



McKinsey: 32% of Companies Killed a Software Purchase Because Coding Agents Could Build It

Cancelling a purchase order is not the same as saving money, and the 2026 numbers show where the difference went. Nearly a third of organizations skipped a software buy because their coding agents could build it instead.

A third of organizations cancelled a purchase they could build in-house

[McKinsey's State of AI: Global Survey 2026](#), published 25 August 2026, ran fieldwork from 4 May to 8 June with 1,719 respondents across 97 countries. The finding that got the headlines: 32% report their organization decided against purchasing at least one software product or feature because they could build the functionality in-house using agentic coding tools.



McKinsey: 32% of Companies Killed a Software Purchase Because Coding Agents Could Build It

The distribution matters more than the headline. Among the roughly 6% of respondents McKinsey classifies as AI high performers (those attributing at least 5% of EBIT to AI), [nearly half skipped at least one purchase](#), against 31% of everyone else. By sector: technology 41%, healthcare 39%, professional services 38%, energy and materials 38%, financial institutions 36%. Scaling is concentrating upward too. Firms above \$1B revenue moved from 27% to 40% on scaling at least one agentic system, while smaller firms stayed flat at 22%.

So the behaviour is real, measurable, and unevenly distributed. The companies most likely to cancel a renewal are the ones with the deepest engineering benches and the most existing AI infrastructure. That is a selection effect rather than a market signal about what a 300-person company should do next quarter.

Two other numbers from the same survey sit awkwardly next to the headline. 37% of respondents report AI contributed positively to EBIT, essentially unchanged from 2025, while around 80% report individual productivity gains. And one in five say their organization is limiting AI use because of operating costs; for coding agents specifically, about one in ten say use is constrained by cost.

That last figure is the one I would put on the first slide of any board deck about this. Roughly ten percent of organizations already hit a run-cost ceiling on the tool they are using to justify cancelling purchases.

McKinsey predicted a \$35 to 40 billion reallocation two years ago

McKinsey's own 2024 analysis anticipated something close to this shape, projecting generative AI would add 1 to 3 percentage points to total SaaS churn and shift 2 to 4 percentage points of spend from buy to build over three to four years, roughly [\\$35 to 40 billion reallocated](#). A few points of reallocation is meaningful for specific vendors in specific categories and close to invisible in any top-line software spend number.

Independent survey data agrees on direction. [Retool's Build vs Buy Report 2026](#) (817 respondents, published February 2026) found 35% have already replaced at least one SaaS tool with a custom build and 78% expect to build more of their own tools this year. Two different samples, two different methodologies, roughly the same third of the market.



! **The number that gets left out**

McKinsey's [agent economics guide](#) estimates \$20,000 to \$30,000 annual run cost for a single-agent workflow at some banks, and \$100,000 to \$200,000 for a multiagent team, driven by fixed infrastructure and orchestration. That is a per-workflow figure, not per-company. A team that replaces four mid-tier SaaS subscriptions with four agentic workflows has not returned to zero.

Code smells, failed security tests, and PII in places it was not before

The cancelled subscription is a clean, visible line in a budget. What replaces it is not clean, and it mostly stays invisible for two or three quarters.

An empirical study published on arXiv in 2026, "Debt Behind the AI Boom", analysed 302,600 verified AI-authored commits across 6,299 GitHub repositories and found 484,366 distinct introduced issues, 89.3% of them code smells. Over 15% of AI commits introduced at least one issue, and 22.7% of tracked issues were still present in the latest version of the repository.

Security looks worse. [Veracode's 2026 testing across more than 150 LLMs](#) found only 55% of AI-generated code passed basic security tests. Security debt now affects 82% of organizations, up from 74%, with high-risk vulnerabilities up 36% year over year. Apiiro's deep code analysis across dozens of enterprises found a 3x surge in repositories containing PII and payment data, and a 10x increase in APIs missing authorization and input validation.

Read those findings together with the build-vs-buy shift and a specific risk profile appears. The functionality most commonly cancelled and rebuilt (internal tools, workflow apps, admin panels, integration layers) is exactly the category that touches customer records and payment data, and exactly the category where an authorization check is one line nobody notices missing. A vendor's compliance posture is a poor substitute for engineering judgment, but it is not nothing, and when you cancel the vendor you cancel that too.

I have written before about how this shows up in the commit history: [refactoring rates fall while duplication climbs](#). The code arrives faster and it consolidates less.



The same survey that reports the shift also refutes the savings story

My take: the 32% figure is being read as evidence that agentic coding pays for itself, and the same survey contains the refutation. AI EBIT impact is flat at 37% year over year while 80% report individual productivity gains. If the build shift were converting into margin, that gap should be closing. It is not.

What I think is actually happening: teams are moving spend from a vendor line item finance tracks to an engineering capacity line item finance does not track per application. The saving is real on the day the renewal is cancelled and unproven twelve months later. [BERI put it more bluntly](#): your engineering team can now build the thing you were about to renew, and that is not the same as saving the money.

Three questions I would ask before cancelling the next renewal

The framework I use with clients has three questions, and none of them are about whether the agent can produce working code. Assume it can.

First: is this a differentiating capability or a commodity? If a competitor buying the same SaaS product gains nothing on you, you gain nothing by rebuilding it. The 41% figure in technology is partly companies rebuilding things adjacent to their actual product, where the marginal cost of maintenance is low because the team is already there. A logistics firm rebuilding its expense tool does not have that.

Second: what is the run-cost line? BERI recommends budgeting 15 to 25% of build cost per year for internally built systems. I would use that as a floor rather than a midpoint for anything with an agentic component in production, because McKinsey's own figures put fixed orchestration infrastructure at \$20,000 to \$30,000 per single-agent workflow annually before you count a single engineer-hour of on-call. Compare that number against the renewal, not the build estimate against the renewal.

Third: who owns it in eighteen months? Vendor software has a support contract and a



roadmap. Internal software has whoever built it, until they leave. **The 22.7% of AI-introduced issues still sitting in the latest repository version is what unowned code looks like measured at scale.**

There is also a sequencing point here. The gap between experimenting with agents and actually scaling them is where run cost, the security review, and the ownership question all get answered, and the survey puts most of the scaling in firms above \$1B revenue. If you have not crossed that gap, you do not yet have the data to make a defensible build decision on anything that stores customer data. This matches something I noted earlier in the year: [developers coding faster has not translated into faster delivery](#), and faster coding is what the build decision assumes.

I expect the 32% to climb and the first quiet re-signings to appear

I expect the figure to keep climbing in the 2027 survey, likely into the high thirties, because the behaviour is easy and the feedback loop is slow. Cancelling costs nothing today. The maintenance bill arrives in year two.

I also expect the first visible reversals within twelve months: companies that rebuilt an internal tool, hit a security finding or a compliance review, and quietly re-signed with the vendor at a worse rate. Unconfirmed, and I have not seen it in any dataset yet, but it is the pattern I would bet on given where the Veracode and Apiiro numbers land.

The number I will actually be watching is not the build-vs-buy percentage. It is whether AI EBIT impact moves off 37%. It has been flat for a year while adoption, productivity self-reports, and agent scaling all rose. Either the value shows up in margin in the next two surveys, or the honest conclusion is that agentic coding changed how the work gets done without changing what it costs, and the cancelled renewals were a transfer rather than a saving. If you are sitting on a renewal decision this quarter and the build case rests on "our agents can do this now", I am happy to [walk through the run-cost side of it with you](#) before the contract lapses.



McKinsey: 32% of Companies Killed a Software Purchase Because Coding Agents Could Build It



WRITTEN BY

Artur Markus

Software engineer in Basel building AI infrastructure and agentic systems, with a background in pharmaceutical automation, GxP compliance, and validation. I help teams turn AI from a chatbot into a reliable operational layer.

Book a meeting →