our participants indicated that they were willing to, or expected to, tweak the outputs of the AI-generated code they used in their professional work.

One participant even likened the need to evaluate and modify the outputs of AI-generated code to “the early days of StackOverflow, [when] you always thought people on StackOverflow are really experienced, you know, that they will know exactly what to do. And then, you just copy and paste the stuff, and things explode” (P2).

Perhaps because this is not a new problem, participants like P3 felt that their companies are not “worried about, like, someone just copy-and-pasting code from Copilot or ChatGPT [because of] having so many layers to check it” with their existing code-quality assurance processes.

We hypothesize that developers do not necessarily expect absolute trust in the accuracy of AI-generated code, nor does absolute trust appear to be required for developers to find AI-generated code useful. Rather, it seems that mostly-correct AI-generated code that can be perfected with some tweaks is acceptable, sufficiently valuable to motivate widespread adoption and use, and compatible with existing quality assurance processes.

Expectations for AI's future

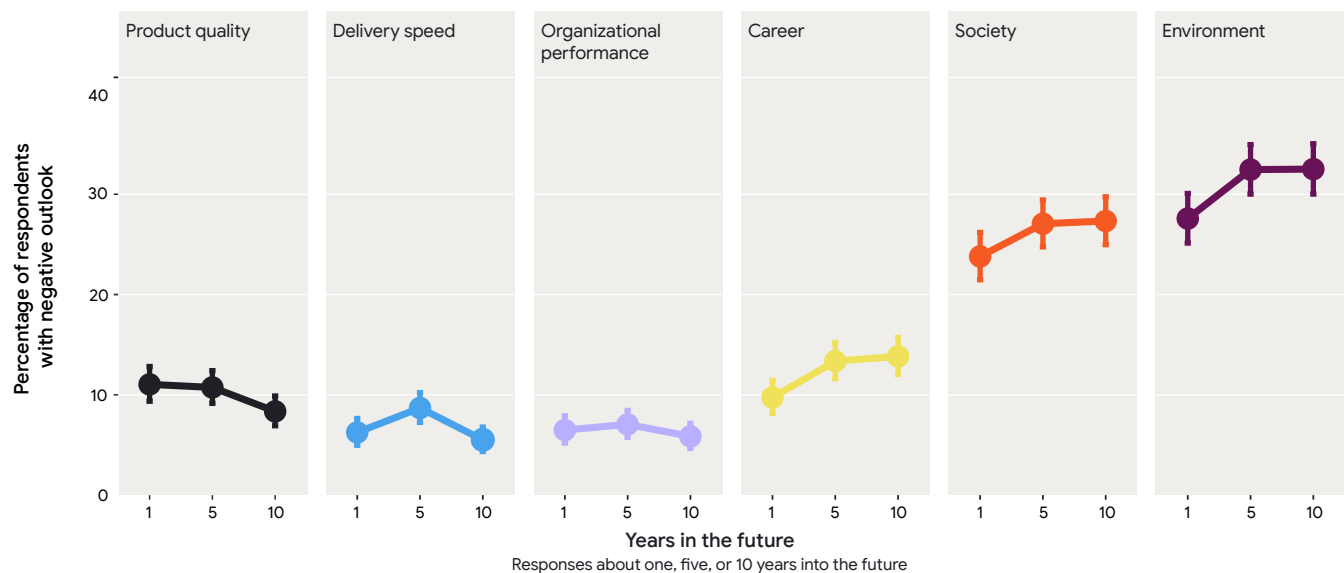
Overall, our findings indicate AI has already had a massive impact on development professionals' work, a trend we expect to continue to grow. While it would be impossible to predict exactly how AI will impact development — and our world — in the future, we asked respondents to speculate and share their expectations about the impacts of AI in the next one, five, and 10 years.

Respondents reported quite positive impacts of AI on their development work in reflecting on their recent experiences, but their predictions for AI's future impacts were not as hopeful.

Optimistically, and consistent with our findings that AI has positively impacted development professionals' performance, respondents reported that they expect the quality of their products to continue to improve as a result of AI over the next one, five, and 10 years.

However, respondents also reported expectations that AI will have net-negative impacts on their careers, the environment, and society, as a whole, and that these negative impacts will be fully realized in about five years time. This data is visualized in Figure 6.

Expected negative impacts of AI



Error bar represents 89% credibility interval

Figure 6: Respondents' expectations about AI's future negative impacts in the next one, five, and 10 years.

Interview participants held similarly mixed feelings about the future impacts of AI as our survey respondents. Some wondered about future legal actions in a yet-to-be-decided regulatory landscape, worrying they might “be on the wrong side of it, if things get decided” (P3).

Others echoed long-held anxieties and asked, “Is it going to replace people? Who knows? Maybe.” (P2), while their peers dismissed their fears by drawing parallels to the past, when “people used to say ‘Oh, Y2K! Everything will be doomed!’ Blah, blah... because it was a new thing, at that time.

[But,] nothing got replaced. In fact, there were more jobs created. I believe the same thing will happen with AI” (P1).

The future effects AI will have on our world remain unclear. But, this year, our survey strongly indicates that AI has produced an unignorable paradigm shift in the field of software development. So far, the changes have been well-received by development professionals.



¹. <https://www.sciencedaily.com/releases/2024/03/240306144729.htm>

². <https://tech.co/news/list-ai-failures-mistakes-errors>

³. <https://klyker.com/absurd-yoga-poses-generated-by-ai/>

⁴. <https://dora.dev/dora-report-2023>

⁵. Rogers, Everett M., Arvind Singhal, and Margaret M. Quinlan. “Diffusion of innovations.” An integrated approach to communication theory and research. Routledge, 2014. 432-44, Tornatzky, L. G., & Fleischer, M. (1990). The processes of technological innovation. Lexington, MA: Lexington Books

⁶. (P[N]), for example (P1), indicates pseudonym of interview participants.

Exploring the downstream impact of AI



Takeaways

This chapter investigates the impact of AI adoption across the spectrum, from individual developers to entire organizations. The findings reveal a complex picture with both clear benefits and unexpected drawbacks. While AI adoption boosts individual productivity, flow, and job satisfaction, it may also decrease time spent on valuable work.

Similarly, AI positively impacts code quality, documentation, and review processes, but surprisingly, these gains do not translate to improved software delivery performance. In fact, AI adoption appears detrimental in this area, while its effect on product performance remains negligible.

Despite these challenges, AI adoption is linked to improved team and organizational performance. This chapter concludes with a call to critically evaluate AI's role in software development and proactively adapt its application to maximize benefits and mitigate unforeseen consequences.

The AI moment & DORA

Estimates suggest that leading tech giants will invest approximately \$1 trillion on the development of AI in the next five years.¹ This aligns well with a statistic presented in the "[Artificial intelligence: Adoption and attitudes](#)" chapter that 81% of respondents say their company has shifted resources into developing AI.

The environmental impacts of AI further compound the costs. Some estimates suggest that by 2030, AI will drive an increase in data center power demand by 160%.² The training of an AI model can add up to roughly “the yearly electricity consumption of over 1,000 U.S. households”.³ It is no surprise that more than 30% of respondents think AI is going to be detrimental to the environment.

Beyond the development and environmental costs, we have the potential for adoption costs.

This could come in many forms, from productivity decreases to the hiring of specialists. These adoption costs could also come at a societal level. Over a third of respondents believe AI will harm society in the coming decade. Given these costs, it seems natural for people to have a deep curiosity about the returns.

This curiosity has manifested itself in a wealth of media, articles, and research whose sentiment and data are both mixed, at least to some extent.



Some believe that AI has dramatically enhanced the ability of humanity,⁴ others suggest that AI is little more than a benign tool for helping with homework,⁵ and some fear that AI will be the downfall of humanity.⁶

Evidence for proximal outcomes, such as the ability to successfully complete a particular task, is largely positive.⁷ When the outcome becomes more distant, such as a team's codebase, the results start becoming a little less clear and a little less positive. For example, some research has suggested that code churn may double from the pre-2021 baseline.⁸

The challenge of understanding these downstream effects is unsurprising. The further away the effect is from the cause, the less pronounced and clear the connection.

Evaluating the downstream effects of AI mimics quantifying the effect of a rock thrown into a lake. You can most easily attribute the ripples closest to the impact point of the rock in the water, but the farther from the entry point you go, the less pronounced the effect of the rock is and the harder it is to ascribe waves to its impact.

AI is essentially a rock thrown into a stormy sea of other processes and dynamics. Understanding the extent of the waves caused by AI (or any technology or practice) is a challenge. This may be part of the reason the industry has struggled to adopt a principled set of measurement and analytic frameworks for understanding the impact of AI.⁹

Our approach is specifically designed to be useful for these types of challenges. DORA is designed to understand the utility or disutility of a practice. We've explored the downstream impacts of myriad practices over the last 10 years, including security practices, transformational leadership, generative cultures, documentation practices, continuous integration, continuous delivery, and user-centricity.¹⁰

We believe that DORA's approach¹¹ can help us learn about AI's impact, especially as we explore the effects of AI across many outcomes.



Measuring AI adoption

The first challenge of capturing the impact of adopting AI is measuring the adoption of AI. We determined measuring usage frequency is likely not as meaningful as measuring reliance for understanding AI's centrality to development workflows. You might only do code reviews or write documentation a few times a month or every couple of months, but you see these tasks as critically important to your work.

Conversely, just because you use AI frequently does not mean that you are using AI for work that you consider important or central to your role.

Given this, we asked respondents about their reliance on AI in general and for particular tasks. The [previous chapter](#) details the survey results and their interpretation.

Using factor analysis, we found our “general” AI reliance survey item had high overlap with reported AI reliance on the following tasks:

- Code Writing
- Summarizing information
- Code explanation
- Code optimization
- Documentation
- Test writing

The strong commonality and covariance among these seven items suggests an underlying factor that we call AI adoption.

AI’s impact on individuals is a story of clear benefits (and some potential tradeoffs)

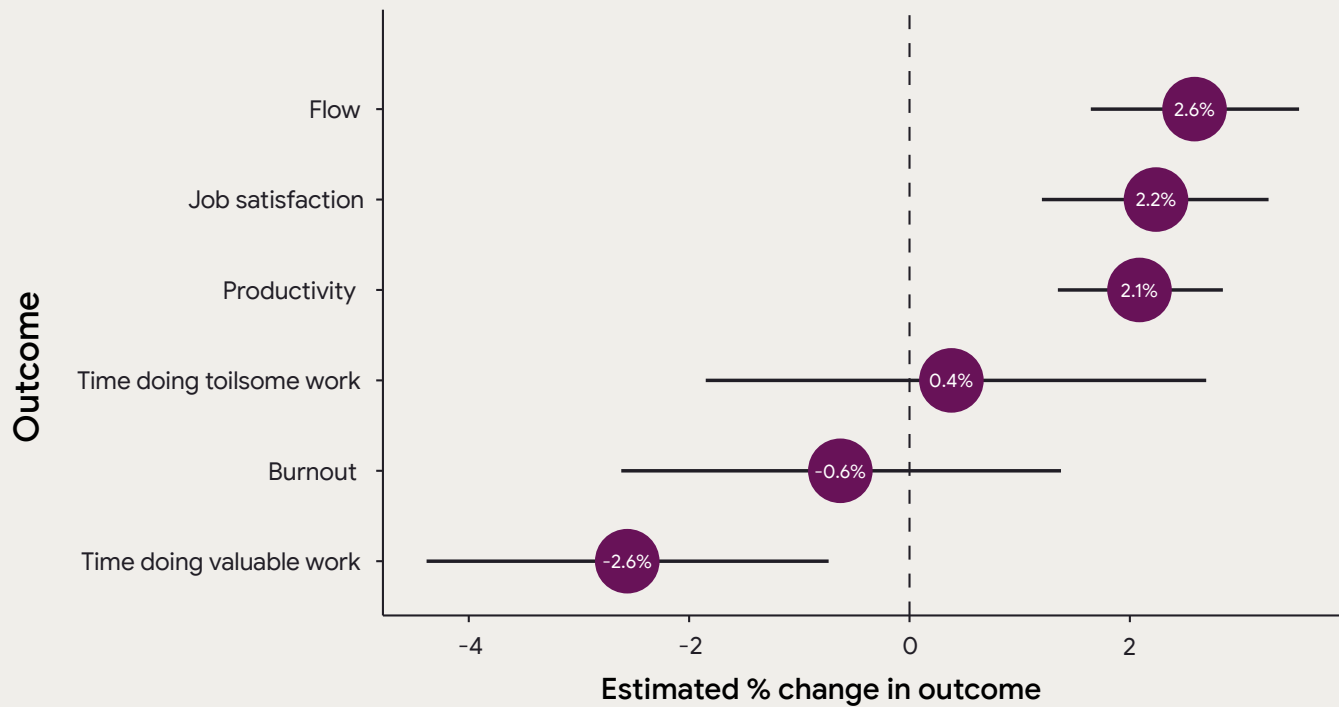
As we do every year, we measured a variety of constructs related to an individual’s success and well-being:

Job satisfaction	A single item designed to capture someone’s overall feeling about their job.
Burnout	A factor that encapsulates the multifaceted nature of burnout, encompassing its physical, emotional, and psychological dimensions, as well as its impact on personal life.
Flow	A single item designed to capture how much focus a person tends to achieve during development tasks.
Productivity	A factor score designed to measure the extent an individual feels effective and efficient in their work, creating value and achieving tasks.
Time doing toilsome work	A single item measuring the percentage of an individual’s time spent on repetitive, manual tasks that offer little long-term value.
Time doing valuable work	A single item measuring the percentage of an individual's time spent on tasks that they consider valuable.

We wanted to figure out if the way respondents answered these questions changes as a function of adopting AI. The results suggest that is often the case.

Figure 7 is a visualization that shows our best estimates about the impact of adopting AI on an individual’s success and well-being.

If an individual increases AI adoption by 25%...



Point = estimated value

Error bar = 89% uncertainty interval

Figure 7: Impacts of AI adoption on individual success and well-being

The clear benefits

The story about the benefit of adopting AI for individuals is largely favorable, but like any good story, has some wrinkles. **What seems clear is that AI has a substantial and beneficial impact on flow, productivity, and job satisfaction (see Figure 7).**

Productivity, for example, is likely to increase by approximately 2.1% when an individual's AI adoption is increased by 25% (see Figure 7). This might seem small, but this is at the individual-level. Imagine this pattern extended across tens of developers, or even tens of thousands of developers.

This pattern is what we expected. We believe it emerged in part thanks to AI's ability to synthesize disparate sources of information and give a highly personalized response in a single location. Doing this on your own takes time, lots of context switching, and is less likely to foster flow.

Given the strong connection that productivity and flow have with job satisfaction, it shouldn't be surprising that we see AI adoption leads to higher job satisfaction.

The potential tradeoffs

Here is where the story gets a little complicated. One value proposition for adopting AI is that it will help people spend more time doing valuable work. That is, by automating the manual, repetitive, toilsome tasks, we expect respondents will be free to use their time on “something better.” However, our data suggest that increased AI adoption may have the opposite effect—reducing reported time spent doing valuable work—while time spent on toilsome work appears to be unaffected.

Markers of respondents’ well-being, like flow, job satisfaction, and productivity have historically been associated with time spent doing valuable work. So, observed increases in these measures independently of decreases in time spent on valuable work are surprising.

A good explanation of these patterns will need to wrestle with this seeming incongruity. A good explanation of a movie cannot ignore a scene that contradicts the explanation. A good explanation of a book cannot ignore a chapter that doesn’t fit neatly into the explanation. Similarly, a good explanation of these patterns cannot just focus on a subset of the patterns that allows us to tell a simple story.

There are innumerable hypotheses that could fit the data, but we came up with a hypothesis that seems parsimonious with flow, productivity, and job satisfaction benefitting from AI while time spent doing valuable work decreases and toil remains unchanged.

We call our hypothesis the vacuum hypothesis. By increasing productivity and flow, AI is helping people work more efficiently. This efficiency is helping people finish up work they consider valuable faster.

This is where the vacuum is created; there is extra time. AI does not steal value from respondents’ work, it expedites its realization.



Wait, what is valuable work?

To make sense of these counterintuitive findings we explored more deeply what types of work respondents judge to be valuable or toilsome.

Traditional wisdom, our past reports, and qualitative data from our interviews suggest that respondents find development-related tasks, like coding, to be valuable work, while less-valuable, even toilsome, work typically includes tasks associated with organizational coordination, like attending meetings. Within this categorization scheme, AI is better poised to assist with “valuable” work than “toilsome” work, as defined by respondents.

We turned to qualitative data from our interviews and found that, when responding to the moderator’s question of whether or not they would consider their work “meaningful,” participants frequently measured the value of their work in relation to the impact of their work on others.

This is solidified by two years of past DORA evidence of the extremely beneficial impact of user-centricity on job satisfaction.

For example, when describing a recent role shift, P10¹² indicated making the decision because “It helps me impact more people. It helps me impact more things.” Similarly, P11 noted “if you build something from scratch and see it’s delivered to a consumer or customer, you can feel that achievement, you can say to yourself, ‘Yeah! I delivered this and people use that!’”

Understanding that the “meaningfulness” of development work is derived from the impact of the solution created—not directly from the writing of the code—helps explain why we observed respondents spending less time on valuable work, while also feeling more satisfied with their jobs.

While AI is making the tasks people consider valuable easier and faster, it isn't really helping with the tasks people don't enjoy. That this is happening while toil and burnout remain unchanged, obstinate in the face of AI adoption, highlights that AI hasn't cracked the code of helping us avoid the drudgery of meetings, bureaucracy, and many other toilsome tasks (Figure 8).

The good news is that AI hasn't made it worse, nor has it negatively affected respondents' well-being.

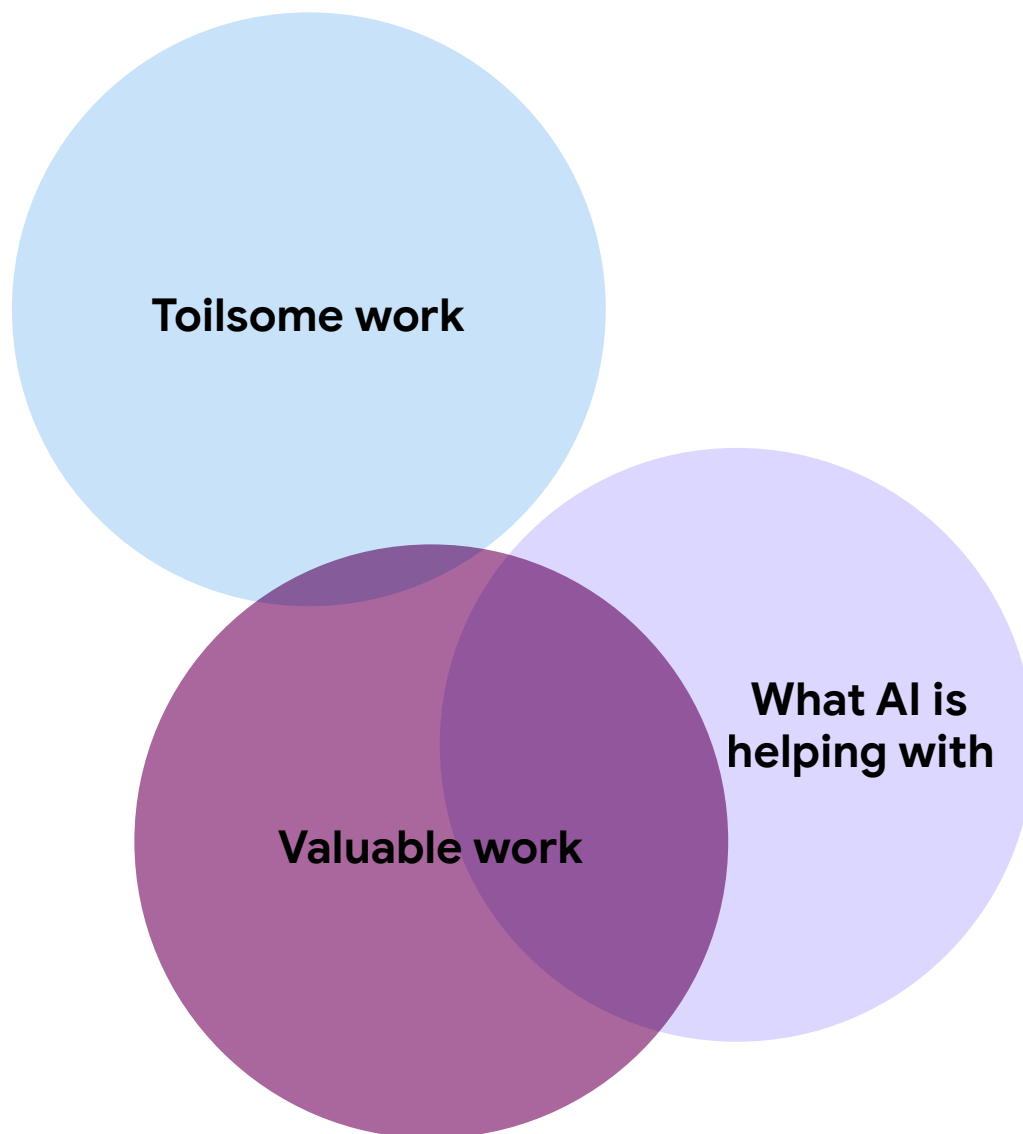


Figure 8: Not data, but a visualization of our hypothesis: AI is helping with our valuable work, but not helping us with our toil.

The promising impact of AI on development workflows

The last section explored outcomes focused on the individual. The next set of outcomes shift focus to explore processes, codebases, and team coordination. Here is a list of the outcomes we measured:

Code complexity	The degree to which code’s intricacy and sophistication hinders productivity.
Technical debt	The extent to which existing technical debt within the primary application or service has hindered productivity over the past six months.
Code review speed	The average time required to complete a code review for the primary application or service.
Approval speed	The typical duration from proposing a code change to receiving approval for production use in the primary application or service.
Cross-functional team (XFN) coordination	The level of agreement with the statement: "Over the last three months, I have been able to effectively collaborate with cross-functional team members."
Code quality	The level of satisfaction or dissatisfaction with the quality of code underlying the primary service or application in the last six months.
Documentation quality	The perception of internal documentation (manuals, readmes, code comments) in terms of its reliability, findability, updatedness, and ability to provide support.

As before, our goal here is to understand if these aspects seem to vary as a function of adopting AI. Figure 9 is a visualization that shows our best estimates of the change in these outcomes in relation to a 25% increase in AI adoption.

Overall, the patterns here suggest a very compelling story for AI. Here are the substantial results from this section.

A 25% increase in AI adoption is associated with a...

7.5% increase in documentation quality

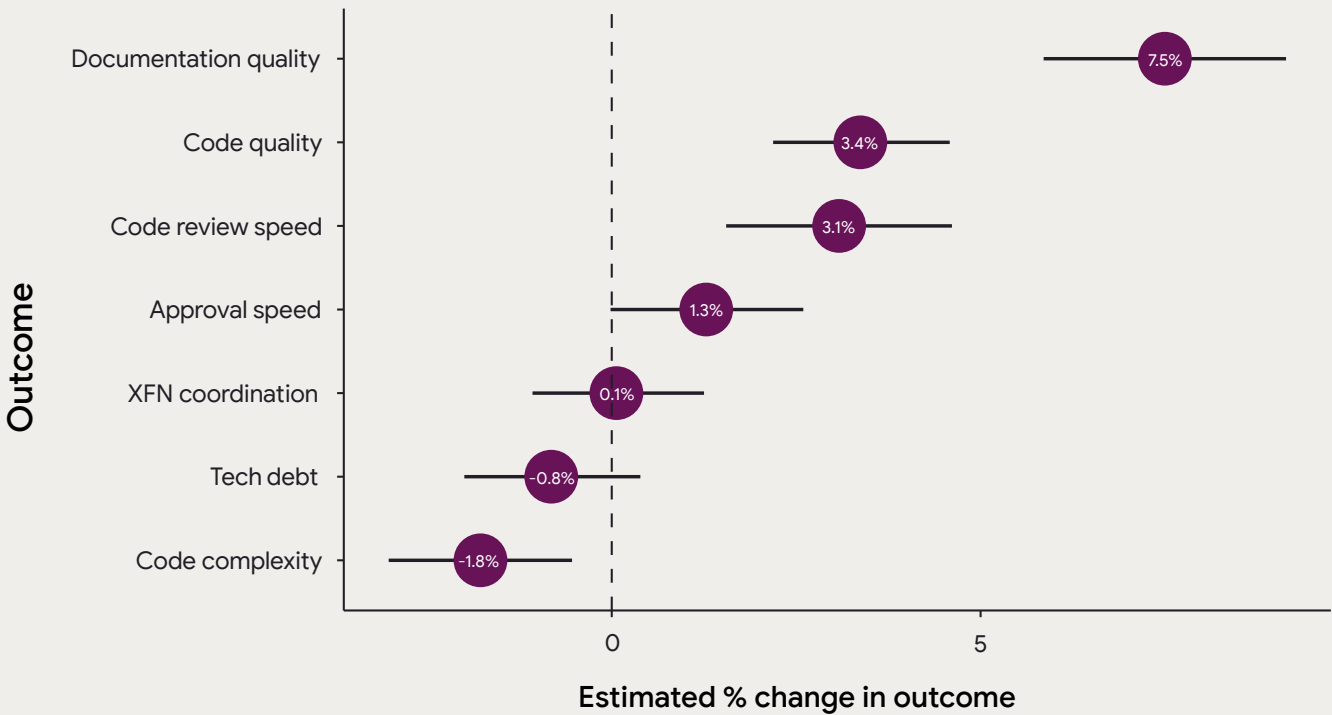
3.4% increase in code quality

3.1% increase in code review speed

1.3% increase in approval speed

1.8% decrease in code complexity

If AI adoption increases by 25%...



Point = estimated value

Error bar = 89% uncertainty interval

Figure 9: Impacts of AI adoption on organizations.

The data presented in the "[Artificial intelligence: Adoption and attitudes](#)" chapter show the most common use of AI is for writing code. 67% of respondents report that AI is helping them improve their code. Here, we see further confirmation of that sentiment. AI seems to improve code quality and reduce code complexity (Figure 9). When combined with some potential refactoring of old code, the high-quality, AI-generated code could lead to an overall better codebase. This codebase might be additionally improved by having better access to quality documentation, which people are using AI to generate (see [Artificial intelligence: Adoption and attitudes](#)).

Better code is easier to review and approve. Combined with AI-assisted code reviews, we can get faster reviews and approvals, a pattern that has clearly emerged in the data (Figure 9).

Of course, faster code reviews and approvals do not equate to better and more thorough code review processes and approval processes. It is possible that we're gaining speed through an over-reliance on AI for assisting in the process or trusting code generated by AI a bit too much. This finding is not at odds with the patterns in Figure 9, but it also not the obvious conclusion.

Further, it isn't obvious whether the quality of the code and the quality of the documentation are improving because AI is generating it or if AI has enhanced

our ability to get value from what would have otherwise been considered low-quality code and documentation. What if the threshold for what we consider quality code and documentation simply moves down a little bit when we're using AI because AI is powerful enough to help us make sense of it? These two ways of understanding these patterns are not mutually exclusive interpretations; both could be contributing to these patterns.

What seems clear in these patterns is that AI helps people get more from the documents they depend on and the codebases they work on. AI also helps reduce costly bottlenecks in the code review and approval process. What isn't obvious is how exactly AI is doing this and if these benefits lead to further downstream benefits, such as software delivery improvements.