



BURAK GALİBA
Thoughts on AI Transformation

ESSAY · ENTERPRISE AI

Prototypers Are Not a Program — the Four AI Roles Enterprises Forget to Staff

Boris Cherny's five work archetypes — prototyper, builder, sweeper, grower, maintainer — read as a diagnostic for enterprise AI programs, which hire and reward the first and leave the other four unstaffed.

LEADERSHIP

TRANSFORMATION

PILOTS



Burak Galiba · October 11, 2026 · 6 min read

<https://burakgaliba.com/blog/prototypers-are-not-a-program>

SCAN TO READ ONLINE

In late June, Boris Cherny — who created Claude Code at Anthropic — posted a short framework that has been passed around ever since. His argument: as engineering, product, design and data science blur together, the useful way to describe people is no longer by function but by the kind of work they're good at. He named five archetypes.

The **prototyper** produces new ideas fast, most of which never ship. The **builder** turns a prototype into something production-grade. The **sweeper** cleans up: simplifies, optimizes, and removes things — including unshipping features that shouldn't exist. The **grower** takes something that works and iterates on it until it fits what users actually need. The **maintainer** owns a mature system and keeps it secure, reliable and efficient as it scales.

Two details in the post matter more than the list. Most people span two of the archetypes, sometimes three, and they don't line up with job titles — there are prototyper engineers and maintainer designers. And the mix a product needs changes with its stage: a new product needs prototyping, building and sweeping; a growing one needs building, sweeping, growing and some maintenance; a mature one needs mostly sweeping, growing and maintenance.

He was writing about product teams at a frontier AI lab. I want to borrow the framework for something narrower, because it describes a problem I see in enterprise AI programs more precisely than anything else I've read this year.

Most enterprise AI programs are staffed for one archetype

Look at who an enterprise AI program actually recruits and celebrates. The enablement workshop, the hackathon, the monthly showcase, the innovation team, the "AI champions" network: every one of them selects for prototypers. They reward the person who can produce a convincing new idea quickly and demo it to an executive.

That's not a mistake in itself. Prototypers are genuinely valuable, and a program with none would never find anything worth building. The problem is that it's the only archetype most programs know how to find, fund or recognize. Which is how a business unit ends up [running an AI pilot factory](#): forty demos, monthly applause, nothing in production. That's the output you'd predict from a team that is all archetype one.

Cherny's stage mapping makes the gap explicit. Even a brand-new product needs builders and sweepers alongside the prototypers. An enterprise AI program that has been running for two years is not a brand-new product, and it's still staffed as if ideas were the bottleneck.

The builder: the person your demo is waiting for

Every stalled pilot is waiting for a builder — someone who can take a working prompt and a document export and turn it into a system with real data access, permissions, error handling and an owner. I've written about [why that gap opens at IT](#) and [what a paved road to production looks like](#). Through this lens, the shorter version is: you have the prototypers, and you didn't staff the next archetype.

Builders are scarce in most enterprises not because the skills don't exist, but because they're locked inside delivery teams working on the existing roadmap. Nobody's job is to pick up the most promising thing from last month's showcase and make it real.

The sweeper: the role nobody gets promoted for

This is the archetype I'd point any executive at first, because almost no enterprise has it and almost no incentive system rewards it.

A sweeper simplifies. They consolidate eight versions of the same assistant into one. They retire the tools nobody uses. They [kill pilots on purpose](#), publicly and without blame, so the people working on them get their time back. They take out features, reduce the number of models in use, and make the estate smaller and cheaper to run.

Now ask how that work shows up in a performance review. The prototyper presents at the showcase. The builder gets the launch announcement. The sweeper's best quarter is one where twelve things stopped existing, which looks, on most scorecards, like nothing happened. So nobody volunteers for it, and the estate grows until someone outside the program — usually finance — does the sweeping for you, with less judgment.

If you want one structural change, make unshipping visible. Report it with the same prominence as launches.

The grower: the owner of the number

The grower takes something already live and keeps improving it until it measurably changes how the work gets done. In enterprise terms, this is the person who [owns the value number](#) — who watches adoption, talks to the users, and iterates on the workflow rather than the model.

In most AI programs, this role doesn't exist after launch. The builders move to the next project, the sponsor moves to the next priority, and the system sits at whatever quality it had on day one. Which is why so many "successful" launches show flat usage six months later: nobody's job was to grow them.

The maintainer: the one who notices it stopped working

The maintainer owns a mature system and keeps it reliable as it scales. For AI, that means the work I described in [Shipped Is Not the Same as Working](#): an evaluation set that's re-run, an incident path, a kill switch, a name next to the system. It's the least glamorous archetype and, for anything customers or money depend on, the one you can least afford to leave empty.

Use the five as a staffing diagnostic

Here's the exercise I'd run. List the people who actually spend time on your AI program — not the steering committee, the people doing the work. Next to each name, write the one or two archetypes they're genuinely good at, using Cherny's definitions rather than their job titles.

Then count.

If the list is mostly prototypers, you've found why things don't ship, and no amount of additional ideation will fix it. If there are builders but no sweepers, your estate is growing faster than anyone can run it. If nobody is a grower or a maintainer, everything you've launched is quietly decaying.

The fix is rarely new hiring. More often, the builders, sweepers and maintainers already work in your organization — in delivery teams, in operations, in the people who keep existing systems running — and the AI program simply never asked for them, because its intake, its events and its recognition were all designed around archetype one. Change what the program asks for, and what it celebrates, and you'll find more of the other four than you expected.

Where to start

Do the count for your AI program this week. Then look at your last two showcases or AI updates and ask which archetypes they recognized. If the answer is "prototypers" twice, you know what to change in the third: put the person who shipped something, the person who retired something, and the person who kept something running on the same stage as the person with the new idea.

I built a free [AI Readiness Score](#) that comes at this from the operating-model side — 20 questions across pilots, data, talent and governance, about ten minutes, no email needed to see the result. It asks how AI work gets proposed, prioritized, funded and staffed, when you last killed a pilot, and how you know a system still works after launch: building, sweeping and maintaining, in different words.

Read the essay online at <https://burakgaliba.com/blog/prototypers-are-not-a-program>

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.