

INNOQ TECHNOLOGY DAY 20.11.2025

High Agency als Überlebensstrategie?

Die Entwicklungsabteilung im Wandel

INNOQ



ROBERT GLASER
HEAD OF DATA & AI



/ai und **jetzt**

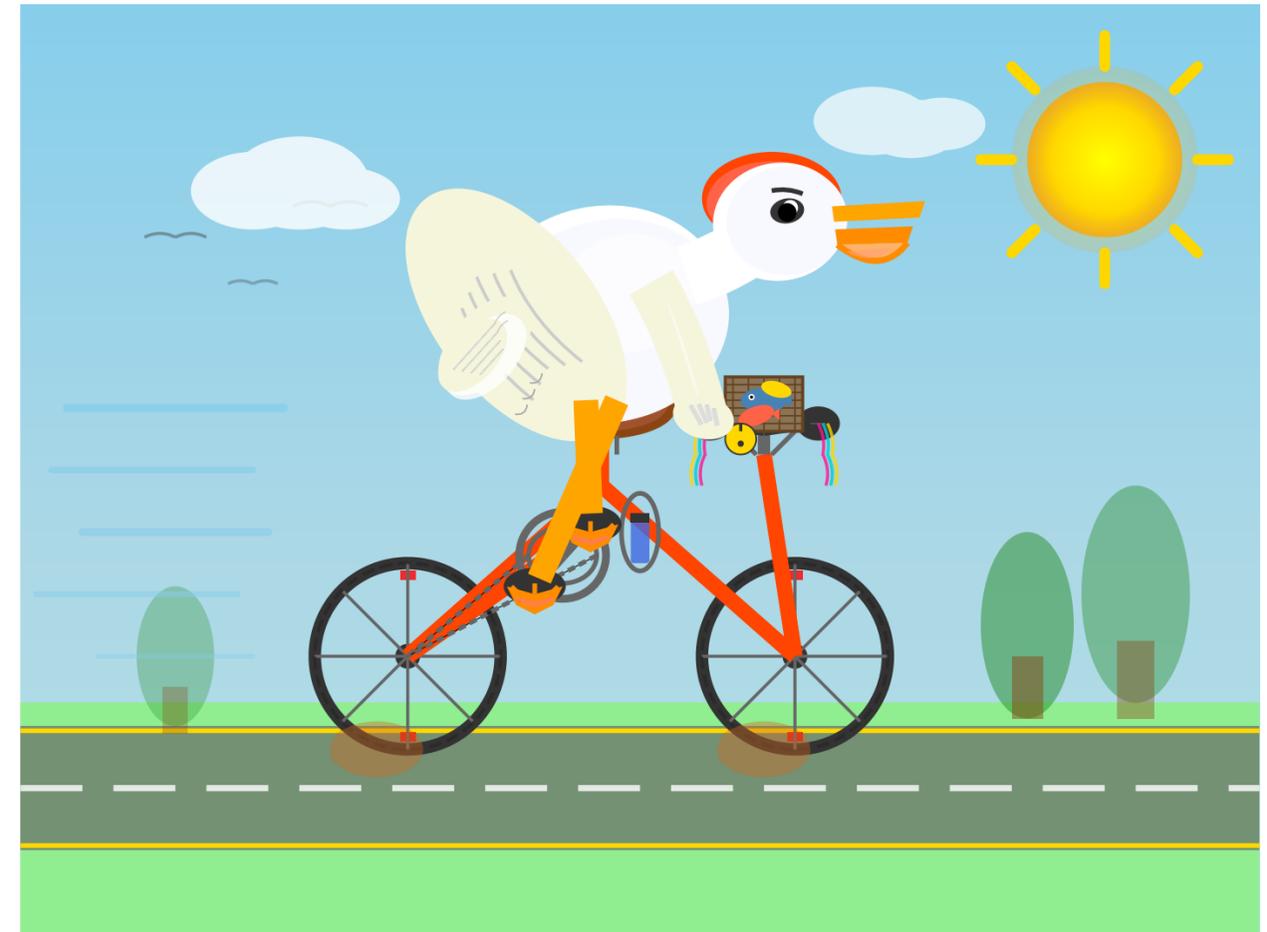
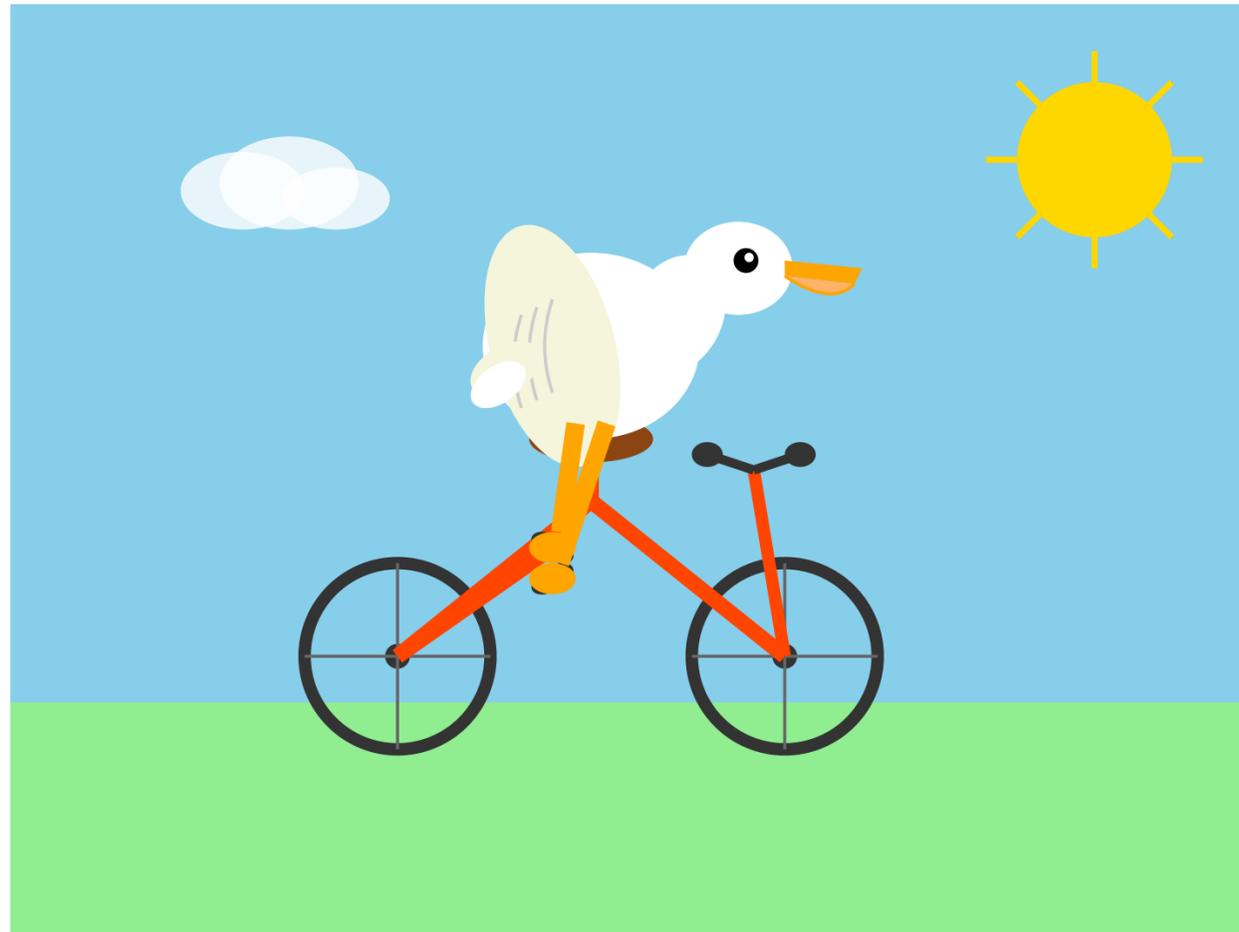
undjetzt.ai



**THIS
WEEK
IN
DATA
& AI**



Agentic Pelican on a Bicycle



<https://www.robert-glaser.de/agentic-pelican-on-a-bicycle/>



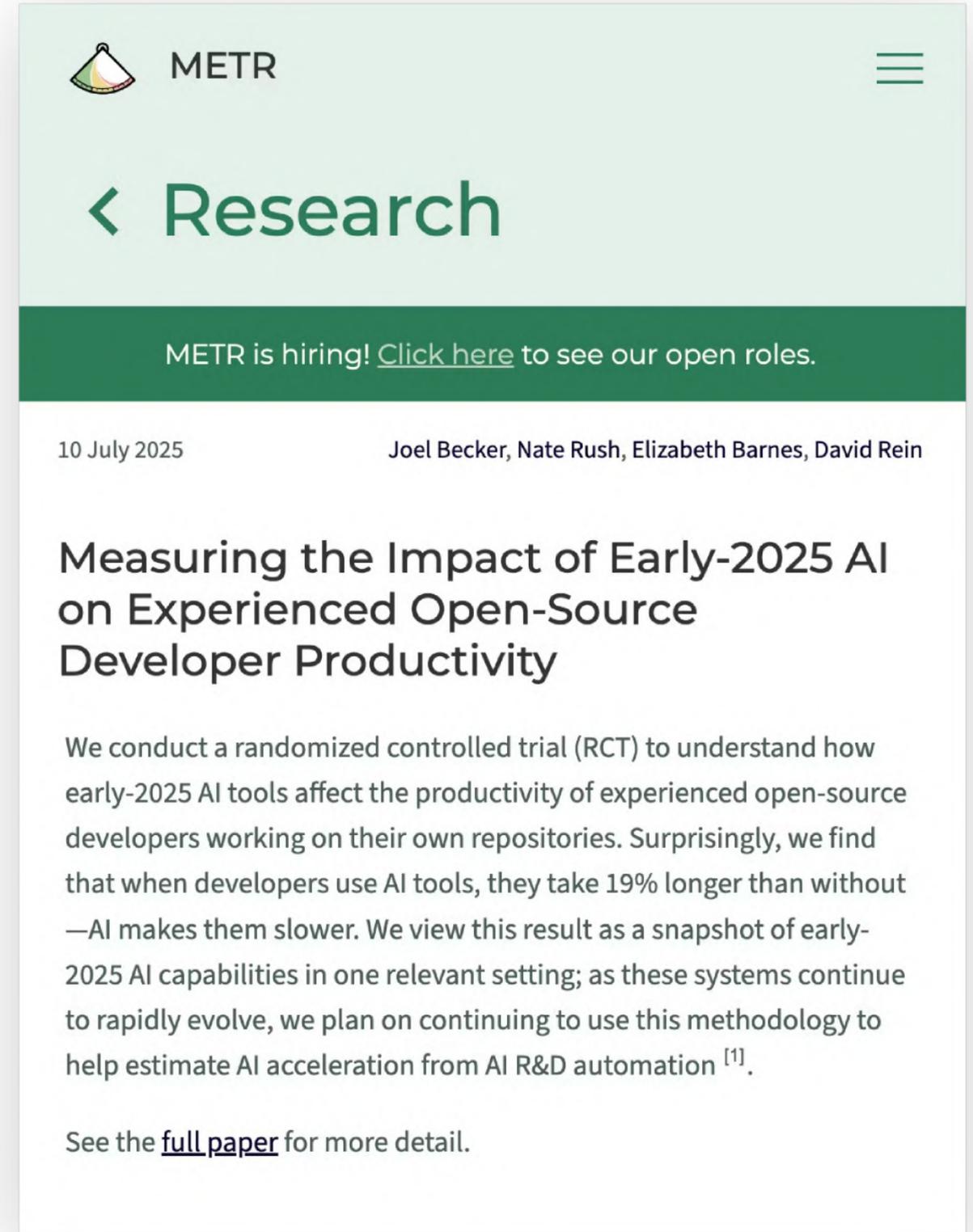
Agentic engineering

Where are we?

Experienced OSS maintainers moved **19% slower** with Cursor and Claude 3.5/3.7 **despite forecasting a 24% speedup.**

- Tacit context locked in heads
- Early agentic capabilities and reliability
- Lack of experience in collaborating with AI

<https://metr.org/blog/2025-07-10-early-2025-ai-experienced-os-dev-study/>



The screenshot shows a blog post from METR. The header includes the METR logo and a navigation menu. Below the header is a green banner with the text "METR is hiring! [Click here](#) to see our open roles." The main content area features the date "10 July 2025" and the authors "Joel Becker, Nate Rush, Elizabeth Barnes, David Rein". The title of the post is "Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity". The abstract text reads: "We conduct a randomized controlled trial (RCT) to understand how early-2025 AI tools affect the productivity of experienced open-source developers working on their own repositories. Surprisingly, we find that when developers use AI tools, they take 19% longer than without —AI makes them slower. We view this result as a snapshot of early-2025 AI capabilities in one relevant setting; as these systems continue to rapidly evolve, we plan on continuing to use this methodology to help estimate AI acceleration from AI R&D automation ^[1]." At the bottom, it says "See the [full paper](#) for more detail."

90 % of teams now use AI. Perceived productivity up, but **delivery instability rises alongside throughput.**

Biggest gains appear only when AI is paired with seven capabilities:

1. clear AI policy
2. small batches
3. strong version control
4. user-centric focus
5. platform
6. healthy data
7. AI-accessible context

<https://cloud.google.com/resources/content/2025-dora-ai-assisted-software-development-report>

The image shows the cover of the DORA 2025 report. At the top, the word 'DORA' is written in large, bold, black letters against a blue background. Below it, the title 'State of AI-assisted Software Development' is written in a smaller, bold, black font. The year '2025' is positioned to the left of a central illustration. The illustration depicts a hot air balloon with a large, circular, clock-like face on its envelope, featuring a needle and hands. Two figures are seated in the basket below, one looking through a telescope. The background of the cover transitions from yellow at the top to light blue at the bottom. At the bottom left, the 'Google Cloud' logo is visible. The bottom section of the cover is dark and contains logos for various sponsors and partners, categorized into Platinum sponsors, Premier research partner, Gold sponsors, and Research partners.

DORA
State of AI-assisted
Software Development
2025

Google Cloud

Platinum sponsors
swarmia /thoughtworks

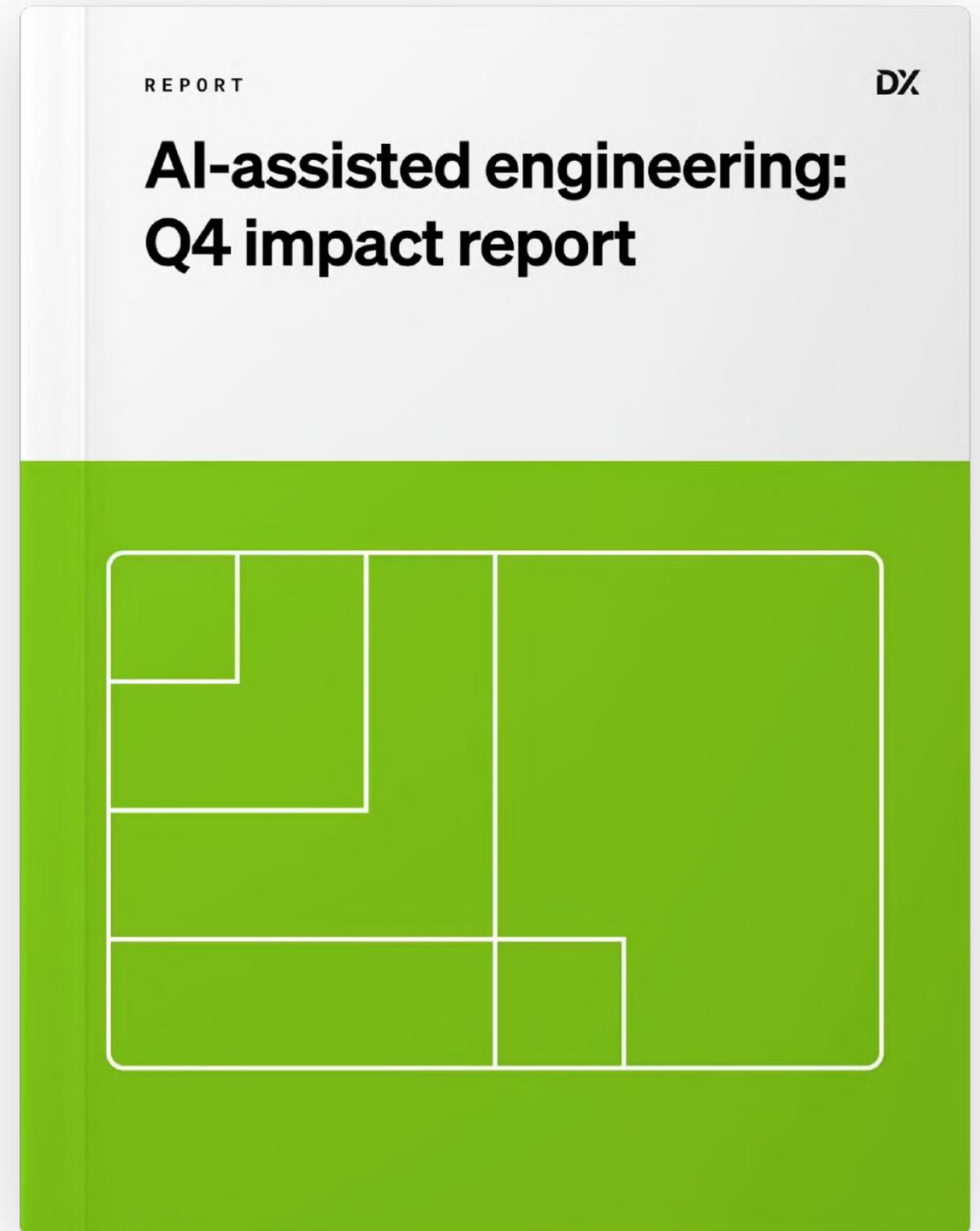
Premier research partner
REVOLUTION

Gold sponsors
Buildkite CodeRabbit DATADOG Deloitte.
harness JELLYFISH Octopus Deploy OPSERA

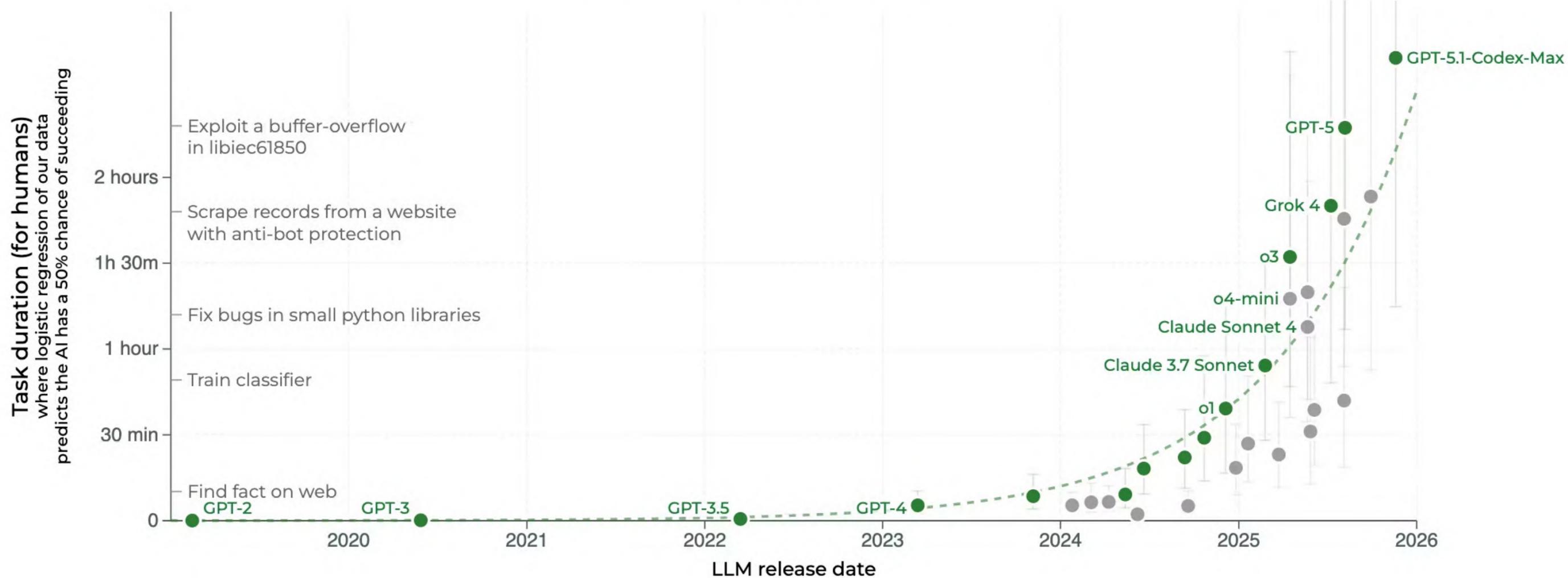
Research partners
GitHub GitLab
skillbench Workhelix

Structured enablement drives measurable ROI. Teams that pair rollout with training, governance, and measurement outperform those treating AI as a plug-and-play tool.

<https://getdx.com/report/ai-assisted-engineering-q4-impact-report/>



The time-horizon of software engineering tasks different LLMs can complete 50% of the time



**The length of tasks AI can do
is doubling every ~6 months.**

This confirms lab reports.

Everyone talks about productivity.
Is no one interested in quality and ambition?

A single person with AI outperforms a two-person human team in solution quality.

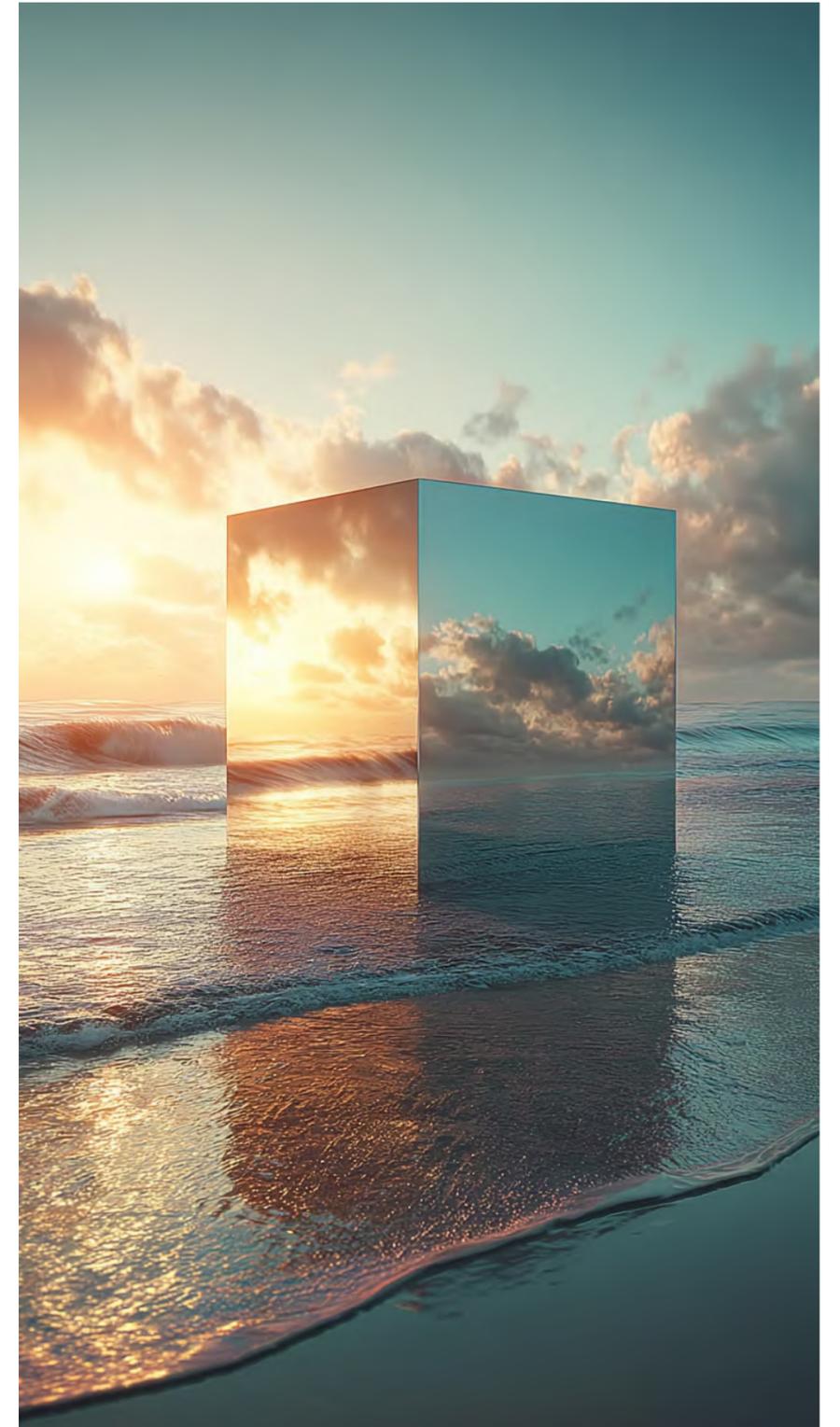
AI blends siloed expertise by breaking the walls between disciplines.

<https://www.hbs.edu/faculty/Pages/item.aspx?num=67197>

The screenshot shows the Harvard Business School Faculty & Research Publications page. At the top left is the Harvard Business School logo. The main header is 'FACULTY & RESEARCH' in a dark teal box. Below it, a breadcrumb trail reads 'Harvard Business School → Faculty & Research'. The page title is 'Publications' in large teal text. The featured article is a 2025 Working Paper from the HBS Working Paper Series, titled 'The Cybernetic Teammate: A Field Experiment on Generative AI Reshaping Teamwork and Expertise'. The authors listed are Fabrizio Dell'Acqua, Charles Ayoubi, Hila Lifshitz, Raffaella Sadun, Ethan Mollick, Lilach Mollick, Yi Han, Jeff Goldman, Hari Nair, Stew Taub, and Karim R. Lakhani. The article is in English, 56 pages long, and available in print format. There are icons for Email, Print, and Share. The abstract begins with 'We examine how artificial intelligence transforms the core pillars of collaboration—performance, expertise sharing, and social engagement—through a pre-registered field experiment with 776 professionals at Procter & Gamble, a global consumer packaged goods company. Working on real product innovation challenges, professionals were randomly assigned to work either with or without AI, and either individually or with another professional in new product development teams. Our findings reveal that AI significantly enhances performance: individuals with AI matched the performance of teams without AI, demonstrating that AI can

If your organisation has never truly empowered humans with clarity and agency, you are not ready to unleash Agents. Agentic AI is a mirror. It does not invent coordination, clarity, or trust. It reflects whatever is already present.

— Stuart Winter-Tear

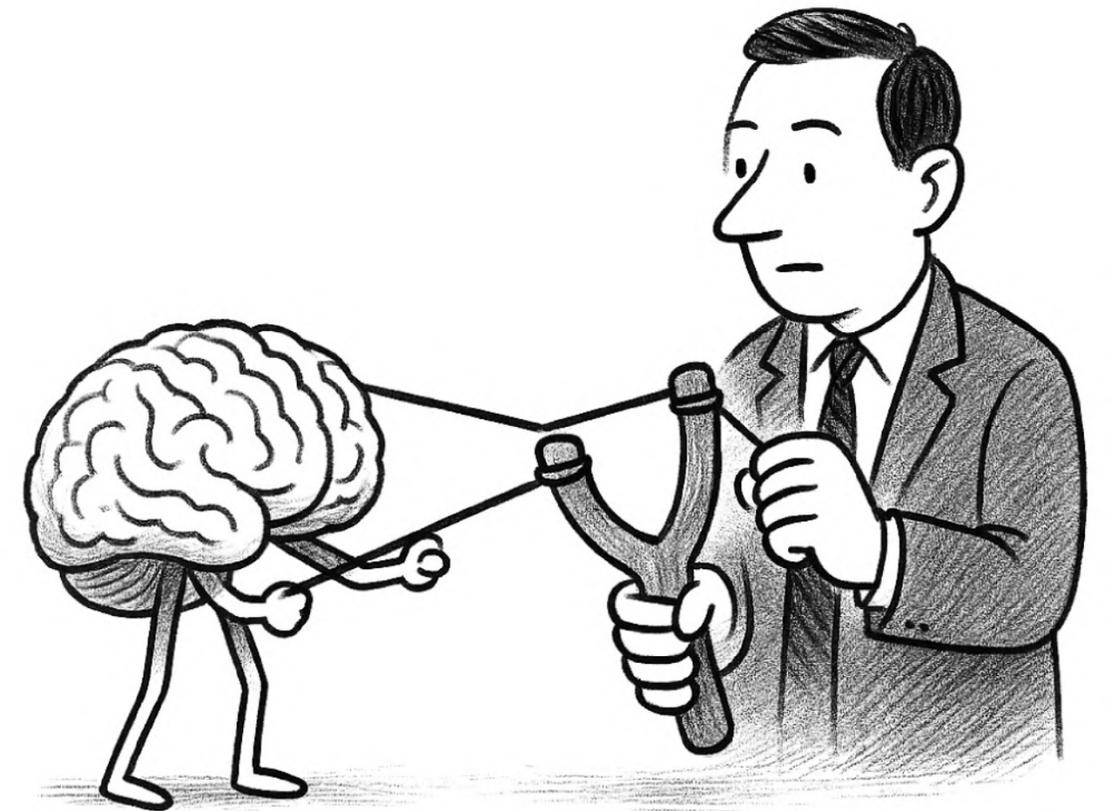


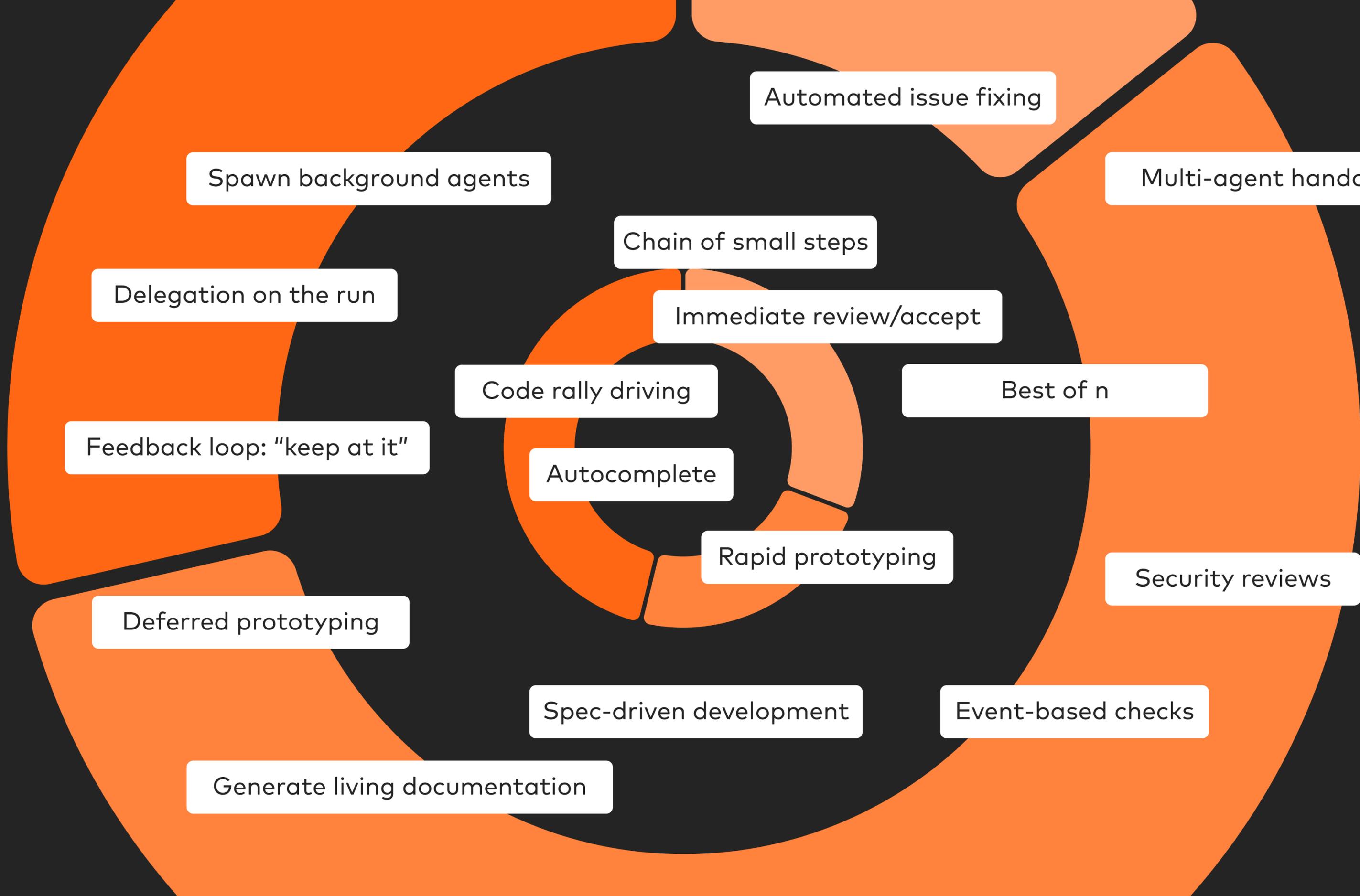


**What does your mirror
currently reflect?**

**If you treat agentic AI as
„another tool in the toolbox“
you'll get waterfall-grade slop in return.**

Enter the elastic loop.





Re-meet the 2001 Agile Manifesto

“collaboration over contract negotiation”

“responding to change over following a plan”

essentially promising tight, continuous feedback.

Those values are only real when **the loop closes every time.**

The elastic loop spectrum

Tight loop

Human stays embedded,
reviews every turn.

High cognitive load.
Maximum control.

When stakes, ambiguity are high.

**When context can be provided
on-demand.**

Loose loop

Human delegates or agent acts
on domain events.

Low (or deferred) cognitive load.
Little control.

When stakes, ambiguity are low.
Or quality control can be deferred.

When context is in place.



Naveen Naidu ✓

@naveennaidu_m



I've fully moved off Claude Code to Codex

Here's how my days run now:

Morning (Codex web):

- Plan the day in Linear
- Turn user feedback into small isolated tasks
- Kick off 3-4 "papercut" parallel tasks in codex web
- Forget them

Deep work (Codex CLI):

- Focus on the one big task
- Hands on coding with the CLI

Evening (review):

- If it's Codex web: create PR + auto-review catches issues. Locally test the change and push to prod
- If it's CLI: /review my local changes, test, PR, merge to main

Web delegates the small stuff. CLI accelerates the important stuff

2:01 PM · Oct 13, 2025 · **404K** Views

Tightening vs. relaxing the loop

Strategic questions

- 1. How volatile is the domain (part)?**
- 2. What's the freshness of context?**

3. **Where does trust live?**

If organizational trust is only earned via peer review, the loop tightens under political pressure.

4. **Which metrics detect drift fast enough?**

Lead time, change fail rate, and loop-closure latency (time from agent output to validated outcome) should inform how far you can stretch.

**This is a weird new situation for all of us.
Build taste, intuition, and experience.**

*For a technology that doubles engineering task length
every ~6 months.*

How does truffle taste?

Ask your team. They need to know.

I'm beginning to suspect that a key skill in working effectively with coding agents is developing an intuition for when you don't need to closely review every line of code they produce. This feels deeply uncomfortable!

— Simon Willison

<https://simonwillison.net/2025/Oct/11/uncomfortable/>

Examples for unclosed loops

1. Drive-by-prompting with lack of context, agent has to prime itself, codebase does not offer enough relevant context, unhappy developer quits tight loop.
2. Papercut best-of-n development tasks delegated to background agents are not reviewed, selection process never happens.
3. Agent opens refactoring PR, engineering never reviews it, change decays behind main—improvements, reward and learnings are all lost.
4. Product discovery agent clusters user feedback and proposes experiments, yet product doesn't validate them with customers.

What everything comes down to:

- 1. Produce work that's faster to validate than to redo.**
- 2. Build things that would not exist otherwise – but deliver value.**
- 3. Always close the loop.**



**The age of the
agent orchestrator**

The new scarcity?

**Markets tend to organize around
whatever is in short supply.**

Expertise gets democratized.

Experience does not.



Air Katakana 

@airkatakana



the choice isnt between "code written by ai" and "artisan handwritten code written by john carmack"

the choice is between "code written by ai" and "something that could have existed, but didnt"

6:40 PM · Oct 13, 2025 · **36.5K** Views

There will still be experts.

We will need people to

- **provide judgment**
- **to set high-level strategy**
- **to handle the weird cases**

But the leverage point moves.

**Knowing how to
break down a task,
set a reward,
audit a run**
is going to be a baseline skill.

Culture eats strategy.

If you don't create an environment where people are comfortable *delegating to machines, measuring the outcomes, and iterating,* you'll miss out on a huge amount of leverage.



**Delegation, feedback,
and agency**

Establish culture readiness

Surface cultural blockers and tackle

Fear of loss of control

Manual sign-off bottlenecks

Hidden "shadow reviews"

Frozen prompts

Limited experiential trust

Leaders haven't shipped with agents

Teams outsource learning

Success stories stay anecdotal

Shallow feedback rituals

Retros as status theater for humans

No telemetry on loop closure

Learnings vanish

Lack of organizational trust

Teams and individuals hoard context

Errors are punished, not mined

Agency collapses into defensive compliance

Change minds

Our job is not to type code into a computer.

Our job is to deliver systems that solve problems.

— **Simon Willison**

Enable autonomy

*Orgs that have worked out how to empower autonomous groups of humans
[...] are already in the best place to use agentic AI.*

— Matthew Skelton, co-author of Team Topologies

Autonomy

The freedom of self-determination.
Agency is the ability to actually
exercise that freedom.

Agency

The ability and willingness to act
autonomously, take ownership of
decisions, and actively shape outcomes —
even in the face of risk and uncertainty.

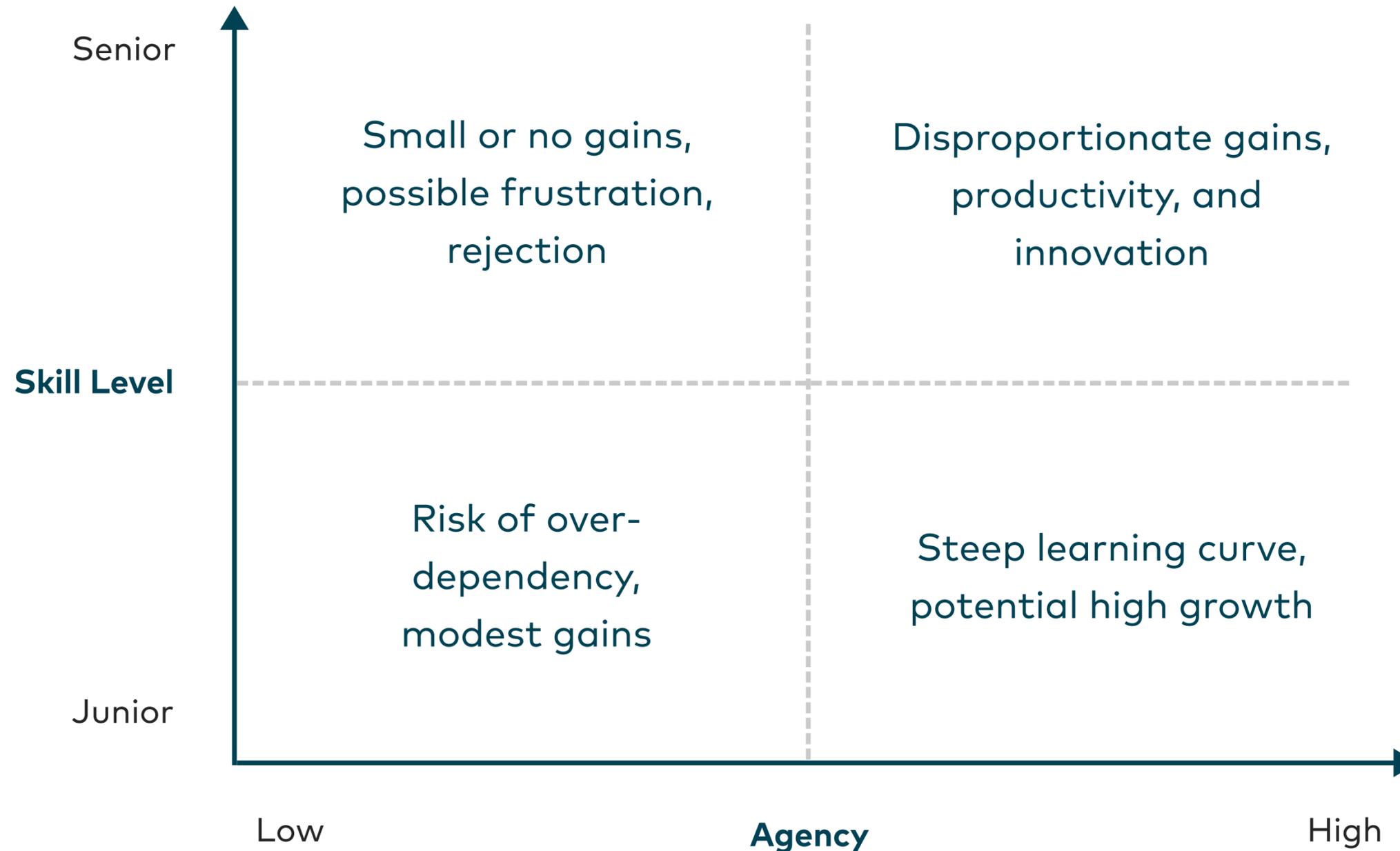
The great divide?

Benefits aren't distributed equally.

High-agency engineers see outsized gains.

Low-agency engineers experience only modest improvements, or may even find themselves falling behind.

A complex, multidimensional divide



How to enable agency?

Build ladders, don't force personalities to flip.

- Drop people into safe loops, celebrate closure, not speed
- Codify playbooks, remove guesswork keeping cautious engineers on the sidelines
- Reward ownership and accountability, so more people reach for it
- Run agency (!) labs, tackle real problems, immediately reflect on what closed the loops

Rebuild teams around agency

High-agency loops call for slimmer, cross-functional squads:

- **One accountable owner** per discipline/value stream/bounded context who can run with their agent, no redundant overlaps.
- **Every duplicate role adds handoffs, waiting, and alignment drag**—exactly the friction that erases the leverage you're trying to buy.
- **Rotational coaching and lightweight guilds** to avoid lone-wolf-syndrome and new silos.



Re-architecting the SDLC

*Tactics and infrastructure for
closed loops*

Platform DX and AX

Agent-ready environments

Standardize reproducible sandboxes and ephemeral stacks so agents can execute, test, and observe work without manual setup.

Instrument loop telemetry

Bake feedback-loop IDs into CI/CD, testing, and review tools so every delegation emits start/stop, drift alerts, and human interventions. Don't use merged PRs as success metric!

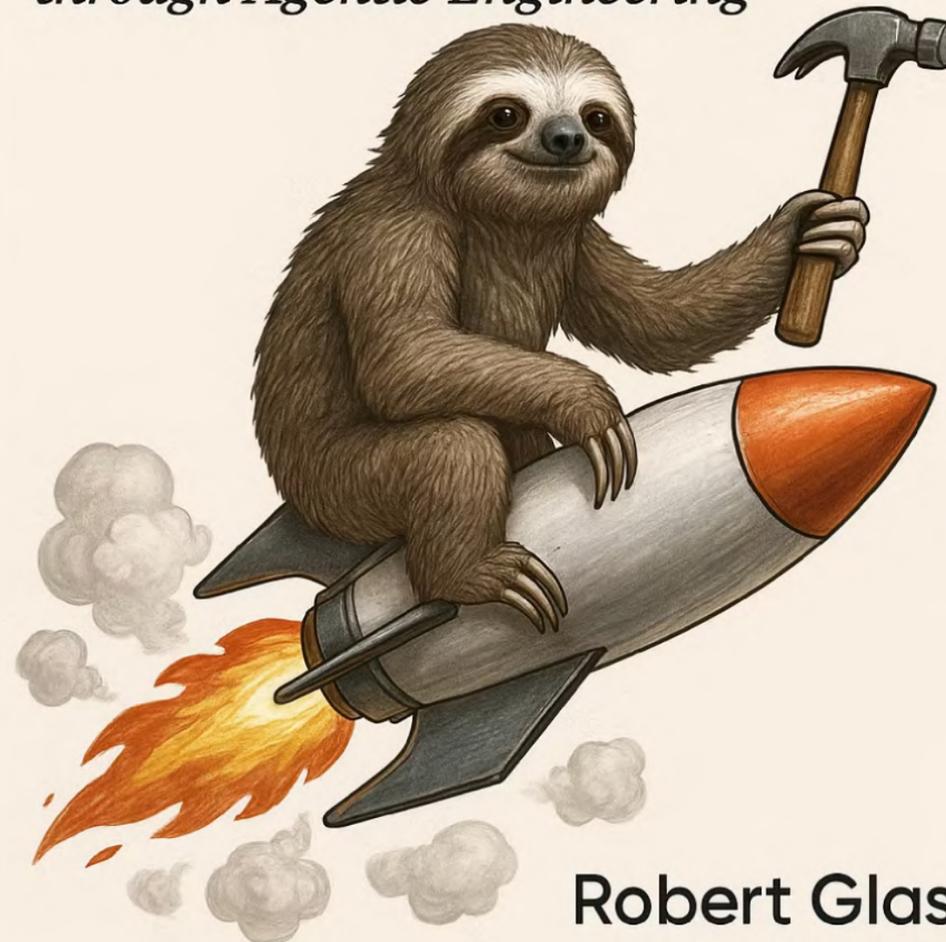
Shareable agent skills

Codify company knowledge, workflows, deterministic tooling as reuseable, model-agnostic agent skills.

O'REILLY®

Re-Learning Software Engineering

through Agentic Engineering



Robert Glaser

Wrapping up

1. Map where your loops don't close
2. Enable agency (this is the hard part)
3. Instrument

Let's talk!



Robert Glaser

Head of Data and AI

+49 160 979 35 254

robert.glaser@innoq.com

robert-glaser.de/linkedin



LinkedIn

innoQ Deutschland GmbH

Krischerstr. 100
40789 Monheim
+49 2173 333660

Ohlauer Str. 43
10999 Berlin

Ludwigstr. 180E
63067 Offenbach

Kreuzstr. 16
80331 München

Wendenstr. 130
20537 Hamburg

Spichernstr. 44
50672 Köln