

GenAI: Easy to use, harder to manage

Turning AI into real business transformation requires more than adopting new tools. It means building the organization around abundant cognition, valuable judgment and compounded learning.



May 7, 2026

By [Sampsa Samila](#)

The first time most people experiment with generative AI, they are struck by how easy the technology is to use and how widely useful it is. Such ease and versatility have fueled widespread adoption of GenAI at the individual level, with [over half of U.S. adults using it in the past year](#). The generalist abilities of the models mean [employees can do tasks beyond their previous capabilities and save real time](#). But what works well for individuals is not the same as what works for companies. At the company level, those [individual gains have yet to show up in profits or productivity](#).

The gap between individual gains and firm-level impact isn't closing by itself. Managers are needed to close it, and they need the same management principles as before, applied to a technology with new properties. As the economists Carl Shapiro and Hal Varian put it a generation ago, "Technology changes, the laws of economics do not." Price theory still predicts what happens when the cost of an input drops, but the input now is cognitive work, which is different from a commodity or a component. Organizational design still matters, but the boundary between human and machine tasks is a new challenge. Companies still need purpose, which, as a guiding principle in AI deployment, helps determine what to automate, what to augment, what to protect and what to refuse.

For broad transformation, companies need repeatable, scalable processes as well as leaders who know how to make them work in their organizations. AI's new leadership challenge

involves thinking in terms of processes and multilayered systems, which is a new framing for many people. It requires building a human organization that continuously renews and develops tacit knowledge around judgment, values and culture. And it demands the managerial agility to do all of this on a fast-moving technological frontier.

Mapping a new territory

Navigating a new territory starts with a map of that territory, an understanding of what's new about the technology itself. What appeals to individuals is that foundation models such as ChatGPT or Claude are extraordinary generalists. They can write, summarize, translate, code, analyze and do almost any task in written language.

But for companies, that generality comes with a cost: the models have no knowledge of a particular company's context, they can produce confident-sounding errors (hallucinations) and their accuracy in any specific domain tops out well below what most business processes require. As the same models move from producing outputs that the user reviews to taking actions on the user's behalf, the stakes of that ceiling change: a confident error is no longer a misleading paragraph but a misguided action with operational consequences. And in a multistep process, those errors compound. What companies need for their operations is not a generalist but something closer to a specialist. Getting there means building a stack of capabilities on top of the foundation model to turn a generalist into something that can do the company's work reliably.

The stack can have several layers:

- Closest to the foundation model, fine-tuning adapts it to a particular domain by training it further on relevant data, reducing its usefulness in other areas and trading generality for accuracy.
- A step removed, retrieval-augmented generation (RAG) gives the model access to proprietary documents and data at the moment of the query, grounding outputs in real company contexts without changing the underlying model.
- Prompting shapes how the model is used, with prompt libraries and templates standardizing the inputs that produce reliable outputs.
- Integration embeds the model into specific workflows, with user interfaces, process design and authorization scoping determining where human judgment enters, where it doesn't and what the system is permitted to do on its own.
- Finally, governance sets the limits: what AI is allowed to do, what it must not do and

how its outputs are reviewed.

Not every company needs every layer, but every company that uses AI at scale has to decide which layers to build, which to buy and how to manage the resulting complexity.

Managing AI's relocated complexity

Every process has a minimum complexity that matches the real-world situation it handles. That complexity cannot be removed, only relocated. To the user, a well-designed AI tool feels simple: you type a question, you get an answer. But the complexity the user no longer sees hasn't disappeared; it has moved into the stack and, more importantly, into the organization that runs it.

The generality-accuracy-simplicity trade-off

In 1976, the Canadian social psychologist Warren Thorngate proposed the *postulate of commensurate complexity*: no theory can be simultaneously general, accurate and simple. A general theory will err on specifics. An accurate theory will be narrow. A simple theory will miss nuance. Any model, theory or even business process has to trade one of the three against the other two.

Five decades later, this [generality-accuracy-simplicity framework is useful for business leaders](#) looking to deploy AI in their companies, even though large language models (LLMs) appear to defy it, since they seem general, reasonably accurate and strikingly simple to use. The apparent resolution is an illusion, however.

User-facing simplicity has been bought by moving enormous complexity into the underlying system, including the parameters, the training infrastructure and the governance layers that make the interface feel effortless.

And the accuracy side of the trade-off has not been resolved; it has been relocated. Because LLMs are prediction machines that generate plausible rather than verified inferences, they operate against an "accuracy ceiling" that becomes more visible as generality expands. Widening scope tends to raise error rates; narrowing scope through fine-tuning or retrieval tends to narrow usefulness.

In short, the simplicity that the user sees is the complexity that the organization has to

manage.

Some of the stack is outsourced to providers: the foundation model, the cloud infrastructure, the commercial tools. The rest has to be built and maintained inside the company: data pipelines, retrieval systems, prompt libraries, evaluation, governance. Each of these requires people who understand what they are doing, processes for keeping them current and accountability when they fail. [The apparent simplicity of AI at the point of use is paid for by a substantial increase in complexity somewhere else](#). Redistributed accountability is the most challenging. When systems take consequential action, the question of who answers for it does not redistribute itself the way technical work does, and absent deliberate allocation it tends to land wherever the harm landed.

Some of the complexity will go in infrastructure that supports the organization broadly, like foundation models, data systems, governance and monitoring. But a significant share of it lands inside the processes where AI is actually deployed, and here companies often discover that AI does not slot neatly into an existing step. Processes are not neutral scaffolding. They were designed around what humans can do: how much context one person can hold, how expensive it is to hand off between people, how long it takes to train someone in a narrow skill. AI has different properties: it can hold more context than any individual but holds no tacit knowledge and forgets between sessions; it can run in parallel at almost no cost and has no fatigue. A task breakdown optimized for humans is not the right task breakdown for humans plus AI. The boundaries sit in the wrong places.

Redesigning business processes

One IESE alumnus reported an order-of-magnitude productivity gain in programming that dissipated because product management and quality control became the new constraints. [Amazon Web Services \(AWS\) has warned](#) that AI coding assistants can overwhelm downstream delivery pipelines unless testing, review and integration are redesigned to match. When one step changes in cost or character, the whole process has to be rebalanced. And rebalancing often means reengineering, not optimizing. It means asking, for each step, whether it should exist at all, who or what should do it, and how the pieces connect.

Redesigning around AI comes down to four decisions: what the AI does, what the human does, what the AI is permitted to do without asking, and how feedback flows between them. Skipping any of them tends to produce what early AI deployments commonly report:

individual tools that work well, firm-level results that don't show up.

Redesigning processes around AI is necessary to see firm-level gains. But it is not, by itself, a source of competitive advantage, because every competitor can redesign too, and the foundation models everyone is redesigning around are the same. If the technology itself is common, differentiation has to come from what a firm builds on top of it: the stack, the processes that surround it, the data it learns from, and the organization that runs the whole system.

The compounding economics

This shifts the economics in a specific way. The stack itself is mostly infrastructure and fixed cost. Foundation models are licensed or built, data systems and governance are expensive to set up, and the organizational capability to manage all of it is not cheap either. After that, the marginal cost of using the stack for one more task, one more customer and one more process is very low. Scale matters. A stack used across many processes and many customers pays for itself many times over; a stack used in one narrow application may not pay for itself at all. Make-or-buy decisions for each layer become central: which layers generate enough distinctive value to justify building in-house, and which should be acquired from providers who can spread their own fixed costs across the whole industry.

But fixed costs are only half the story. The technology also learns with use. More interactions produce more data for fine-tuning, more examples for retrieval, more information on where prompts and workflows fail, more opportunity to evaluate and improve. A stack that is used heavily gets better faster than one that is used lightly. That compounding matters because the competitor's stack is also learning, and whoever learns faster widens the gap.

Part of what compounds is operational: experience with where the system can be trusted, where it cannot, and what the failure distribution looks like in the firm's specific context. That kind of knowledge is bespoke, accumulates only with use and cannot be acquired by waiting. There is no comfortable steady state.

The growth imperative follows directly. Using AI at scale is how the fixed costs get leveraged and how the feedback loops get fed. Companies that find ways to deploy AI across many processes and to run it hard generate both the economics and the learning that early leaders need. Companies that pilot cautiously may find that by the time they are ready to scale, the distance to catch up has grown.

The human organization

Underneath all of this sits a more fundamental distinction. AI operationalizes codified knowledge: the things an organization can write down, formalize and represent in a system. Tacit knowledge is different in kind. It lives in people and in the relationships between them: the judgment built through experience, the sense of what matters in a specific context, the shared standards a team develops over time, the organizational memory of what has been tried and what has failed. A competitor can buy the same foundation model and build a similar stack, but they cannot buy the organization that has learned to work with it. AI handles the codified work, while humans hold the tacit knowledge that directs it, evaluates it, corrects it and gives it meaning. As systems become more autonomous, more of that directing has to happen ahead of time, encoded into how the work is set up, because there is less opportunity to intervene in the moment. The tacit knowledge stays human; its point of application moves earlier.

However, tacit knowledge does not maintain itself, and the conditions that produce it are the conditions that AI use can erode. Tacit knowledge has always required specific conditions to develop and transmit: apprenticeship, mentoring, struggle with hard problems and exposure to other people's judgment in real time. [Experience at companies on the frontier of AI deployment](#) is showing that employees work more independently and get less mentoring. [Experiments are showing](#) that employees using AI can defer to AI and engage less with struggle in solving problems. These same aspects — independence, self-sufficiency and the quick use of tools to solve problems — can increase productivity in the short run, but hinder the development of tacit knowledge in the long run.

The [skills needed in leadership](#) thus shift toward creativity, collaboration and stakeholder management — the human capabilities that produce and transmit judgment. Previously, tacit knowledge developed as a byproduct of doing the work: junior employees struggled with problems and sought advice as it was the only way to get the job done. Now, these conditions have to be actively designed and implemented: structured mentorship, deliberate exposure to hard problems without AI assistance, environments in which struggling productively is permitted and rewarded, evaluation criteria that distinguish AI-mediated competence from earned competence.

The leadership task

This is the new territory: the stack, the relocated complexity, the redesigned processes, the compounding economics and the human organization that holds the tacit knowledge. While the territory is genuinely new, the principles for navigating it are ones most leaders already have. Systems thinking, economies of scale, organizational design and purpose are not new ideas. They are the durable frameworks of management, applied to a context where the input is cognition, where the assets compound through use, and where the boundary between what machines do and what humans do has to be drawn rather than inherited.

The leadership task is to decide where cheap cognition should go, where machine action is acceptable, where scarce judgment must concentrate, and how the organization learns faster than its competitors. That means thinking in systems and compounding rather than tools and pilots; investing in the human organization as its value rises alongside cheaper machine capability; and anchoring the whole system in purpose, because feedback loops compound in whatever direction they are pointed. And it requires a [board that can govern](#) this work as a fiduciary discipline, not delegate it to management as a technology project.

Every AI decision is a bet on a trajectory. AI capability is increasing and cost is decreasing on a trajectory that shows few signs of leveling off. That means AI may be twice as capable and half the cost a year from now. The firms where individual benefits become firm-level gains will be the ones that have already built the organization that is ready for this moving frontier.

MORE INFO:

[“From model design to organizational design: complexity redistribution and trade-offs in generative AI”](#) by Sharique Hasan, Alexander Oettl and Sampsa Samila.

[“AI adoption and the demand for managerial expertise”](#) by Liudmila Alekseeva, Jose Azar, Mireia Gine and Sampsa Samila is published in the *Strategic Management Journal* (2026).

[“Power steering, not a brake: how boards should actually govern AI”](#) by Henk de Jong, Robert Maciejko, Sampsa Samila and Christoph Wollersheim.

This article is included in the annual publication, Insight for Global Leaders No. 2 (2026).



Sampsa Samila

Professor in the Strategic Management Department at IESE Business School, where he is the Academic Director of the [Artificial Intelligence and the Future of Management Initiative](#).

www.iese.edu/insight