



**MENA
LEADERS
COMMUNITY**
PARTNERS OF 

August 2026

AI. Served Fresh.

AI breakfast report

What five tables of builders, CTOs and operators actually said when you got them talking

salt[™]

The one-line version

Every table thought it was talking about a different problem: engineering workflow, enterprise agents, ROI measurement and they all landed on the same three words:
ops, trust, data.

Nobody at this breakfast is stuck on model capability anymore.

Everybody is stuck on the plumbing underneath it and on whether anyone believes the output enough to act on it without checking.

If there's a single sentence that captures the room, it's the one a fintech-heavy table wrote down almost as an aside:

"Ops is the lever and trust is the denominator."

What to expect in this report

A synthesis of five roundtable discussions from a recent AI leadership breakfast, drawing together perspectives from engineering, fintech, payments, cost and data leaders across the room.

01. The SDLC didn't evolve. It inverted, in five months
02. The wall everyone hits isn't technical. It's regulatory, organisational, or about trust
03. Measuring AI's value: why finance's usual metrics miss AI's real wins
04. The data underneath the agents is the actual problem, not the agents
05. AI didn't remove cognitive fatigue, it moved it
06. The skills gap is real, immediate, and nobody is training for it
07. Where the room actually disagreed
08. The action items people wrote down

Theme 01

The SDLC didn't evolve. It inverted, in five months.

This was the most energised topic across every group and the timeline is the shocking part. Multiple tables, independently, described the exact same shift happening on the exact same clock: February to now.

Before: write the product roadmap, review it, write the code.

Now: write the requirements, test, push the code.

The requirements-writing step used to be a formality that preceded engineering. Now it is the engineering bottleneck and it's being done by people who never used to write specs at all. PMs at one company prototype directly in Lovable, then hand off to Claude for design. No engineer touches the sandbox until it's time to harden it. Some organisations now let PMs commit code in sandboxed branches. One team went further and built an internal "Replit" that let a PM, not an engineer, ship a custom feature for a single large B2B client in two days, collapsing the time from idea to customer feedback to almost nothing.

That's the superpower. Here's the problem it created, in the room's own words.

Better spec'ing bloats scope.

When anyone can write a detailed spec, "detailed" doesn't mean "well-scoped." Non-technical people can now describe everything they want, in depth, which is a new failure mode, not a solved one. The fix people landed on was more upfront research to narrow scope before spec'ing, not after.

The architecture pattern that kept coming up:

A deterministic wrapper around a probabilistic, multi-model core (teams cited setups like Onyx or Pi routed through OpenRouter). The system around the LLM has to be deterministic, or, as one participant put it, you're just burning tokens.

Autonomy causes breakage.

The time that used to go into writing code now goes into testing, backward compatibility and human-in-the-loop QA. The instinct is to fail fast: get in front of the real customer as early as possible rather than polishing in isolation.

The uncomfortable one:

Research and design automation is quietly abstracting PMs away from the customer and domain understanding that used to be their entire job. On the engineering side, the honest answer to "who knows the answer here?" is increasingly "Claude" and more than one person flagged that as itself the problem, not the solution. Product quality still has to be owned by a human PM, or nobody owns it.

This is the same phenomenon the enterprise-agents table called context drift: PMs and engineers slowly losing the domain expertise, market knowledge and deep customer understanding that used to accumulate as a side effect of doing the work by hand. When the work is delegated, the side effect disappears too.



Theme 02

The wall everyone hits isn't technical. It's regulatory, organisational, or about trust

Three separate tables, three separate domains, same conclusion.

Fintech's wall is dev-to-prod, not dev.

The value is real and already proven in ops: one team cut ticket volume 65–70% automating KYC, reconciliation and remittance workflows. But getting from a working prototype into production is blocked by regulation and institutional trust, not by model quality. The common workaround right now is running on-prem.

Payments themselves are quietly working, though.

One team's automated validation agents, checking deposits, withdrawals and fraud indicators, now handle roughly 92% of payment approvals with a human layer on top. Payment validation is one of the more mature, trusted agentic use cases, while payment processing itself is where clients get nervous.

Payroll clients are drawing hard lines.

Some large global clients explicitly forbid AI anywhere near payment processing, full stop, no exceptions for "just the validation layer." The complication: modern codebases touch AI implicitly at this point, which makes a clean compliance guarantee genuinely hard to give.

Agentic commerce is forming, but adoption hinges on trust.

Visa and Stripe are actively building rails for agents to transact on a human's behalf. The room was split: some see clear momentum, others were openly sceptical the adoption curve is anywhere close. Agentic payments look like an easier problem than agentic commerce broadly, since moving money is more constrained and auditable than judging what a shopper wants.

Trust isn't uniform across a single value chain: it's earned function by function. Across all three domains, the pattern repeats. Technical capability is rarely the constraint; regulation, institutional confidence and compliance clarity are.



Theme 03

Measuring AI's value: why finance's usual metrics miss AI's real wins

This came out of the cost-focused table almost as a warning to the rest of the room. Traditional finance has exactly three lenses: revenue up, budget down, headcount down. AI's actual wins—better customer experience, preserved institutional knowledge, faster onboarding, happier employees—don't map cleanly to any of the three. That mismatch creates real organisational resistance: nobody wants to sponsor a transformation project whose success will be judged by metrics that don't reflect what the project actually did.

R&D and experimentation suffer worst here, because early-stage AI work is supposed to fail sometimes and that's the hardest thing to get budget approval for in a finance-first culture.

The case study that cut through this most clearly was an events-industry company (chairs, event rentals—not a company anyone expected to be a case study). Cost savings from AI were almost beside the point. What actually moved:

01. A 700+ SKU catalogue (just chairs) became instantly searchable, so staff could match inventory to a specific event's needs in seconds instead of institutional memory or guesswork.

02. Vendor selection shifted from relationship-driven picks to objective price/quality comparison, which incidentally improved compliance and killed a favoritism problem nobody had budgeted to solve.

03. Agents could prep staff for oddly specific customer requests (the example given: a Russian family's birthday party in a Dubai villa) using LLM research that would otherwise take real staff-hours.

04. New hires, especially the weaker performers on sales and communication, could perform at a much higher level immediately, because the knowledge base and scripts did the heavy lifting.

None of that shows up as "cost cut." It shows up as service quality and customer satisfaction, which is exactly the kind of intangible improvement traditional finance frameworks aren't built to value. The table's read: organisations need genuinely new methods to measure and communicate this and the honest framing of "AI digitisation vs digitalisation" is the right conversation to have, but it's a politically expensive one to start internally.

The trap sitting right next to this: token-maxing without output. One case saw AI make customer support meaningfully faster — and customers responded by just using the channel more. Satisfaction went up. Spend didn't. The team had to actively adjust model output just to keep the economics sane. Faster isn't automatically better if the underlying business doesn't capture the upside.



Theme 04

The data underneath the agents is the actual problem, not the agents

This is the point where "enterprise AI" stopped sounding exciting and started sounding like a systems-integration problem, which — depending on who you ask — is either deflating or the most useful reframe of the morning.

The core complaint, nearly verbatim from the room:

"Not the agents, the data underneath them. Spread across Notion, Slack, SharePoint, Jira, and nobody updates documentation once a project moves. PRDs go stale, people update the ticket and not the doc, and then you're asking an agent to decide on data that's already wrong."

Underneath that, two more layers:

- **Access control.** An agent that can synthesise across every system in the company can also, by accident, tell someone what a client paid you. Role-based gating has to exist inside the agent layer, not just at the door.
- **The human bottleneck.** When agents can generate ten times the PRs, drafts, or recommendations a human could produce, review capacity doesn't scale with them. This is where the fatigue actually lives — not in doing the work, but in checking work you can't physically keep pace with. One striking real example: a sales rep told a customer they'd spent Dh50,000 when the real number was Dh5,000. An order-of-magnitude hallucination that made it all the way to a customer because no one caught it in time.

The best answer anyone had for this was Daniel's team's approach: build the "second brain" as an index, not a store. Instead of copying everything into one dumped repository (the instinctive move and the wrong one), point the AI layer at the existing sources of truth, code repos and approved docs. Then let it reference rather than replicate. It's a cleaner fix for the centralisation problem than the more common instinct to funnel everything into a shared drive and hope.

Two other approaches surfaced in parallel: role-based access controls built directly into the agent, with a single gated access point (Slack, in one team's case) rather than open querying; and Claude-based aggregation across Notion, Linear and internal databases, letting the model build its own working context on the fly rather than maintaining a static index at all. Nobody claimed a definitive answer—three credible teams, three different architectures, none obviously wrong.



Theme 05

AI didn't remove cognitive fatigue, it moved it

A study came up that stopped the room: radiologists using AI for cancer detection saw a 30% performance drop after three months. The instinctive read is "AI made them worse." The counter-argument that won more of the room: AI absorbs the easy cases, so the humans left in the loop are disproportionately handling the hard ones. The team isn't necessarily worse — they're facing a harder distribution of problems than before, which will look like a performance drop on average difficulty-adjusted metrics.

The reframe that got repeated: fatigue doesn't disappear with automation, it shifts in nature. Process-driven, low-effort work moves to agents. The humans left are making more active, higher-stakes decisions, more frequently, with less of the "easy win" work to recover between them. And the real cognitive load was never in the execution anyway—it's in the preparation: scoping the problem, defining the PRD, deciding what to even ask for. AI accelerates execution and leaves prep exactly as hard as it always was.

One line from this conversation is worth sitting with on its own:

"You can outsource your thinking, but you should not outsource your understanding."

There's a wrinkle nobody had a clean answer for: the conceptual leaps that used to happen in hallway or water-cooler conversation (tone, sarcasm, gesture, the stuff that makes two people suddenly see a problem differently) aren't things transcription or ambient-capture tools (Meta glasses got a specific mention) can replicate yet. AI can surface information. It can't yet make the associative leap two humans make talking past each other in a hallway



Theme 06

The skills gap is real, immediate and nobody is training for it

Two sides of the same coin came up here.

The junior engineering problem:

Juniors are now producing work — in one case, a genuinely better architecture than a senior engineer had proposed — without being able to explain why it works. They have output without understanding. This is the SDLC-inversion problem and the context-drift problem meeting at the individual level

The macro problem:

engineers are meaningfully ahead of every other department on token-cost optimisation and efficient prompting — sales teams in particular burn tokens far less efficiently, mostly from unfamiliarity rather than laziness. Multiple teams have responded with company-wide budget caps plus regular prompt-engineering coaching sessions, on the logic that a budget without training just produces frustrated non-engineers, not better habits.

Zoomed out further, the room's honest read was uneasy: institutions haven't defined what the future jobs even are, governments aren't training towards AI-adjacent roles (QA, AI management, prompt engineering) and there's a real risk of a genuinely societal problem — people displaced with no defined retraining path, not because AI destroyed their job outright but because the surplus productivity has nowhere obvious to go.

The optimistic counter-frame, also voiced in the room: every prior technology wave (the wheel, the industrial revolution, the car) displaced jobs and still raised aggregate wealth. The stated worry this time is that wealth concentration looks more extreme — a first trillionaire is imminent — though the counter-counter-argument is that concentration itself may not be the problem if the floor is also rising. Someone brought up Hans Rosling's point that global living standards have broadly, measurably improved over decades — with the caveat that his data predates social media and depression/social-comparison effects now complicate any simple "things are getting better" story.

The bluntest line of the morning belongs here too, unfiltered:

"Even AI does not have a chance until you root out the cause of inefficiencies: corruption, hiring family members, incompetence. I'm not holding my hopes up."

Worth including precisely because it's the one voice in the room that thinks the bottleneck was never technological at all — and given everything above about ops, trust and organisational structure, that view isn't easy to dismiss.



Not everything converged.
A few genuine fault lines:

Where the room disagreed

Is agentic commerce close or overhyped?

Some cited Visa/Stripe infrastructure as evidence adoption is underway; others were flatly sceptical the trust problem gets solved soon. Nobody resolved this.

Are humans getting worse or just facing harder problems?

The radiology stat split the room between "AI degrades skill" and "AI redistributes difficulty."

Is AI reducing tech's importance or increasing it?

One view: business requests stay fundamentally simple (KYC docs, propagating consistent data across systems) and always were — AI just makes the plumbing faster, not the requests smarter. Another view: the tech stack and architecture decisions (deterministic wrappers, indexing strategy, access control) are now more central to getting any of this right, not less.

Inequality: problem or side effect?

Raised without resolution: if AI-driven wealth concentration accelerates but the global floor also rises (per the Rosling framing), is that actually the crisis it's assumed to be?



Moving forward

Action items people wrote down

Engineering & workflow

Three-tier task system: small/low-risk changes go to PMs or designers, engineers review and deploy after AI-assisted validation.

Expand QA automation into load testing & continuous automation suite maintenance, gated by AI code-review blockers.

Sync internal documentation directly with tickets; auto-generate test scenarios and acceptance criteria from that sync.

Separate documentation into draft/verified/archived tiers. Keep unverified and legacy content out of what the AI actually consumes.

Governance & cost

Middle-layer filtering to stop sensitive data (especially payment/finance data) reaching AI systems unfiltered.

Company-wide token budgets, paired with actual training. A cap without coaching just produces friction.

Push multi-model orchestration (a strong planning model for architecture, a cheaper model for implementation) to control cost without losing quality.

Start new sessions for code review specifically. Models defend their own prior output inside the same session and lose that bias in a fresh one.

Strategy

Align AI use cases to existing OKRs so prioritisation has a shared reference point (understanding not all orgs have an OKR framework).

Treat the "second brain" as an index into sources of truth, not a copy of them.

Build role-based access control into the agent layer itself, not just at the system perimeter.

Actively route freed-up engineering capacity at technical debt before new features — several people described this as the actual point of AI velocity, not shipping faster.

The line to end on

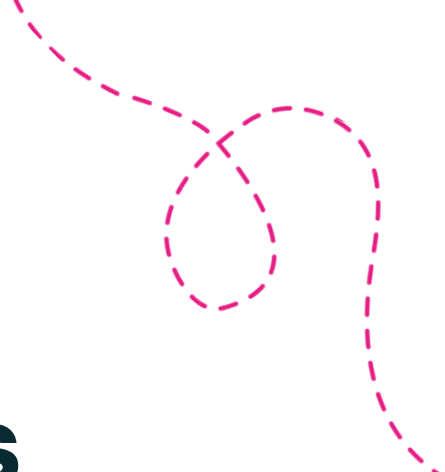
The real conclusion

Someone summed up the epistemic state of the entire room better than any single insight did:

***"The world hasn't actually changed, it's just happening faster.
Same human problems with AI amplifiers."***

Every hard problem discussed that morning—trust, data quality, org structure, who gets credit, who gets blamed when the model is wrong—predates AI by decades.

What's changed is the clock speed. Five months to invert an entire development lifecycle is not normal, and nobody in that room, across three tables and a dozen industries, sounded like they'd fully caught up to their own timeline yet.



Salt Solutions

Need AI talent that can actually deliver?

Salt specialises in placing AI and digital transformation leaders across the Middle East, Asia Pacific, Africa and Europe. We work with organisations at every stage. from first AI hire to scaling a global team.

Global talent solutions



welovesalt.com/insights



Connect on LinkedIn