BGL BURAK GALİBA
Thoughts on AI Transformation

ESSAY · ENTERPRISE AI

Your AI Agent Has a Launch Date. It Needs an End Date.

Most enterprises have no criteria for retiring AI that already shipped. Gartner found only 17% of CIOs do. Three retirement triggers to set at launch, an end date that makes renewal the decision, and who signs it.

AGENTIC AI GOVERNANCE ROI

Burak Galiba · November 14, 2026 · 8 min read
https://burakgaliba.com/blog/when-to-retire-an-ai-agent

SCAN TO READ ONLINE

What happens to an AI agent that shipped, still works, and isn't worth running?

In most organizations I'd expect the honest answer to be: nothing. Nobody decided to keep it. Nobody was ever asked.

I've written about killing pilots before they ship and about checking whether a shipped agent still works. This is the question that comes after both, and I skipped it both times.

Gartner put a number on it in October. In its 2027 CIO and Technology Executive survey of 2,002 CIOs and technology executives, 31% of CIOs said they have guidelines for deciding when an AI pilot should not move to production. Only 17% have guidelines for abandoning low-value AI initiatives already in production. In the same survey, only 13% said their enterprise has realized significant or transformational value from AI so far.

Read the first two figures together. Roughly one CIO in three has a rule for keeping a pilot out of production. Roughly one in six has one for anything that got through. A missing rule doesn't mean nothing ever gets retired. But without one, I'd expect production to behave like a one-way door: things go in, and getting one out takes an argument nobody is assigned to make.

The survey also hints at what the rule is worth. Organizations with guidelines for abandoning low-value AI in production were 2.6 times more likely to be in the group Gartner calls vanguards, its strongest performers. That is a correlation, not proof that the rule produces the result. It's still the direction I'd bet on.

Why production AI never gets switched off

Pilots are hard enough to stop: there's sunk cost, a sponsor's name and a demo that still looks good. But a pilot at least runs into a forcing event. Sooner or later it needs more funding or has to pass a production gate, and at that moment someone has to say yes.

Production has no such moment. Once an agent is live, it continues unless somebody argues for it to stop, and making that argument is nobody's job.

Four things keep it that way:

- **The launch was the finish line.** The project closed, the team was reassigned, and the business case was filed. Nobody reopens a business case for something already delivered.
- **The cost is spread too thin to see.** The API bill sits in one budget, the platform licence in another, and the time people spend maintaining it in none. No single line is large enough to question.
- **Retiring it contradicts an announcement.** A production agent had a launch email with a sponsor's name on it, and switching it off reads as admitting that email was wrong.
- **Nobody has the authority.** IT runs it but didn't ask for it. The business asked for it but doesn't run it. Each assumes the decision belongs to the other.

None of this requires anyone to behave badly. It only requires that continuing costs nothing to decide and stopping costs a meeting.

An idle agent is not like an idle dashboard

There's a fair objection here. Enterprises are full of applications, reports and dashboards nobody would rebuild today, and they've lived with that for decades. Why treat an agent differently?

I see two reasons. A forgotten dashboard sits unopened. A forgotten agent keeps answering whoever asks, fluently, from sources nobody is updating. And an agent holds live credentials: it reads data and, increasingly, acts on it. An unused report is clutter. An unowned agent is an exposure that still talks to people.

Working and worth running are two different tests

When I wrote about day-90 monitoring, the test was quality: are the answers still right? That test needs an owner and a living evaluation set.

Worth is a separate test, and an agent can pass one while failing the other. It can score well on its evaluation set every month and still have a fraction of the users its business case assumed. The process it supported may have been reorganized. The platform it sits on may now do the same job natively. Or the value was never baselined in the first place, so nobody can say what it earns.

The quality review asks whether the agent does what it was built to do. The worth review asks whether you would build it again today. Going by Gartner's 17%, few organizations have any rule behind the second.

What an agent nobody would rebuild actually costs

The run bill for any one agent is rarely the line anyone questions, which is part of why it survives. The costs that matter are elsewhere.

Upkeep. A production agent needs its evaluation set maintained, its sources kept current and its owner's attention every month. On an agent nobody values, you either pay for that or you skip it and get the decay I just described.

Attention. An inventory of two hundred agents gets a thinner review per agent than an inventory of sixty. Every agent you keep without a reason dilutes the scrutiny on the ones that matter.

The portfolio total. One agent is cheap. A few hundred, each with a run bill, a licence and a slice of somebody's time, is a budget line. In the same Gartner survey, 51% of CIOs expected AI to increase the total cost of ownership of their technology, and 58% said they are under increased pressure to deliver cost improvement from AI. Both pressures land on the same portfolio, and one that only grows makes the second harder to meet.

I've argued before that capping everyone's tokens is the wrong fix. Retiring what isn't earning is a more targeted one. And just as kills fund ships at the pilot stage, retirements fund the upkeep of the agents worth keeping.

Three retirement triggers to set at launch

The time to decide what "not worth running" means is before launch, when nobody is defending anything yet. I'd set three thresholds for every agent and write them next to its business case.

A usage floor. The number of active users or transactions below which the agent isn't worth maintaining. Take it from the business case. My starting line would be half of what the case assumed, two quarters after launch. Four hundred planned and forty actual isn't a close call.

A value floor. The baselined metric the agent was supposed to move, and the minimum movement that justifies keeping it. The simplest floor is the improvement the business case used to justify the build. An agent launched without a baseline can't pass this test, which is a useful thing to discover before launch rather than after.

A cost-per-outcome ceiling. Total run cost, including upkeep and the owner's time, divided by the outcomes it produces: tickets resolved, documents processed, requests handled. When that figure crosses what the next-best option costs, whether that is the manual process it replaced or a platform feature that now does the same job, the agent has lost its argument.

Any one of the three is enough to trigger a review. An agent doesn't have to fail all of them to deserve a conversation, and a trigger that trips in month three shouldn't wait for month twelve.

Then add the piece that makes the thresholds matter: **an end date.** Every agent gets one at launch. I'd set it six months out for anything new and twelve months out for an agent that has already been renewed once. On that date the agent stops unless someone renews it.

That reverses the default. Today, retirement needs a signature and continuation needs nothing. With an end date, renewal is the decision that needs a signature. It's the difference between a contract that auto-renews and one somebody has to sign again.

Who signs the renewal, and what it costs them

The business unit that owns the value number signs. IT brings the cost and usage figures to the review. It shouldn't make the call on worth, for the same reason it shouldn't own the benefit line.

A signature that costs nothing gets given every time. So the renewal has to carry the bill: the agent's full run cost, upkeep included, shows on the signer's budget for the period they're renewing. Signing then means agreeing to pay for another year, not agreeing that the agent seems fine.

Three rules keep the end date from being either a formality or a threat:

- **Notice, not a cliff.** I'd give the owner sixty days' notice, with the three thresholds and where the agent stands against each. Nobody should cause an outage by calendar.

- **A named hand on the switch.** If the date passes unsigned, IT disables the agent. That isn't IT judging its worth. It's IT carrying out a decision the business made by not renewing.
- **Security keeps its own veto.** The business signs for worth. Security can still pull an agent for risk at any time, renewed or not.

The renewal conversation is short when the agent is earning: here are the thresholds, here is where it stands, signed. It's also short when the agent isn't. If no one in the business will put their name and their budget on another year, that is the decision, and nobody had to be the villain who proposed it.

This is different from the kill switch I described for incidents. A kill switch is for an agent that is doing harm this afternoon. A retirement is planned, scheduled and unremarkable.

Retire it properly

Switching an agent off is the easy part. A retirement that's actually finished has four more steps:

- **Tell its users** what is going away, when, and what to use instead.
- **Revoke its access.** Service accounts, API keys and data permissions go with it. An agent that's off but still holds credentials is still an exposure.
- **Keep what it taught you.** The evaluation set and the reasons it missed its thresholds are worth more to the next business case than the agent was.
- **Take it off the inventory**, with the date and the reason.

Where to start

Every agent you already run was launched with no thresholds and no end date, so start with the inventory. If you have one, add a column: end date. If you don't, four columns on a spreadsheet is enough to begin, and this makes five. Give every existing agent a date inside the next two quarters, oldest first.

Then take your three oldest production agents and ask one question of each: *if this didn't exist, would we build it today, at today's cost?* The agents where the room goes quiet are your first retirements.

If you want a harder version of that question, I built a free Ship / Fix / Kill triage: 15 questions written for pilots, and the ones on value evidence and unit economics apply just as well to an agent that shipped a year ago.

Read the essay online at https://burakgaliba.com/blog/when-to-retire-an-ai-agent

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.