Astraios Software

THE "INSTITUTIONAL CORE" REALITY CHECK

WHY THE PATH TO AUTONOMOUS AI BEGINS WITH EXCAVATION, NOT REPLACEMENT

The state of adoption

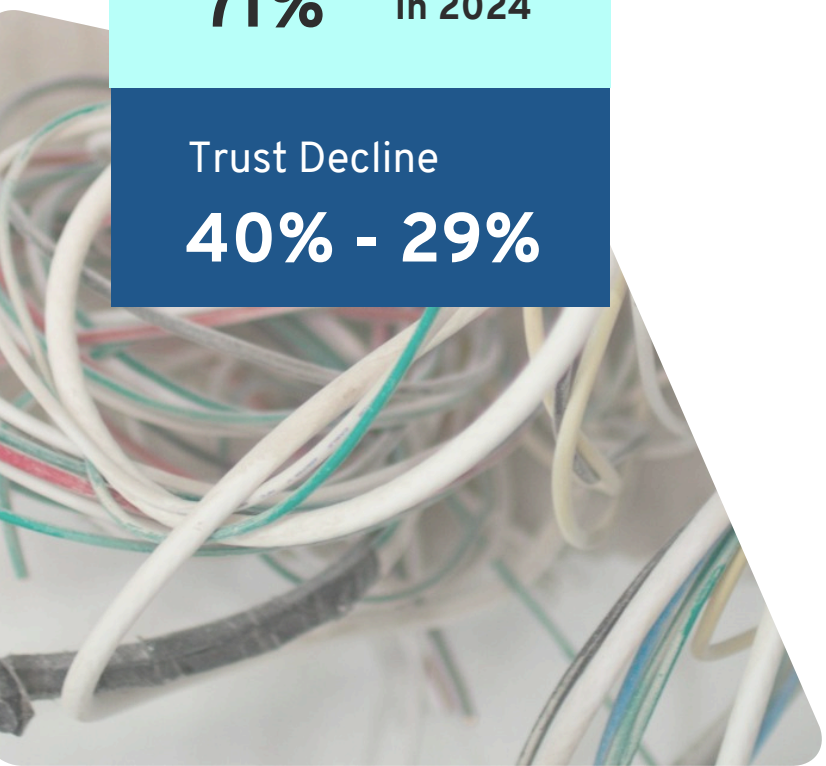
AI coding tools have crossed from experimental to default infrastructure faster than almost any prior enterprise technology wave. Generative AI use in business functions jumped from 33% to 71% between 2023 and 2024 alone (McKinsey). At Google and Microsoft specifically, AI now generates 25–75% of new code, by their own CEOs' account — all of it still reviewed by human engineers before deployment. A16z's Q1 2026 survey of Global 2000 CIOs found 81% of enterprises now run three or more AI model families, up from 68% a year earlier — multi-tool stacks (Claude, Codex, Figma-to-code pipelines) are the norm, not the exception.

AI Use In Business

71% In 2024

Trust Decline

40% - 29%



What teams report back

Adoption and trust are moving in opposite directions. Developer trust in AI-generated output fell from 40% to 29% between 2024 and 2025*. The top frustration, cited by 66% of developers, is AI output that looks almost right but isn't, and 45% say debugging AI-generated code now takes longer than writing it manually would have. At the leadership level: 79% of organizations report real challenges adopting AI, and 54% of C-suite executives admit the effort is straining their company internally — despite 59% of companies investing over $1 million annually in it.

- Stack Overflow, 2025 Developer Survey (data collected May–Aug 2025; published Dec 2025)

WHERE IT ACTUALLY BREAKS DOWN

The gains don't compound

OpenAI's 2025 enterprise data found a fourfold productivity gap between AI power users and typical employees inside the same organization, using identical tools. Only 29% of organizations see significant ROI from generative AI, despite individual gains reaching 5x for top users. The productivity is real. It just stays trapped at the individual level instead of compounding into the business — and the reason is structural, not a tooling gap.

The reality check

The original goal was productivity, and on paper it's delivered: a week of work compresses to two days, code and specs land fast, exactly as promised. But the speed shows up entirely on the generation side. Testing, orchestration, and issue management haven't kept pace — so teams increasingly ask one AI tool to test another AI tool's output. That sounds efficient. It isn't, because it skips the real question: tested against what? The original Figma design? The original business requirement? A probabilistic system has no native concept of a deterministic business outcome — it can check whether code runs, not whether it does what the business needed. It runs for a while. Then one morning it doesn't, and the team ends up debugging the debugger.

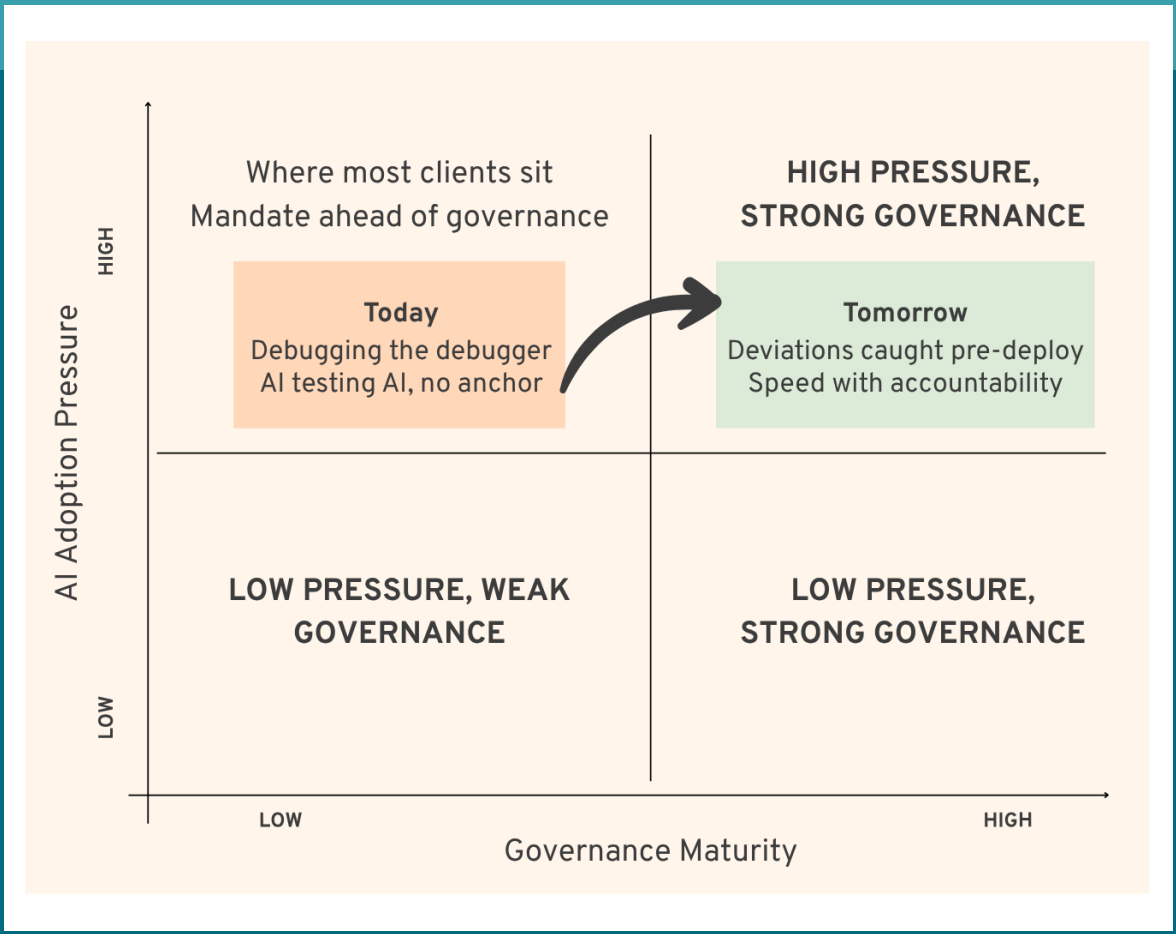
We bridge that exact gap — deterministic business outcomes on one side, probabilistic technical execution on the other.

THE SPECIFIC FAILURE MODE WE CATCH

Our work sits inside organizations — healthcare delivery groups, GCCs, and other enterprises under AI-adoption pressure — where leadership has mandated AI tools (Claude, Codex, Figma-to-code) faster than governance has caught up. The friction isn't in the tools. It's in the gap between the mandate and the team executing it without a map.

The root issue starts before a single line of code is written: business problems rarely get translated into a use case written in a language business, tech, testers, and the AI engine itself can all read the same way. Ambiguity at the prompt becomes ambiguity in the output.

From there, the gap compounds. AI-generated code is probabilistic, not deterministic — it will always carry some deviation from what a human would have written by handMost team leads have no reliable way to tell whether a given deviation is harmless or a defect. Fewer still are checking for a third risk: whether the code is genuinely original, or substantially reproduced from somewhere else, with the licensing and liability questions that brings. Relevance, correctness, and legitimacy — none of the three gets checked, because no one owns checking them.



This is precisely the layer we install: a shared use-case language that keeps business intent legible all the way through to the AI engine, paired with a review discipline that validates AI output against that original intent rather than against itself.

OUR APPROACH

A three-phase framework: Excavate → Architect → Activate

1. Excavate
Shared use-case language business, tech, testers, AI

Fixes: prompt-to-use-case translation gap

→

2. Architect
Review discipline vs. deterministic ground truth

Fixes: probabilistic deviation, unowned by anyone

→

3. Activate
Phased rollout, checkpoints on relevance & legitimacy

Fixes: unchecked code relevance, accuracy, origin

1. Excavate — fix the prompt & case gap

Before any code gets written, we build the shared use-case language that lets business, tech, testers, and the AI engine itself read the same intent the same way. This is where the ambiguity that later shows up as bad output actually gets removed — at the source, not after the fact.

2. Architect — fix the unowned deviation

We install the deterministic ground truth that probabilistic output should be measured against — the original business requirement, not another AI's judgment of "looks right." This is the review discipline that tells a team lead whether a deviation is harmless or a defect, instead of leaving it to guesswork.

3. Activate — fix the unchecked code

Phased rollout with explicit checkpoints on the three things nobody currently owns: is the code relevant to the actual requirement, is it correct, and is it legitimately original. Each phase clears these checks before the next phase expands scope.

WHY US

Astraios Software

AI Archeology

Every enterprise runs on institutional logic that nobody fully wrote down. It lives in spreadsheets, in the heads of senior staff, in the workarounds that survived three system migrations. When AI is layered on top without understanding it, the automation fails in ways that are hard to diagnose and expensive to fix.

AI Archeology is need of the hour. We unearth the past with AI-powered precision. We surface the operational logic, map the decision flows that actually run the business, and identify what is genuinely automatable versus what requires human judgment to remain accountable.

#46/4, Groud Floor, Novel Tech Park
Hosur Road, Bengaluru, KA - 560068

sales@astraios.in

www.astraios.in