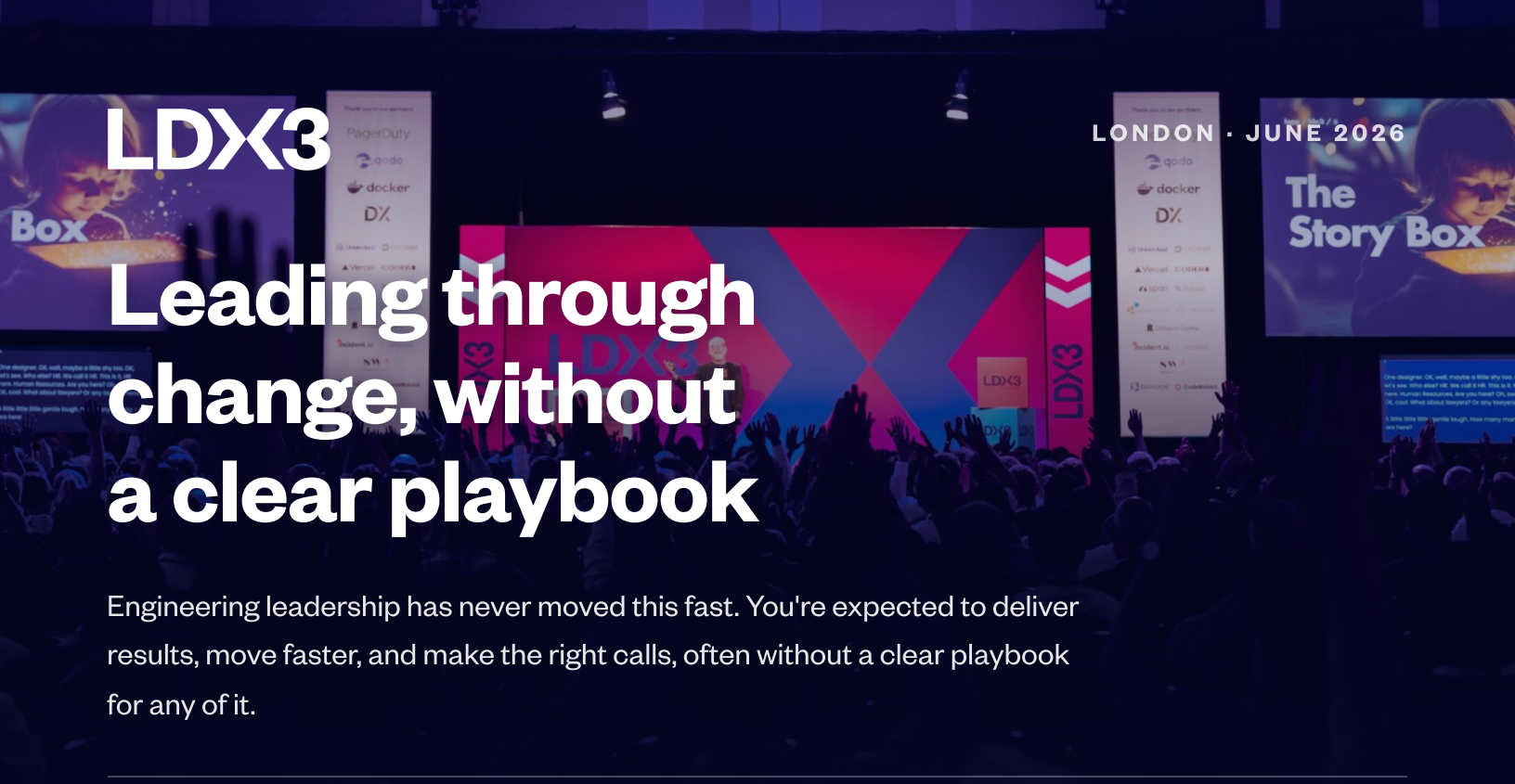




LDX3

LONDON · JUNE 2026

Leading through change, without a clear playbook

Engineering leadership has never moved this fast. You're expected to deliver results, move faster, and make the right calls, often without a clear playbook for any of it.

— POST-EVENT DEBRIEF · FOUR-MINUTE READ

This document is the practical answer to that pressure. It pulls together the four themes that came up again and again across our LDX3 London 2026 event, from people who are already writing that playbook themselves. **Not predictions or hype.** What actually worked, who it matters to, and what to do about it.

Each section names the speakers whose talks it draws on, with a link through to the full video.

01 Closing the context gap between AI tools and how your organization actually works

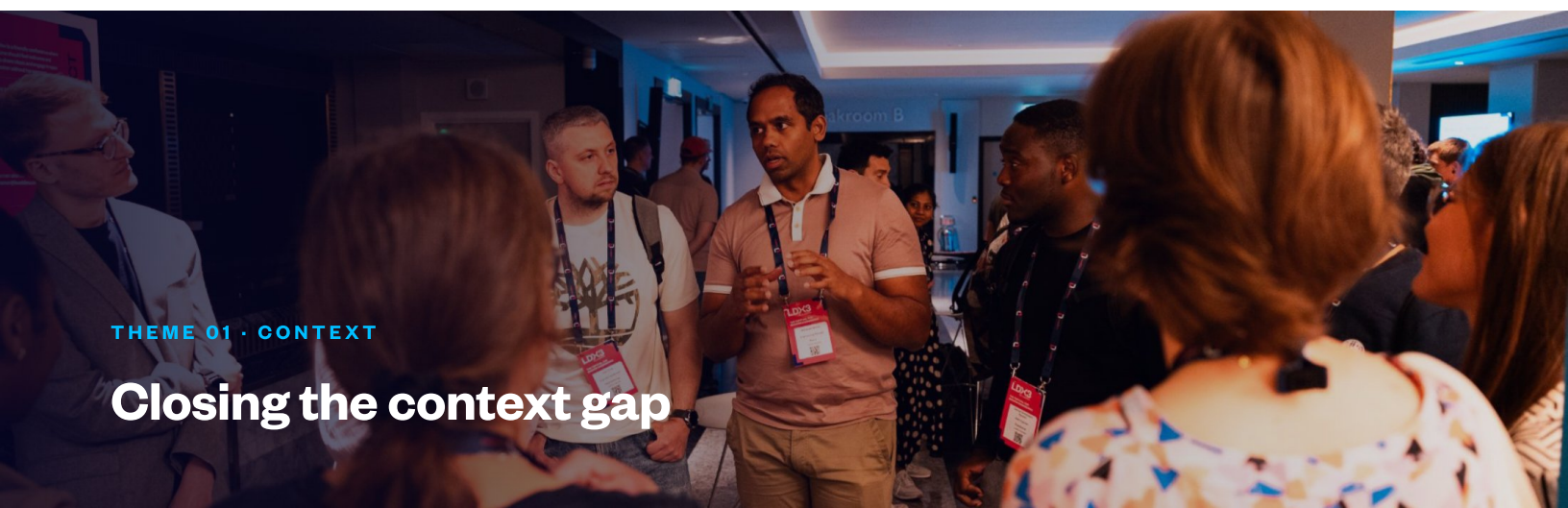
02 Replacing manual approval gates with automated guardrails

03 Redesigning leadership as the management layer flattens

04 Treating technical debt and system resilience as a business risk, not an engineering preference



LDX3 is coming to New York in September. [Find out more →](#)



THEME 01 · CONTEXT

Closing the context gap

1 THE PROBLEM UNDERNEATH THE TOOLS

AI coding tools are fast, but they don't know your organization. They can read a file, sometimes a repository. They don't know why a service was built a certain way, who owns what, or which shortcuts were taken under deadline pressure three years ago. Dennis Pilarinos, founder and CEO at Unblocked called this the "context gap," and argued it's the single biggest blocker to AI actually working in production, not model quality.

Lawrence Jones, an engineer at Incident.io, backed this up with numbers. For every \$1 spent on AI tools, his data showed organizations reclaim \$3.70 in equivalent headcount cost, but only once teams had the context and process in place to use the tools properly. Justin Reock, deputy CTO at DX, added the adoption-side view: teams using AI daily ship 60% more pull requests, but AI-generated code reaching production with no record of who used it or how (shadow AI), is already happening at scale.



Dennis Pilarinos

Unblocked



Lawrence Jones

incident.io



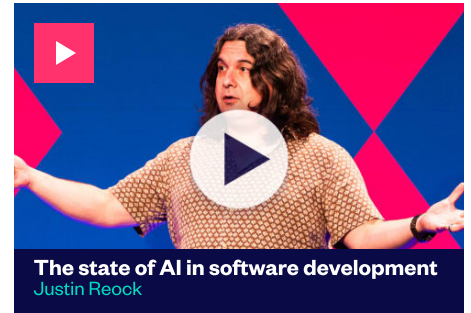
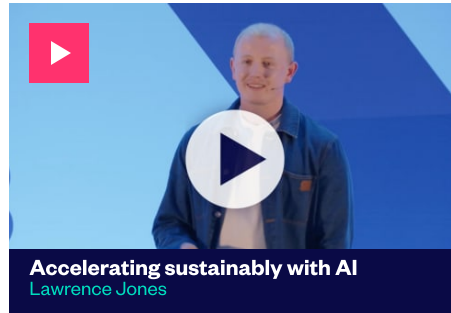
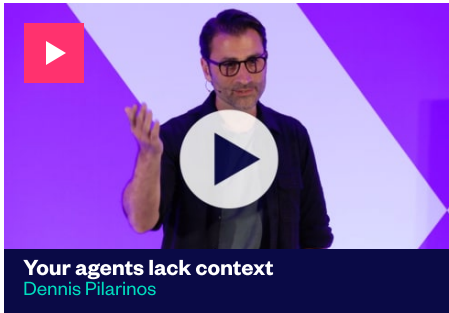
Justin Reock

DX

MATTERS MOST TO

Engineering managers setting AI norms for their team with no company-wide policy to lean on, and **Staff+ engineers** who are expected to be the de facto AI authority without a mandate or a map.

WHAT TO WATCH



WHAT TO DO

- Write down the things your team already knows but has never documented: architectural intent, the "why" behind past decisions, who owns what. Put it somewhere an AI tool (and a new joiner) can actually find it.
- Work out which of your engineers are supervising AI carefully, which are checking every output line by line, and which are trusting it without checking. Each group needs different support, not the same rollout.
- Track how much AI-generated code is reaching production without going through your normal tooling. If you don't know the number, assume it's higher than you think.
- Before your next context window fills up on a long AI session, ask whether you're feeding it the right information or just more information. More isn't the same as better.

"Shadow AI is definitely happening. AI-generated code is making its way into production, but with no record of telemetry from those users actually using the enterprise account."

Justin Reock, deputy CTO, DX



Gergely Orosz, Camille Fournier, and Will Larson are among the speakers in New York this September. [Find out more →](#)

LDX3

By LeadDev

THEME 02 · PROCESS

Guardrails, not gates

LDX3
By LeadDev



2 WHEN THE CODE MOVES FASTER THAN THE PROCESS

Developers can now move at 10x speed. Most organizational systems, approvals, reviews, sign-offs, are still moving at 1x. Nicole Forsgren, senior director, GM for developer intelligence at Google put it plainly: coding was never the majority of our time. Now that coding is faster, the manual steps around it are the thing slowing everyone down.

Rick Clegg, developer experience lead at Wise, shared what replacing those manual gates with automation actually looks like in practice, and why it matters most for the people newest to the team. At Wise, junior engineers have used AI to patch vulnerabilities across five services in a single session, something that would once have needed a senior engineer's sign-off at every step. His argument: guardrails let you say yes to more people, more often, without lowering the bar.



Nicole Forsgren

Google / Microsoft



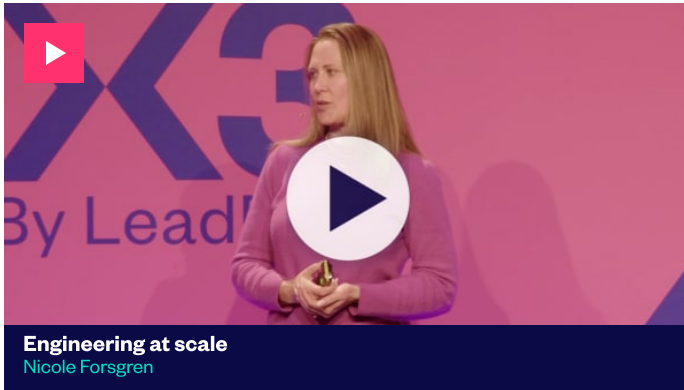
Rick Clegg

Wise

MATTERS MOST TO

Engineering managers trying to keep quality up while shipping faster with the same headcount, and **Director+ leaders** who own the governance decisions that either enable or block this.

WHAT TO WATCH



WHAT TO DO

- Move from manual approval to automated rollback. If a change fails on a defined metric, it rolls back automatically. Nobody has to catch it in a meeting.
- Write your architectural boundaries down in a format AI tools can read, not just in a senior engineer's head. This cuts down on the informal "let me explain how this system works" conversations that create bottlenecks.
- Stop treating "looks good to me" as a sufficient reason to approve a change. Ask agents and engineers alike to record why a change passed, not just that it did.
- Look specifically at where your early-career engineers are blocked by process rather than by skill. That's usually where guardrails pay off fastest.

"When your seniors leave, who is going to carry the torch? We've watched juniors one-shot five services to patch vulnerabilities."

Rick Clegg, Developer Experience Lead, Wise



The conversation moves on in New York. [Find out more →](#)

THEME 03 · LEADERSHIP

Leadership under the flattening

3 WHERE THE PRESSURE IS ACTUALLY LANDING

Budget that used to go to headcount is going to compute instead, and middle management is absorbing most of the pressure. Danit Nativ Navon, senior engineering manager at Meta argued that the leaders who'll do well here aren't the ones who can out-code a 20-year-old with an AI tool. They're the ones who create enough psychological safety that people are still willing to learn and ask questions.

That pressure shows up in very different ways depending on where you sit. Diane-Lee Pretorius, director of engineering at Paymentology pointed out that a typical cross-domain kickoff meeting can cost roughly £700 an hour in senior salaries and often produces no written record at all. Her fix, the "simmer method," replaces the meeting with a shared list of open questions people can think through asynchronously before a decision gets made.

Dominika Rogala, founder of Good Job Coffee, looked at the same pressure from the burnout angle: AI reduces the effort a task takes, but it doesn't reduce the number of things pulling at someone's attention. Demands go up even as individual tasks get easier, and burnout follows when the resources to cope don't keep pace.



Danit Nativ Navon
Meta



Diane-Lee Pretorius
Paymentology

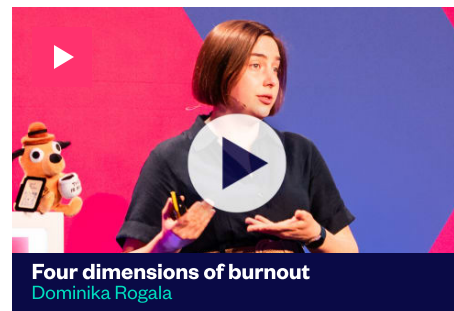
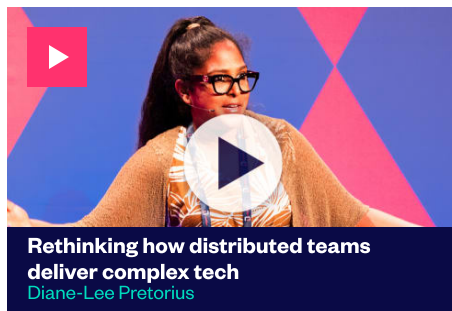
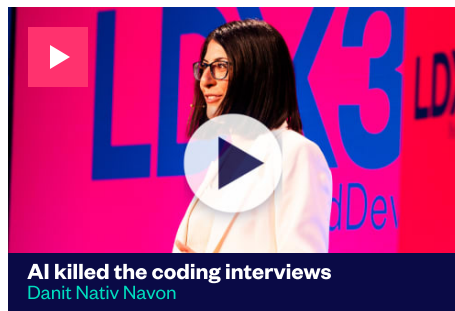


Dominika Rogala
Good Job Coffee

MATTERS MOST TO

Director+ leaders managing the cost and structure of this shift to AI-augmented coding, and **engineering managers** watching their teams absorb the pressure.

WHAT TO WATCH



WHAT TO DO

- Before booking your next cross-team kickoff, work out what it would actually cost in salaried time. Then ask if a shared async document could get you 80% of the way there.
- Name the norm explicitly: dissenting views are expected, not just tolerated. Leaders should speak last in a discussion, not first.
- Watch for the quiet signals of burnout, not just the loud ones: people going quiet in channels they used to be active in, or messages creeping later into the evening.
- Separate what people need to do their job (support, clarity, tools) from what's being asked of them (workload, context switching). If the second is rising and the first isn't, that's the gap to close.

"In the age of AI, your value is your ability to ask questions. We don't need leaders who can vibe code better than a 20-year-old. We need leaders who can create the psychological safety for teams to learn."

Danit Nativ Navon, Engineering Manager, Meta



You're not facing this alone. The same community, the same problems, will be in New York this September. [Find out more →](#)



4 THE COST THAT STAYS INVISIBLE UNTIL IT ISN'T

Faster code generation means more code, and more code sitting on top of systems that were never designed to move this fast. Sam Newman argued that the instinct to fix this by adding more safety mechanisms often backfires. Every extra mechanism is itself something else that can fail. The way to make systems genuinely more resilient is to reduce how tightly everything is connected, not to bolt on more protection.

Eli Sullivan, senior staff engineer at Dojo, described what happens when that complexity is left unaddressed: 90 separate Kubernetes clusters across the organization, each slightly different, making it impossible to roll out anything safely, AI tooling included. Their fix was a single standard template that every team builds from.

Ejber Ozkan, director of engineering at ITV, tackled the harder problem underneath all of this: technical debt is mostly invisible to the business, so it never gets planning capacity until it becomes an outage. His approach: ring-fence a fixed share of planning time for debt work, and require every debt item to record who found it, what it is, when it started, and why the shortcut was taken in the first place.



Sam Newman
Independent



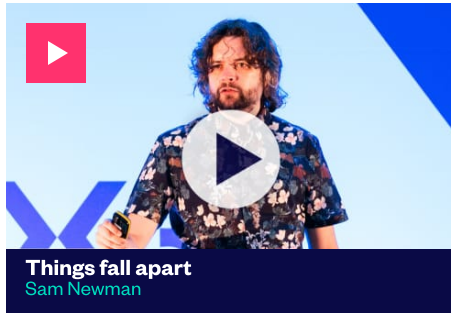
Eli Sullivan
Dojo



Ejber Ozkan
ITV

MATTERS MOST TO **Staff+ engineers** who own architecture decisions across teams, and **Director+ leaders** who need to defend that investment in language the business understands.

— WHAT TO WATCH



— WHAT TO DO

- Set aside a fixed share of planning capacity, not a "when we get to it" promise, for the technical debt nobody can see on a dashboard.
- Before adding a new safety mechanism to a system, ask what new failure mode it introduces. More protection isn't always more resilient.
- If your teams are each running a slightly different setup, standardize on one template before trying to roll out AI tooling across the board. AI tools need predictability to be useful.
- For every piece of debt you log, record why the shortcut was taken at the time. Without that, it's much harder to make the case for fixing it later.

"Increasing a system's complexity to handle more failure scenarios can actually introduce more sources of failure."

Sam Newman, Independent Consultant

WHERE TO START

Four questions to ask your org

Take these questions back to your team and your org, and get the conversation started on what needs to change.

1 What does your team know that lives only in someone's head, and where would an AI tool, or a new joiner, actually go to find it?

2 Which manual approval step are you keeping out of habit rather than need, and could an automated check do the same job?

3 What does your next cross-team meeting really cost in salaried time, and could a shared async document get you most of the way there?

4 What invisible technical debt are you carrying, and do you still know why the shortcut was taken in the first place?

None of this needs a full re-platforming or a company-wide mandate to start. It needs one team, one decision, and a written record of what happened.

UP NEXT FROM LEADDEV

The same shift is happening everywhere. New York is where it gets unpacked next.

Everything above came out of a room full of engineers who are already rewriting how this job gets done. That's what LDX3 New York exists for: the same practical, no-hype approach, the next stop on the same story.

■ September 15 & 16, 2026 ■ Javits Center, New York

SPEAKERS INCLUDE

- **Gergely Orosz** — The Pragmatic Engineer
- **Camille Fournier** — VP of Engineering & Author
- **Will Larson** — CTO, Imprint

[View the full agenda →](#)



leaddev.com →

ABOUT LEADDEV

LeadDev is the home of modern engineering leadership: articles, videos, research, and events for the people building and leading software teams. The work is practical and experience-backed, made for senior engineering leaders and influential ICs navigating delivery pressure, AI disruption, and organizational change.

LDX3 is its flagship conference brand, running in London, New York, and Berlin. Every talk referenced here is free to watch at leaddev.com.

