

ESSAY · ENTERPRISE AI

Your Business Unit Became an AI Pilot Factory — How to Govern It Back Into Production

Monthly workshops, showcase meetings, thirty agents that are really system prompts, nothing live. How a business unit governs its own AI work into production.

PILOTS

GOVERNANCE

LEADERSHIP

Burak Galiba · August 3, 2026 · 7 min read
<https://burakgaliba.com/blog/the-ai-pilot-factory>



SCAN TO READ ONLINE

I've written about how [applause becomes the intake process](#) when nobody builds a real one. This is what happens one level down, after the applause — because business units don't respond to that incentive once. They industrialize it.

The pattern looks like this. A monthly enablement workshop where teams build agents. A showcase where employees present those agents to executives. A channel full of screenshots, a spreadsheet with forty entries, a leaderboard of who built the most. From the outside it is the best-looking AI program in the company: broad participation, genuine enthusiasm, visible output every month.

Then someone asks the question that ends the good mood. *How many of these are in production?* The honest answer is usually zero — sometimes "one, sort of, if you count the one that three people use." A unit that has become very good at manufacturing pilots has become good at manufacturing exactly one thing, and it isn't value.

What the factory actually produces

Look closely at what gets demoed at these showcases. In most cases it's a carefully written system prompt on whatever chat platform the company licensed, sometimes with a few documents uploaded alongside it.

I want to be fair to that artifact, because there is something genuinely valuable inside it. A good system prompt is a domain expert writing down how the work is actually done — the exceptions, the escalation rules, the four cases where the official procedure is quietly wrong. That is tacit knowledge, it is scarce, and nobody in IT can write it.

But it's a specification, not a product. Everything expensive still sits outside it: where the data comes from when it isn't a PDF someone exported, who is allowed to see which records, what happens when the answer is wrong and who notices, who maintains it in month seven when the policy changes. Writing a fortieth prompt advances none of that. The factory has one machine and it stamps out the same part.

The queue is served at a fixed rate

Here is the arithmetic that most business unit leaders never do. Production capacity — engineers who can build the integrations, the permissions, the evals — is roughly fixed this quarter. Call it three or four projects. Pilots enter your queue at ten a month. Nothing about the entry rate changes the exit rate, so the only thing that grows is the wait.

And the wait isn't the whole cost. Every unshipped pilot carries a bill: it's usually in unofficial daily use by the team that built it, quietly answering from data that [ages from the moment it was exported](#). It holds an expectation with an executive who saw it in March. It keeps a standing claim on IT's attention every time its sponsor asks for a status. Forty of those claims is a permanent low-grade denial-of-service on the three things that could actually ship.

That's the AI graveyard, and it isn't filled by failures. It's filled by successes nobody had the capacity to finish.

The second cost doesn't show up in any report

The first cohort builds something good, and waits. The second cohort watches them wait. By the third workshop, the smartest people in the room have worked out the actual rule of the game: building is celebrated, shipping is not available, and the answer to "when does mine go live?" is a change of subject. So they stop showing up — not loudly, nobody writes a memo. Attendance drifts, submissions thin out, and the people who arrived with the most energy disengage first, because they're the ones with somewhere better to spend it.

This is the part I'd put in front of any executive still counting workshop attendance as a KPI. Volunteer enthusiasm is the scarcest raw material in an AI program, and it is not renewable on demand. Spending it on a pipeline you have no capacity to drain converts your best people into your most cynical ones, and the trust that burns is what the next initiative inherits.

Say out loud what the workshop is for

Before any of the governance below, do one honest piece of accounting: divide the number of agents built by the number in production.

If that ratio is thirty to zero, you are not running a delivery pipeline. You are running an AI literacy program — and that is a legitimate, valuable thing to run. People who can write a precise prompt, judge a bad answer, and spot a workflow worth automating are worth more than any single agent you'll ship this year.

So call it a literacy program, measure it like one, and stop presenting its output as a portfolio. Then, separately and much smaller, run the thing that *is* a portfolio.

Govern it like a portfolio you're accountable for

None of this requires central permission. All of it sits inside a business unit's own authority:

- **Cap the queue, not the ideas.** Ask engineering one question — how many of ours can you take this quarter? Whatever the number is, that's how many active builds you sponsor. Everything else stays an idea in a register: written down, ranked, alive, unbuilt. Ideas are free to hold. Half-built systems are not.
- **Rank by workflow, not by wow.** Score candidates on how often the workflow runs, how many people run it, how long it takes them, and how much a wrong answer costs. A dull, high-frequency task beats a dazzling rare one every time. The demo that got the loudest applause is frequently the one with the worst arithmetic.
- **Require a process owner, not a builder.** Every slot needs a named person who owns the workflow and will commit to changing how their team works when the system lands — not the person who

wrote the prompt. If nobody will change a process for it, you don't have a project; you have a toy with a fan.

- **Make each candidate answer three questions before it gets a slot.** Where does the data come from when it's live, and who's allowed to read it? What happens when it's wrong, and who finds out? Who owns it in month seven? A team that can't answer these isn't blocked — it just needs a conversation with IT that it hasn't had yet. That conversation is the real work.
- **Consolidate before you build.** Forty prompts usually contain about five patterns. One governed assistant over policy documents can serve eight teams that each demoed their own. The graveyard is often eight versions of the same product, none of which got finished.
- **Close things on a schedule.** Every quarter, everything not in production and not in the active queue gets closed — publicly, without blame, with one paragraph on what was learned. [Killing pilots on purpose](#) is not a punishment; it's how you give someone their Friday afternoons back.
- **Change what the showcase shows.** Two standing agenda items: what went live and what measurably changed, and what we closed and why. Demos stay, but only in a clearly-labelled "asking for a slot" segment, with the data source stated on the slide. Whoever ships gets the recognition the demo used to get, or your incentive system is still instructing people to stop at demo day.

The most valuable thing your unit can build isn't an agent

Here's the move I'd make with the next workshop. Don't build a thirty-first agent. Take the top-ranked candidate and spend the session writing its evaluation set: a hundred real questions from real cases, each with the correct answer and a note on why, written by the people who actually know. Include the ugly ones — the ambiguous cases, the ones where the right answer is "escalate this," the ones where two systems disagree.

That artifact is the difference between a system people trust and a system people quietly stop using. Engineering cannot produce it. A vendor cannot produce it. It is the one input only your unit can supply, it turns "make it live" into something with a finish line, and it takes an afternoon.

It's also a much harder test of whether the use case is real than any demo. Teams that can't fill the sheet usually discover they were automating a workflow they had never precisely defined.

Where to start

Two artifacts, both smaller than the spreadsheet you already maintain. A ranked register of every AI idea in the unit, with the four workflow numbers next to each. And one number from engineering: how many slots you get this quarter. The register minus the slots is your backlog; the slots are your portfolio; everything else stops being called a pilot.

Then send the next workshop invitation with a different assignment on it.

If you want a read on whether your organization has the machinery or just the enthusiasm, my free [AI Readiness Score](#) takes about ten minutes — 20 questions across pilots, data, talent, and governance, including the ones this essay is really about: who decides what advances, how many things you run at once, and what happens to the ones that don't make it.

Read the essay online at <https://burakgaliba.com/blog/the-ai-pilot-factory>

I built a free AI Readiness Score — 20 questions across pilots, data, talent and governance, about ten minutes, no email required: <https://burakgaliba.com/readiness>

© 2026 Burak Galiba · <https://burakgaliba.com> · Views are my own.