

DX

Leadership in AI Assisted Engineering

LeadingEng

NEW YORK

The impact of GenAI

Sundar Pichai says AI is making Google engineers 10% more productive. Here's how it measures that.

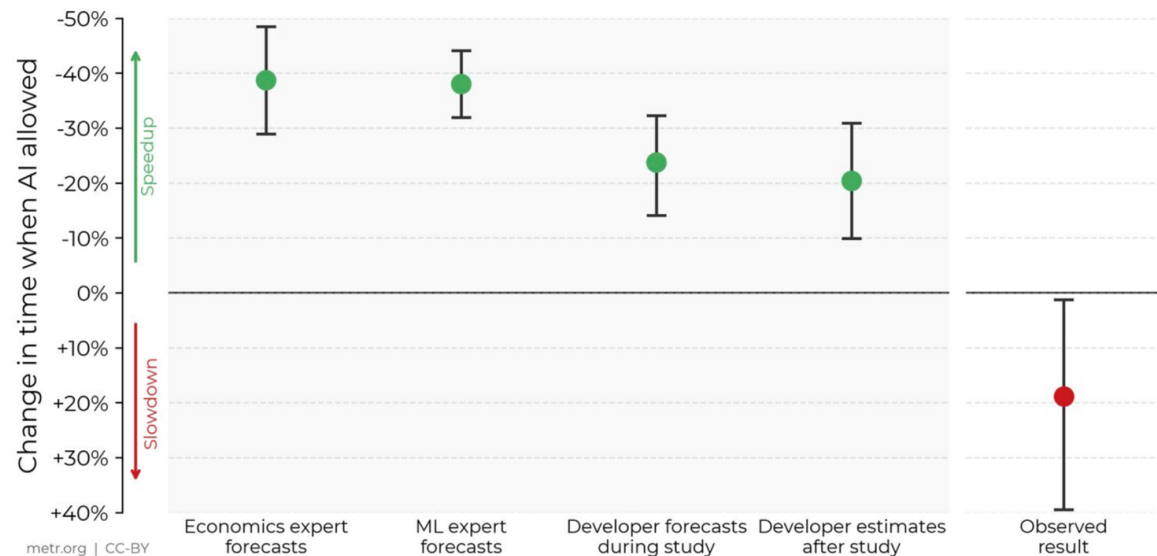
By [Hugh Langley](#)



Against Expert Forecasts and Developer Self-Reports, Early-2025 AI Slows Down Experienced Open-Source Developers



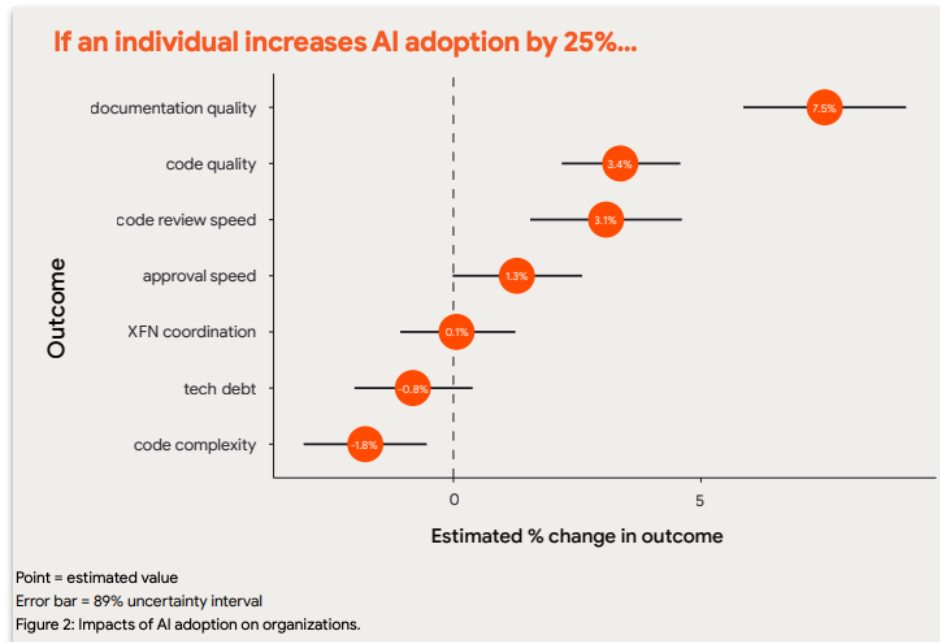
In this RCT, 16 developers with moderate AI experience complete 246 tasks in large and complex projects on which they have an average of 5 years of prior experience.



GenAI is significantly impacting development

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

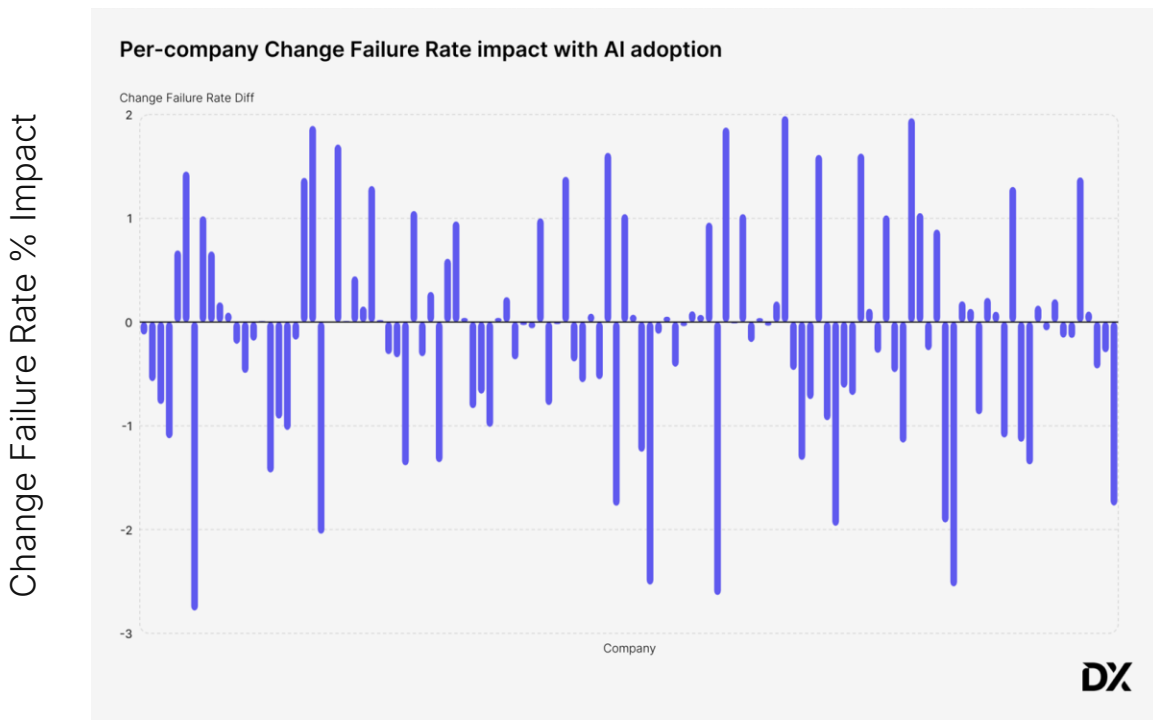
- 7.5% increase in documentation quality
- 3.4% increase in code quality
- 3.1% increase in code review speed
- 1.3% increase in approval speed
- 1.8% decrease in code complexity



Industry averages hide the big picture of quality impact

Per-company Change Failure Rate% impact for AI Users vs. Non-AI Users

Data from 61 companies



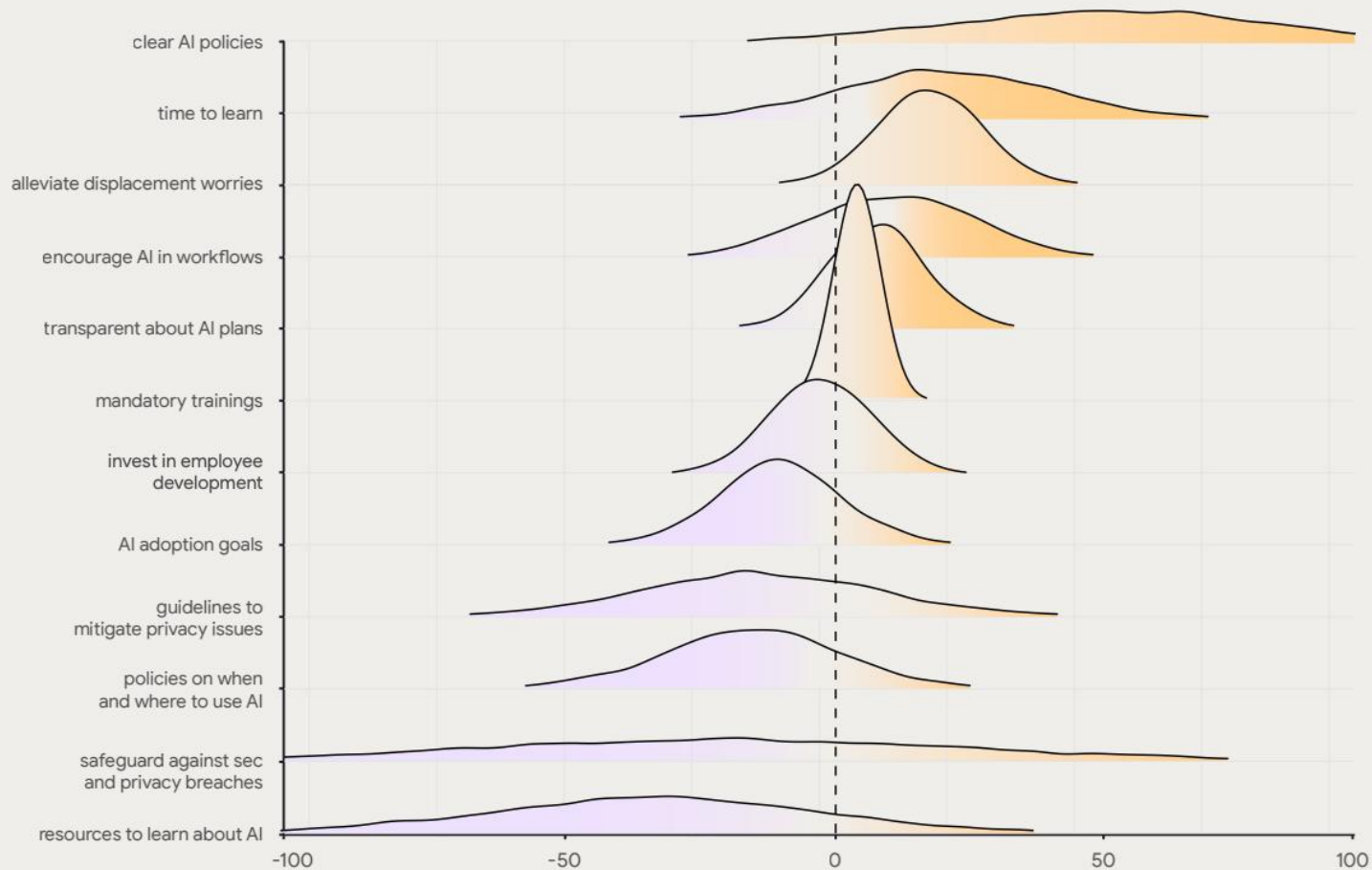
GenAI ROI not evenly distributed

Lack of education and enablement on best practices and use cases.

Organizations often just “turn on” the tech and **expect users to be proficient.**

Difficulty measuring the impact and knowing what to measure.

Strategy



Bayesian Posterior Distributions of AI Adoption Strategies: <https://dora.dev/research/ai/gen-ai-report/>

Top Distribution Analysis

High Likelihood (80-95%)

Clear AI policies: ~90%

Time to learn: ~85%

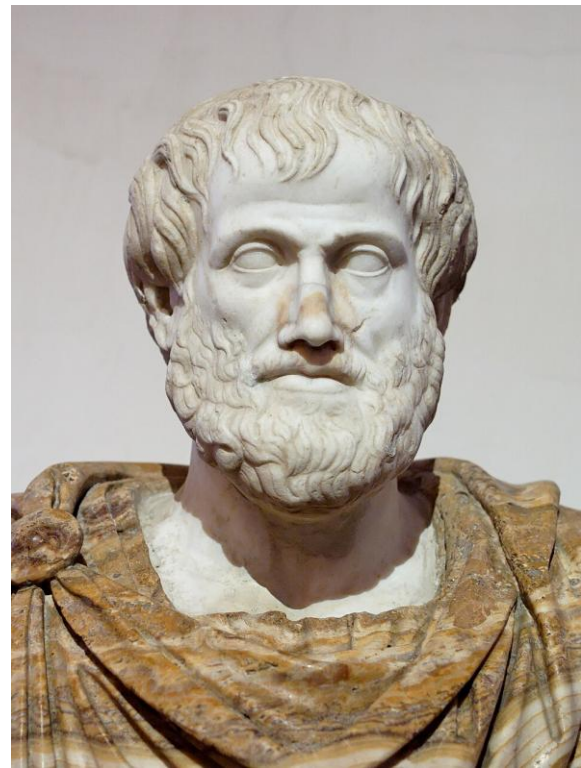
Alleviate displacement worries: ~80%

As leaders, we should drive...

Reducing fear of AI

Google's Project Aristotle

- Assumed that the **recipe for a successful team** would be a combination of **high performers, an experienced manager and unlimited free resources.**
- They were wrong. **The high performing teams were the ones with greatest psychological safety.**



Transparency – Set Clear Intent and Expectations

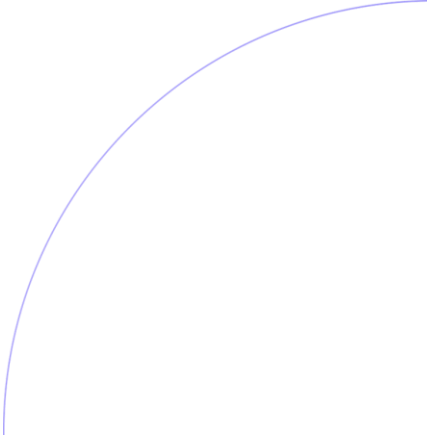
Explain the “why”: Frame GenAI adoption as a way to **augment, not replace, engineers’ skills**. Show data and case studies illustrating benefits and pitfalls.

Address fears directly: Proactively acknowledge concerns about job security, performance monitoring, or AI errors, clarify boundaries on how AI outputs will be evaluated and used.

Open metrics discussion: If you’ll measure AI usage or impact, share exactly what’s being tracked and why, avoiding “black box” surveillance.

As leaders, we should drive...

Open metrics discussions



Measuring GenAI impact

Metric Type	Good For	Not Good For	Challenges
Telemetry metrics	Measuring impact on developer output	Quantifying ROI Understanding how tools are being used	Limited, possibly inaccurate insight Incomplete story
Experience sampling	Quantifying ROI Identifying best use cases	Collecting large amounts of data at once	Difficult to set up Must be run over period of time
Self-reported	Measuring adoption, developer satisfaction, productivity	Quantifying ROI	Can only be run periodically Participation rates

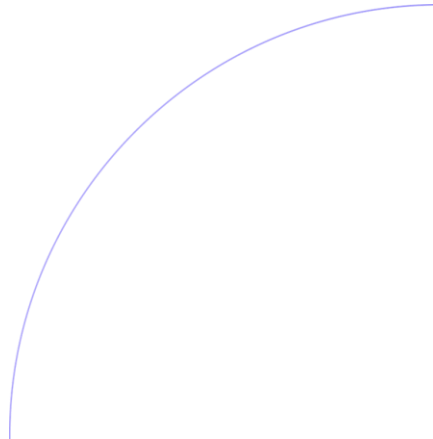
Table 1: DX AI Measurement Framework

* Metrics for autonomous AI agents

Utilization	Impact	Cost
How much are developers adopting and utilizing AI tools?	How is AI impacting engineering productivity?	Is our AI spend and return on investment optimal?
- AI tool usage (DAUs/WAUs) - Percentage of PRs that are AI-assisted - Percentage of committed code that is AI-generated - Tasks assigned to agents *	- AI-driven time savings (dev hours/week) - Developer satisfaction - DX Core 4 metrics, including: - PR throughput - Perceived rate of delivery - Developer Experience Index (DXI) - Code maintainability - Change confidence - Change fail percentage - Human-equivalent hours (HEH) of work completed by agents *	- AI spend (both total and per developer) - Net time gain per developer (time savings - AI spend) - Agent hourly rate (HEH / AI spend) *



**Foundational DevEx and DevProd
metrics still matter the most.**



As leaders, we should drive...

Compliance and trust

System prompt update loop

Most enterprise AI solutions will allow you to update the “system prompt” that underlies every prompt sent to the assistant. This is almost like templating, in that the rules contained in this prompt will be applied to every prompt sent into the assistant.

When the model creates inaccurate or suboptimal output, in many cases that output can be corrected going forward by changing the system prompt.

Example usage

[Include Simple System Prompt]

You have been observed making the following errors:

1. Proposing outdated Spring Boot versions (older than 2.6).
2. Suggesting or using deprecated methods in the java.* libraries.
3. Returning code snippets containing syntax errors such as missing parentheses or braces.

Your new rules:

- Always provide code snippets that use Spring Boot 3.x or newer, unless the user explicitly requests otherwise.
- Verify that you are not using deprecated methods from the Java standard library (check the current Java LTS for deprecation).
- Double-check that any code snippets are syntactically valid (balanced parentheses, braces, etc.).

When responding:

- Provide relevant explanations for your code choices.
- If uncertain, indicate possible methods or approaches rather than returning an incorrect snippet.
- Do not include references to internal or proprietary APIs beyond standard library or Spring Boot dependencies.



Guardrails – Protect Quality and Trust

Clear validation steps: Require human review of AI outputs in critical areas to ensure code quality and compliance.

Bias and security training: Teach engineers how to spot hallucinations, data leaks, or biased outputs in AI-generated code or documentation.

Feedback loops: Maintain open channels (Slack channels, retros, office hours) for reporting AI-related issues without fear of blame.

As leaders, we should drive...

Employee Success

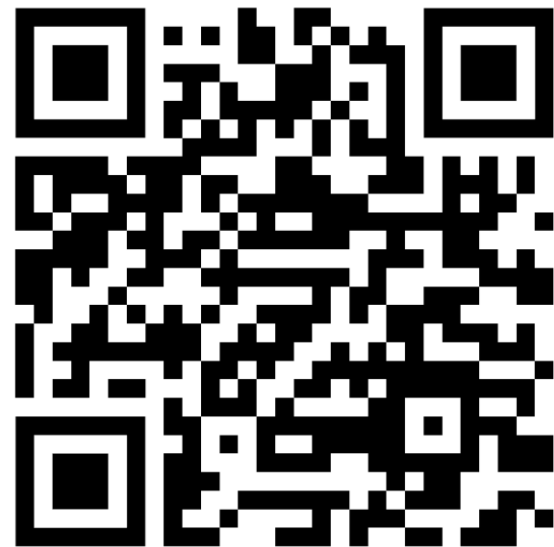


Provide both education and adequate time to learn.

GUIDE

DX

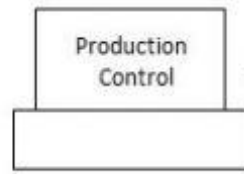
Guide to AI Assisted Engineering



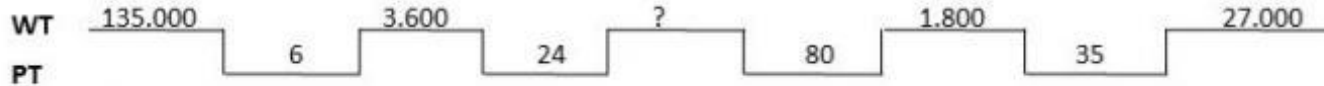
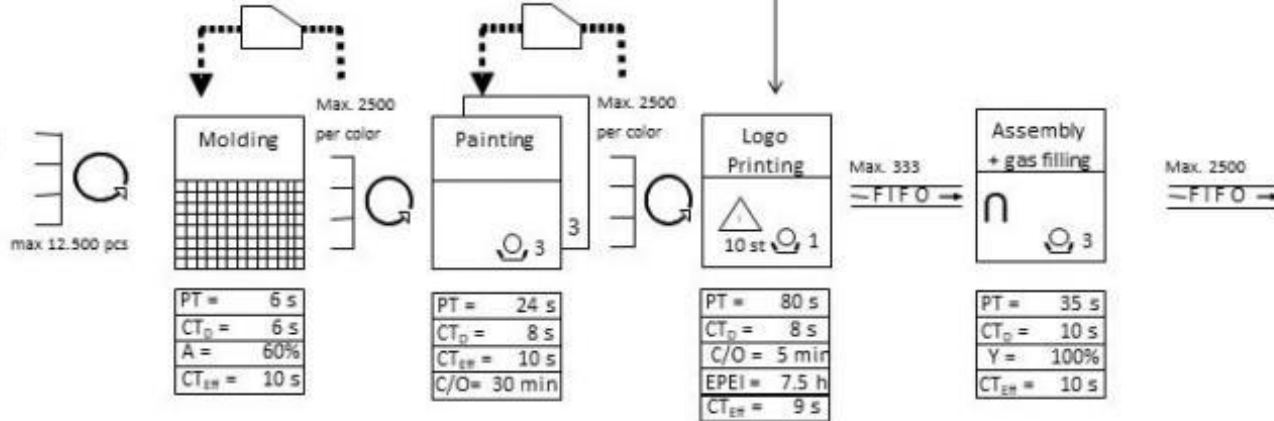
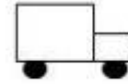
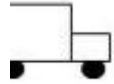
As leaders, we should drive...

Integration across SDLC

"An hour saved on something that isn't the bottleneck is worthless" - Eli Goldratt



**Find the bottleneck.
Fix the bottleneck.**



WT: 144.000 sec
PT: 151 sec
PLT: 144.151 sec

Faire - Automated Code Review

"Fairey" triggers off of GitHub PR and provides always-on review

Solution pulls context, surrounding code, and documentation, and provides PR comments

Completes roughly 3,000 reviews a week

Spotify - Incident Management

Internal incident management agent platform drafts remediation steps directly into SRE channels

Monitors logs, correlates alerts, and suggests runbook steps

Currently handling 90% of incidents at Spotify

Next Steps

Distribute the AI guide as a reference for integrating AI into your development workflows

Determine a method for measuring and evaluating GenAI impact

Track and measure AI adoption and iterate on best practices and use cases

Running data-driven evaluations of AI engineering tools

Nov 13, 9:00am PT / 6:00pm CET

AI tools are one of the biggest bets engineering leaders are making, but choosing the right ones requires more than a trial run. A successful POC depends on clear goals, structured evaluation, and the right success metrics. Join DX CEO Abi Noda and CTO Laura Tacho as they share how to run AI tool evaluations that deliver actionable insights and lasting impact.

In this session, you'll learn:

- How to choose the right AI tools to evaluate
- Which developers should be included in the evaluation
- What to measure in order to accurately evaluate tool performance

Speakers



Abi Noda
CEO, DX



Laura Tacho
CTO, DX



Register Here!