

BUSINESSGPS

EARLY ADOPTER SIGNAL

Early Adopter Signal

Who repairs each change on your work package, and can it be undone? What Anthropic's own code team does, 15 October

Week of October 12, 2026

PUBLISHED

OCTOBER 19, 2026

By the end of this week you can hold three things for every change on your own work package: whether it can be undone, who repairs it, and what it is for, in one sentence you can say.

On 15 October 2026 Anthropic said it had moved most of the reading of its own code to machines, and asked: “If machines do most of the reading, how does anyone learn the code?” For its own codebase, from June to September, it reports that “the share that had to be reverted fell by more than 40 percent” while merged changes more than doubled: its own figure, a share of merged changes that had to be undone, not a verdict on quality, not a statement of cause, and not a client’s programme.

If machines, or people who later move on, land changes on your work package, you can keep the reason each one is safe in a note of your own.

On 15 October 2026 Anthropic described how its own team keeps track of code when machines do most of the reading

Anthropic’s engineering post of that day, How the Claude Code team does code review, by Addy Osmani, Morgan Westlee Lunt and Qing Wang, describes the vendor’s own team and its own product. It is not a client’s programme with a vendor and an integrator.

The authors write: “We didn’t remove review. But we did move most of the reading to machines, optimized for human time and attention, and invested heavily in making it fast and easy to undo mistakes.”

On review’s second job: “Reading a teammate’s diff is how you learn what changed and how the pieces fit.” Their answer: “We don’t have a formal process for this. Nobody on the team keeps a complete mental model of Claude Code in their head, and we don’t think anyone needs to.”

What the team says it does. Anthropic does not say these practices caused the fall in reverts:

- When someone wants to understand a part, they ask Claude for an explainer; the post’s example is the module they are about to change.
- Changes are small, and “Everything is reversible.”
- “Blast radius sets the bar” (blast radius: how much a change could break).
- “If you land a small or medium change and something still goes wrong, it’s a team norm that you usually fix it.”

Between 12 and 16 October 2026 HCLTech, Accenture and McKinsey each named the forward-deployed engineer. HCLTech promises help for the client’s people and Accenture sells it; in the release, transcript and text introduction we read, none of the three gives a measure of what those people can run once such an engineer leaves. So a change on your work package may be landed by someone who later moves on.

We think the part of the work worth keeping is knowing why a change is safe to land, and it stays with you only if you keep it on purpose.

Anthropic’s post says the team “now tracks rounds-to-pass alongside recall” (recall, in the usual sense: the share of real problems a reviewer finds), and prints no recall figure. We expect Anthropic to publish no recall figure for its code reviewer by Monday 2 November 2026. We are wrong if it publishes one.

Ask one question of whoever lands a change

If this goes wrong, who fixes it, and what is it for, in one sentence?

Ask it of whoever lands the change: an engineer placed with your team, a colleague, or the person running the agent that wrote it. A good answer is a name and a sentence you could repeat to someone who was not there.

Keep a note, one row for each change

One week on one work package. A filled-in example: everything in it is made up.

Change	Can it be undone, and how	Who repairs it if it fails	What it is for, in one sentence	Who wrote that sentence
New field on the supplier record	Yes: the previous version is kept and can be re-stored	The engineer who landed it	Holds the supplier's tax number, so invoices stop coming back	The engineer, checked by you
Payment run moved	Yes: switch back to the old schedule	The payments analyst who asked for it	Moves the run to Thursday, so it clears before the weekend	You
Rule that blocks duplicate invoices	Yes: the rule can be switched off	The person running the agent that wrote it	Stops the same invoice being paid twice	The person running the agent
Approval report rewritten	Not without rebuilding the old one	Not yet agreed	"Tidy-up"	
Refund test scripts updated	Yes: the old scripts are kept	The test lead	Covers the new refund limit agreed this month	The test lead

In the fourth row, the change hardest to undo has no repairer, no real sentence and no name. That is the gap, and the row to ask about first.

Count what the note takes from your week

With example figures, which are illustrative: fifteen minutes a change and six changes a week is 90 minutes, an hour and a half a week. Put your own count in their place.

Keep it, one change at a time

1. **Ask** for the one-sentence purpose before the change lands. Done when: the sentence is in your note, with the name of whoever wrote it.
2. **Check** that the change can be undone, and how. Done when: the route back is written beside it.
3. **Write** down who repairs it if it fails. Done when: there is a name in that column.
4. **Read** the explainer for any part you do not know, or ask for one. Done when: you can say what that part does without it.

Your programme lead has a piece this week too: the Way Pointer at capability.ai/blog/a-name-beside-every-engineer, a page that puts a name beside each engineer placed on the programme, with hours a week and cover for the day job. When you are asked how much of your week the note takes, your own count is the sentence to carry up.

BusinessGPS publishes the Early Adopter Signal. Get the Signal by email each week at capability.ai/newsletter.

Sources

1. Anthropic, How the Claude Code team does code review, by Addy Osmani, Morgan Westlee Lunt and Qing Wang, 15 October 2026: <https://claude.dev/blog/solving-code-review-at-anthropic/>. The reverted share is Anthropic's own figure for its own codebase, June to September 2026, a share of merged changes; it is not a measure of quality, and the post gives no figure for how much understanding the team retains.
2. HCLTech, press release, HCLTech and Anthropic sign Global Premier Partnership to scale governed Agentic AI across global enterprises, 12 October 2026: <https://www.hcltech.com/en-us/press-releases/hcltech-and-anthropic-sign-global-premier-partnership-scale-governed-agentic-ai>. HCLTech says it will help clients “train their workforces and build the organizational capabilities needed”.
3. Accenture, Investor Day FY27 conference transcript, 14 October 2026; Kate Clifford, Chief Leadership and Human Resources Officer, on the forward-deployed engineer, a role she says Accenture did not have 18 months ago: <https://investor.accenture.com/~media/Files/A/accenture-v4/investors/documents/accentures-fy2027-investor-an-analyst-conference-transcript.pdf>. Accenture sells client upskilling through LearnVantage.
4. McKinsey & Company, What is a forward-deployed engineer?, 16 October 2026: <https://www.mckinsey.com/featured-insights/mckinsey-explainers/what-is-a-forward-deployed-engineer>. McKinsey says the engineer builds “alongside the people who will use it”, and that the label “may be a buzzword” for a service offered for years. Read for its text introduction; the video was not watched.