

Resolving the Category Error in the Enterprise

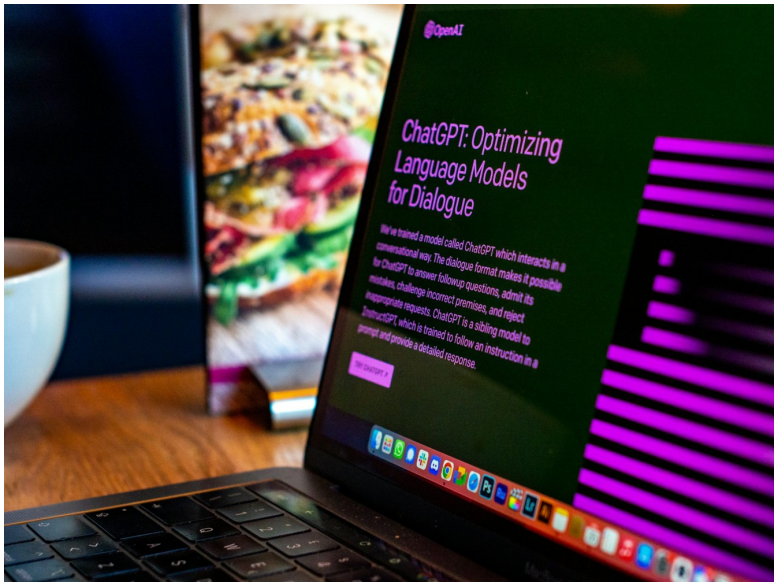
Davin Patton
Building products and strategies from the intersection of finance, engineering and GTM | Founder, Special Missions Group & Edge.VI

July 6, 2026

In Q1 2026, AI-driven investments accounted for the majority of all US real GDP growth. A collapse in spend threatens a market-wide correction that extends well beyond the technology sector. In the midst of this, a 2026 survey of nearly 3,000 executives found that roughly three in four say their AI investments have failed to meet expectations, backing broad anecdotal evidence of corporate America struggling to find ROI on AI spend.

You might think this is happening because these initiatives are poorly conceived, but they're not. This is happening because a category-level error is being made that existing playbooks aren't designed for. That's why token usage mandates proliferated, why AI directives issued from the top of organizations consistently fail to result in profitable output, and why the next corrective steps of establishing ownership, limiting model usage, and implementing change practices within this construct won't work either. Across industries, this category error has created significant risk exposure to both top line revenue and SG&A efficiency that hasn't yet been priced in because the market broadly sees AI as a product if not a solution.

Operators who remain disciplined against this market pressure and understand that AI is a non-deterministic technology that only performs as well as the fundamentals surrounding it are the only ones positioned to extract long term value.



The Category Error: AI is Not a Product

AI is fundamentally and broadly misunderstood, not only in theory but also in practice. Walk into any boardroom today and ask how growth will happen. The answer is AI. Ask what the new internal tooling strategy is. AI. Ask how efficiency will improve. AI.

These answers are placeholders presented as strategies: unless you are a model provider, AI is not a product and it is certainly not a solution.

A product has definition. It has a customer with a specific problem, a lifecycle, roadmap management, a go-to-market plan, a financial strategy, stakeholder alignment, and risk mitigation built in from the start. It is stress tested across every one of those dimensions before it earns the right to be called a solution. AI is none of that.

AI is a technology. It is a capability that must be shaped, governed, and deployed against a specific problem by people who understand the problem, the tool, and the people who face the problem. Treating it as anything else doesn't accelerate results. It converts necessary hard work into debt while burning capital on the mistaken assumption that adoption alone is the answer.



When Adoption Outpaces Outcomes

Evidence that adoption is outpacing outcomes is now showing up across multiple firms. Meta and Amazon both had internal leaderboards ranking employees by token consumption, the implicit logic being that the volume of AI usage was a proxy for productivity. In a 30-day window at Meta alone, total usage exceeded sixty trillion tokens. Amazon's version had to be shut down after employees began running AI tools on trivial tasks purely to climb the rankings, driving up compute costs with no corresponding output. Microsoft documented the same pattern independently. What these three multi trillion dollar firms are each showing, by running the same experiment and arriving at the same result, is that when you measure adoption instead of outcomes, you get adoption without outcomes.

Uber is publicly describing what happens when the same confusion hits corporate budgets. After incentivizing its five-thousand-strong engineering workforce to adopt Claude Code, including through internal leaderboards tracking usage, the company burned through its entire annual AI tools budget in four months. Its COO called the moment of discovery "head-exploding." In a subsequent interview, he was direct about what the spend had actually produced: "That link isn’t there yet. Maybe implicitly there's more that is getting shipped, but it's very hard to draw a line between one of those stats and 'Okay, now we're actually producing like twenty-five percent more useful consumer features.'" During a hands-on demonstration the CTO ran, he consumed $1,200 worth of tokens in two hours. The engineers generating these costs, the COO noted, weren't thinking about the invoice because they never saw it.

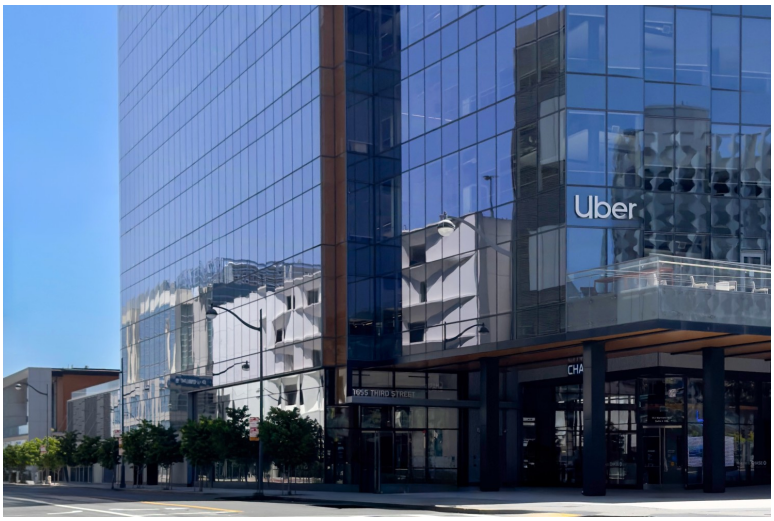


Photo by Dale Cruse, licensed under CC BY 2.0 (cropped)

These are among the most elite technology operators in the world. But they all made the same mistake: they treated a technology as though it were a solution, assumed that adoption was evidence of value, and built incentive structures that rewarded the former while never defining the latter. Silicon Valley is beginning to wake up to this, but the correction being seen among early corporate adopters isn't happening because leadership recognized the category error being made. It's happening because the bills arrived. Buyer's remorse isn't a strategy though, and it leaves the underlying problem intact. To be clear: it is still progress, but it is incomplete. You cannot integrate any technology into workflows without evaluating its cost structure, establishing governance, understanding its adoption implications, and defining what success looks like in terms a CFO can defend to a board.

Big Tech is at the forefront of this, serving as a leading indicator. Token usage mandates and adoption leaderboards remain active practice across industries well outside of the Bay Area, with non-technology firms including major retailers and consumer brands tying compensation and performance evaluations directly to AI adoption metrics.

The risk ahead is that either corporate America continues struggling to find AI profitability because it still isn't asking basic questions like "What is the ROI of our deployed model vs the nearest alternative?" or "What happens to our unit economics if API pricing normalizes?" or "What does our token spend actually produce?" Or they overcorrect. They mandate a blunt, one-size-fits-all model across all use cases, find it underperforms expectations, and conclude that AI is overrated. That conclusion, reached at scale, is dangerous because of the scale of the US's investments, and planned investments, into AI.



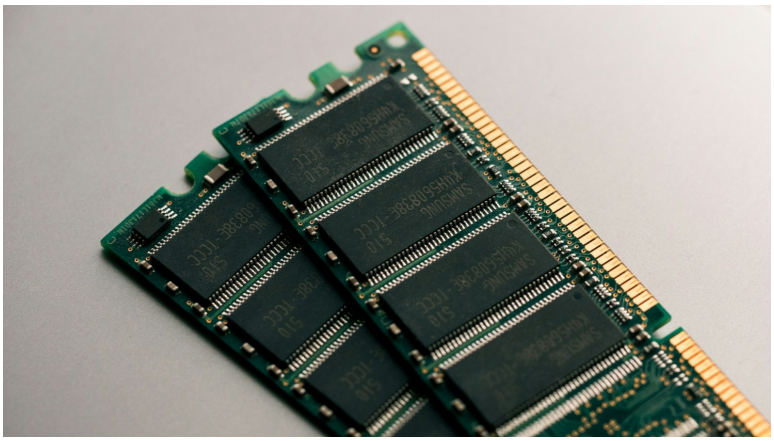
The Hard Unit Economics of LLMs

As high as the stakes are at a macro level, the challenge of profitable AI deployment is no easier at the micro level.

Companies run virtualization servers to provide digital services for employees: email, web browsing, teams, and so on. An Exchange server delivering Outlook to over a thousand users may be able to get by on 128GB of memory. And for the typical knowledge worker, if a dedicated virtual environment is required, Microsoft recommends 4-8GB of system memory per person.

Running a language model is fundamentally different. Inference requires loading an entire model into GPU memory. Today this can be 70 to 140GB for a mid-range model, not counting the inference engine itself (e.g. vLLM or similar). For every token generated, the GPU's processor must, in the simplest case, work across all of that memory. And as the conversation grows, so does the memory required to hold each conversation. A *single* conversation can easily require 30% or more in additional memory, even if the last message in the context window is simply "Thank you." All of this is on top of conventional infrastructure a company still has to run - databases, session handling, logging, orchestration, etc, all sitting on top of an OS/VM. Meanwhile, models are getting larger, not smaller. Today's proprietary frontier models are widely believed to be 1,000GB or much more, and conversation context windows have grown at least 5x in a single generation.

Batching prompts improves efficiency at scale, and model architectures can reduce how many of a model's parameters are actively computed against on any single inference pass. But agentic workflows, which are becoming increasingly common, require large sustained memory allocations that cannot be shared regardless of how many GPUs you have. And regardless of architecture, the entire model and its caches still have to sit in concurrent memory, whether or not every parameter gets used on a given token. The net result is that a single knowledge worker can consume 10 to 100 times the memory they would in any other corporate system, all while using specialized memory that costs at least 10 times more.



This means the largest tech firms have an advantage in the era of the LLM. They can afford the systems. They can maintain the infrastructure. And they have the scale to bring the unit economics down. But it's still hard.

This difficulty is exactly why, even as OpenAI scaled users and revenue dramatically, power consumption rose in lockstep. Despite some of the world's best engineers applying and inventing every optimization possible, from extremely aggressive response caching to infrastructure tuning, there's little public evidence its per-user economics have substantially improved. If anyone could have engineered their way out of this cost structure it would have been them, backed by nearly $200B in funding to date. This is also why OpenAI announced new inference hardware. They urgently need to improve efficiency, so they have designed their own ASIC, even if its adoption potentially limits significant future model architecture changes.

Most organizations have none of these efficiency levers. An additional user, especially at a smaller company, is dramatic extra cost. You don't have the same scale. Custom hardware is completely out of the question. So you have to over invest into costly infrastructure relative to your budget in order to even run an LLM to begin with: remember, if you cannot load the model, you cannot have an LLM, regardless of how short you keep your conversations. And that's not your only problem. Because the model providers and hyperscalers are buying so much memory, the costs of your non-AI systems and services are becoming dramatically more expensive as well, further eroding per user economics.



The Limits of the Disruption Narrative

Even if you did solve the infrastructure problem, you will likely have a harder time solving for what incumbents have built over decades. A large firm already has a product that can benefit from AI. A startup seeking to compete with DocuSign, believing the market's story that DocuSign is in systemic decline, will find that right after they code their app in days, they need to hire dozens of lawyers, each barred in every single jurisdiction they hope to operate in. Then, they'll need to hire a sales team, invest into passing audits, and more. Suddenly, it becomes apparent that AI is not simply 'the answer'. Worse, because of their smaller scale, their access to model rates will be worse than Docusign. Because they don't have formal processes, their code will likely have more vulnerabilities. And because their team will not be as connected to their own work, zero day security fixes and feature changes will require significantly more analysis and rework. DocuSign can indeed be written in days, but the business that is Docusign is built on hundreds of experts leveraging thousands of working hours of experience. That isn't easy to replace. The catch, of course, is that the incumbency must effectively operate its moat, and markets have struggled to understand and price this.

Google was briefly treated as an existential casualty of ChatGPT, the assumption being that AI was a threat to search. But the market had confused the interface with the stack. Google isn't just search, Google is one of the top hyperscalers, operating datacenters at scale. Its ad platform is one of the most important in all of advertising. YouTube has more video than any other platform on earth. Android is actively used by billions. Google itself originated the architecture modern LLMs run on, has a tier-one model in Gemini, and is now on its eighth generation ASIC built to run it efficiently. In Q1 2026, Alphabet reported $109.9 billion in quarterly revenue, up 22% year over year, with operating margins expanding to 36%. No startup is credibly attacking more than one of these simultaneously, and the numbers are what owning all of them looks like.

If a startup did succeed, among the best and most likely outcomes is acquisition by an incumbent, not the replacement of one. In the current macro environment, markets assign proportionately larger valuations to larger firms, creating a structural incentive for incumbents to acquire rather than compete. And an outsized raise now makes M&A more likely, not less, as larger firms are increasingly using acquisitions to rapidly absorb new technologies and desirable names in ways that markets reward with immediate price target increases, effectively subsidizing the financing of the deal itself.



But incumbency advantage does not a win make. The incumbent's advantage is structural, not operational. Having the moat doesn't mean that the people inside are doing the work correctly and protecting, or even recognizing and leveraging, the moat that they have.

As Meta and Amazon laid bare in a very clear fashion, major firms aren't strategically running the unit economics on their tokens. Internal employee sentiment shows that many companies aren't even performing model evals for ROI. Most likely, their teams are overworked and forced to adopt workflows they don't understand. They may even be forced to burn tokens to display productivity. This exposes real highly leveragable opportunities across multiple industries in deployment strategy and execution.



The Answer Hasn't Changed

The operators who will win are not the ones with the largest models or the biggest budgets. They are the ones who can answer a simple question that most organizations have never

formally asked: “What does good actually look like?” They’ll be closing real deals with real people who have real needs, one profitable engagement at a time, while everyone else is marketing the signing of blank terms with random names in random places at pricing no one’s checking. Is it fancy work? No. In the short term, markets may not even recognize it. But it’s work that yields a strategic advantage that is difficult to compete with, regardless of how large a competitor is and regardless of what happens in the markets.

What this work looks like in practice is less complicated than the industry has made it. AI is a technology. So it must be implemented like one. That means the same principles that govern every serious technology implementation apply here, not the bureaucratic overhead that passes for governance and not the compliance theater that substitutes for strategy, but the actual mechanisms by which any technology generates value. The first question isn’t "How do we integrate AI?” It is "What, exactly, is the objective and what does success look like?” A surprising number of organizations have abandoned discipline and aren’t asking this before moving straight to implementations.



Before any AI enabled launch, these key questions must be answered:

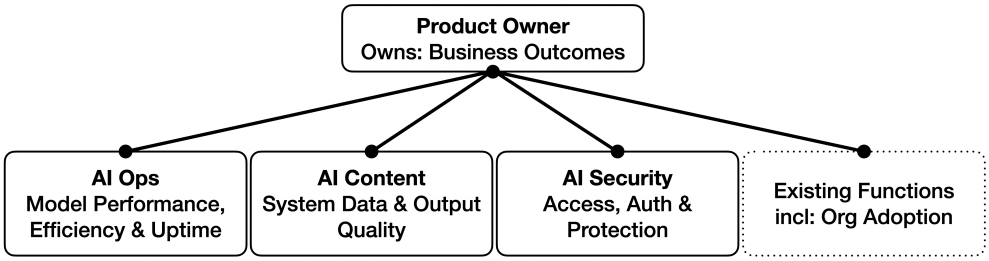
- Who owns this? Specifically, who will be notified of issues and is responsible for addressing them?
- What does good look like? Specifically, can the team create examples of what acceptable and unacceptable responses look like on the spot?
- Who does this change the work of, and what is the plan to prepare them?
- What are the reversions? Specifically, APIs fail, models fail, and prompting strategies fail, so what happens in these scenarios?
- What is the follow up plan to ensure the system is working as intended after 30 days, 60 days, 90 days, and so on?

All of these questions require answers prior to launch. Failing to answer them exposes the project to failure. Not the dramatic kind that's obvious right away. More often it's the quiet kind: the deployment underperforms, leading to it going unused, and spend behind it never shows a return.

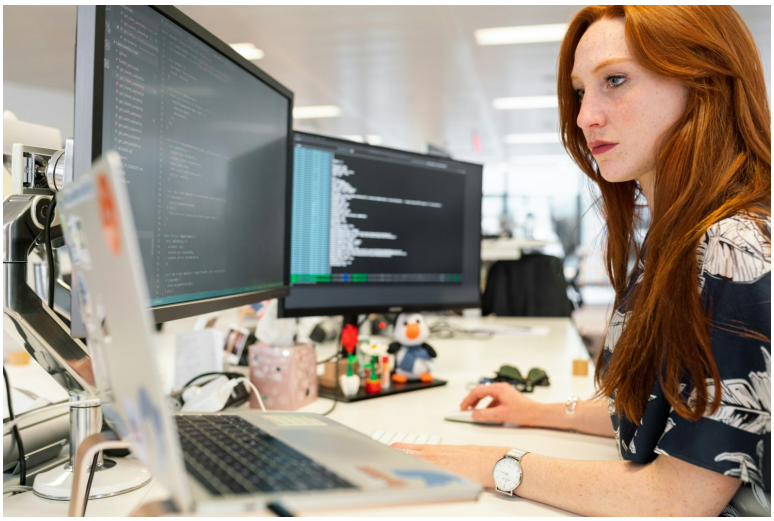


Discipline Must Meet the Org Chart

Technical implementations, while necessary, aren't where many deployments fail. Contrary to popular expectations, it has been my experience that implementations fail organizationally at an alarming rate. To address this, any team deploying an AI-enabled solution needs a product owner supported by three distinct AI specialist roles: AI operations, AI-facing content, and AI security.



The product owner is the role that owns the high level problem that AI is being deployed against. They define the objective, hold the success criteria, and are accountable to stakeholders for whether the deployment is delivering value for the customer. This is the role that ensures the AI enabled solution remains connected to the business problem it was built to solve, and that every governance decision made around it serves that purpose rather than substituting for it.



AI Ops is a virtual infrastructure and reliability function, typically operating across a full stack ensuring model consistency and performance across the defined requirements. For example, one of the most reliably expensive mistakes I've seen in AI deployments is defaulting to the most capable model available. You wouldn't retain a partner at an international law firm to contest parking citations, so why would you mandate a frontier model do textual summaries and classifications? This is a common mistake made industry-wide. On the surface, it would seem to incur a 10x cost penalty but in reality it is much more severe. This is because modern frontier models are trained to "think" about their answers. So they do that a lot, even for simple questions. The resulting difference between a fast, well-suited model and a frontier model for such a simple task could easily be 25x the cost. At single user scale, under a million tokens, this may only amount to a handful of dollars but across a busy team this can easily amount to thousands of dollars per day. AI Ops catches this by running evals on model performance against requirements.

Contrary to how it's viewed in many enterprise settings, AI Ops isn't a subject matter role. It doesn't own whether outputs are good enough, or whether the underlying data is stale, as that is the AI Content role's responsibility. And it doesn't own low usage as that is a delivery-level concern that sits outside of these AI specific roles being outlined. Assigning these roles to AI Ops is a governance failure because AI Ops cannot fill these roles, causing the system to fail quietly for reasons no one else on the team can clearly explain never mind correct.

AI Content is separate because it's one of the most common silent failure scenarios: it addresses situations where a solution works technically but quietly becomes useless because the underlying data has drifted and no one with both the responsibility and the knowledge to correct it was ever assigned.



The following dynamic that I've observed illustrates how this plays out: consider a B2B enterprise using an internal AI assistant to generate preliminary Service Level Agreements (SLAs) for its sales team. At the start of the quarter, leadership decides to modify its hypercare initial support period from three months to three weeks, with paid extensions available. However, the master playbook residing in the company's knowledge base is not updated to reflect this change. The AI continues to confidently auto-populate three-month hypercare periods into dozens of agreements.

From an AI Ops perspective, the deployment is a success: latency is low, token costs are optimized, and the model is retrieving data exactly as instructed. Yet, the business is unknowingly taxed on every newly closed deal. Triggered by the near-signing of an agreement on terms the business no longer intended to offer, staff now has to triple-check the most fundamental aspects of every SLA in a way they never had to when using static internal templates. You'd think this would create a lot of noise and formal corrective action. That's actually not what I've seen. In reality, the sales team will quietly stop using the system altogether. And AI Ops can't catch or correct this because they don't dictate the company's SLA strategy; they only manage the technical pipeline.

The AI Content role is non-technical and belongs to a domain expert with both the taste and subject matter knowledge to define what good looks like, maintain the examples that prove it, identify when outputs have drifted from that standard, and communicate specifically why in ways the technical team can act on. Without this role, there is no stable definition of quality in the system, and no one positioned to notice when that quality disappears.

A unique aspect of AI is that from the system's perspective, everything is identical content. A language model processes every token through the same probabilistic mechanism, which means the system cannot enforce a boundary between what it is permitted to do and what it has access to. A mathematical proof announced by NIST in June 2026, published in the peer-reviewed IEEE paper 'Robust AI Security and Alignment: A Sisyphean Endeavor?', makes the implications of this clear: there can exist no fixed set of guardrails that will be universally robust against adversarial prompts. There will always be a way in, it's just a matter of finding it. This is a problem that requires a new role: AI Security.



Consider a public-facing AI assistant integrated with an organization's internal knowledge base. A user who understands LLM architecture can surface data the system was never intended to expose, with no zero day exploits required. The model would retrieve exactly what it had access to, the pipeline would run as designed, and the content quality check would pass. The failure would be invisible to every role previously defined because none of them own that boundary.

The following example is more personal. A public-facing AI integration exposes metered billing infrastructure to any user of the system. Because no complete defense exists, a bad actor can drive sustained request volume against the team's API keys, effectively stealing their compute, and directly hit their budget. With no existing role monitoring external consumption, the attack would run until a billing alert triggers, if one exists, by which point the damage is already done. This is something I've personally observed, where an adversary used a public deployment to not only gain free usage, but also do so at a higher usage tier with higher end models (the company was using a not yet publicly available frontier reasoning model for a simple website chatbot). The bot was performing as architected with low latency and high reliability, all content was generated using models that had passing evals, and on the IT side nothing was wrong: no keys were compromised, access had been correctly enforced for the VM servers on internal subnets, and the servers hitting the API endpoints were correctly doing so via static public IPs tightly whitelisted on the provider side.

The role of AI Security addresses these risks. It governs data access scoping, authorization architecture, and the interfaces through which users reach systems, with continuous monitoring post-launch and a defensive posture that evolves as attack vectors do. It is distinct from AI Ops, which manages the technical pipeline and has no mandate over authorization logic. It is distinct from AI Content, which governs output quality. And it is distinct from IT, which provisions infrastructure to specification and is not positioned to govern the AI-specific attack surface that specification created.



AI Ops, Content, and Security are new AI-specific roles without precedent. But existing roles must be integrated into this delivery structure from inception as well, few as critically as organizational adoption planning: in the typical enterprise implementation, it arrives after the technical build is underway, at which point it becomes remediation rather than strategy, because the decisions that determine whether adoption happens at all have already been made. For AI, that sequencing is especially damaging. A non-deterministic system produces scenarios that require both early context and technical fluency to consistently recognize. A user who had a good experience on Monday and a poor one on Thursday is not resistant to change. They are responding rationally to inconsistency. Distinguishing that from a communication problem requires enough technical knowledge to know the difference. Bringing this function in late, without that fluency, sets both the practitioner and the project up for a failure no one in the room will be positioned to explain.

A team implementing an AI-enabled solution must have a governance structure responsible for overseeing the entire stack, tracking output quality across failure scenarios while keeping the core platform within financial bounds. With a full team in place on a system performing as intended, this means aggressively tracking token spend against the value of both outcomes and the cost of alternative implementations, on a per model, per use-case, and per user basis. This is frequently overlooked because organizations assume two things: that users will self-report when a solution isn't working, and that an AI-enabled answer will simply be a profitable one. Both assumptions are wrong. The more likely outcome, as I've observed it, is the quieter and more damaging one: the internal user simply stops using the solution. No dramatic failure. No incident report. Just silent abandonment while no one who can see both the usage data and the business outcome puts two and two together, leaving a company with a significant investment, no return, and a deficit to external expectations.



The principles laid out here exist across every serious discipline from consulting to financial risk management to cybersecurity. They were not invented for AI. They were developed over decades precisely because every powerful technology creates debt, exposure, and insidious failure when deployed without them. AI is not exempt from this. It is subject to it much more acutely than most technologies that preceded it because it is more expensive, more opaque, more unpredictable, and more consequential at the margin. This structure doesn't build itself. Ownership must be named, success must be defined, and accountability for standing it up must sit somewhere before the first deployment begins.



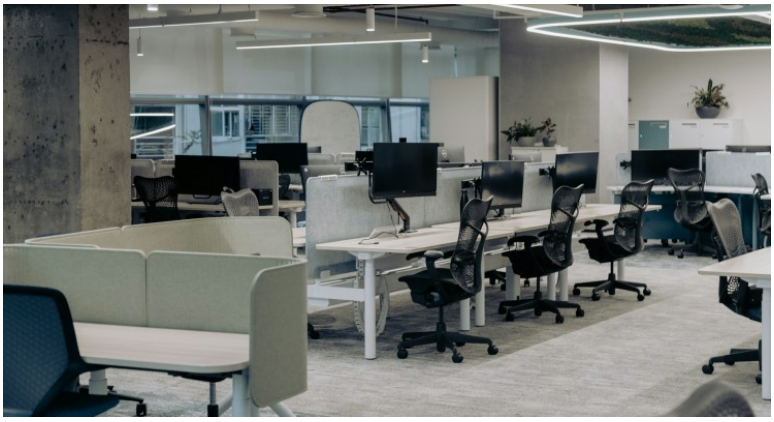
The Cost of AI Theater

The organizational cost of getting AI wrong in the current macro environment is potentially dramatic. The United States has made an outsized investment in the technology: AI-related investment accounted for the majority of last quarter's real GDP growth, at a moment when the remaining drivers of growth don't rest on robust retail activity or broad-based business investment. Contrary to popular narratives, much of recent equities growth has come on the

back of genuinely strong earnings, but that's built on pricing that's fully exposed to the AI investment thesis. A failure to deliver, or even a perceived failure to deliver, threatens to compress valuation multiples in a way that threatens firms far removed from the Bay Area and the tech sector altogether. That's just the first-order effect. The knock-on effects widen from there: a broad market correction would push corporations to cut capital expenditures. Capital-gains tax revenue and stock-based compensation income tied to AI-exposed firms, both the hyperscalers funding the buildout and the suppliers receiving that spending, have become an increasingly important share of recent federal tax receipts. So a valuation pullback here would hit the federal tax base harder than a similarly sized correction would in a more diversified market.

This dynamic matters even more in the context of the current consumer environment. The top 10% of earners are responsible for somewhere between roughly a third and half of all consumer spending, depending on methodology, with the higher end of that range being the most concentrated consumer spending has been in nearly 40 years. Much of that spending is discretionary, and because it's funded substantially by unrealized or recently realized capital gains rather than paychecks, it's highly sensitive to market movements and to this group's perception of their own wealth. A market correction due to a failure of the AI investment thesis threatens that spending directly. Cuts at the top would hit businesses that serve this segment first, producing earnings misses and hiring freezes or layoffs. That, in turn, would compound existing weakness in the labor market, itself already strained by declining real wages, and further erode the consumer spending base the economy is now disproportionately leaning on.

But even if none of these macro worst-case scenarios come to pass, every failed AI deployment still represents a heavy tax on an organization. Worse, it's a tax that de-optimizes business processes and harms product effectiveness, all while reducing internal confidence. Wall Street is growing less patient with AI narratives unsupported by discrete results. Firms that stopped breaking out AI performance as a separate line item are now facing direct analyst scrutiny about what the spend actually produced. The pressure is being felt inside organizations as well: from conversations I've had with people inside these firms, leadership is beginning to demand an explicit ROI case in ways they weren't a year ago. For the individual firm, failure in AI today represents a material threat to finances, talent, brand equity, and increasingly, enterprise value.



The human cost of getting this wrong is no less significant and it is already impacting real people in real ways. When a company cannot demonstrate AI ROI but still needs to protect margins and its own narratives to the market, it cuts headcount. It then attributes the reduction to AI efficiency gains that, by its own admission, have not materialized. The employees being

laid off are told, or left to infer, that they were replaced by a technology. In many cases though, what's happened is that this same technology is going unused two floors up in the same building, monitored by dashboards no one tracks anymore. The company gets short term margin relief and credit from Wall Street for the AI narrative simultaneously. But the worker gets neither the job nor an accurate explanation for losing it, remaining teams work under increasing pressure and are less likely to perform, and leaders remain on the hook for expectations they still haven't actually met. This is an ongoing category error masquerading as progress on the way out the window.

The fundamentals were never optional. The current AI environment has only made the cost of disregarding them too severe to ignore.