

AI Cuts Those Jobs Which Were Never There in the First Place

Let us accept it: most of what we were doing in these massive knowledge companies was too simple to survive. It was never meant to run forever. What remains after the AI apocalypse is the work that was actually worth doing.

Let me say something that will make some people uncomfortable: most of the jobs that AI is eliminating were never real jobs in the first place.

They were organizational padding. They were roles that existed because a structure needed a certain number of nodes, because a manager needed a certain number of reportees, because a process was never questioned, because nobody asked whether the output of that role could have been produced differently — or not at all.

AI did not kill these roles. AI simply made it impossible to pretend they were necessary.

I say this as someone who has designed the systems that eliminate tasks, and who has led the teams whose tasks were eliminated. I have sat on both sides of that table. What follows is not a policy paper or an economist's projection. It is what I see from the inside, as of May 2026, told without the euphemisms that make boardroom presentations comfortable and delivery floors anxious.

The Hierarchy of Displacement: Documentation First, Code Last

AI is not replacing all roles equally. It is replacing them in a very specific order, and that order tells you everything about what was real work and what was transcription.

First to go: the people who document other people's thinking. The business analyst who sits in a meeting, captures requirements, and produces a 40-page specification document. The presentation designer who converts a strategy discussion into a PowerPoint. The technical writer who documents what the architect decided.

And here is one that cuts closer to the bone: the technology practice solution architect who has a set of guidelines and fits those guidelines onto each different assignment the same way, based on three to five parameters. That is not architecture. That is templating. And AI can template faster than any human alive.

Second: the people who create visual artifacts. Charts, data models, wireframes, dashboard layouts. The skill that remains valuable is the judgment to know whether the output is right — not the ability to produce it.

Third — and this is where the story changes — coders. Yes, Google and Anthropic are loudly declaring that their coding models can complete organisation-level code. And they can — in demonstrations. In greenfield environments. In showcases where the applause comes before the SLA review.

But here is what I see in production: as of May 2026, all the stories we hear about code completion with agents are not SLA-bound for the outcome. When your code has to meet a contractual SLA inside a regulated banking environment, with integration points to systems that were built before some of your developers were born, the human coder is still the one signing the delivery note.

To set the context, to create the intent, to frame the problem — we still need senior architects and analysts. Then we need developers and testers for inference evaluation. Enterprise integration and refactoring create a large part of development work that still requires many hands, many eyes, and many years of hard-won judgment.

The pattern is clear: the further you are from the thinking and the closer you are to the typing, the more exposed you are. AI does not replace thinkers. It replaces transcribers.

The Tree Diagram: How the IT Services Machine Actually Works

To understand what AI is really displacing, you have to understand how the IT services industry is actually structured — not the way it appears in annual reports, but the way it operates in practice.

Most large technology service companies can be expressed as a tree diagram. The company grows around a particular product or platform. Around that product, a structure forms: a partnership manager who maintains the vendor relationship, a go-to-market team that creates collateral and pursues deals, a training enablement function, a hierarchy of senior leadership for staffing and allocation, and a few hundred junior or mid-level individual contributors who do the actual billable work.

The entire model runs on one economic principle: maximise revenue per hired junior with optimal training investment. The competency centre is not a knowledge hub — it is a billing engine. Its primary function is to place those individual contributors into engagements to work on a slice of delivery under someone else's governance.

And here is how the pyramid economics work in practice. Every time a project needs one FTE, the competency centre sends a pyramid — not one person, but three. Why? Because every project opportunity for one of their team members is a learning opportunity for the juniors. The competency centre justifies the presence of two extra FTEs with arguments like: “What if the senior falls sick for two days? He needs a backup.” And the most junior person — a fresh graduate or someone with two to three years of experience, the person every service company has a fancy name for — that person earns the highest operating margin for the company on fixed-bid, output-based projects.

This model runs identically whether the company is based in the US, France, Ireland, or India. The most junior fellow is deployed because it is an organisational mandate to deploy rookies on every engagement. Eventually, in a couple of years working on the same project or account, that rookie becomes a standard IC, then assumes the senior's position, and the cycle continues. This is not a bug in the system. This is the system.

AI is about to remove the middle person from this pyramid. And deploy the senior for Agentic strategy, enabling the junior to run on their own and validate the inferences. The three-person pyramid becomes a two-person team with an AI agent in the middle.

The Go-To-Market Disruption Nobody Is Discussing

AI will also displace the go-to-market layer — if not fully, then substantially. Here is why: a frontier model does better solution design and estimation than most of the individual contributors in technology companies. After consuming two years of past solutions, an LLM can yield better results than keeping a team of moderate-to-pre-leadership-level solution architects on the bench.

This is not a theoretical claim. I have seen it happen. A solution that would have taken a team of three architects two weeks to design, estimate, and package can now be

produced in a day with AI assistance and one senior architect's judgment. The two weeks of effort did not represent two weeks of thinking. They represented two weeks of formatting, cross-referencing, and organizational coordination. AI compresses all of that to near zero.

The implication is significant: the go-to-market teams that grew during a product's high tide will shrink during AI's high tide. And many of the leaders who rode that wave will find that their career trajectory was correlated with market timing, not with unique capability.

The Defensive Middle: Career Protection as Organizational Drag

When revenue is good and a specific product or platform is riding a high tide, the senior leaders of the competency team grow fast on the company ladder. They get promoted. They build teams. They accumulate organizational capital.

Then something predictable happens they become defensive. They avoid all decisions that carry delivery risk. They optimize for career safety, not for client outcomes.

Innovation becomes threatening because innovation introduces uncertainty, and uncertainty threatens the metrics that justified their promotion. This cycle continues until the day the market sends a signal to executive leadership and the investors — and the team gets trimmed.

This pattern has repeated across the industry multiple times. Some technologies see high tide while others see low tide. When companies moved from custom business applications to ERP, the SAP ECC-HANA and Oracle eBiz teams inflated while Java developers scrambled. Then came 2019 to 2021, when the SAP teams wanted to reinvent their identity and the full-stack engineering teams were having a celebration. Architects and developers held two jobs per head — some got caught, some did not.

Super-refined specialisation practices emerged. There is no single Low-Code-No-Code practice, but fifteen different sub-practices for fifteen different LCNC and rapid application development platforms. Each one a small tree in the larger forest, each one with its own partnership manager, GTM team, training enabler, leadership hierarchy, and army of juniors.

AI does not care which tide you rode. It compresses all of them simultaneously.

The Roles Disappearing Next Month: PMOs and the Orchestrator Class

PMOs are next. The people who collect data from multiple service delivery streams of a large account and compile it into QBR reports, monthly dashboards, or status decks. The AI capability for dashboard creation and data synthesis is superhuman. What used to take a team of three a week can now be done in an afternoon.

But the delivery leads will still keep someone on the team for manual data collection, even if they build a pipeline for automated engagement data collection. Why? Because the trust transfer — from “I trust this person’s judgment about the data” to “I trust this system’s output” — takes longer than the technology deployment. The last mile of automation is not technological. It is psychological.

Then there is the Orchestrator Class. People whose only job is to integrate information collected from various sources and present it upward. The human “Backend for Frontend” — collecting inputs from five teams, producing a consolidated output for the leadership layer. These integration nodes can be replaced with agents, initially with human-in-the-loop. Within six to eight months, the loop closes. Fully autonomous.

Were these ever real roles? Or were they artifacts of an organizational design that assumed information could not flow without a human relay? AI did not eliminate these jobs. It revealed that the jobs were workarounds for a system design problem that nobody had bothered to fix.

The Elephant Nobody Will Name: The Vague KRA

Every tech service company has leaders — middle management and above — who are not directly contributing to delivery, not creating innovations, and not mining revenue. They exist because the organizational structure requires a certain number of people at a certain level, and because somebody above them needs a certain number of reportees underneath.

These are the people with vague KRAs. Key Result Areas that sound important in a performance review but produce nothing that a customer would pay for. AI will replace

those roles with vague KRAs — not by automating what they do, but by making it visible that what they do does not need doing.

When the PMO team that reported to them is gone, when the orchestration layer they managed is automated, when the status reports they reviewed are generated without human input — what remains? The answer, for some of these roles, is: nothing that could not be absorbed by the layer above or the layer below.

The Ecosystem Nobody Wants to Disrupt

Here is the part of this story that nobody in the technology industry wants to say out loud.

This model worked well for everybody in the schema. The companies made money. The employees had careers. And more importantly — for governments — it created a perception of a thriving economy. Kids came out of engineering colleges and landed decent jobs at nice office buildings. Middle-aged professionals were doing acceptably well, buying one or two mid-segment cars, shopping at malls, ordering from quick commerce apps — which in turn employed another layer of less-educated workers in the delivery and logistics chain.

The IT services pyramid was not just a business model. It was an economic ecosystem. And that ecosystem is now under AI pressure — uniformly, simultaneously, across every layer.

But this time it will be different from the previous technology cycles — the SAP-to-cloud migration, the Java-to-full-stack shift, the LCNC proliferation — because this time we cannot simply relabel the competency centre and retrain the pyramid. The displacement is structural, not cyclical.

We did not acknowledge one hard truth: everything we were doing collectively — placing real-life business flows into systems of software — is apparently quite easy to simulate with software with minimal human effort.

The complexity was never in the technology. It was in the coordination. And coordination is exactly what AI agents are designed to do.

New Roles That Did Not Exist Last Year

The AI Adoption Coach. Not a trainer. A coach who works alongside senior leaders, helping them integrate AI into their daily decision-making. These leaders built their careers on human judgment. They are not going to learn AI from a course. They need someone who understands both the technology and the psychology of a 50-year-old delivery leader who has been successful for decades and is now being told that AI is essential. This is a change management role that requires deep empathy and technical fluency in equal measure.

The Enterprise AI Auditor. This role does not exist in any organizational chart I have seen, but every enterprise that has deployed more than three AI projects needs one. This person extracts the success and failure data from AI projects across the enterprise, compiles them into structured assessments, identifies patterns, and closes the feedback loop. Without this role, enterprises keep making the same mistakes across different departments.

The jobs AI creates are harder, rarer, and more valuable than the ones it eliminates. That is the uncomfortable truth that the “AI will create more jobs” narrative glosses over.

What Comes After the Scaffolding

I have spent this entire article describing what AI displaces. Let me end with what it releases.

When you remove the organizational padding — the pyramid staffing, the defensive middle management, the vague KRAs, the orchestrator class that existed to relay information that should have flowed automatically — something surprising happens to the people who remain. They start working differently. Not because someone mandated a culture change programme. But because the structural reasons for fear, subterfuge, and passivity are gone.

The middle manager who spent their energy protecting their position instead of taking risks? That behaviour was rational inside a system that punished failure and rewarded caution. Remove that system, and the same person — freed from the organisational drag

— often becomes someone you did not expect: energetic, curious, willing to experiment. The passivity was never a character trait. It was a survival strategy for a structure that penalised initiative.

I am genuinely hopeful about what the remaining workforce in IT will look like. Not because AI makes everything better — it does not — but because AI removes the architecture of mediocrity. The layers that forced talented people to operate below their capability, to wait for approvals from people who added no judgment, to perform work that existed only because the system assumed it was needed.

The people who remain will work with more self-organisation, more autonomy, and more direct connection to the outcomes they produce. They will not work under fear and subterfuge, but under pleasure and curiosity — because for the first time, the structure around them will be designed for value creation, not for headcount justification.

AI did not just cut the jobs that were never there. It cleared the space for the work that always should have been.

The Job That Was Never There

When we say “AI is cutting jobs,” we are using the language of violence — as if AI is an external force that attacked a stable system. But what if the system was never stable? What if many of these roles were temporary solutions to permanent problems — the problem of information not flowing, the problem of systems not talking to each other, the problem of organisations not trusting their own data?

A person who spends three days compiling a status report from five delivery streams is not doing knowledge work. They are doing plumbing — connecting pipes that should have been connected at the infrastructure level. A solution architect who fits the same template onto every assignment based on three parameters is not doing architecture. They are doing pattern-matching — and AI pattern-matches better, faster, and without the politics.

These jobs existed because the alternative — building systems that connect information automatically, designing processes that do not require human relay nodes, trusting machines to do what machines can do — was not yet available. Now it is.

AI did not cut these jobs. AI made it visible that these jobs were scaffolding — necessary during construction, but not part of the building.

The question is not whether the scaffolding should come down. The question is what we owe the people who stood on it — and how we build the next structure so that the scaffolding is not needed in the first place.

I have published my frameworks on this and other questions at reshminandy.com — because I believe the honest conversation is worth more than the comfortable one.

Reshmi Nandy is a Transformation Lead and GCC Builder with 27 years of experience in enterprise AI, deal architecture, and operating model design. She holds a USPTO patent in AI-based value stream optimization. She writes her thoughts at reshminandy.com.