SILVERCREST
ASSET MANAGEMENT GROUP

September 22, 2026

Same Question, Different Answer: Probabilistic AI in the Enterprise

Alexander I. Waldorf, CFA
Managing Director,
Portfolio Manager

Since the "computer" arrived in Corporate America, there has been a simple expectation: if you give the system the same inputs, it should produce the same answer. Most existing systems lean on that promise. However, AI changes that expectation. A large language model does not simply execute a set of hardened instructions; it interprets context and generates a response based on statistical prediction. In other words, the same properties that make AI ***flexible and powerful*** also make it ***harder to rely on***. The same question can produce a different answer. Worse, a correct answer and an incorrect one can arrive in the same fluent language and format.

This is becoming an operating issue. In consumer settings, the stakes are lower, and if you get a mediocre travel suggestion or an awkward draft email, you just try again. In an enterprise setting, the output may feed a critical process (customer, compliance, financial), and the cost of being "almost right" can be quite high.

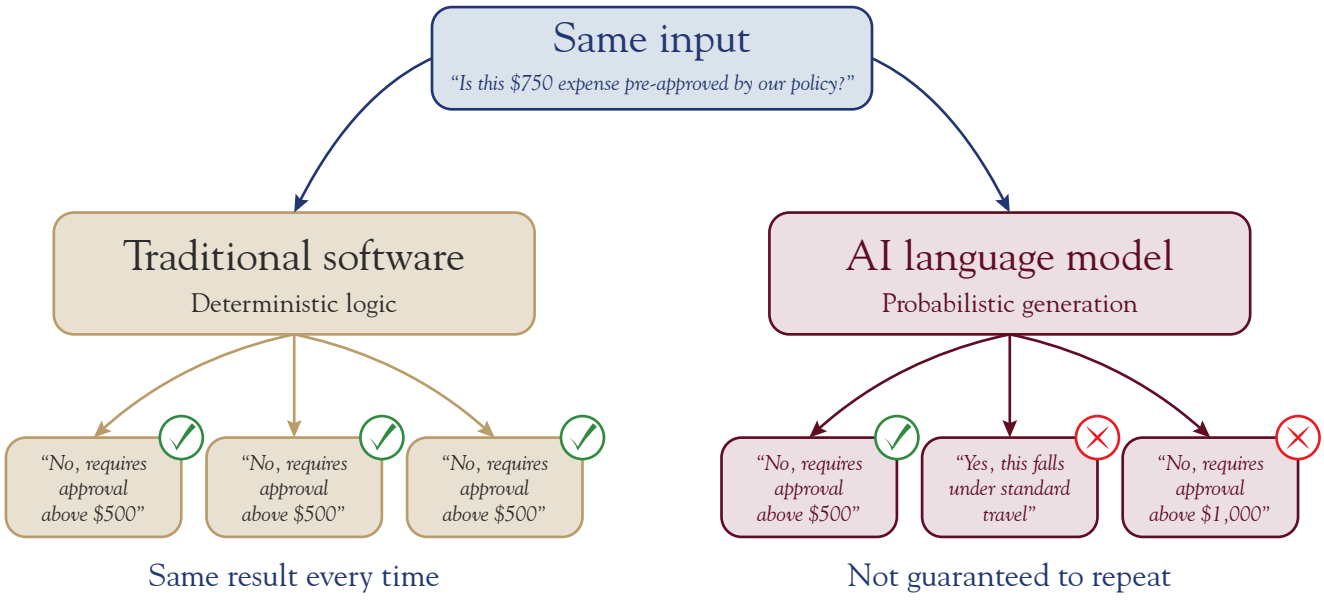
Meanwhile, financial markets have arrived at a remarkably confident view: that AI will disintermediate not only large parts of the software industry but also the professional services firms that have historically helped companies adopt new technology. Many consulting and IT services stocks, for example, have been repriced as if the "help" layer of the economy is becoming obsolete. The initial logic is understandable: if the tool can now do the work itself (including even its own installation), then isn't everyone more self-sufficient?

It is a fair question, but it might be the wrong one. A better set of questions might be: (1) what kind of work is AI being asked to do, (2) when is it allowed to act rather than assist, and (3) who is responsible for making a probabilistic system dependable inside an institution that requires control? None of this argues for slowing adoption; in many cases, companies likely need to move faster than they are. But speed does not create effectiveness on its own. Success in the enterprise probably looks something like being aggressive in exploring use cases but conservative in handing over authority.

The Promise Traditional Software Makes

To be fair, traditional software is by no means perfect or magically reliable. Systems fail; they need testing, monitoring, and maintenance. Data changes. However, traditional software is built on explicit logic (same inputs, same outputs, same path, every time). Therefore, ***you can verify the software once and trust it indefinitely***. Once a payroll system is tested against its specification and passes, that guarantee holds until someone changes the code. The entire apparatus of enterprise IT relies on the assumption that correct behavior, once demonstrated, stays demonstrated.

Large language models work differently at the root. A model does not retrieve an answer; it generates one, predicting each next word from a probability distribution learned across enormous amounts of text. Given the same question, the system produces a range of plausible responses and samples one. This is the source of the technology's power: the ability to handle messy things like ambiguity, language, and judgment. However, the very thing that makes it capable is also what makes it inconsistent. Industry leaders say as much themselves: OpenAI's documentation describes reproducible output as a best effort, not a guarantee.[1]



However, ***reproducibility is not correctness***. Even a "perfectly" deterministic model can reliably give you the same wrong answer every time. Beyond answer variance, the deeper issue is that a model's behavior cannot be pre-specified. With conventional software, an engineer can establish what the system will do for every class of input, because a human authored every branch. With a learned model, no one authored the behavior; it was "grown" rather than written. A slightly rephrased question can send it down a different path.

What does this mean in practice? With deterministic software, passing a test suite is proof. With a probabilistic system, testing becomes sampling: you probe the distribution of behaviors and estimate a rate of correctness, and you cannot enumerate every case.

"Correctness" ceases to be a state of the system and becomes a statistic you must actively maintain.

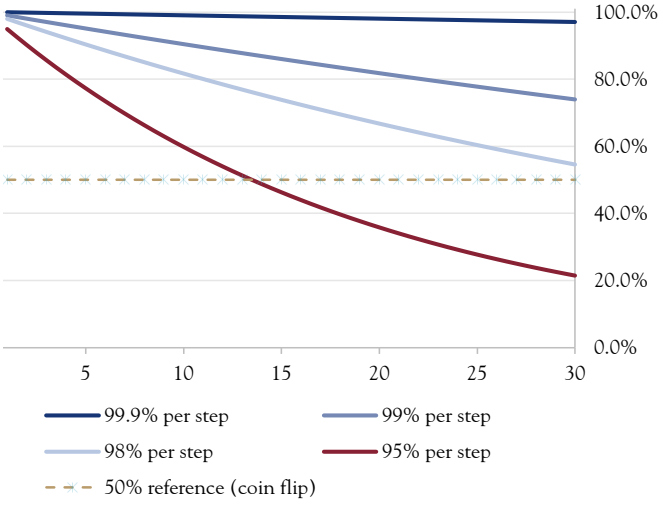
In the enterprise, useful is not the same as reliable. Business processes are less forgiving by nature. A company exists, among other things, as a machine for producing repeatable outcomes. Introduce a system that is usually right and occasionally, confidently wrong, and you have new problems.

The first is just arithmetic. A model that completes a single task correctly 98% of the time sounds excellent. But real business processes are often chains. String ten such steps together, and the odds that the whole chain completes correctly fall to about 82%. At twenty steps, roughly 67%. This may be part of why AI demonstrations impress, but deployments sometimes disappoint.

FIG. 1

The Arithmetic of Chained Probabilistic Steps

End-to-end reliability falls quickly as probabilistic steps are chained.



Per-step reliability compounded across sequential steps (r^n).

The second is the character of the errors. Traditional software fails loudly (a crash, an error, usually something obvious). A language model fails fluently. Mistakes arrive in a confident, well-formatted result that looks like its correct output, which is the kind of error existing controls are least equipped to catch. And even tiny error rates compound at scale (e.g., across a million customer invoices).

Lastly, technology shifts don't change ultimate accountability. Courts in some jurisdictions have already held that a company owns what its software says, even if a chatbot invented the policy.[2] In 2025, Lloyd's of London began offering insurance against losses from AI errors, and FINRA has reminded member firms that using AI does not relieve them of existing regulatory obligations, with hallucinations cited as a specific concern.[3]

The lesson here is not "don't deploy." The bridge between a capable model and a dependable process is simply longer than it looks, and crossing it is real work. Which raises the obvious question: what does that work actually look like?

What Successful Deployments Do Differently

You can't "make" AI deterministic. Instead, thoughtful deployments establish systems to contain it. The model output is fenced within guardrails, and much of the effort goes into deciding where the fence sits. Even the AI labs' own guidance says as much: hard-coded workflows for well-defined tasks, and open-ended autonomy only where flexibility justifies the loss of predictability.[4]

Where the fence belongs depends on what the AI is asked to do, as not all work carries the same risk. A system can observe, retrieve, summarize, draft, recommend…or it can act. Risk rises as the system moves from helping a person to acting on behalf of the company. In other words, a chatbot can be wrong in a window, but an agent can be wrong in the business. Therefore, autonomy has to be earned as reliability is demonstrated. In that way, it is not so different from hiring a human employee. You don't "test" an employee once at onboarding and then trust his or her work unsupervised forever. You review the output, adjust responsibilities, and scale autonomy as trust develops. AI adoption is much more like hiring an extremely skilled but erratic employee than it is like installing a program.

Then comes the part that distinguishes this technology from its predecessors: the "fence" might require a more permanent staff. Because the system's correctness is a statistic rather than a proven state, someone must continuously measure it

(curated test sets or "evals," scoring live outputs, watching for drift, and re-validating the whole system whenever the underlying model changes, which is not always announced). In this world, "deployment" is an operating capability that someone has to manage, not a process with a fixed end date.

What The Market Has Decided (For Now)

Through the first half of 2026, many consulting, professional, and IT services firms have been repriced as structural losers with major names down by 20 to 30% or more. And the reasoning is intuitive enough. These firms sell human hours, and AI may compress hours and/or lower the barrier to competitive entry. The same logic has weighed on application software, where investors question whether per-seat pricing survives a world in which agents, not employees, do the clicking.

Within software, the answer is probably greater dispersion rather than extinction. Thin interfaces and generic workflow layers are genuinely exposed. However, systems of record that own the data, permissions, integrations, and audit trail may become more valuable as agents multiply (because agents inherently need something trustworthy to anchor to). Smart people disagree on whether agents will shrink or expand the software market as a new interface for knowledge work. We're happy to observe and learn.

And on services, some of the evidence genuinely supports the market's current read. Basic managed-service type work (billed by the hour, focused on repeatable or tightly testable tasks) is under real pressure, and the labor intensity of that kind of work seems to have already fallen. That part of the market's reaction isn't just narrative, and it should be taken seriously.

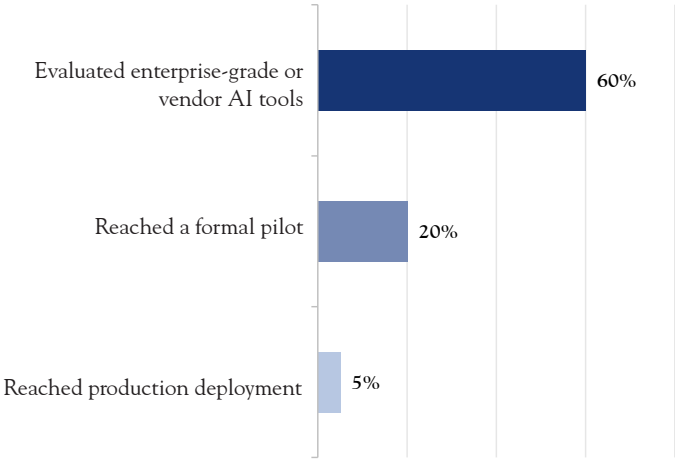
But now put that verdict next to some of the deployment evidence, because the two do not obviously rhyme. MIT's widely cited 2025 study of enterprise AI, based on hundreds of deployments, found that around 95% of enterprise generative AI pilots produced no measurable P&L impact. Roughly 60% of organizations evaluated enterprise-grade AI tools, about 20% got as far as a pilot, but only about 5% reached production.[5] The authors' diagnosis was not that the models were too weak. The failures were more about integration, as the tools did not reliably learn context, adapt to workflows, or survive contact with enterprise data. The externally partnered deployments, notably, succeeded at roughly twice the rate of internal builds.[6]

The market suggests that the help is at risk of becoming obsolete, while some field data indicate that companies struggle to deploy successfully without it. Both cannot be entirely right. How about the parties with the best visibility into the technology's actual trajectory? Within days of each other, Anthropic announced an enterprise-services joint venture reportedly valued at more than $1.5 billion, and OpenAI announced a deployment company capitalized at more than $4 billion that acquired a consulting firm whose goal is to embed engineers inside client organizations to make AI work.[7] Both ventures are built around the same "forward-deployed engineer" (FDE) model, embedding engineers directly inside client organizations, on the premise that the hard problems live inside the customer's walls, not inside the software.[8]

FIG. 2

Where Enterprise AI Initiatives Stall

Share of surveyed organizations



Source: MIT NANDA, "The GenAI Divide: State of AI in Business 2025," July 2025.

So, at the very moment public markets conclude that AI removes the existing services layer, Anthropic and OpenAI put real capital into entering the services layer. We can debate whether the incumbents or the upstarts will execute better against the new opportunity. However, it is harder to dismiss what this reveals about where the companies closest to the technology believe a real bottleneck (and real revenue) may sit.

Looking at the prior waves of enterprise computing deployment, it's hard to find examples where companies didn't need outside help. Mainframes built the original outsourcing industry. The ERP era of the 1990s generated multi-year implementation engagements and birthed the modern systems-integration industry. Cloud computing was supposed to eliminate infrastructure complexity, but instead it created a migration and managed-services economy that continues to grow. Even SaaS spawned its own ecosystem of implementation partners. Each time, a technology was

sold as simplifying, yet enterprises paid outsiders not just to install it but to keep it running.

Therefore, we think the burden of proof actually sits with the claim that this pattern now breaks, especially when the new technology is harder to make dependable than anything that came before it.

What Is Genuinely Different This Time (And What Is Not)

This wave differs from all prior waves in one respect: the tool participates in its own deployment. AI writes integration code, generates test cases, drafts documentation, and migrates data. The hours required to implement a system will probably continue to fall. Some of today's one-off containment work and guardrails will be productized and shipped as features. You can imagine a new checkbox on an enterprise contract requiring that inference output be repeatable. And if a firm's business model was, at the core, reselling routine implementation labor at a markup through junior staff, that model is in genuine trouble. No amount of "clients still need trusted advisors" messaging changes that reality.

However, two distinct claims are being conflated. "Demand for outside help disappears" and "the traditional business model of outside help is under pressure" are different statements. The second is well supported, but the first does not necessarily follow from it and might, in fact, be exactly wrong.

In the old world, you could at least in principle be "finished" with your software. You implemented it, verified it, and, beyond routine maintenance, moved on. However, AI systems are arguably never finished, because "correctness" is something you have to keep earning. It becomes something of a standing operational discipline, which sounds eerily similar to the bread and butter of a services franchise. So, there's a world where certain work goes away while the operating relationship might actually grow.

There is also a component of this work that cannot be automated even in principle, because it is not labor: accountability. At the end of the day, someone has liability for a system's behavior, even if that system is "agentic."

The management team of one of our long-time holdings has a useful phrase they have said to us several times over the years: "We always reserve the right to get smarter." In that vein, we come to this with humility and claim no unique critical insight. However, at the moment, here's where we come out:

The probabilistic nature of AI makes the services layer more necessary, not less—while simultaneously disrupting the way that layer has traditionally priced its output.

Total demand for "help making AI dependable" feels like a market likely to grow considerably over the next five-plus years. But the value likely shifts away from labor arbitrage and toward outcome-based pricing, proximity to the models, useful domain knowledge, and willingness to own results. At the company level, therefore, dispersion within the affected sectors is likely to be high. Are you selling hours, or are you selling accountability?

What would make this wrong? If pilot-to-production conversion rates improve dramatically without growth in deployment and operations spending, that would suggest the technology might be correcting its own limitations. Or, if enterprises demonstrate they can internalize the eval-and-monitoring discipline as routinely as they internalized, say, cybersecurity protocols, then whoever is charging for that work is in trouble.

What we do not expect to see—because over 50 years of enterprise computing history and the observable behavior of the AI vendors argue against it—is the version of the future where a technology this powerful, this unpredictable, and this much harder to make dependable than anything that preceded it turns out to be the one companies deploy successfully without help.

Mr. Waldorf is a Managing Director and Co-Portfolio Manager for the firm's domestic value equity capabilities and the firm's Energy Infrastructure portfolio.

Endnotes

1. OpenAI, "Advanced Usage—Reproducible Outputs," OpenAI API documentation, accessed July 2026, https://developers.openai.com/api/docs/guides/advanced-usage
2. Moffatt v. Air Canada, 2024 BCCRT 149 (British Columbia Civil Resolution Tribunal); see also Marisa Garcia, "What Air Canada Lost in ' Remarkable' Lying AI Chatbot Case," Forbes, February 19, 2024. https://www.forbes.com/sites/marisagarcia/2024/02/19/what-air-canada-lost-in-remarkable-lying-ai-chatbot-case/
3. "Courts to Companies: You Own What Your Chatbot Says," PYMNTS, July 2026, https://www.pymnts.com/news/artificial-intelligence/chatbot-tracker/2026/courts-tell-companies-they-own-what-their-chatbot-says; FINRA, "Regulatory Notice 24-09: FINRA Reminds Members of Regulatory Obligations When Using Generative Artificial Intelligence and Large Language Models," June 27, 2024, https://www.finra.org/rules-guidance/notices/24-09
4. Erik Schluntz and Barry Zhang, "Building Effective Agents," Anthropic (engineering blog), December 2024, https://www.anthropic.com/engineering/building-effective-agents; OpenAI, "Building Agents," OpenAI Developers, accessed July 2026, https://developers.openai.com/tracks/building-agents
5. MIT NANDA initiative, "The GenAI Divide: State of AI in Business 2025," July 2025; funnel figures as reported in "MIT: Why 95% of Enterprise AI Investments Fail to Deliver," AI Magazine, September 2025. https://aimagazine.com/news/mit-why-95-of-enterprise-ai-investments-fail-to-deliver
6. Sheryl Estrada, "MIT Report: 95% of Generative AI Pilots at Companies Are Failing," Fortune, August 18, 2025. https://fortune.com/2025/08/18/mit-report-95-percent-generative-ai-pilots-at-companies-failing-cfo/
7. Janakiram MSV, "AI Giants Bet Billions on the Most Expensive Job in Enterprise," Forbes, May 28, 2026. https://www.forbes.com/sites/janakirammsv/2026/05/28/ai-giants-bet-billions-on-the-most-expensive-job-in-enterprise/
8. "Palantir's Forward Deployed Engineering Playbook: The Original Model Anthropic and OpenAI Are Copying," Perspective AI, May 14, 2026. https://getperspective.ai/blog/palantir-forward-deployed-engineering-playbook-anthropic-openai-copying

Disclosures

This communication contains the personal opinions, as of the date set forth herein, about the securities, investments and/or economic subjects discussed by Mr. Waldorf. No part of Mr. Waldorf's compensation was, is or will be related to any specific views contained in these materials. This communication is intended for information purposes only and does not recommend or solicit the purchase or sale of specific securities or investment services. Readers should not infer or assume that any securities, sectors or markets described were or will be profitable or are appropriate to meet the objectives, situation or needs of a particular individual or family, as the implementation of any financial strategy should only be made after consultation with your attorney, tax advisor and investment advisor. All material presented is compiled from sources believed to be reliable, but accuracy or completeness cannot be guaranteed.

© Silvercrest Asset Management Group LLC