



AN INDUSTRY BRIEFING FROM GRAIL

What's Changing in the Moat You Ship

Your product has ten years of customer data behind it and you have never treated it as an asset. An AI-native at twenty people can reach feature parity in eighteen months. A briefing for CTOs, CPOs, and CPTOs at mid-market European vertical-software firms.

PREPARED BY

Johan Grönstedt · GRAIL

FOR

European mid-market vertical-software technology and product leaders

DATE

April 2026

Monday 9:15, R&D review

Every CTO at a mid-sized European vertical-software firm has had the same Monday. Sprint velocity up twenty-two percent quarter on quarter. Copilot adoption at seventy-one percent. Cycle time flat. Your VP Engineering opens the review with the dashboard. He is mid-sentence when your phone buzzes. It is the CTO at your biggest customer. One sentence: "We just watched a demo from an AI-native in our space. Can we talk this week?" Your VP Engineering mentioned a recruiter call from a well-funded entrant in your vertical over coffee Thursday. Your CEO has a rebuild-or-buy decision on his desk from last week's off-site. Your CFO forwarded the Procore AI-agents announcement this morning: "Worth discussing before Q2 planning."

You have run this function for ten years. The pattern used to be simple. Hire engineers who understand the vertical. Hold the review bar. Ship the roadmap. Protect the moat built on domain depth. The business was understandable.

Something changed. A well-funded AI-native at twenty people can reach feature parity in 12 to 18 months where your five-year incumbent build took 150. In construction alone the named entrants are already shipping: Unlimited Industries on design configuration, PermitFlow on permitting, Buildots on site monitoring, Procore's own AI agents on contract generation. Every vertical has its version. Ninety-five percent of corporate AI deployments produce no measurable P&L impact. Your VP Engineering carries architectural reasoning that has never been written down. Two founding engineers have had recruiter calls this quarter.

You are not running one R&D function. You are running two, and only one of them is on your scorecard. One ships the current product. The other encodes what has to exist by 2028: the ten years of customer telemetry you have never turned into a training dataset, the domain reasoning sitting in three founding engineers' heads, the integration depth across your top customer systems that an AI-native cannot replicate without your customer base. On one hurdle rate the first wins every quarter. On one scorecard the second does not exist.

This briefing covers what is happening in your sector, what it means for the R&D function you run, and the three questions your team should answer before the next board cycle.

Vertical software just lost the equation that priced the moat

The AI-natives in your vertical are not experimenting. They are building. In construction alone: Unlimited Industries raised twelve million in seed funding from a16z to generate hundreds of thousands of design configurations on AI. PermitFlow raised fifty-four million Series B on AI permitting. Buildots is cutting site-monitoring delays in half. Procore announced AI agents in November 2024 compressing days-to-hours on contract generation, RFI resolution, and daily logs. Every vertical has its version. Whatever your vertical, there is already a well-funded entrant, or there will be one before the year closes.

The moat your product ran on for a decade has moved. The technical barrier to building a competing product collapsed. A capability that used to take a 150-person team five years to ship can be rebuilt by a 20-person AI-native in 12 to 18 months, on a modern architecture, priced at a fraction of your per-seat number. Your feature edge was never going to hold. What can hold is the layer underneath: the accumulated understanding of your vertical. Why the data model works this way. What the regulator actually enforces vs what they publish. What customers need but have never articulated. How your top-twenty accounts actually configure the product. That intelligence compounds. It takes years to accumulate and it cannot be bought. An AI-native cannot ship it because they do not have your customer base.

The feature you ship this quarter on GPT-5 is on an eighteen-month obsolescence curve. Your ten years of customer telemetry is the asset the AI-native cannot replicate without your customer base.

The industry's speed mismatch is the third force. AI-natives build from scratch with no legacy, no five-year roadmap cycles, no regulated customer environments they have to learn before they can ship. You have all three. You cannot pivot at their speed on their terms. What you can do is identify what is genuinely defensible in your product, stop defending what is not, and encode the domain knowledge the AI-native cannot build without you. The firms that are already moving are quiet about it. The ones that are shipping "AI-powered" badges on existing features are the ones the board replaces in 2028.

Your R&D function is feeling it every day

Your build engine. Copilot adoption at seventy percent. Cycle time has not moved. Velocity is up double-digit on the dashboard and people are pushing ten-thousand-line PRs barely reviewed. The foundational layer rots. Twelve months in, your team cannot execute a rebuild when the board asks for one. The review discipline that worked in 2022 is training the wrong practice on agent-generated volume now, and in a regulated vertical the production bar is higher than it is in horizontal SaaS.

Your platform and data. Spotify's best developers have not written code since December because Spotify spent four years on the platform foundation. You cannot replicate four years in eighteen months. What you can do is turn ten years of operational exhaust into an asset. Usage logs, support tickets, CS notes, QBR transcripts, renewal-negotiation files, churn signals. None of it is on the finance team's radar. None of it is structured as training-ready data. All of it compounds the moment you start treating it that way. The model layer commoditises. The customer-intelligence layer compounds.

Your product thesis. Your board asked for the firm's three-sentence product thesis last quarter. Your answer was the one from three years ago. Your domain model assumes AI-native entrants will still need five years to learn your vertical. They do not. Procore shipped AI agents in November 2024. The rebuild-vs-layer-vs-ride decision is on your desk. Three paths. One of them is already the default, and it is the one that does not compound.

Your R&D bench. You stopped hiring juniors in 2024. The routine work that trained juniors into seniors is now running on agents. No routine work, no training ground, no senior bench in 2030. And your three founding engineers carry why the data model is shaped the way it is. OpenAI and Anthropic both resumed junior hiring in late 2025 specifically to pair juniors with seniors on judgment-intensive work. The firm that cuts for this year's number has a cheaper function in Q4 and an empty bench in five years.

Your board-level authority. Thirty percent of European AI-strategy ownership sits with the CIO, twenty-three percent CDAO, twenty-one percent CAIO, one percent COO. You are not on the list. Not because you abandoned the mandate. You have not yet claimed it. Your CEO has a rebuild-or-acquire decision on his desk. Every month without your authored thesis, the decision drifts toward whoever shouts loudest. Your Q1 boundary agreement with the CEO lands before the sponsor asks who owns the architecture.

The moat in vertical software is your dataset, not your features

Your vertical is repricing from below because AI-natives reach feature parity in eighteen months at a fraction of your per-seat economics. Your function is compressing because five engineers now produce what fifty did, and the review bar that worked for fifty cannot hold the volume five produce with agents. Same force, two altitudes. But there is a third thing happening that most vertical-software CTOs have not named yet.

The moat in your vertical is not above the model layer. It is in the ten years of customer telemetry your product has been generating and you have never treated as an asset. The feature you ship this quarter on GPT-5 commoditises when the next foundation model rolls in 2027. What does not commoditise: the usage logs, the support tickets, the QBR transcripts, the renewal-negotiation files, the domain reasoning sitting in three founding engineers' heads. A twenty-person AI-native cannot build that layer because they do not have the customer base. You do. The window closes in roughly eighteen months, which is the time between now and when the AI-native has enough of their own customer footprint to start building the same layer.

In 2015, a vertical-software firm ran the hospitality point-of-sale category. Twenty-plus years of domain depth. The top chains ran on it. Oracle bought them in 2014 for five point three billion dollars. A newer firm shipped a cloud product on the same feature set, with an architecture the incumbent could not retrofit. By 2021 the newer firm was public at roughly twenty billion. The incumbent had become a line in Oracle's segment report. The difference was never the point-of-sale. One firm treated real-time sales data, payroll data, and ordering data as the asset. The other treated them as operational exhaust. Same vertical. Same customer base. One firm turned the data into a moat. The other did not.

You can tell which camp a firm is in from a single question. Is the customer data your product has generated for the past decade a structured asset on its way to becoming a training corpus, or an unstructured exhaust nobody has audited? Most vertical-software incumbents cannot answer. The ones that can have already made the shift.

You have twelve to eighteen months. The CTOs who claim the authorship of this decision in Q1 author the decade. The ones who wait until the AI-native signs the next biggest-customer pilot inherit a seat that has narrowed to release cadence and feature velocity, and a scorecard that stopped mattering the quarter their product team started arguing with the entrant's demo.

What most firms are doing. What the winners are doing instead.

Every vertical-software CTO has had the AI conversation. Most are making the same four decisions. A smaller group is making different ones.

THE DEFAULT PATH

Ship Copilot, track adoption

Seventy-percent adoption. Cycle time flat. Large PRs barely reviewed. Foundation rots under a green dashboard.

Layer AI features on the legacy stack

Ships this quarter. Demo-able to the board. Moat does not compound. Three years in you are the 2022 product with AI wrappers.

Treat domain knowledge as implicit

"We know our vertical." The knowledge is in three founding engineers' heads. Two of them just had recruiter calls.

Cut the junior layer for the Q1 headcount win

Cheaper function this quarter. Empty bench by 2031. External CTO hire at a premium the firm cannot absorb.

THE WINNERS' PATH

Buy the tool layer, redesign the review bar

Review-time floor per line. ADRs mandatory on agent-generated changes. Eval harness tuned to your vertical's constraints.

Encode the dataset before the AI-native has one

Ten years of usage, tickets, QBRs turned into training-ready data. Data-rights renegotiated on renewal.

Extract the reasoning before the notice period

Agent-assisted interviews with founding engineers. Why this data model, why this regulatory edge case, why this workflow.

Pair juniors with seniors on composite work

Eighteen-month apprenticeship on domain judgment, not five years on routine. OpenAI resumed junior hiring for this reason.

Same engineering team. Same twelve months. Opposite outcomes. The difference is whether the CTO authors the architecture decision or signs the requisition.

THREE QUESTIONS

Build Velocity. Product Defensibility. R&D Capital Allocation.

Three questions every vertical-software CTO is tracking. The second is the crux. The first and third are how you earn the right to answer it.

01 · BUILD VELOCITY

Is our engineering speed shipping production-grade output, or demos that fall over?

Copilot adoption at seventy percent. Cycle time flat. Ten-thousand-line PRs barely reviewed, foundation rotting. Spotify spent four years building the platform that lets agents ship cleanly. You cannot replicate four years in eighteen months, and in a regulated vertical the production bar is higher than in horizontal SaaS. Buy what the tool layer can carry. Own the discipline that makes it ship.

02 · PRODUCT DEFENSIBILITY

What does our product have that an AI-native at ten percent of our economics cannot copy?

Your feature edge commoditises every eighteen months when the next foundation model rolls. Your ten years of customer telemetry compounds. Your domain reasoning sits in three founding engineers' heads, and two of them had recruiter calls from AI-natives last month. The moat sits underneath the model layer, not above it. The window to encode it is eighteen months.

03 · R&D CAPITAL ALLOCATION

Rebuild, layer, or ride? Which path does your capital actually fund?

Three paths compete for the R&D budget. Rip and rebuild on foundation-model architecture. Layer AI on legacy. Ride current ARR and acquire an AI-native tuck-in. On one hurdle rate the layer-on option wins every quarter. On one scorecard the rebuild does not exist. The CTO who walks into the board with one budget runs the same programme every peer is running.

What the firms winning this play are actually doing

Four concrete moves any vertical-software CTO can start this quarter. Each addresses one sub-function. Pulled together, they become one architecture the firm builds on.

1**Build engine: absorb agent velocity without the foundation rotting**

Instrument review-time-per-line as a floor, not just cycle time. PRs above five hundred lines require two senior reviewers and thirty minutes of documented review. Architectural decision records mandatory on every agent-generated change above a complexity threshold: no ADR, no merge. Evaluation harness coverage tuned to your vertical's regulatory constraints as a first-class metric. Rebalance the senior-to-junior ratio around the composite rather than the 2022 team shape. The dashboard is no longer velocity. It is review depth against output volume.

2**Platform and data: build the customer-intelligence corpus**

Ten years of operational exhaust turned into a structured asset. Usage logs, support tickets, CS notes, QBR transcripts, renewal-negotiation files. Audited for customer consent, redacted where required, structured for retrieval and for model grounding. Data-rights renegotiation across your top-twenty customer contracts tied to your next twenty-four months of renewals. Integration depth graph published for your top-twenty customer systems, maintained as structured metadata. The AI-native cannot build this layer without your customer base.

3**Product thesis: encode the domain knowledge before notice periods**

Agent-assisted extraction interviews with the three to five founding engineers who carry why the data model is shaped the way it is, why this regulatory edge case exists, why the top-twenty customers configure the product the way they do. Not documentation for the wiki. Institutional infrastructure. Structured, versioned, queryable, tied to the data model and the roadmap. Your three-sentence product thesis updated every quarter, published to the board before the board asks for it.

4**R&D bench: rebuild the junior pathway around senior and agent pairing**

Your 2024 decision to stop hiring juniors saves Q1. It empties the 2031 senior bench. Reverse it. Juniors in 2026 do not start on routine code. The agent layer absorbed that work. They start on audit, exception-handling, and paired composite work with a named senior for their first eighteen months. The career curve compresses from five years to eighteen months because the senior's judgment is now legible in the agent-and-junior review loop. By 2029 you have a mid-tier bench the competition cannot hire away.

The only firm that joins R&D authorship to working infrastructure

The vertical-software CTO has four places to go. None can do what we do.

The big tech-transformation practices ship a standardised framework scaled across a sector book. Your seat is idiosyncratic. A reusable framework on it produces governance theatre you cannot defend.

The hyperscaler professional services arms price the engagement as enablement for cloud consumption. Every build-or-buy decision routes through their pricing for three years.

The developer-productivity tooling vendors have a quote on your desk. Cursor, Copilot, Claude Code. Their product is a build-velocity intervention measured on seats. Telling you to spend freed capacity on the dataset instead of expanded seats is structurally impossible for them to say.

A tuck-in acquisition of an AI-native in your vertical looks clean on the sponsor slide. The authorship transfers to whoever you bought. Your founding-engineer dataset does not come with it.

The four options you have already looked at. None can run the architecture conversation your vertical requires. GRAIL is the fifth.

WHAT YOU GET

Senior advisory that speaks engineering-review, not transformation-deck

Working infrastructure your team uses the same week

Custom agents built on your code, your customers, your stack

The start of the customer-intelligence corpus your product owns

No vendor lock-in at any phase of the engagement

WHY IT MATTERS

Founded by operators who have carried CPO-adjacent seats

Business first, technology second

Three weeks from first meeting to working infrastructure

Your team keeps the infrastructure, you own the IP

Every next step earned by what just shipped, never assumed

Three sessions with your R&D leadership. Working infrastructure from week one.

We start with your R&D leadership team because the change cannot outrun the team's personal understanding of what is possible. Three workshops. Each delivers standalone value. Each earns the right to the next.

WORKSHOP 01

How to think

Walk through what is happening in your vertical and inside your R&D function. Build the first piece of infrastructure in the room: a rebuild-vs-layer-vs-ride agent that takes your real R&D budget and runs the alternatives analysis against your named AI-native competitors. Your leadership team leaves with a shared view and the first working artefact already in hand.

WORKSHOP 02

Your R&D infrastructure

A half-day build. Your VP Engineering, Head of Product, Head of Platform, and Head of Data each leave with five to ten pieces of working infrastructure tuned to their function. Review-bar audit agent. Architectural-decision scaffolding. Domain-knowledge extraction interview agent. Customer-telemetry corpus prototype. Infrastructure your team is using the same week, not decks about infrastructure.

WORKSHOP 03

Ready to run

Refined infrastructure based on real use. The board document that names the rebuild-vs-layer-vs-ride path with its own hurdle rate and kill criterion. The boundary agreement with the CEO on who authors the architecture decision. The eighteen-month encoding plan for the domain knowledge currently sitting in founding engineers' heads.

Three weeks. No IT project. No board approval needed to start. The infrastructure your R&D function is already running on when we finish.

For your next R&D review

The questions where your R&D leadership cannot agree on an answer are the ones worth an hour on the agenda. Ask these honestly.

Is your R&D budget today one instrument or two? What does the board see that splits running the current product from encoding the dataset, and what is the kill criterion on each?

Name the AI-native in your vertical. What is their funding round, what customer did they just pilot, and at what percent of your unit economics are they quoting? If you cannot name the top three, that is the first thing the briefing changes.

If your two founding engineers left for an AI-native tomorrow, how many months would it take to reconstruct the reasoning behind your data-model and your top-twenty customer integrations?

Copilot adoption is at seventy percent. Cycle time has not moved. Where did the freed hours go, and is your review bar still the one you want the foundation layer trained on in a regulated vertical?

Rebuild, layer, or ride. Which path has your CEO already committed to in the hallway, and is your written architecture thesis in his hand before the next board pack, or after?

If any of these produces a disagreement in your R&D leadership, that is the conversation worth having before the board asks.

IN THIS SERIES

There's a version of this briefing for every seat at the table.

GRAIL has produced industry briefings calibrated to the specific reality of each C-suite role.

CEO

Revenue. Margin. Valuation.

Three numbers your board reads. AI is rewriting all three in your industry.

CFO

Capital. Forecast. Story.

What you allocate this year is the story your board reads next year.

COO

Cheaper. Better. Faster.

The triple threat that used to be a tradeoff is now table stakes.

CHRO

Capacity. Capability. Promise.

The biggest talent transition of a career. Compete on capacity, not headcount.

CPTO

► YOU ARE READING

Moat. Speed. Allocation.

The platform moat that survives 2028 is being chosen this year.

CCO

Margin. Pace. Position.

The whole GTM playbook is being rewritten in real time, across every commercial team at once.

Visit grail.works for the full series.

GRAIL

An AI-native management consultancy helping European mid-market leadership teams figure out what fundamentally changes about their business as AI enters their industry.

www.grail.works