



# The AI Last Mile Problem

What It Actually Takes to Move Enterprise  
AI from Pilot to Production

## Introduction

Enterprise AI stalls at the point where a prototype has to become a production system. With model capability and generation no longer the limiting factors, the difficulty lies in proving that a system works. This requires tying AI's output to a decision someone will stand behind, and getting business and technology leaders to agree on what defines success.

This paper examines the vendor response taking shape around this gap: small data and engineering teams placed inside a customer's environment against a defined outcome and a fixed clock. The category is splitting over a question most buyers ask too late: what remains with the customer when the engagement ends, and what that costs to run and extend. The analysis draws on Futurum Research survey data covering 600 enterprises that develop AI applications in-house, and on the engagement models now on offer, Snowflake's among them. It closes with what a CIO and a business-unit owner should require before signing the engagement agreement.



# The Constraint Moved from Building to Proving

Among enterprises that develop their own AI applications, 85% expect to increase that spending over the next 12 months, and just over half cannot point to a return on investment (ROI).<sup>1</sup>

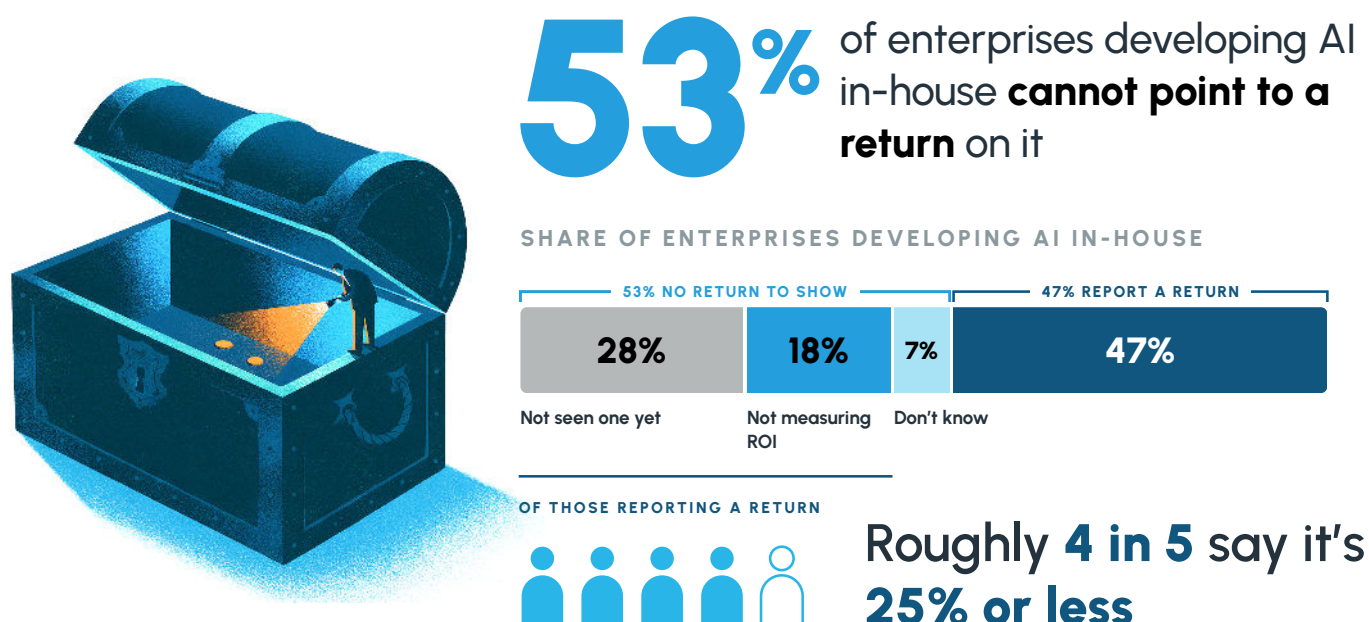
The usual explanation for that gap blames model capability or immature tooling. The Futurum data locates the problem inside the organization.

Producing a working prototype became cheap and fast. The difficulty shifted to proving the system works, integrating it with the systems and processes the business already runs, and naming the person who owns the decision the system informs. Those are organizational tasks, and adoption without reorganization stalls at the pilot stage.

Enterprises unable to point to a return fell from 60% in March 2025 to 53% in March 2026, so the trend is improving from a low base (see Figure 1). Yet, the sharper number inside it is the 18% measuring no return at all. A pilot with no measure attached can run for quarters, even years, before anyone writes it off, and the spending it consumes does not appear in a variance review.

The executive approving the next AI budget is buying an operating-model change, whether the invoice explicitly states it or not.

Figure 1. Where the Return Goes Missing



<sup>1</sup> Source for all ETR datapoints: ETR. (2026, March). AI Product Series, Theme 3: Building AI Internally. Enterprise Technology Research. Base n=600 respondents at organizations developing AI applications in-house; 27% CxO, 52% VP, director, or manager level; 62% large enterprise. Trend figures reference the March 2025 (n=600) and September 2025 (n=600) waves of the same series. Figures for enterprises unable to point to a return, respondents expecting increased spend, and absence of enterprise-wide governance are sums of the relevant response categories.

# The Last Mile Breaks in Four Places

Enterprises developing AI in-house name the same three obstacles in every Futurum wave on the subject. Lack of skilled personnel leads at 52%, data quality and availability follows at 51%, and integration with legacy systems holds at 44%. Budget constraints rank seventh of eight substantive options at 29%.<sup>1</sup>

Those three led in March 2025, September 2025, and March 2026, an 18-month pattern across three consecutive survey waves. And integration is consistent across the three waves.

Calling this one obstacle hides four separate breaks: evaluation and data, integration, ownership, and value definition. Table 1 assigns each one an owner and a test.

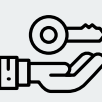
They compound and disguise each other. A break at any one of the four still presents as a technology problem at the next steering or project review, which routes the fix to the wrong owner. Some projects never cleared production; some shipped but cost more than they returned; and some were written off as impossible when the real problem was that nobody owned them.

Production AI systems introduce distinct operational requirements. The system must have observability built in for nondeterministic output, an identity model that specifies which permissions an agent inherits when it acts for a user, cost governance at each invocation, and a rollback path when out-of-bounds or inaccurate outcomes occur. A system that files the order and books the follow-up must be governed accordingly, or it is not governed at all.

Governance of AI has not kept pace with our ability to use AI. Of these enterprises, 58% run without an enterprise-wide AI governance framework, and AI-specific security risks entered the challenge list in March 2026 at 44%, level with integration in its first wave.<sup>1</sup>

The failure is organizational, and it shows up in the production readiness review or in an engineering ticket in production.

Table 1. The Four Failures and Their Owners

| FAILURE                                                                                                                                            | WHAT BREAKS                                                                                                                         | OWNER                          | TECH → BUSINESS                                                          |
|----------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|--------------------------------|--------------------------------------------------------------------------|
| 01  <b>Evaluation and data</b><br><b>51%</b>                    | The 20%-to-90% climb runs on evaluation sets and feedback loops that agent teams skipped.                                           | <b>Data and AI engineering</b> | <div><div></div><div>Tech</div><div>Joint</div><div>Business</div></div> |
| 02  <b>Integration</b><br><b>44%</b> <small>FLAT 18 MO.</small> | Legacy systems were not built for a nondeterministic actor to write to.                                                             | <b>IT, platform</b>            | <div><div></div><div>Tech</div><div>Joint</div><div>Business</div></div> |
| 03  <b>Ownership</b>                                            | Business and technology work the same problem from opposite ends. Ownership rarely survives the departure of the executive sponsor. | <b>Both, jointly</b>           | <div><div></div><div>Tech</div><div>Joint</div><div>Business</div></div> |
| 04  <b>Value definition</b>                                     | Nobody sized the outcome, so nothing got prioritized.                                                                               | <b>Business owner</b>          | <div><div></div><div>Tech</div><div>Joint</div><div>Business</div></div> |

Source: Futurum ETR, AI Product Series, Theme 3 (Building AI Internally), March 2025, September 2025, and March 2026 surveys (n=600 each).



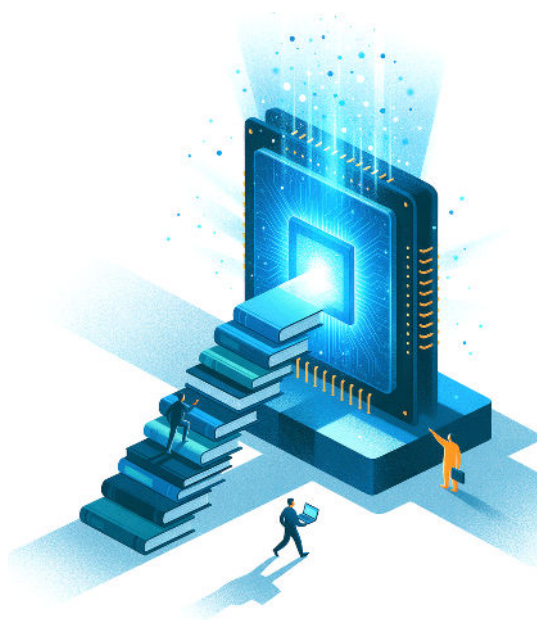
## Internal Teams Rarely Close the Gap Alone

The gap is structural, so hiring harder does not close it. Scarce talent tops the list of challenges and is the least interesting reason for it. The harder problem is knowing what is worth building, and no team can scope that from within a single company's history alone.

Enterprises are already paying to build that muscle in-house. Upskilling existing developers rose from 64% in March 2025 to 73% in March 2026, while third-party AI consulting held roughly flat, between 33% and 36%<sup>2</sup> (see Figure 2). These figures reflect direct buyer budget allocations rather than vendor-reported adoption metrics.

Decision rights are split, too. Nobody convenes the workshop where business and platform owners commit to the same success criteria on the same date, and buying authority now spans both.

*Figure 2: How enterprises address the need for specialized AI skills*



**73%** **upskilled or retrained** existing developers in March 2026, up from **64%** a year earlier

SHARE OF ENTERPRISES, BY APPROACH

○ March 2025 ● March 2026

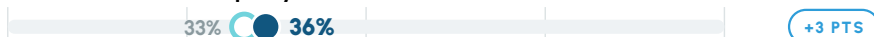
**Upskilled or retrained existing developers**



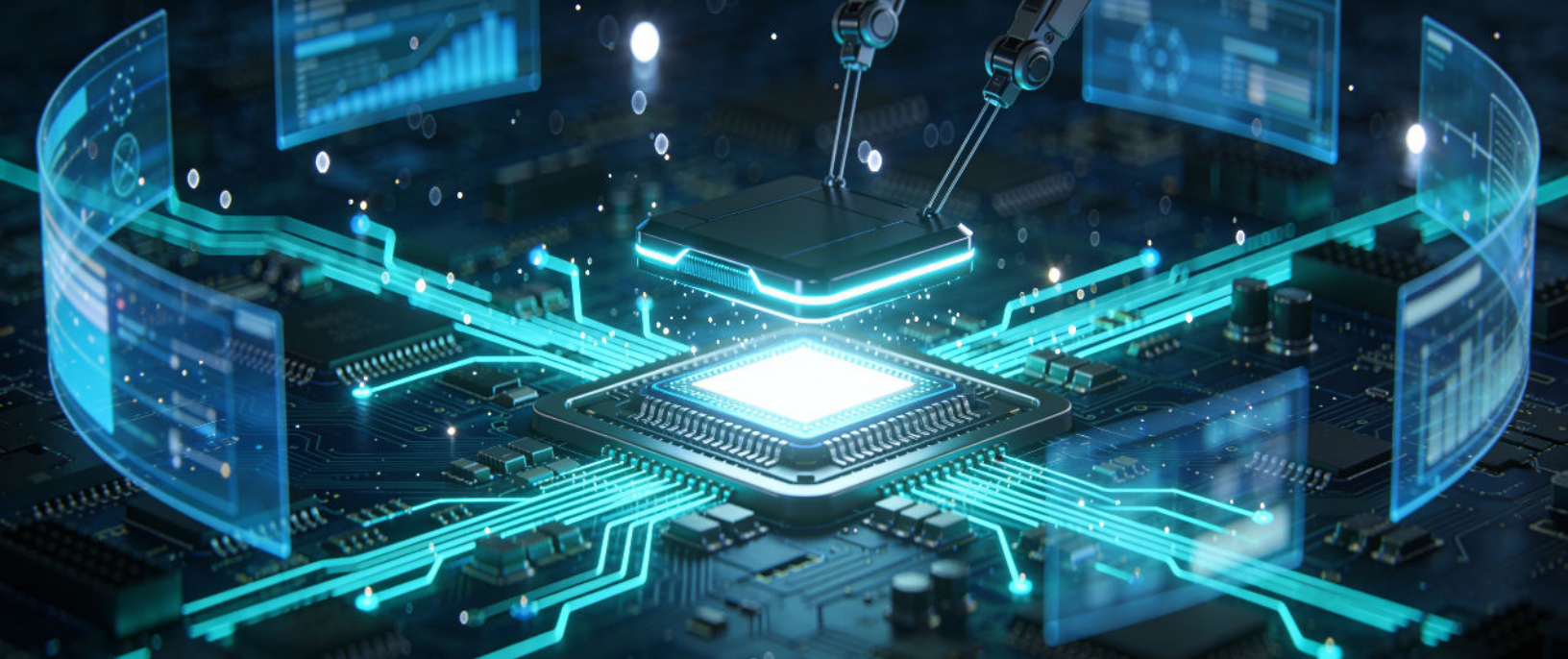
**Hired new AI-skilled developers**



**Partnered with third-party AI consultants**



<sup>2</sup> ETR, AI Product Series, Theme 3 (Building AI Internally), March 2025 and March 2026 surveys (n=600 each).



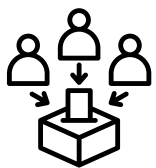
# Embedded Engineering Programs Share a Label But Differ in Structure

## Offerings Under Three Operational Patterns

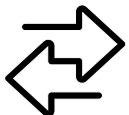
Several platform vendors now field forward engineers inside customer environments. Offerings fall into one of three structures, and the structure predicts what the customer is left with at project-end. Table 2 scores each pattern against the four tells.



**Permanent Embedded Teams:** The vendor team remains on site indefinitely. The underlying system remains opaque to the internal engineering staff, preventing internal capabilities from compounding and increasing annual operational expenses.



**Staff Augmentation Models:** The vendor bills engineering hours against a rate card without committing to a specific business outcome. The customer bears the delivery risk while treating the contract as a strategic partnership.



**Transfer and Exit Engagements:** The vendor scopes work to a defined deliverable with fixed pricing and an explicit end date. The customer retains the production system and the internal operational knowledge required to maintain it.

These are not points on a maturity curve. A vendor picks one because of how its services motion is funded, and the buyer inherits that choice.

# Four Tells Separate a Real Exit from a Slow Subscription

Each tell is answerable from a contract before any work starts (see Table 2).

- **Contract Payment Structure:** Contracts bill either for engineering hours or for a defined deliverable. Outcome-based pricing forces the vendor to qualify the work before accepting it, and the strongest versions set a business-value floor during scoping. That floor is the mechanism that keeps an engagement from becoming another pilot.
- **Asset and Intellectual Property Ownership:** The agreement must establish whether the customer owns the working system, the data assets, and the right to operate and extend it without the vendor present. The carve-out on co-developed IP is where a clean exit quietly turns into a subscription.
- **Whether the Vendor Will Build Outside Its Own Stack:** A team that builds only on its employer's platform is selling the platform, and the outcome is the pretext.
- **Whether the Vendor's Own Product Changes as a Result:** A vendor that generalizes field work into its roadmap has a reason to make the solution durable. A vendor that does not is running a services line under a product logo. The mechanism cuts both ways: a team with roadmap authority has a standing incentive to turn a customer's problem into a shipped feature, and the customer whose differentiated logic becomes that feature has funded someone else's roadmap. Price that risk in the IP clause.

Table 2. Four Tells Scored Against the Three Patterns

| TELL                                                                                                                                           | ASK THIS                                                                                                                                       | WHY IT MATTERS                                                                   |
|------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|
| 01  <b>What the contract pays for</b>                       |  <b>Hours, or a named deliverable?</b>                      | Outcome pricing forces the vendor to qualify the work first.                     |
| 02  <b>Who owns what is built</b>                           |  <b>Is the system and data operable without the vendor?</b> | The co-developed IP carve-out is when an exit turns into a subscription.         |
| 03  <b>Whether the vendor builds outside its stack</b>      |  <b>Or only on its own platform?</b>                        | Only its own platform means the outcome is the pretext for selling the platform. |
| 04  <b>Whether the vendor's product changes as a result</b> |  <b>Does fieldwork generalize to the roadmap?</b>           | Roadmap authority is durability and appropriation risk in one clause.            |

## Where Snowflake Sits

Snowflake is a public instance of transfer and exit. Its Forward Deployed Engineers, a team spanning AI engineers and researchers, full-stack and data engineers, industry specialists, and AI product managers, embed directly inside customer teams to co-build AI solutions.<sup>3</sup>

At Sanofi, that team built Concierge for Field, an agent on Snowflake Cortex AI that assembles a sales representative's pre-call plan in a single conversation. It surfaces the highest-priority physician by specialty and prescribing history, reviews past engagement, and emails the finished plan. Work that took hours of manual research now takes seconds. Sanofi is deploying agents across R&D, procurement, IT, HR, and field sales.<sup>3</sup>

Of the four tells, the public record addresses only capability transfer. Snowflake states that its engineers co-build with the customer's own teams, which speaks to capability transfer. Contract structure, IP ownership at exit, willingness to build outside the Snowflake stack, and whether field work generalizes into the product are unaddressed in public material. Snowflake described its deal minimums and offboarding terms to Futurum in a briefing, and those specifics are not independently verified, so this paper applies the four tells only to what is published. Two questions remain open: whether a motion staffed by a product organization scales beyond a handful of concurrent engagements, and whether it remains outcome-priced as it grows.

<sup>3</sup> Snowflake. (2026, June 2). [Sanofi chooses Snowflake to accelerate its AI-powered drug development \[Press release\]](#). Published at Snowflake Summit 26.



# Five Requirements to Set Before Scoping

Five requirements separate an engagement that leaves capability behind from one that leaves an invoice. Table 3 states them as a checklist that a buying committee can walk through in a single meeting.

- **A named business outcome with a number attached, agreed before scoping:** Without the number, success gets decided retroactively by whoever is still in the room.
- **One accountable owner on the business side and one on the technology side, in the same document:** Two owners named separately in two documents is an ownership failure with better paperwork.
- **Evaluation criteria written down before the build starts, including how the system will be judged wrong:** A system with no defined failure mode can only be demonstrated.
- **An explicit exit date tied to deliverable handoffs:** The contract must enumerate the specific assets transferred to the customer on that date, including source code, data pipelines, intellectual property rights, and runbooks.
- **A capability transfer plan naming the internal people who will maintain and extend the work:** A plan that names roles instead of people has not been agreed to by anyone.

Table 3. Checklist of Five Buyer Requirements

| REQUIREMENT |                                                                                                                                                                                | PREVENTS |                                                                                                                                             |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|---------------------------------------------------------------------------------------------------------------------------------------------|
| 01          |  A named business outcome with a number attached, agreed before scoping                     | →        |  No way to prove it worked                             |
| 02          |  One accountable owner per side, business and technology, in the same document              | →        |  Ownership that does not survive the sponsor's leaving |
| 03          |  Evaluation criteria written down before the build starts, including how it is judged wrong | →        |  A moving definition of success                        |
| 04          |  An exit date, and what the customer holds: code, data, IP, and the ability to operate      | →        |  A transfer that never transfers                       |
| 05          |  A transfer plan naming who internally maintains the work                                   | →        |  A capability that leaves with the vendor              |

The constraint moved, and the operating model is what is being purchased. Embedded engineering is one credible answer, and the test that matters at the end is whether the capability stayed behind when the team left. The requirements in Table 3 exist to make that answerable on a date. Watch whether these programs hold their outcome pricing as they scale, or drift back toward the retainer.

# Important Information About This Report

## Authors

### Mitch Ashley

Vice President & Practice Lead, CIO & Technology Buyers,  
and Software Lifecycle Engineering | The Futurum Group

### Brad Shimmin

Vice President & Practice Lead, Data Intelligence,  
Analytics, & Infrastructure | The Futurum Group

## Publisher

### Futurum Research

## Inquiries

Contact us if you would like to discuss this report and  
The Futurum Group will respond promptly.

## Citations

This paper can be cited by accredited press and analysts,  
but must be cited in context, displaying author's name,  
author's title, and "The Futurum Group." Non-press and  
non-analysts must receive prior written permission by The  
Futurum Group for any citations.

## Licensing

This document, including any supporting materials, is  
owned by The Futurum Group. This publication may not be  
reproduced, distributed, or shared in any form without the  
prior written permission of The Futurum Group.

## Disclosures

The Futurum Group provides research, analysis, advising,  
and consulting to many high-tech companies, including those  
mentioned in this paper. No employees at the firm hold any  
equity positions with any companies cited in this document.



## About Snowflake

Snowflake is the platform for the AI era, making it easy for  
enterprises to innovate faster and get more value from data.  
More than 14,500 customers around the globe, including  
hundreds of the world's largest companies, use Snowflake's  
AI Data Cloud to build, use and share data, applications  
and AI. With Snowflake, data and AI are transformative for  
everyone. Learn more at [snowflake.com](https://snowflake.com) (NYSE: SNOW).

# Futurum

## About The Futurum Group

The Futurum Group is an independent research, analysis,  
and advisory firm, focused on digital innovation and  
market-disrupting technologies and trends. Every day our  
analysts, researchers, and advisors help business leaders  
from around the world anticipate tectonic shifts in their  
industries and leverage disruptive innovation to either gain  
or maintain a competitive advantage in their markets.

# Futurum

**CONTACT INFORMATION:** The Futurum Group LLC | [futurumgroup.com](https://futurumgroup.com) | (833) 722-5337