

# The Structural Problem Nobody Is Solving

Miguel Guevara · 2026 · [ignitena.com/the-structural-problem](https://ignitena.com/the-structural-problem)

---

*Organizations don't fail at change because their people resist it. They fail because nobody redesigns the conditions that make the old behavior easier than the new one.*

We went live on a Tuesday. By Thursday, half the org was back in the legacy system.

Not because they didn't understand the new process. Not because the training was bad. Because the environment still made the old way easier. The workaround was faster. The manager's scorecard still measured the old metric. The approval chain hadn't changed. Nobody had closed the path back.

That pattern has shown up in every engagement across nearly thirty years of Fortune 500 ERP rollouts, post-merger integrations, and frontier-tech companies scaling faster than their operating models can absorb. SC Johnson, Grainger, Johnson Controls, Beam Suntory, IDEX, Edelman, Anduril Industries. Different industries, different scales, different technologies. The same breaking point.

The communications went out. The training ran. The system went live. Six months later, the workforce was working around it instead of in it.

The gap is environmental. It closes when the environment changes. It doesn't close when you add more training. It doesn't close when you send another email.

This isn't a people problem. It's an architecture problem.

## 01 THE SCALE OF THE PROBLEM

McKinsey surveyed nearly 2,000 participants across 105 countries for their 2025 State of AI report. 88% of organizations report using AI in at least one business function. Only 6% qualify as high performers getting real returns.

That's not an adoption problem. That's a structural one.

OpCo Intelligence confirmed the mechanism in early 2026, surveying 123 senior operators from companies including Stripe, Anthropic, Databricks, and Microsoft. General-purpose chatbots have near-universal adoption. The tools exist. The binding constraint is organizational, not technical. Roughly 70% of respondents have no formal implementation metrics. They use adoption as a proxy without measuring whether behavior change is holding.

S&P Global found that 42% of companies abandoned most of their AI initiatives in 2025, up from 17% the prior year. RAND puts overall AI project failure rates above 80%, roughly twice the rate of non-AI technology projects.

This isn't new. Gartner data puts ERP project failure rates above 75%. BCG studied 225 transformation programs and published the results in 2005. They found that the primary drivers of failure were not strategy, technology, or vision. They were four hard-side factors: project duration, team capability, management commitment, and the additional effort the change required from employees beyond their day jobs. Twenty years ago, the data was already pointing at the operating environment, not at the people.

The conditions problem buried ERP rollouts, merger integrations, and workforce transformations for decades before AI arrived. AI compounds it at a faster cycle and a higher cost. The thesis is not an AI story. AI is the accelerant. The conditions gap is the constant.

## 02 BEHAVIOR FOLLOWS STRUCTURE

If the environment still makes the old behavior easier than the new one, that's the behavior you'll get. This is the principle underneath every failed transformation I've seen.

Communications change what people know. Training changes what people can do. Neither one changes what the environment rewards, measures, or makes easy. Every change program reaches that limit. The question is whether anyone recognizes it as a structural constraint or just asks for more training.

Consider what happens when a new ERP system goes live but the approval chain hasn't changed. The system is designed for a three-step process. The organization still runs a seven-step approval. People don't resist the new system. They comply with the existing authority structure. The behavior follows the structure, not the training.

Matt MacInnis, COO of Rippling, arrived at the same conclusion from a different direction. Rippling is an \$11 billion company that scaled 11x in four years. His principle: leaders make a fundamental error by over-rotating on outputs, watching dashboards instead of engineering the inputs that produce those outputs.

His analogy is clean. Screaming about the score doesn't change the plays. Change the plays and the score follows.

That's the structural argument in operator language. The organizations getting results treated deployment as an architectural question, not an adoption question. McKinsey's data confirms it: AI high performers are nearly three times more likely to have redesigned workflows before deploying. The other 94% are still trying to add AI to the architecture that was already failing them.

Ben Horowitz frames it differently but lands in the same place. Organizational design is a communications architecture. Every design optimizes some paths at the expense of others. The architect's job is to decide which paths matter for the transformation at hand, then build explicit processes for the ones you didn't prioritize.

Most transformations skip that step entirely. They deploy the tool, launch the new process, announce the new model. They don't redesign the operating conditions around it.

The twenty-year gap between BCG's 2005 research and the Rippling team's 2025 framing tells you something. The data has been pointing at operating conditions for two decades. Academic proof, operator validation, current field data. All three converge on the same conclusion. The enterprise buyer saw BCG confirm it. The frontier-tech operator saw MacInnis live it. The PE operating partner sees the failed portfolio company thesis prove it by negative example.

The pattern has always been there. The market just hasn't built around it.

### 03 WHAT THE CHANGE PROGRAM CAN AND CANNOT DO

The change management discipline has known for decades that the people side determines outcomes. Prosci studied 3,000 change practitioners. 88% of projects with excellent change management met their objectives. 13% with poor change management did.

The discipline is right about the problem. The standard response is incomplete.

Prosci's ROI model distinguishes two categories of project benefits: those independent of adoption and usage, and those dependent on it. The right question: what percentage of this project's expected return depends on people actually changing how they work?

The market answers it with communications and training.

Communications and training are necessary. They are not sufficient. The speed and depth of today's transformations exceed what they were designed to handle.

Here's where it breaks. Organizations invest in executive sponsorship and frontline readiness. Nobody prepares the person in the middle.

Middle managers didn't design the change. They can't control the timeline. They absorb every question and fear from the frontline while still being held to the same operational targets they had before the transformation started. They get pulled into four extra roles at once: subject matter expert, project champion, communicator, change advocate. Nobody reduced their existing workload to make room.

A briefing deck tells them what the change is. It doesn't prepare them for the conversations walking through their door. "My team doesn't know what to do Monday morning." "The new system doesn't handle our exception process." "I was told one thing in training and something different by my VP." Those conversations land on the middle manager. Every one of them is an operating condition the program didn't address.

The most common workaround in any transformation starts here. When the middle manager can't resolve the tension between what the program promised and what the field is experiencing, they improvise. The improvisation becomes the default. And the default becomes the reason the old behavior persists six months after go-live.

That's not a training gap. It's a conditions problem. The role was never redesigned for the weight it carries.

The change program handles communications, stakeholder alignment, training, and readiness. It does that well. But there's a ceiling. Above it sit decisions that restructure how the organization works. Closing the legacy system instead of running both in parallel. Tying leader accountability to sustained behavior change, not go-live completion. Removing the workaround before it becomes the default.

These aren't program interventions. They're decisions only the sponsor can make. Most sponsors have never been asked to make them explicitly.

#### 04 WHY AI BREAKS THIS FASTER

Every previous technology disruption gave organizations time. ERP rollouts stretched across years. Merger integrations had transition periods measured in quarters. The workforce adapted slowly because the technology moved slowly.

AI doesn't wait.

The gap between what the technology enables and what the organization's structure permits is wider than it has ever been. It's widening faster than any change program can close through communications and training alone.

McKinsey's high performers, the 6%, are 2.8 times more likely to have redesigned workflows rather than adding AI to existing ones. They're also far more likely to have defined human-in-the-loop validation processes: 65% versus 23%.

The gap between the 6% and the 94% is not model quality, budget, or talent. It's workflow redesign. It's structural.

And the projects that don't collapse often become something worse: compliance theater.

I've seen this at close range. The dashboard shows 85% adoption. The executive sponsor presents it to the board. Beneath the number, the frontline has found three workarounds that let them use the new system to execute the old process. The reports look different. The behavior is identical.

Usage metrics don't measure behavior change. They measure login frequency. Nobody is asking whether the workflow has actually shifted, whether the decision rights have moved, whether the accountability model reflects the new operating reality. The old way of working persists under a layer of new tools.

That's the invisible failure mode. The market hasn't named it.

And AI makes it worse for a specific reason. Previous technologies replaced discrete tasks. AI changes the relationship between a person and their work. It redistributes judgment, collapses decision layers, and moves information access from hierarchical to flat. Those are structural changes to the operating model. Treating them as adoption challenges — something communications and training can handle — guarantees the workaround.

## 05 83 CONDITIONS ACROSS 14 DOMAINS

The conditions are not abstract. They are specific and measurable.

Over thirty years of field work, I've cataloged 83 structural conditions organized across 14 domains. These conditions define everything an organization was built to reward before the change was introduced: the incentives, the decision rights, the accountability models, the escalation paths, the metrics, the role definitions, the approval chains.

The domains ask whether the program itself is structurally sound. Whether the technology design accounts for how people actually work. Whether the operating model has been redesigned. Whether the metrics measure the right things. Whether the governance structure can absorb the decisions the transformation requires. Whether the sponsors are making structural decisions or just approving budgets. Whether people leaders are prepared for the weight they'll carry. Whether the people affected know what good looks like on Day One. Whether the people affected helped shape the solution. Whether the organization can adjust when the plan meets reality.

Scan all fourteen domains before go-live. Every unaddressed condition predicts where the workaround will form.

Three diagnostics measure readiness against these conditions.

The Sponsor Assessment measures current state. Twelve to fourteen questions in enterprise language. Red, amber, or green readiness across each domain.

The Operator Stress Test measures the same domains at a specific point in time, using operator language for frontier-tech environments.

The Practitioner Diagnostic helps practitioners identify root causes of recurring problems by opening the relevant conditions from the full inventory.

These are not surveys. They are structural maps. They show where the environment will resist the new behavior before you find out at go-live.

## 06 WHAT STRUCTURAL ARCHITECTURE LOOKS LIKE

Four questions every transformation has to answer.

1. Are the conditions producing the behavior you want? This is the domain-level assessment. If the metrics still measure the old process, the incentives still reward the old behavior, and the role definitions haven't changed, the answer is no. It doesn't matter how good the training was.

MacInnis calls this engineering your inputs. Identify what needs to change at the atomic level of how work gets done. Design the process around that. Then measure whether the outputs follow. When you change what the organization measures, you change what people optimize for. When you change what people optimize for, you change behavior.

2. Has the sponsor made the structural decisions? Closing the legacy system. Changing the scorecard. Reassigning decision rights. Removing the workaround path. These are irreversible commitments that signal the organization is serious. Without them, the change program reaches its ceiling and stays there.

Most sponsors have been trained to approve budgets, review status updates, and show up at town halls. That's executive visibility, not structural sponsorship. The structural decisions are harder. They require the sponsor to close optionality. To make a choice that cannot be undone quietly. To put their name on a commitment that will be visible when it works and visible when it doesn't.

3. Is the operating model designed for the new way of working? Not the technology. The operating model. Who reports to whom. How decisions get made. What gets escalated and what gets resolved locally. Where information flows and where it stops.

Horowitz's point is direct: the authority structure is a communication path. When new technology changes who has information and who has decision rights, the old communication paths become liabilities. The structural move is to redesign them deliberately, before the old paths reassert themselves as informal workarounds.

4. Can the organization adjust when the plan meets reality? No transformation survives contact with the field exactly as designed. The question is whether the governance structure, the feedback loops, and the decision authority are fast enough to adapt without reverting.

These four questions map to the 14 domains and 83 conditions. They can be assessed before go-live. They can be tracked through deployment. And they predict, with high accuracy, whether the change will hold ninety days after the consulting team leaves.

## 07 THE TEST

The test at close is simple.

We leave on day one post-launch and return ninety days later. The conditions are still holding. Not because the right people are paying attention. Because the environment makes it easier to work in the new way than to go back.

That's permanence. Not because someone is enforcing compliance. Because the structure makes the new behavior the path of least resistance.

Most change programs measure go-live completion. The dashboard goes green. The team celebrates. The consulting firm writes the close-out report. And six months later, the CFO is asking why the technology isn't delivering the returns the business case promised.

The answer is almost always the same. The behavior changed for the duration of the program. Then the conditions pulled it back.

We measure what happens after the program ends. That's the only metric that matters.

---

## 08 ABOVE PERMANENCE

Above permanence is a harder question most organizations aren't ready to ask: whether the fundamental design of the enterprise itself is compatible with the new way of working.

This goes beyond any single transformation. It asks whether the organizational architecture — the thing that predates every initiative, every program, every technology deployment — was designed for the operating model the organization is trying to become.

I spent fifteen months embedded with a frontier-tech defense company scaling across seven functions and three geographies simultaneously. The conditions problem showed up at a speed and intensity I hadn't seen in traditional enterprise work. Every structural gap that takes six months to surface in a Fortune 500 ERP rollout surfaced in six weeks. The company was moving too fast for communications-heavy change programs and too deliberately for adoption to be left to chance. The operating model itself had to be redesigned continuously, not once at program start and once at close.

That experience clarified the frontier question. The conditions problem is not limited to individual transformations. It lives in the architecture of the enterprise. And the organizations that recognize this early — that treat structural design as a continuous discipline rather than a program activity — are the ones pulling away from everyone else.

Most organizations haven't asked this question. The ones that have are the ones moving fastest.