

Adoption and Fragmentation of AI in CI/CD—and Operational Readiness

by [Guy Currier](#), Analyst, Futurum Group

SPONSORED BY



A Techstrong PulseMeter survey finds most organizations looking to expand AI across software delivery workflows but are encountering familiar operational barriers.

Research Insights

The AI conversation in software delivery tends to move at a different speed than the underlying reality. Vendors announce integrations, analysts publish adoption projections, and the assumption embedded in much of the public discourse is that the enterprise is actively operationalizing AI across their CI/CD pipelines.

We fielded a Techstrong PulseMeter poll among 148 individuals familiar with the use of AI in their organizations' CI/CD pipelines, and that data tells a more complicated story.

The dominant posture toward AI in CI/CD appears to be active interest running ahead of operational readiness. Most organizations are only at the stage of experimenting with AI or beginning to expand their use in selected areas. Far fewer have achieved anything resembling end-to-end AI integration across the software delivery lifecycle. Many are attempting to operationalize AI within environments that were already architecturally complex before AI entered the picture, and that complexity is the central variable that helps us understand the true state of this market.

Where we stand

Asked to characterize their current AI adoption in CI/CD, respondents divided roughly into thirds. About 35% report no active use of AI in the pipeline. Another 34% describe themselves as experimenting in isolated areas. A smaller group (12%) reports broad adoption across

Key Takeaways

- **AI adoption in CI/CD is real, but most organizations are still in early or transitional stages.** Experimentation is widespread; fully integrated AI-driven software delivery remains the exception.
- **Current AI use targets developer productivity.** Code generation and coding assistance dominate adoption. Deployment automation, monitoring, incident response, and security automation remain materially less mature.
- **Organizations plan to expand AI across the pipeline.** Future adoption plans span testing, operations, deployment, monitoring, and security, suggesting that AI is increasingly understood as an operational capability rather than just a developer tool.
- **Barriers to broader AI adoption are operational, not attitudinal.** Fragmented toolchains, integration complexity, governance concerns, and data silos are the primary constraints. The case for AI is widely accepted, of course, but the challenge now is engineering and organizational execution.
- **Organizations expect AI to improve delivery performance more than reduce costs.** Speed, software quality, testing efficiency, and reliability lead the expected outcomes, positioning AI as a capability investment rather than a cost-cutting exercise.

multiple pipeline stages, while 11% describe limited production use in specific areas and 8% say AI is fully integrated across the CI/CD lifecycle.

The aggregate figures also obscure something worth slowing down to examine: even “experimentation” covers a wide range of organizational commitment and readiness. The number that stands out sits at the far end of the adoption spectrum. Only one in twelve respondents works in an environment where AI is truly woven into the delivery process. That reflects current operational maturity—not intent—and the gap between the two is where the most consequential questions reside.



Part of what makes that gap difficult to close is the infrastructure organizations are working with. The poll asked respondents to characterize their CI/CD environments, and the results suggest that most software delivery architectures themselves are in transition. Thirty-two percent describe hybrid environments spanning multiple CI/CD platforms and toolchains. Another 23% are running Jenkins-centered setups with modern tooling layered on top; 17% remain primarily legacy or Jenkins-based. Only 11% describe a fully integrated, platform-driven architecture.

Most organizations, in other words, are trying to operationalize AI within delivery environments that still reflect fragmented tooling, inconsistent integration, and years of accumulated architectural

decisions. The data suggests this is not coincidental. Fragmented delivery infrastructure creates exactly the kind of operational complexity that makes AI harder to deploy, govern, and scale.

Where AI is being used and where it is not

Among respondents who are using AI in their pipelines, adoption clusters heavily at the developer end of the delivery lifecycle. Sixty-four percent use AI for code generation or developer assistance, by far the most widely adopted use case. Automated testing and test generation follow at 38%, with build and pipeline optimization at 34%.

Adoption drops off sharply when the use cases shift toward operational workflows. Just 27% use AI for deployment automation. Security and compliance automation sits at 22%. These are not marginal capabilities; they represent significant portions of the software delivery risk and cost surface. Their relatively low adoption reflects where organizations currently feel confident trusting AI with consequential decisions, and where they do not.

Poll respondents selected just over two current AI use cases. That is a small number, and it says something useful. Most organizations have found one or two areas where AI is tractable enough to use in production. Threading it reliably through the delivery process as a whole is a different, harder undertaking.

Future plans look materially different. Asked which areas they are considering for AI, respondents spread their answers considerably more broadly: 57% for code generation, 54% for automated testing, 51% for monitoring and incident response, 50% for build and pipeline optimization, 45% for deployment automation, and 41% for security and compliance automation. The average number of areas selected rose to more than two and a half. Organizations are not planning to stop at developer assistance. Translating those plans into operational reality will require confronting the same barriers that have limited broader adoption so far.

Where organizations expect AI to deliver and what's standing in the way

When asked which AI use cases are delivering or expected to deliver measurable value, respondents ranked improvements in delivery performance ahead of cost reduction. Sixty-five percent expect faster development cycles; 62% expect improved code quality; 53% expect reduced testing effort; and 50% expect improved reliability and fewer incidents. Reduced operational costs, often the primary metric foregrounded in AI business cases, ranked last among the options at 44%.

The relative ordering is noteworthy, suggesting that organizations are evaluating AI in CI/CD primarily as a capability investment, something that improves how software gets built and released rather than as a mechanism for cutting operational spend. This distinction matters because it changes how teams should frame AI ROI and how executives should measure whether the work is succeeding.

The barriers to realizing that value are, on the evidence, largely structural. Asked what is limiting progress with AI in CI/CD, respondents pointed most often to security and governance concerns (33%), followed by fragmented toolchains and lack of integration (30% each). Unclear ROI was cited by 26%; data silos and data quality concerns each appeared in roughly a quarter of responses. Skill gaps, workflow integration difficulty, and performance and reliability concerns each registered in the low 20s. Organizational resistance to change was cited by 14%, and 17% say nothing is currently limiting their progress.

Notably absent from that list: any version of “we do not believe AI will be useful.” Interest in AI is evident. The constraint is operational. Getting AI to function reliably across a delivery environment is not a motivational but an organizational and infrastructural problem.

Where we go next

Despite those barriers, the market is not pausing to reassess. Asked how they expect AI use in CI/CD to change over the next 12 to 18 months, 35% plan

to expand into selected additional areas, 20% plan to scale across multiple pipeline stages, and 17% expect AI to become core to their delivery strategy. Only 7% report no plans to adopt AI.

Most organizations, by this reading, expect to keep advancing despite the friction. Less clear from the data is whether the infrastructural work required to support that advancement—including toolchain consolidation, integration improvement, and governance framework development—is receiving equivalent attention. The poll findings suggest it should be. Urgently, but not hastily.



Analyst View

We are not describing a market that is confused about AI's potential role in software delivery. The PulseMeter poll reveals a market that understands that potential reasonably well but is struggling with something considerably harder: operationalizing AI inside environments built before AI was a practical consideration.

The specificity of the constraint matters here. The barriers respondents cite—including fragmented toolchains, integration complexity, governance concerns, and data silos—are not abstract organizational challenges. They are characteristics of how enterprise software delivery was constructed over the past decade, often under pressure, across acquisitions, platforms, and team structures that accumulated without a unifying architecture. AI does not make those characteristics

disappear. In some respects, it makes them more consequential: AI models operating on inconsistent or siloed data tend to produce inconsistent or unreliable outputs. The quality of an AI integration is limited by the quality of the environment in which it operates.

The concentration of current AI adoption in developer productivity use cases also reflects something real about organizational risk tolerance. Code generation and developer assistance are high-feedback, low-blast-radius applications of AI. A developer can evaluate a code suggestion in seconds; a bad suggestion takes seconds to discard. The same logic does not hold for deployment automation or security governance workflows, where the feedback loop is longer and the consequences of errors are more significant. Why would a platform team hand over higher-order

operational judgment before the surrounding data, controls, and escalation paths are ready?

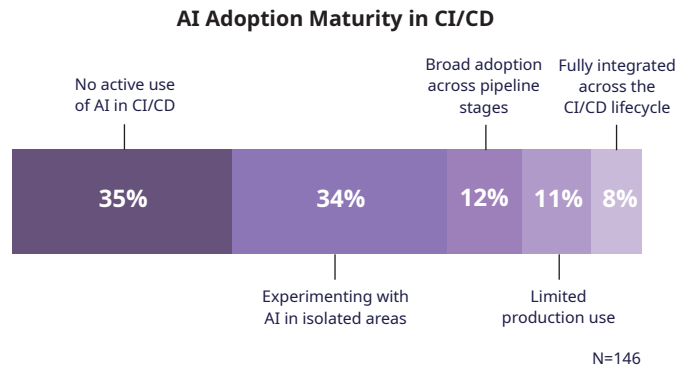
The next phase of AI adoption in CI/CD will likely depend less on organizational intent than on organizational infrastructure. Governance frameworks that can span fragmented delivery environments, integration architectures that give AI models access to consistent pipeline data, and platform strategies that reduce toolchain complexity are the investments that will determine how effectively organizations can scale beyond the developer productivity use cases where most adoption currently lives. The trajectory from our findings is reasonably clear. The path is considerably less so. It's not whether AI belongs in software delivery but whether delivery environments are ready for AI.

Recommendations

- **Be deliberate in AI expansion across the pipeline.** Many organizations are successfully beginning with lower-risk developer productivity use cases, such as code generation and testing assistance. But most recognize AI's potential for deployment, monitoring, operations, or security workflows. Subject to the recommendations that follow, if you have already stepped into AI adoption in your pipeline, think expansively, but with operational discipline.
- **Address fragmentation before scaling AI operationally.** Multiple toolchains, isolated or ad hoc integrations, and siloed workflows increase operational complexity and limit AI effectiveness, especially at production scale. Organizations should simplify delivery architectures and improve platform consistency as a prerequisite to broad AI operationalization.
- **Prioritize operational improvements alongside AI adoption.** Expanding AI use in CI/CD will require more than adding AI features to isolated tools. Organizations should evaluate the overall maturity of their pipeline with equal attention, including integration, orchestration, governance, and workflow consistency across the software delivery lifecycle.
- **Use measurable delivery outcomes as metrics.** Organizations most often expect to quantify AI initiatives by improvements in delivery speed and software quality, as well as by testing efficiency, operational reliability, and workflow effectiveness. The market has moved beyond treating AI adoption itself as the primary objective; you can instead focus on outcomes that matter most to your organization (not just efficiency or cost improvements).
- **Strengthen governance, workflow consistency, and delivery visibility from the outset.** As AI becomes more integrated into software delivery operations, organizations will need stronger governance models, better cross-pipeline visibility, clearer operational controls, and more consistent workflows to ensure AI can operate safely and effectively at scale.

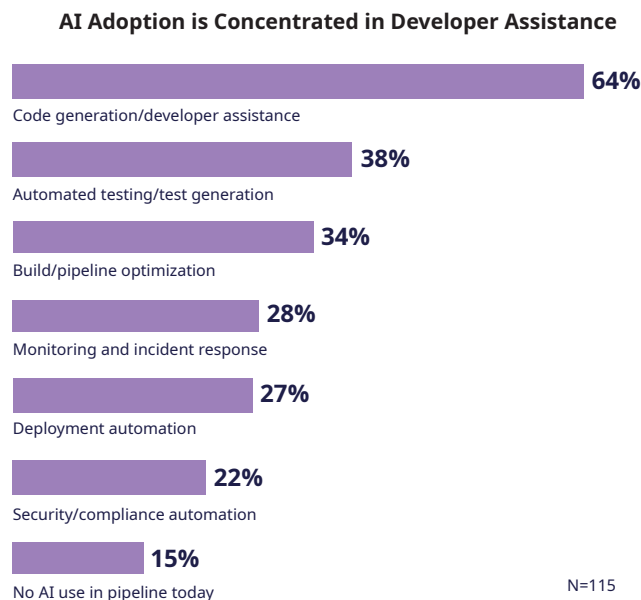
Research Results

How would you describe your organization's current level of AI adoption within your CI/CD pipeline?



The distribution of AI adoption in CI/CD looks less like an adoption curve and more like a holding pattern. Among qualified respondents, 35% report no active AI use and 34% describe isolated experimentation, meaning roughly seven in ten organizations have not moved AI into production delivery workflows in any meaningful way. At the far end, just 8% describe AI as fully integrated across the CI/CD lifecycle. The middle of the distribution (12% reporting broad adoption across multiple stages, 11% reporting limited production use in specific areas) represents the territory most organizations are trying to reach. Most are not there yet.

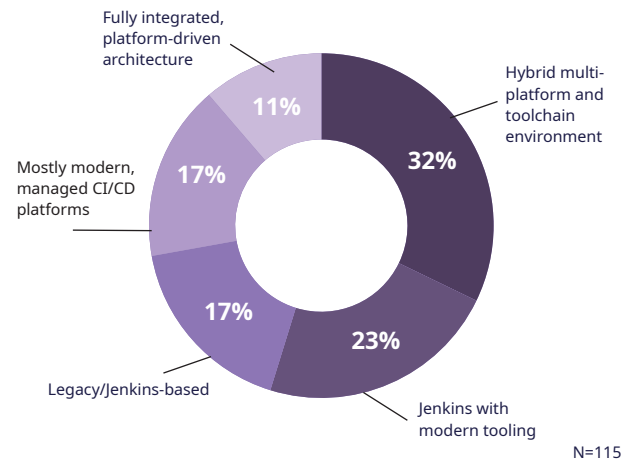
In which areas of your software delivery pipeline are you currently using AI? (Select all that apply)



Code generation and developer assistance are the most widely adopted AI use cases, cited by 64% of respondents. Automated testing follows at 38% and build and pipeline optimization at 34%. Deployment automation (27%), monitoring and incident response (28%), and security and compliance automation (22%) trail substantially. The pattern suggests organizations are deploying AI first in lower-risk, developer-adjacent workflows with shorter feedback loops. On average, respondents selected just over two current use cases, a measure of how selective and concentrated current adoption remains.

How would you describe your current CI/CD environment?

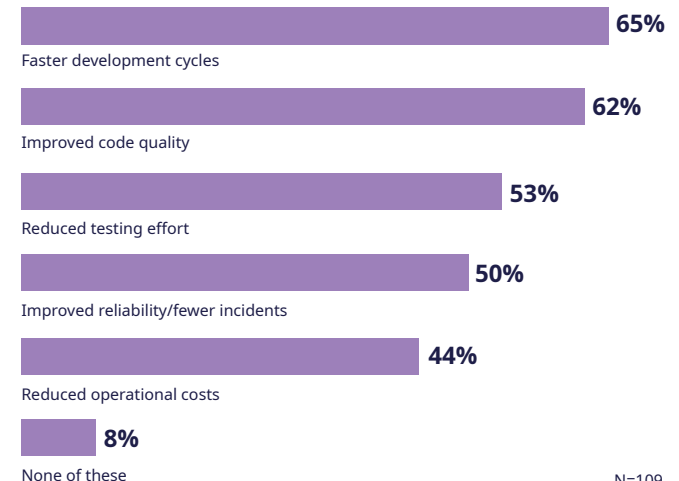
CI/CD Environments Remain Highly Fragmented



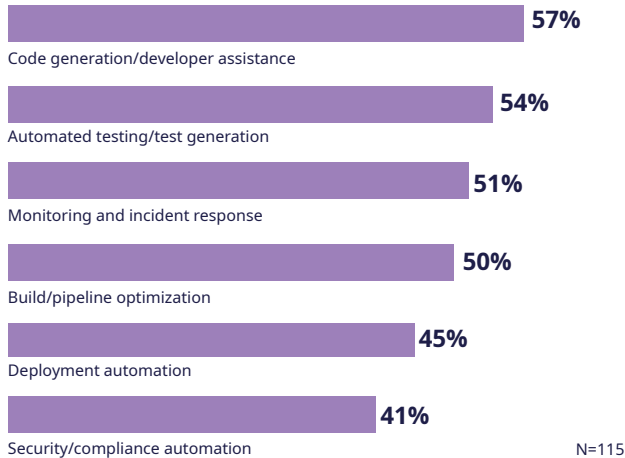
Software delivery environments are themselves works in progress. Thirty-two percent of respondents describe hybrid, multi-platform CI/CD environments; another 23% run Jenkins-centered setups with modern tooling layered on top; 17% remain primarily legacy or Jenkins-based with custom integrations. Only 11% describe a fully integrated, platform-driven architecture. The cumulative picture is one of organizations managing multiple platforms, toolchains, and integrations simultaneously, an environment that creates operational complexity and limits the consistency that effective AI deployment requires.

Which AI use cases are delivering, or do you expect to deliver measurable value in your CI/CD pipeline? (Select up to three)

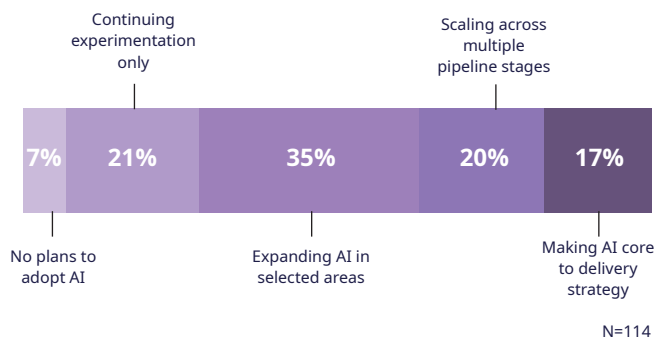
Organizations Prioritize Speed, Quality, and Testing Efficiency



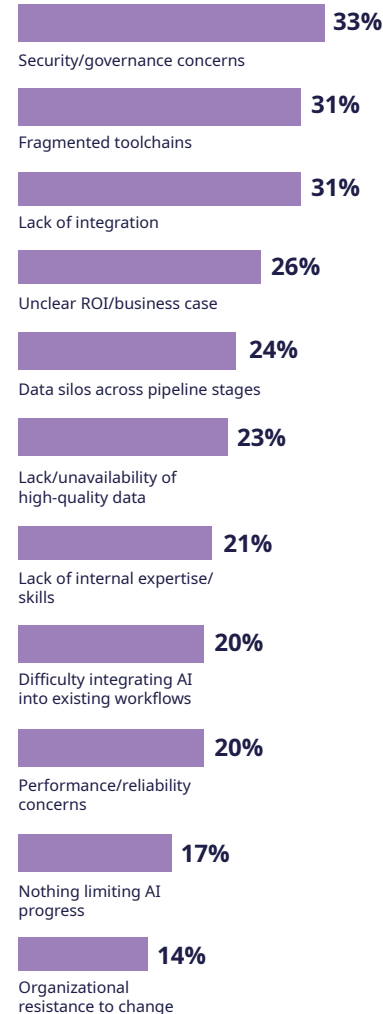
Respondents ranked delivery performance improvements ahead of cost reduction as their primary expected AI outcomes. 65% expect faster development cycles; 62% expect improved code quality; 53% expect reduced testing effort; and 50% expect improved reliability and fewer incidents. Reduced operational costs, often the metric most prominently cited in AI business case discussions, ranked last among the options at 44%. The ordering suggests organizations are evaluating AI in CI/CD to improve how software is built, not primarily as a mechanism to cut operational spend.

In which areas of your software delivery pipeline are you considering using AI? (Select all that apply)**Organizations Expect AI Adoption to Expand Across the Pipeline**

Future adoption plans are considerably more distributed than current use. Respondents plan to expand AI into code generation (57%), automated testing (54%), monitoring and incident response (51%), build and pipeline optimization (50%), deployment automation (45%), and security and compliance automation (41%). The average number of areas selected rose to more than two and a half. Organizations are not planning to stop at developer assistance. Closing the gap between current and planned adoption is the central operational challenge that these findings surface.

How do you expect your use of AI in CI/CD pipelines to change over the next 12-18 months?**Most Organizations Expect AI Use in CI/CD to Grow**

Most organizations anticipate continued AI expansion over the next 12 to 18 months. Thirty-five percent plan to expand into selected additional areas, 20% plan to scale across multiple pipeline stages, and 17% expect AI to become core to their delivery strategy. 21% expect to continue experimenting at current levels. Only 7% report no plans to adopt AI. The data suggests the market has moved past questioning whether AI belongs in software delivery. The question now is operational: how to advance AI adoption safely, effectively, and at a pace that organizational infrastructure can sustain.

What, if anything, is limiting your organization's progress with AI in CI/CD? (Select up to three)**Security, Fragmentation, and Integration are Limiting AI Progress**

Security and governance concerns lead the list of barriers at 33%, followed by fragmented toolchains and lack of integration (31% each). Unclear ROI was cited by 26%; data silos and data quality concerns each appeared in roughly a quarter of responses. Skill gaps, workflow integration difficulty, and performance and reliability concerns each registered in the low 20s. Organizational resistance to change was cited by 14%; 17% report no significant barriers to their current progress. The results frame AI adoption in CI/CD as primarily a systems integration and operational readiness problem, not a motivation or awareness problem.



This Techstrong PulseMeter is sponsored by CloudBees.
To learn more about CloudBees, visit cloudbees.com.