APTLY TECHNOLOGY | INDUSTRY REPORT

Forward-Deployed AI Engineering

Industry Report 2026

How Enterprises Are Moving AI from Pilot to Production



A guide to evaluating, contracting, and governing forward-deployed engineering engagements

Published August 2026

A research report from Aptly Technology

Table of Contents

Table of Contents 2
TL;DR: The Report in Ninety Seconds 4
Executive Summary 5
The Pilot-to-Production Gap 6
What Is Forward-Deployed AI Engineering? 8
FDE Is Not IT Professional Services, and It Is Not Staff Augmentation 9
Why Enterprises Are Adopting the FDE Model 10
Who's Actually Offering FDE Today 12
Where Aptly Sits in This Landscape 12
The FDE Engagement Lifecycle 13
Five Phases, From Discovery to Independence 13
Aptly's Seven Connected Service Tiers 14
Why the Discovery Phase Takes Longer Than Buyers Expect 14
The Weekly Working Increment 15
The Aptly FDE Readiness & Value Framework 15
The Seven Pillars 16
Scoring and Using the Framework 17
How Aptly Applies the Framework in Practice 18
Benefits, Risks & Governance Guardrails 19
The Risk Taxonomy 19
Governance Guardrails to Put in Place Before Day One 21
Building the Guardrails In, Not Bolting Them On 22
The Regulatory Backdrop Just Got a Lot More Concrete 22
Holding Vendors and Partners Accountable 22
Four Contract Terms Procurement Can't Define Without You 22
Classifying Commodity Work vs. Differentiating Work 23
Preventing Dependency That Looks Like Success 24
Coordination Drift in Multi-Partner Engagements 24
Talent, Skills & the New Engineering Persona 25
The Market Has Priced This Scarcity In 26
What an FDE Actually Needs to Know 27
What This Means for the Build-vs-Buy Decision 28
How Aptly Addresses the Talent Constraint 28
Measuring Success: Outcomes and ROI 29

Measure the Outcome, Not the Model 29
The Evidence That Most Organizations Skip This Step 29
Buying Beats Building, By a Wide Margin 30
The Outcomes Aptly Is Built to Deliver 30
The Future of Forward-Deployed AI Engineering 30
The Billion-Dollar Validation Nobody Saw Coming a Year Ago 30
Systems Integrators Are Racing to Match It 31
Two Structural Shifts Already Underway 31
Forward Deployed Engineering Case Studies | Aptly 32
Case Study 01: Enterprise Knowledge Intelligence Platform 33
Case Study 02: Intelligent Customer Support Automation 35
Case Study 03: AI Productionization & Enterprise Deployment 37
Executive Recommendations 39
Short Term (0-6 Months) 40
Mid Term (6-18 Months) 40
Long Term (18+ Months) 40
Where Aptly Fits 41
Who Aptly Is 41
Where Aptly Focuses First 41
The Full Lifecycle, Not the Last Mile 41
How Aptly Addresses the Risks This Report Documents 43
How to Start an Engagement With Aptly 44
Why This Report Exists 44
Frequently Asked Questions 44
Appendix: Research Methodology & Sources 46
Methodology 46
Primary Sources Referenced 46
About Aptly Technology 47

TL;DR: The Report in Ninety Seconds

- Somewhere between 80% and 95% of enterprise AI initiatives never reach production or reach it without moving the P&L. RAND puts the AI project failure rate above 80%, twice the rate of ordinary IT projects. MIT's Project NANDA found 95% of generative AI pilots show zero measurable profit-and-loss impact, against $30-40 billion in enterprise spending. Gartner's 2024 AI Mandates survey (Gartner, G00847066) puts prototype-to-production failure at 59%.
- The fix that the market has converged on is Forward-Deployed Engineering (FDE): embedding vendor engineers directly inside the client's business/operations environment, building against production data and real workflows, rather than handing over a platform and a roadmap.
- This is not a niche delivery model anymore. In the first five months of 2026 alone: Deloitte launched a service line literally named "Forward Deployed Engineering", AWS committed $1 billion to the model, Accenture launched two separate FDE practices with Microsoft and ServiceNow, and both Anthropic and OpenAI formed multi-billion-dollar, private-equity-backed enterprise services ventures built around the same embedded-engineer model, within a day of each other.
- FDE done well compresses deployment timelines and de-risks regulated rollouts. FDE done badly creates the exact dependency it was hired to prevent: Gartner's strategic planning assumption (Gartner, G00844355) is that by 2028, 70% of enterprises running FDE-led agentic AI engagements will be forced to abandon what was built, because of vendor cost and a lack of internal skill to carry it forward.
- Talent is the constraint behind all of it. FDE job postings grew over 700% between April 2025 and April 2026, senior FDE compensation at frontier labs now clears $485,000, and Microsoft's 2026 Work Trend Index finds only 16-19% of AI users qualify as advanced enough to be called "Frontier Professionals" in the first place.
- This report introduces the Aptly FDE Readiness & Value Framework, a scorecard covering the seven conditions that separate an FDE engagement that builds lasting capability from one that builds dependency, plus a practical governance and contracting playbook drawn from current Gartner research, primary regulatory sources, and Aptly's own delivery experience.

STAT	WHY IT MATTERS	SOURCE
80-95%	of enterprise AI initiatives fail to reach production or deliver measurable value	RAND / MIT NANDA / Gartner
70%	of enterprises running FDE-led agentic AI will be forced to abandon it by 2028 without the guardrails in this report	Gartner, G00844355
$1.5B-$10B	committed by Anthropic and OpenAI in May 2026 alone to build dedicated enterprise FDE ventures	Blackstone / TechCrunch
700%+	growth in FDE job postings, April 2025 to April 2026	Perspective AI

STAT	WHY IT MATTERS	SOURCE
$200K-$400K	typical FDE consulting cost per use case, per quarter	Gartner, G00844355
2x	higher success rate for externally sourced AI builds vs. internal-only builds	MIT NANDA

Executive Summary

Every enterprise buyer we work with is somewhere on the same journey right now. They ran the pilot. The demo worked. The board got a good slide. Eighteen months later, the thing that worked in the demo still hasn't shipped to production, and nobody can quite explain why.

This isn't a failure of ambition, and it mostly isn't a failure of the models either. It's a delivery problem. AI platforms are powerful in the lab and stubborn in the enterprise, where the real obstacles are fragmented data, legacy systems, unclear ownership, and governance processes that were never designed for software that acts on its own. Closing that gap requires engineering judgment applied inside the client's actual environment, not a slide deck handed over at the end of a strategy engagement.

The delivery model built for exactly this problem is Forward-Deployed Engineering. Palantir invented the role in the mid-2000s to solve integration problems traditional consulting couldn't touch, and for a decade it stayed mostly inside Palantir. That changed in 2026. Every major AI lab, most of the large systems integrators, and now two of the largest professional services firms in the world have stood up FDE practices of their own, several of them within the same few weeks of each other. The model has gone from a Palantir curiosity to the default way serious AI vendors sell into the enterprise.

That shift is the reason for this report. CXOs evaluating AI vendors in the second half of 2026 are going to be pitched FDE, whether or not they use that name for it, and the vendors doing the pitching have far bigger research and marketing budgets than the buyers trying to evaluate them. This report exists to close that information gap: what FDE actually is, why the model works when it works, what it costs when it goes wrong, and how to structure an engagement so it builds capability your organization owns rather than a dependency you're stuck renting.

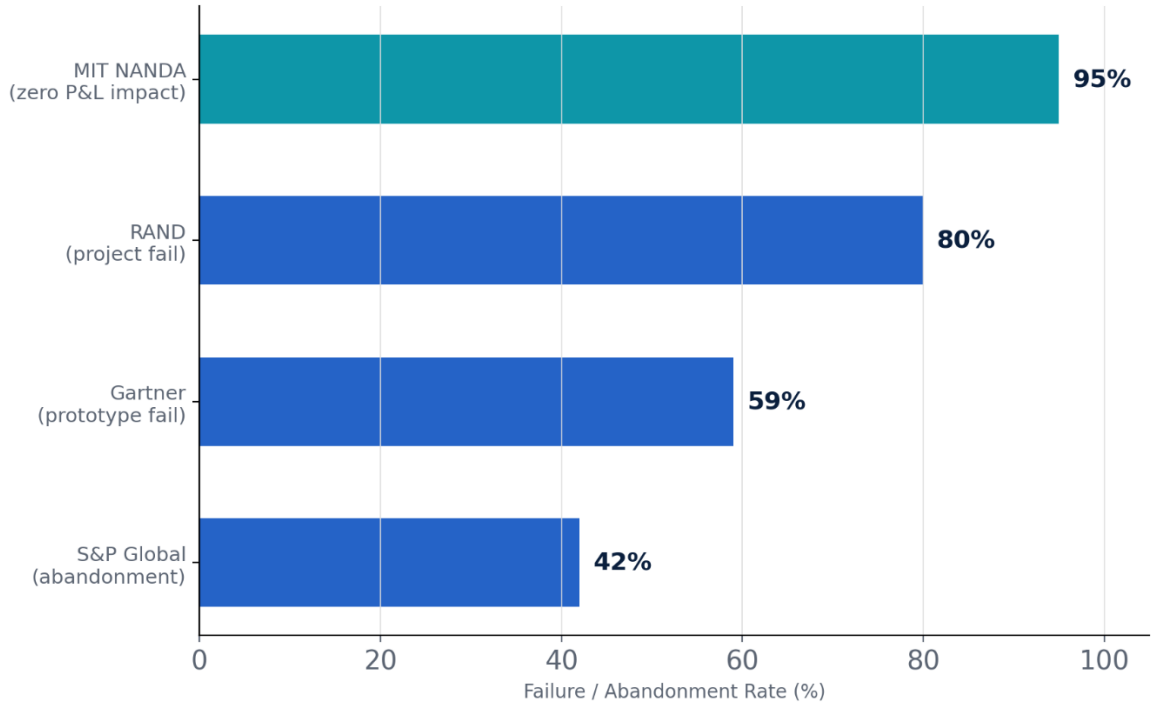
The report also introduces the Aptly FDE Readiness & Value Framework, a diagnostic model that scores an organization's readiness to run a successful FDE engagement across seven dimensions, and closes with a chapter on how Aptly's own Forward-Deployed AI Engineering Services are built around the governance and knowledge-transfer principles this research recommends.

01

The Pilot-to-Production Gap

Every dollar an enterprise spends on an AI pilot is a bet that the pilot eventually becomes something the business runs on. That bet is losing more often than it's winning, and three independent research organizations, using three different methodologies, arrived at roughly the same number.

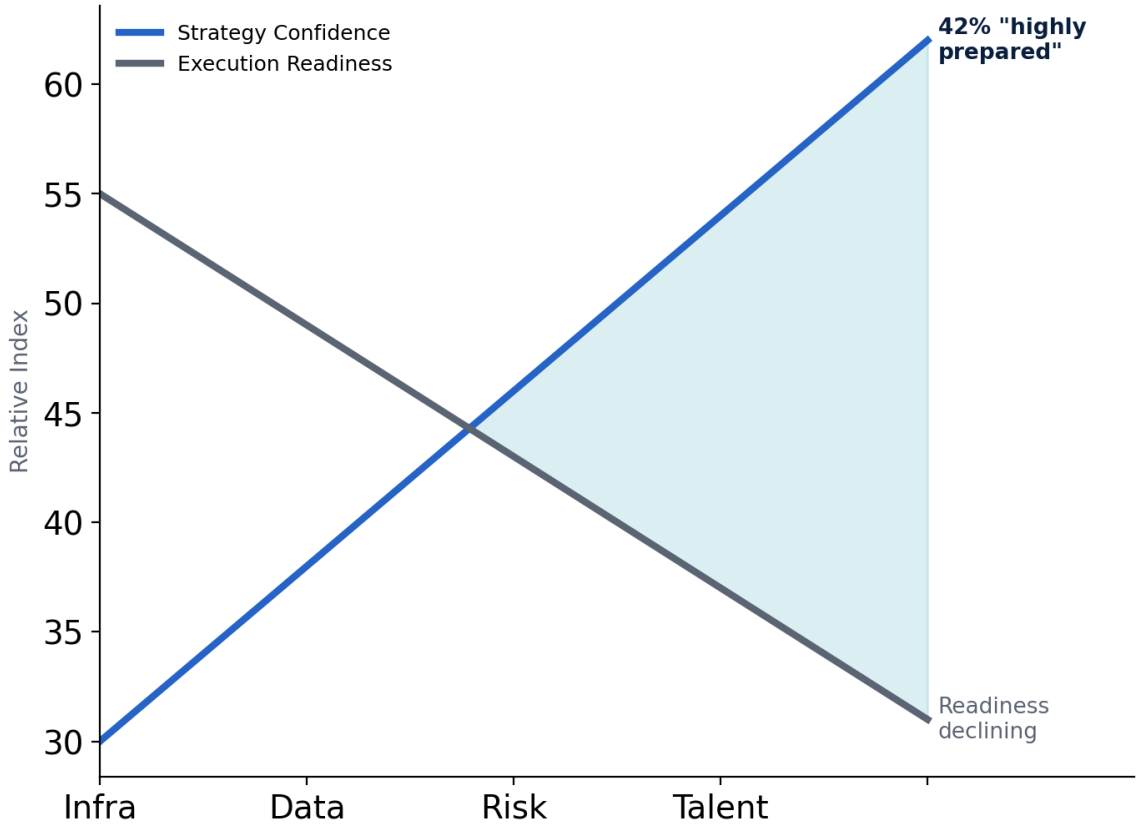
RAND Corporation surveyed 65 data scientists and engineers in 2024 and found that more than 80% of AI projects fail, twice the failure rate of comparable IT projects that don't involve AI at all. MIT's Project NANDA went deeper into the mechanism: a 2025 study covering 300 enterprise AI deployments, 150 leadership interviews, and 350 employee surveys found that 95% of generative AI pilots show zero measurable profit-and-loss impact, despite $30-40 billion in enterprise investment.



Only 5% of organizations studied extract real value at scale. Gartner's own research puts a similar number on the earliest stage of the funnel: the 2024 Gartner AI Mandates for the Enterprise Survey (Gartner, G00847066), which screened for C-level and VP-plus respondents at organizations already running at least one AI use case in production, found that on average 59% of generative AI prototypes never make it into production at all.

None of these numbers describe a technology problem. RAND's respondents pointed to unclear success metrics, weak data foundations, and fading executive sponsorship, not model quality. A separate 2025 survey from S&P Global Market Intelligence, cited in a Forbes Technology Council analysis, found that 42% of companies abandoned most of their AI initiatives that year, up sharply from 17% the year before. The direction of travel is the opposite of what you'd expect from a technology that keeps getting more capable: abandonment is rising, not falling, as the models improve.

Confidence hasn't caught up to this reality, which is its own problem. Deloitte's 2026 State of AI in the Enterprise survey, which polled 3,235 leaders across 24 countries, found that 42% now call their AI strategy "highly prepared." The same group reports feeling less prepared on infrastructure, data, risk, and talent than they did a year earlier. Strategy confidence is climbing faster than execution capability, and that gap is exactly where pilots go to die.



The market has priced this in. Gartner forecasts worldwide AI spending will reach $2.59 trillion in 2026, up 47% year over year. That much capital chasing a 5-to-20% success rate is precisely the condition that makes a new delivery model attractive to enterprise buyers and lucrative for the vendors who can credibly close the gap. The rest of this report is about the model the market has converged on to do that, what it gets right, where it goes wrong, and how to run it so it works for you rather than on you.

Every one of these numbers points to the same unresolved question: if 59% to 95% of AI initiatives never reach production using the standard delivery models enterprises already have, and abandonment is rising rather than falling as the underlying models improve, something structurally different has to happen between the pilot and the production system for the odds to change. That “something” in question here is what we call Forward Deployed Engineering. The rest of this report picks up exactly where this chapter leaves off: what FDE is, why it emerged as the market's answer to the gap just described, and how to run it well.

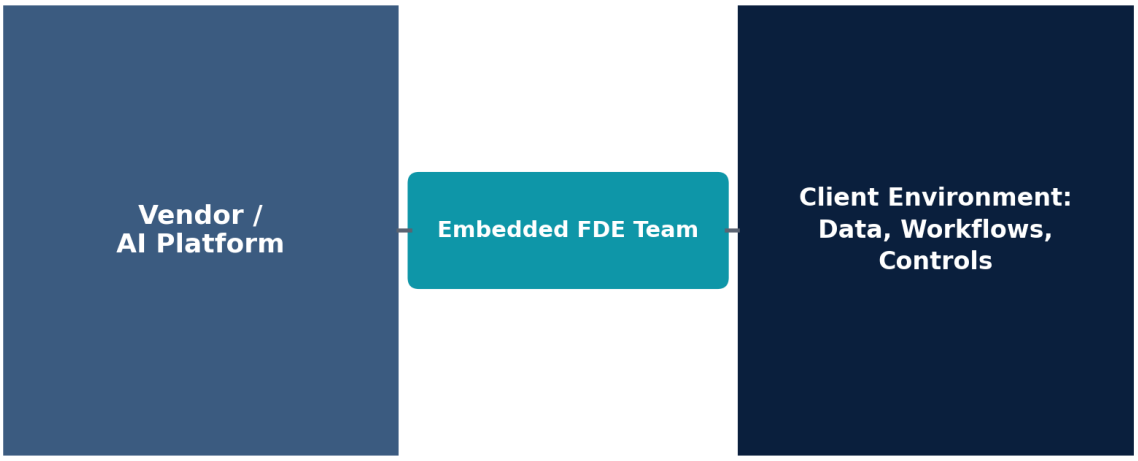
02

What Is Forward-Deployed AI Engineering?

Most enterprise AI initiatives never get past the pilot stage in the first place: RAND Corporation research found that more than 80% of AI projects fail to reach production, a rate twice that of comparable non-AI IT projects, which is precisely the gap that made a delivery model like Forward-Deployed Engineering necessary.

The term "Forward-Deployed Engineering" gets used loosely enough right now that it's worth being precise before going further, because the substance of the model matters more than the label.

Gartner's working definition (Gartner, G00847066) is a delivery model where engineers from a technology vendor or IT services provider are embedded within the client's teams, often on-site. Their job is to accelerate the journey from AI pilot to production by tailoring solutions to the client's own data, workflows, controls, and architecture. The goal isn't just closing the "last mile" of an AI project. It's operationalizing the project entirely: value, integration, and workflow orchestration, not just a working prototype.



Palantir built the model to solve a problem nobody else had solved. The company's forward-deployed engineering approach traces back to its earliest intelligence-agency clients in the mid-2000s: organizations that couldn't fully articulate their own requirements and, for security reasons, often couldn't hand over their data for a normal vendor to study offline. The only way to build something useful was to put engineers inside the building, working directly against production data and alongside the analysts who'd use the result. By 2016, Palantir employed more forward-deployed engineers than traditional software engineers, and the model became the company's defining go-to-market motion.

Concrete examples make the shift easier to picture. CIO.com's 2026 survey of the emerging discipline describes forward-deployed engineers building a claims-triage agent embedded directly in an insurer's adjuster workflow, a supply chain copilot wired into a manufacturer's existing ERP rather than a standalone dashboard, and a compliance-monitoring agent built inside a bank's own case-management system so examiners never have to leave the tool they already use. In each case, the engineer's job wasn't to hand over a model. It was to make the AI disappear into a workflow the business already ran, which is the pattern this report keeps returning to.

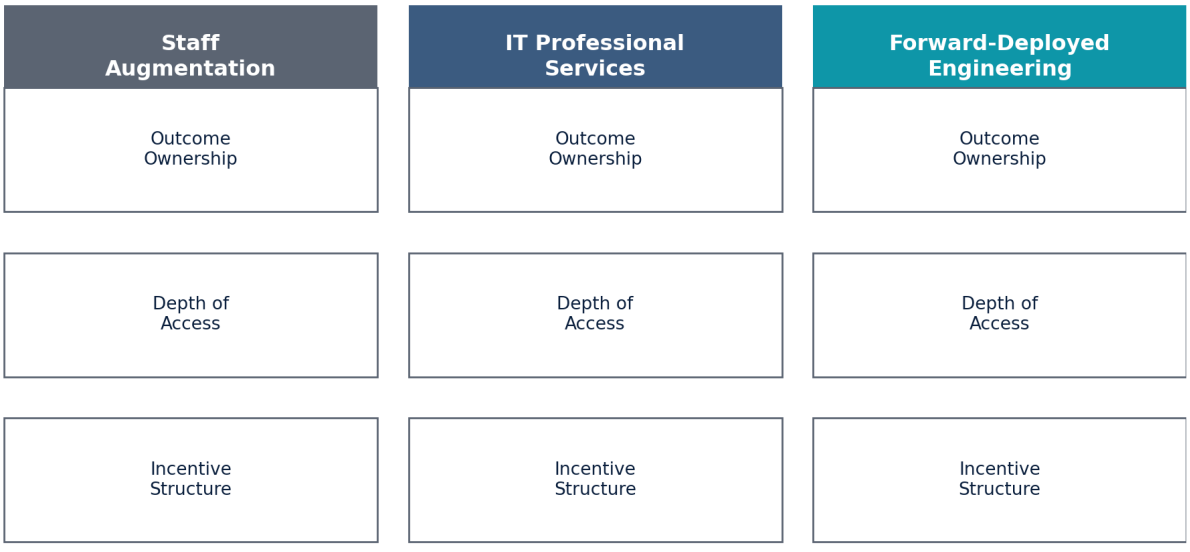
The idea broke out of Palantir's orbit in 2025, largely through Andreessen Horowitz's "Trading Margin for Moat" essay, which reframed FDE as a deliberate business strategy rather than an operational quirk. The argument: ServiceNow and Workday went public with gross margins around 63% and 54% precisely because they were willing to do the unglamorous implementation work that pure product-led-growth companies refuse to touch, and that willingness became a durable competitive moat. a16z's bet was that AI companies willing to do the same thing would win enterprise accounts no API-only competitor could close. The Pragmatic Engineer's explainer on the role, published around the same time, captured why the hiring market took off immediately after: FDE sits at the intersection of three skill sets that rarely live in one person, which makes it both extremely valuable and extremely hard to hire for at scale.

FDE Is Not IT Professional Services, and It Is Not Staff Augmentation

The distinction matters because procurement teams default to treating any embedded vendor engineer as one or the other, and both comparisons undersell (or mischaracterize) what a well-run FDE engagement actually looks like.

Traditional IT professional services operate with a service-provider mindset: scope the work, staff the project, bill the hours, hand over the deliverable. Gartner (Gartner, G00847066) draws the FDE distinction on three points. First, outcome ownership: a true FDE is measured on actual business impact and production stability, not delivery of a milestone. Second, depth of access: FDEs typically have full-stack access to both the client's environment and the vendor's own codebase, which lets them build and iterate in real time rather than filing change requests. Third, incentive structure: success is measured by long-term platform adoption and value delivery, not billable hours or a fixed project close date.

Staff augmentation is a simpler comparison to dismiss. An augmented staff member works under the client's direction, effectively becoming a temporary employee. A forward-deployed team brings a defined delivery model, multidisciplinary expertise (engineering, architecture, and something close to management consulting), and accountability for an agreed outcome, all while staying embedded with the client and focused on one specific business problem rather than general capacity.



A more accurate way to describe the relationship is team expansion rather than staffing or professional services: the FDE joins as a temporary extension of the client's own engineering organization, sharing its goals, its data access, and its accountability for the outcome, rather than sitting apart from it as an external contractor executing a fixed scope.

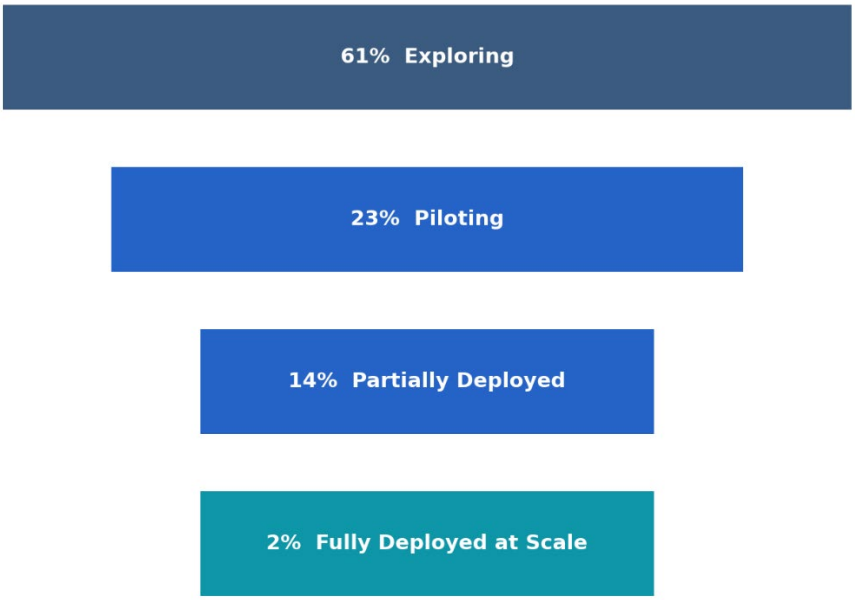
03

Why Enterprises Are Adopting the FDE Model

Put the last two chapters together and the adoption logic writes itself. Enterprises are sitting on a pilot-to-production failure rate north of 80%, a governance and infrastructure readiness gap that's widening even as strategy confidence rises, and a technical integration problem, connecting AI to fragmented data, legacy systems, and existing workflows, that's consistently harder than building the model in the first place.

FDE exists because the standard alternatives, generic advisory consulting on one end and a self-service platform on the other, don't touch any of that. An advisory engagement produces a roadmap; a self-service platform produces an API key. Neither one ships production code against a client's actual data, which is precisely the step where 59% to 95% of AI initiatives stall out.

The demand-side data confirms the gap is real and getting more urgent, not less. Deloitte's 2026 research shows agentic AI usage climbing from 23% today to an expected 74% within two years. Capgemini's Rise of Agentic AI survey shows why that curve won't be smooth: 61% of organizations are still just exploring agentic AI, 23% have launched pilots, only 14% have deployed agents even partially at scale, and a mere 2% have reached full enterprise deployment. That collapse from 61% down to 2% is the entire adoption argument for FDE compressed into one statistic. The market isn't short on ambition. It's stuck exactly where the pilot-to-production gap predicts it would get stuck, and FDE is the model built specifically to move organizations through that choke point.



Deloitte: agentic AI usage projected to reach 74% within two years

Gartner's own framing (Gartner, G00847066) of the benefits is worth stating plainly, because it maps directly onto the failure modes in Chapter 1:

- Faster time-to-production: FDEs compress the "stuck in POC forever" cycle by building directly inside the client's own environment and constraints, rather than a generic sandbox.
- Higher ROI probability: a good FDE translates business intent into a working system, covering data readiness, evaluation, enterprise guardrails, and observability, and keeps iterating until the outcome actually lands.
- A closed feedback loop: enterprise use cases become product feedback for the vendor, which means the platform itself improves faster in response to real deployment friction, not hypothetical requirements.
- De-risking in regulated settings: enterprises in regulated industries need a stronger emphasis on testing, monitoring, and deployment discipline than a generic rollout provides, and FDEs combined with internal governance teams are built to deliver exactly that.

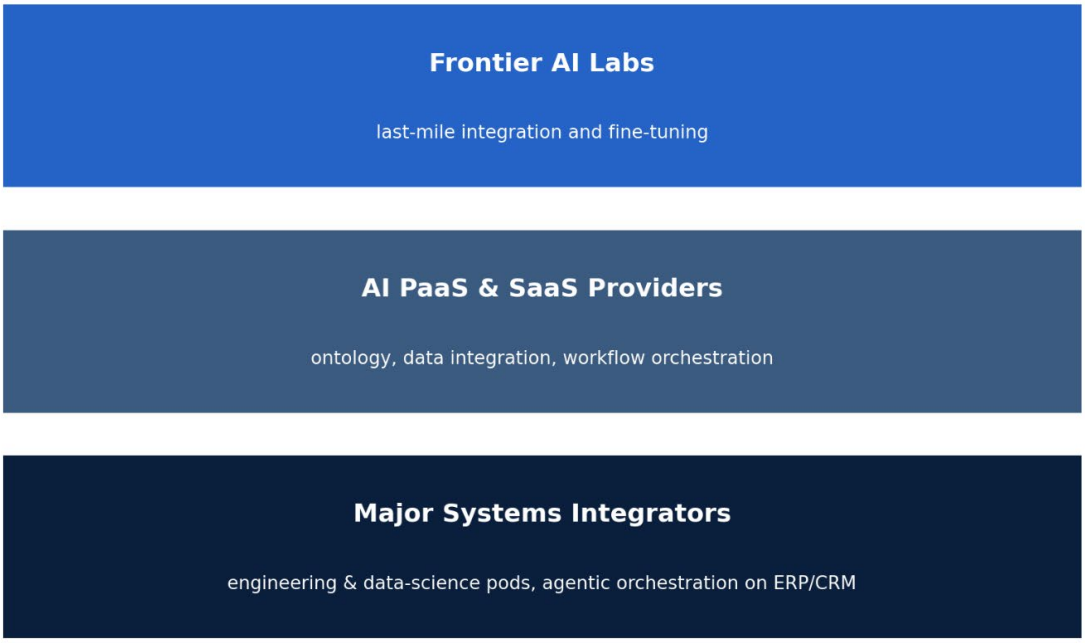


The proof points are no longer hypothetical, either. OpenAI's own FDE case documentation cites a 98% adoption rate on its Morgan Stanley deployment, alongside efficiency gains of 20-50% across finance, manufacturing, and telecom clients, including T-Mobile and Klarna. Numbers like that are what a CFO actually forwards to the board, and they're the reason FDE has moved from an interesting staffing model to the default enterprise AI sales motion in under two years.

Who's Actually Offering FDE Today

Three distinct categories of vendors now offer some form of FDE, and Gartner's research (Gartner, G00847066) maps them cleanly:

- Frontier AI labs, including Anthropic, Cohere, Google, Mistral AI, and OpenAI, focused mostly on last-mile integration: fine-tuning their models against a client's proprietary data and building custom agentic workflows on top.
- AI PaaS and SaaS providers, including Adobe, AWS, Databricks, Microsoft, Palantir, and Salesforce, with a focus that varies by vendor but usually centers on ontology, data integration, and agent-driven workflow orchestration.
- Major systems integrators, led by Accenture and Deloitte, who structure their offering as "pods" of engineers, data scientists, and industry experts, typically focused on agentic orchestration layered onto critical ERP and CRM systems and vertical-specific AI.



Most AI startups and large vendors will offer some version of FDE within the next year or two, as the alternative, a slide deck and a roadmap, keeps failing to move enterprises past the numbers in Chapter 1.

Where Aptly Sits in This Landscape

Aptly Technology occupies a distinct position in the vendor map above: an independent, technology-agnostic Forward-Deployed AI Engineering provider that is neither a frontier model owner defending a single platform, nor a large systems integrator layering FDE onto a legacy consulting practice. Aptly's forward-deployed teams

originate from the AI research and engineering group behind AptlyStar.ai, the company's own generative AI agent platform, which means engineers embedding inside a client's environment bring direct product-building experience rather than a generic advisory playbook applied to AI for the first time.

That independence matters because of a pattern this report keeps surfacing: FDE engagements sold by a frontier lab or a platform vendor tend to steer, subtly or otherwise, toward that vendor's own stack. Aptly recommends models, platforms, and architectures based on client requirements alone, working across commercial and open-source models and infrastructure including Microsoft Azure, Azure AI Foundry, AWS, and Kubernetes, which directly limits the vendor lock-in exposure documented later in Chapter 6.

04

The FDE Engagement Lifecycle

Every buyer who's been sold on FDE eventually asks the same question: what does this actually look like, week to week, once the contract is signed? The answer matters more than the sales pitch, because a well-structured lifecycle is what separates an engagement that builds lasting capability from one that quietly becomes permanent, expensive staff augmentation.

Five Phases, From Discovery to Independence

Aptly's internal delivery methodology structures every FDE engagement around five phases, a structure that mirrors what's become the industry standard for embedded engineering work:

- **Phase 1: Technical Discovery & Audit:** Before any code gets written, the FDE team audits the client's data environment, obtaining read-only access to source systems, mapping schemas, and identifying structural quality issues. The critical output of this phase isn't a document; it's alignment on the Minimum Viable Value, or MVV: the smallest scope of integration that proves concrete economic value. That's a deliberately different question from the Minimum Viable Product question of what the smallest set of features is that the team can ship. The MVV question is what the smallest integration is that proves the initiative is worth doing at all, and getting that scope wrong is the single most common way engagements burn budget without producing anything a business unit will actually adopt.
- **Phase 2: Landing Zone Setup & Security Hardening:** The FDE team provisions the secure cloud infrastructure the platform will run on, working directly alongside the client's network and security operations teams to establish private clusters, secure network perimeters, and infrastructure defined as code for reproducibility. This phase is where a disorganized engagement typically gets stuck: waiting on access approvals and provisioning requests that have nothing to do with the AI itself.
- **Phase 3: Ingestion & Integration:** With secure infrastructure in place, the FDE builds the client-specific integration layer: data pipelines that clean and deduplicate source data, identity configurations that bypass fragile username-and-password credentials in favor of proper workload identity, and custom orchestration connecting the platform to legacy client APIs.
- **Phase 4: System Evaluation & User Acceptance Testing:** Once data is flowing, the system gets evaluated rigorously against defined accuracy thresholds, while early business users run it through real operational scenarios rather than synthetic test cases.
- **Phase 5: Operations Handover:** The final phase is the one most vendors underinvest in: packaging the system, writing detailed operations runbooks, configuring monitoring and alerting, and training the client's own IT staff to run the platform without the FDE in the room. An FDE who never leaves hasn't succeeded. They've just become expensive.



Aptly's Seven Connected Service Tiers

Aptly structures the five-phase lifecycle above into seven connected service tiers that a client can enter or exit at the appropriate point, rather than being sold a single, undifferentiated block of embedded headcount. The tiers are Discovery (determine what to build), Prototype (build a working use case), Production (move the solution into live, secure use), AI Agents (develop and deploy copilots and automation), Platform (configure and deploy GPU, models, and infrastructure), Governance (ongoing monitoring for safety and compliance), and Managed Ops (manage, maintain, optimize, and control costs).



This tiering matters directly to the readiness conversation in Chapter 5: an organization that only needs a Discovery and Prototype engagement to validate an MVV should not be sold a full Managed Ops commitment, and a regulated enterprise that needs Governance built in from day one should not have to bolt it on after Production. Letting the client match engagement scope to actual need, rather than defaulting to the largest and most open-ended option, is one of the more concrete ways Aptly's model works against the dependency risk this report spends Chapter 6 describing.

Why the Discovery Phase Takes Longer Than Buyers Expect

The phase breakdown above understates just how much of an engagement's real time goes into discovery rather than building. Gartner's vendor-accountability research (Gartner, G00844355) quantifies it precisely: discovery, requirements-gathering, and stakeholder communication typically consume 40% to 60% of a well-run engagement, quality evaluation consumes another 15% to 20%, and pure technical implementation, the part most buyers assume is the whole job, is closer to 15%.



That ratio isn't inefficiency. It reflects how FDEs actually produce useful results: by observing how work really happens, not by reading a requirements document. Gartner's guidance is specific here, FDEs need to be placed alongside the people who do the daily work, claims adjusters, loan officers, supply chain analysts, because that's

where the undocumented workarounds and tribal knowledge live that no system of record captures. A single workflow can require hundreds of questions before an FDE can write a meaningful specification. Route the FDE through IT as their primary working relationship instead, and discovery quietly stalls, even though IT remains essential for provisioning access and enforcing security controls.

There's a human dimension to this that's easy to overlook from a systems-architecture view. An engineer mapping a claims adjuster's workflow in granular detail is, from that adjuster's perspective, potentially capturing the exact expertise their job depends on, to hand it to an AI agent. An employee who reaches that conclusion will cooperate visibly and withhold selectively, sharing the official process rather than the real one, and an agent built on the sanitized version will fail on precisely the edge cases the expert chose not to mention.

Resolving this is explicitly the engagement sponsor's job, not the FDE's: before kickoff, someone with executive standing needs to answer, clearly, what happens to the people whose workflows get automated and what role they play in shaping the outcome. Position domain experts as co-designers rather than subjects, and they become the people who share the workarounds that make the system actually work.

The Weekly Working Increment

The single mechanism that keeps an FDE engagement honest is a disciplined weekly cadence. Gartner's guidance (Gartner, G00844355) is unambiguous: a well-run weekly increment includes code running against production data (not a staged environment), a domain expert testing that increment and providing direct feedback, and the requirements document being updated to reflect what was actually learned. If the FDE is building the wrong thing, this cadence surfaces it at week two, not month four, and it's what distinguishes a genuine FDE engagement from consulting wearing an FDE label. If weeks pass without a working increment, the engagement has drifted into advisory work, regardless of what the invoice calls it.

One counterintuitive piece of this discipline is worth calling out directly: when an increment misses its target, the right response is not to extend the timeline quietly. It's to reset the requirements and redirect the next sprint. Requirements update mid-engagement isn't a failure signal. It's evidence that the engagement is grounded in reality rather than a fixed plan written before anyone understood the problem. Gartner goes further still: a "no-go" finding, discovering during discovery that a project isn't viable as scoped, is itself a successful engagement outcome, arguably the most cost-efficient result an FDE can deliver, because it saves the far larger cost of building the wrong thing at full engagement price.

05

The Aptly FDE Readiness & Value Framework

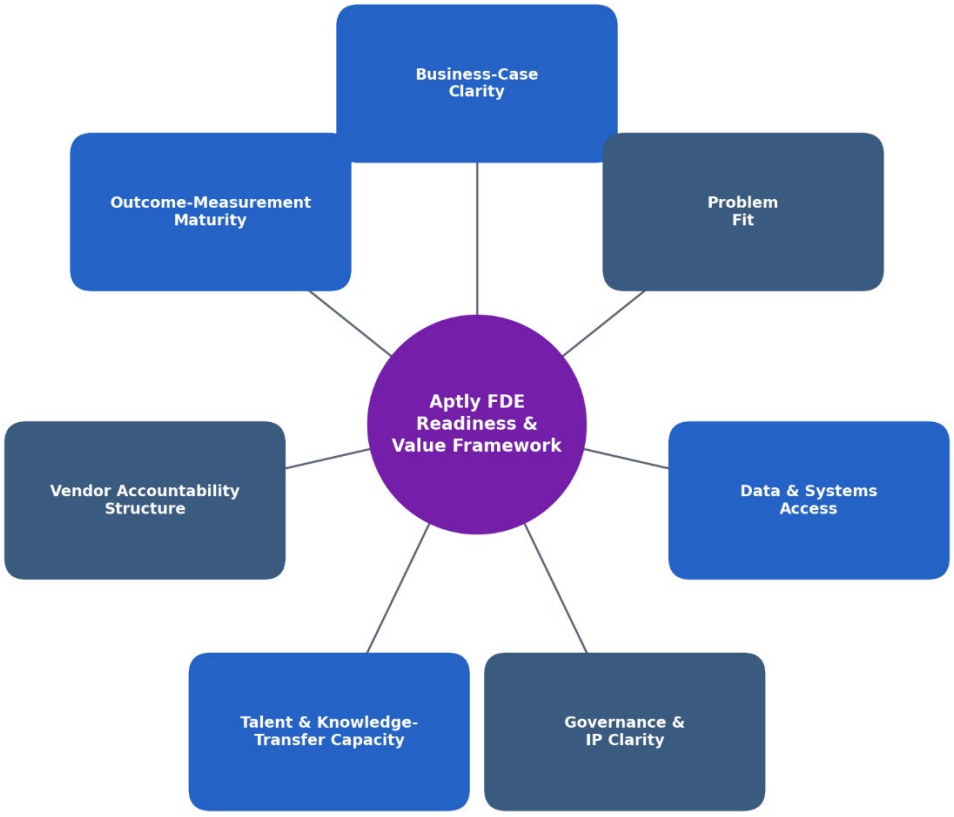
Most organizations evaluating an FDE engagement focus their diligence on the vendor: their technical depth, their reference clients, their pricing. That's necessary but insufficient. The research in this report points to a harder truth: whether an FDE engagement succeeds depends at least as much on conditions the client controls as on which vendor gets hired. Gartner's own framing (Gartner, G00844355) makes this explicit: the vendor FDE

determines whether the solution is technically sound, but the engineering leader creates the conditions that determine whether it reaches production at all.

The Aptly FDE Readiness & Value Framework scores an organization across seven dimensions before an engagement begins, giving CXOs a structured way to identify which conditions need to be in place before signing a contract, not after.

The Seven Pillars

1. **Business-Case Clarity:** Is there a named executive sponsor with the standing to accelerate provisioning and remove internal blockers? Without one, the engagement stalls in access requests before productive discovery ever begins.
2. **Problem Fit:** Does the target problem require new workflows or legacy-system integration that off-the-shelf tools genuinely can't deliver, and is it bottlenecked by operational complexity rather than by missing data or unclear ownership? FDE resources aimed at problems a standard AI deployment could already solve produce a working system at full engagement cost that never needed one.
3. **Data & Systems Access:** Has the organization estimated the full integration burden, security reviews, data-sharing agreements, and access approvals, before committing the engagement? If that burden will consume most of the available window, the problem isn't viable as currently scoped, and that should disqualify it before kickoff, not become a mid-engagement risk to manage.
4. **Governance & IP Clarity:** Is there a clear, upfront classification of which artifacts are commodity capability (that should flow into the vendor's platform for everyone's benefit) versus differentiating capability (that represents this organization's competitive advantage and needs explicit IP protection)?



5. **Talent & Knowledge-Transfer Capacity:** Is there an internal engineer assigned to work alongside the FDE from the first sprint, not just during a final handoff, so the organization is building capability in parallel with the build itself?

6. **Vendor Accountability Structure:** Does the contract contain the four terms procurement typically can't define without technical input: IP classification, deliverable definitions, handoff criteria, and provisioning SLAs? Chapter 7 covers this in full.

7. **Outcome-Measurement Maturity:** Are success metrics defined at the business-outcome level, not the model-performance level, before the engagement starts? "The agent's accuracy score" is not a business outcome. "Discount adjustments executed automatically" or "time from bug report to deployed fix" are.

Scoring and Using the Framework

Score each pillar honestly on a five-level scale, from Not Ready through Production-Ready, and resist the urge to average the results. A single weak pillar, most often governance clarity or talent capacity, caps what the other six can deliver, in the same way a single missing safety guardrail caps the value of an otherwise well-engineered system.



Score your organization across all seven pillars — a single weak pillar caps the rest

The pattern is worth naming directly, because it echoes a gap this report keeps surfacing from a different angle: Deloitte finds that only 21% of enterprises have a mature governance model for autonomous AI agents, even as agentic usage is on track to triple within two years. High ambition and low governance maturity is not a stable combination, and it's the exact condition an FDE engagement inherits if the client hasn't closed the gap before the engineers show up.

How Aptly Applies the Framework in Practice

Aptly runs this same seven-pillar assessment as the opening step of nearly every new engagement, before a statement of work is drafted, because the research behind this framework is drawn from Aptly's own delivery experience as much as from Gartner's published guidance. A prospective client whose Business-Case Clarity or Data & Systems Access pillar scores “Not Ready” is told that directly, in the discovery conversation, rather than being sold an engagement scoped to fail. That willingness to recommend a smaller Discovery or Prototype tier, or in some cases to recommend waiting, is a direct byproduct of Aptly's outcome-based engagement model: the company is structured to be measured on production outcomes and customer ownership, not on billable hours logged against an open-ended contract.

06

Benefits, Risks & Governance Guardrails

FDE is not a free lunch, and the vendors selling it have limited incentives to lead with that fact. The risk side of this model is where a CXO's diligence earns its keep, and it's also where a well-governed vendor differentiates itself from one selling speed without a plan for what happens after the engagement ends.

The Risk Taxonomy

Gartner's risk analysis (Gartner, G00847066) groups the exposure into four categories, and it's worth sitting with each one rather than treating them as a generic disclaimer.

Vendor dependency. Faster AI deployment can come at the cost of permanent reliance on vendor talent for every subsequent change and iteration, creating substantial lock-in and real barriers to exit.

Security and data exposure. Integrating external engineers into core processes and data pipelines creates specific new problems: privileged access to production environments and sensitive datasets, compliance drift when an FDE circumvents internal procurement or security protocols to move fast, and IP contamination when it's unclear who owns the custom code or artifacts an FDE writes.

Talent atrophy. This is the risk Gartner calls out as the most significant long-term exposure. When vendor engineers handle the hardest last-mile integrations, internal teams can end up as operators of a black box rather than architects of their own systems, critical system knowledge concentrates with the vendor rather than the client, and high-caliber internal talent may leave if they feel like they're shadowing vendor engineers rather than building anything of their own.

Shadow architecture and technical debt. Fast-moving FDE-generated artifacts can bypass enterprise standards entirely unless the client enforces reference architectures, rigorous evaluation, and clear platform usage guidelines from day one.



Gartner's own strategic planning assumption puts a number on how often this goes wrong at scale: by 2028, 70% of enterprises (Gartner, G00844355) running FDE-led agentic AI engagements will be forced to abandon what they built, driven by high vendor costs and a lack of internal skill to evolve the solution independently. That's not a marginal failure rate. It's a majority outcome for engagements that don't build the guardrails this chapter describes from the start.

The underlying cost structure makes the stakes concrete: FDE consulting engagements typically run $200,000 to $400,000 per quarter, per use case (Gartner, G00844355), before platform and integration costs, which means a mismanaged engagement doesn't produce a partial result. It produces the wrong outcome, at full price, with the vendor left as the only party capable of maintaining what was built.

70%

of FDE-led agentic AI engagements
abandoned by 2028

200K – **400K per use case, per quarter**

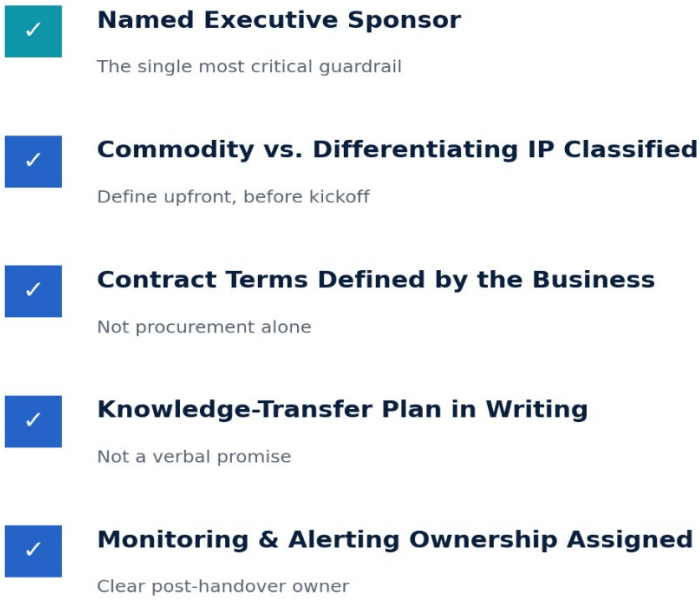
There are three specific structural failure modes worth naming, because Gartner finds (Gartner, G00844355) they consume an engagement's entire value before the technical work even has a chance to fail on its own merits: misaligned problem scope (FDE resources pointed at a problem off-the-shelf tools could already solve),

ungoverned access (provisioning delays that consume the engagement window before productive discovery even starts), and unmanaged handoff (a solution nobody but the vendor can operate once the FDE leaves). All three are organizational failures, not technical ones, and all three are preventable before the FDE ever arrives on site.

Governance Guardrails to Put in Place Before Day One

Gartner's recommended controls translate directly into a pre-engagement checklist:

- **Context-engineering guardrails:** Define exactly what data an FDE can access, how exceptions get handled and documented, and what the audit trail looks like.
- **Shadow-AI prevention:** Require FDEs to build inside the organization's sanctioned AI orchestration platform, so every agent they create is discoverable, auditable, and usable by others after the engagement ends.
- **Co-ownership of the ontology layer:** Insist that data models and semantic layers are built in a format the internal team can manage after the FDE leaves, not buried in a proprietary structure only the vendor understands.
- **IP structured upfront:** A useful rule of thumb: at least 70% of the code or integration patterns an FDE develops should be integrated back into the client's own internal developer platform, not left as a one-off script that only the original engineer can maintain.
- **An exit plan defined on day one, not negotiated at the end:** The most successful engagements set exit criteria before work begins, and those criteria need to be both value-bound and time-bound: a target number of internal team members certified through knowledge transfer, full transfer of ownership by a defined date, and a demonstrated time-to-value for the specific use case. Without this, FDEs quietly become permanent staff augmentation, and nobody notices until the invoices have been arriving for two years.



Building the Guardrails In, Not Bolting Them On

Every guardrail in the checklist above is easier to enforce on a vendor that was built around them from the outset than on one retrofitting them under contractual pressure. Aptly holds ISO/IEC 42001 certification for AI governance, alongside ISO 27001 and SOC 2 certifications covering information security and operational controls, which means context-engineering guardrails, access logging, and audit trails are standard practice on an Aptly engagement rather than a negotiated add-on. On the IP question specifically, Aptly's engagement contracts classify commodity versus differentiating work upfront, consistent with the 70% internal-platform-integration benchmark this chapter recommends, and the exit plan, covering knowledge-transfer milestones and a defined handoff date, is negotiated before the engagement starts rather than at its conclusion.

The Regulatory Backdrop Just Got a Lot More Concrete

Governance guardrails aren't only a best-practice recommendation anymore; they're becoming a compliance requirement, and the pace of that shift over the past year changes how urgently this chapter should be read. Singapore's Infocomm Media Development Authority launched the Model AI Governance Framework for Agentic AI in January 2026, the world's first governance framework built specifically for autonomous agents, structured around assessing and bounding risk upfront, ensuring meaningful human accountability, and implementing technical controls across the agent's full lifecycle.

The following month, NIST's Center for AI Standards and Innovation launched its AI Agent Standards Initiative, with an explicit research focus on agent identity: the problem of reliably distinguishing an AI agent from a human user inside enterprise systems, and ensuring agents can only act within delegated scopes of authority. The EU AI Act's existing human-oversight and robustness requirements already apply to autonomous systems operating in high-risk domains.

None of this is abstract for an organization already running ISO/IEC 42001-certified AI governance. It's a signal that the controls this chapter recommends aren't just good practice today; they're converging with where mandatory compliance is heading over the next eighteen months, and organizations building these guardrails into their FDE engagements now won't be scrambling to retrofit them later.

07

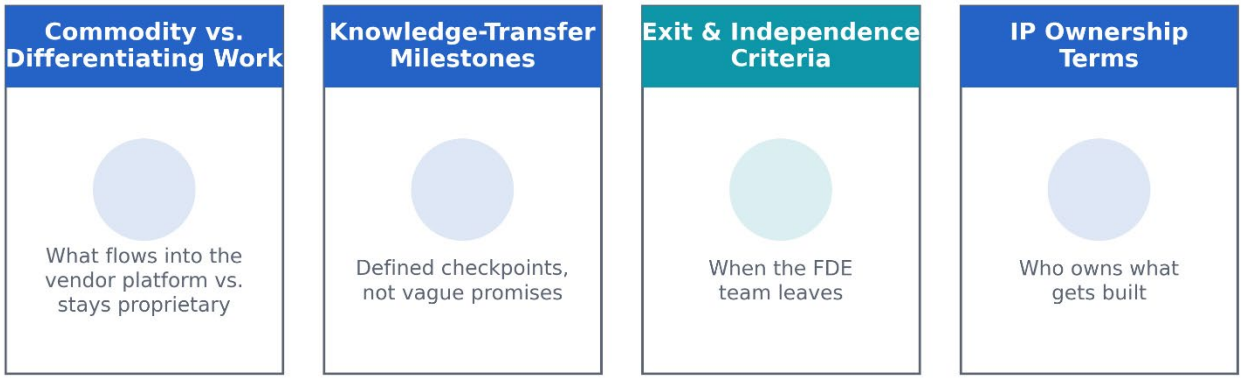
Holding Vendors and Partners Accountable

If there's one chapter in this report worth printing and bringing into a vendor negotiation, it's this one. Everything in Chapter 6 describes what can go wrong. This chapter is the contractual and operational discipline that prevents it.

Four Contract Terms Procurement Can't Define Without You

Before an engagement starts, verify the contract contains four specific terms, and escalate to the engagement sponsor if any are missing (Gartner, G00844355). A contract that reaches signature without these terms gives the client nothing to enforce once the engagement is underway.

- **Intellectual property classification** that distinguishes commodity work from differentiating work, in writing, before the engagement begins.
- **Deliverable definitions** that name specifically what the FDE will produce and what form it will take, not a vague reference to "production-ready AI capability."
- **Handoff criteria** that specify, concretely, what internal independence looks like before disengagement, not a calendar date picked in advance of understanding the work.
- **Provisioning SLAs** that set response times for access requests, so the engagement doesn't quietly lose weeks to internal IT ticket queues.



Aptly builds all four of these terms into its statements of work as a matter of standard practice, not as a response to buyer pushback. IP classification, deliverable definitions, handoff criteria, and provisioning SLAs are agreed upon during the Discovery tier, before Prototype work begins, which means a client evaluating an Aptly engagement can hold the proposal against this exact checklist during procurement rather than discovering gaps once the engagement is already underway.

Classifying Commodity Work vs. Differentiating Work

At every milestone review, before asking a vendor to generalize custom work, classify what was delivered. Commodity capabilities, standard integrations, common authentication patterns, generic AI workflows, should flow into the vendor's platform, and the vendor should be required to name the specific time and release when that will happen. Differentiating capabilities, proprietary workflows, unique operational logic, industry-specific edge cases, represent real competitive advantage. Invoke the IP terms in the contract before the engagement closes, and verify independently that the vendor's own platform roadmap doesn't already include them (Gartner, G00844355).

There's a useful early-warning signal buried in this process: when generalization signals go flat, meaning an FDE can no longer name a product counterpart on the vendor's side for the work they're doing, that's evidence the FDE is building custom work in isolation from the vendor's own product team. The response isn't to wait and see. It's to request a working session with the vendor's product manager, not the FDE, and ask them to trace exactly how sprint deliverables are queued for platform integration. If the vendor can't demonstrate a direct line from the client's deployment to their own roadmap, escalate before the engagement window closes.

Preventing Dependency That Looks Like Success

The hardest failure mode to catch is the one that looks like progress. Require the FDE to work alongside at least one internal engineer for the entire engagement, not only during a final handoff phase. By the last sprint, that internal engineer should be able to run the solution, diagnose failures, and deploy changes without the FDE in the room. Require a production runbook and architecture documentation as contracted deliverables, not optional artifacts a vendor produces only if time allows.

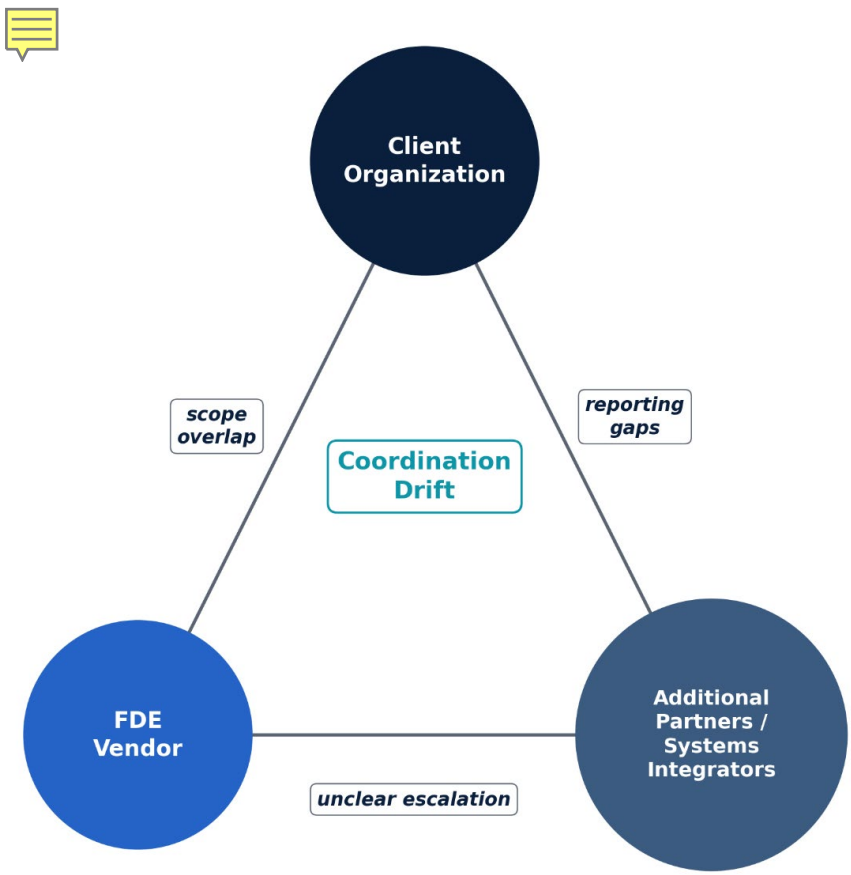
The clearest single diagnostic Gartner offers for this entire chapter is worth quoting directly: if the only person who understands the system at disengagement is the FDE, the engagement has produced a dependency, not a capability (Gartner, G00844355). Flat FDE effort across successive deployments, meaning the vendor's time investment doesn't shrink as use cases mature, is the specific signal that this has already happened.

When that pattern shows up, audit with the engagement sponsor exactly which deliverables remain exclusively FDE-owned; if the FDE is still required for routine operations rather than new capability development, the engagement has drifted from adoption into dependency, and it's worth naming that drift out loud before it becomes the new normal.

Coordination Drift in Multi-Partner Engagements

Larger engagements often involve three parties at once: the customer, a consulting or systems-integration partner, and the vendor's own FDE team. Without named owners inside each party, accountability quietly falls to organizations rather than individuals, and the most common failure mode follows directly from that ambiguity: the FDE builds a capability the consulting partner was scoped to integrate, or the partner documents a workflow the FDE is still actively building against. Both parties are working. Neither is working on the right thing, and by the time the mismatch surfaces in a joint review, the engagement has already absorbed weeks of rework (Gartner, G00844355).

The fix is a single-page ownership map, produced before the engagement begins, specifying exactly which partner owns each deliverable, down to a named individual rather than a firm. On the customer side, name an adoption owner with real credibility in the business unit where the solution will operate, not a project manager in IT, and give that person explicit responsibility for stakeholder alignment and operational rollout. Weekly joint reviews across all three parties aren't optional overhead in this structure; they're the primary mechanism for catching drift before it becomes rework, and a skipped review should be treated as a risk event worth documenting, not a scheduling inconvenience.



This chapter matters more, not less, as the vendor landscape gets more crowded. Deloitte has stood up its own FDE practice. Accenture now runs two, one with Microsoft and one with ServiceNow. AWS has committed nine figures to embedding its own engineers with enterprise customers. A bigger vendor name on the contract is not a substitute for the accountability structure described above; if anything, larger multi-partner engagements are exactly where coordination drift is most likely to appear, because more parties means more places for ownership to go unnamed.

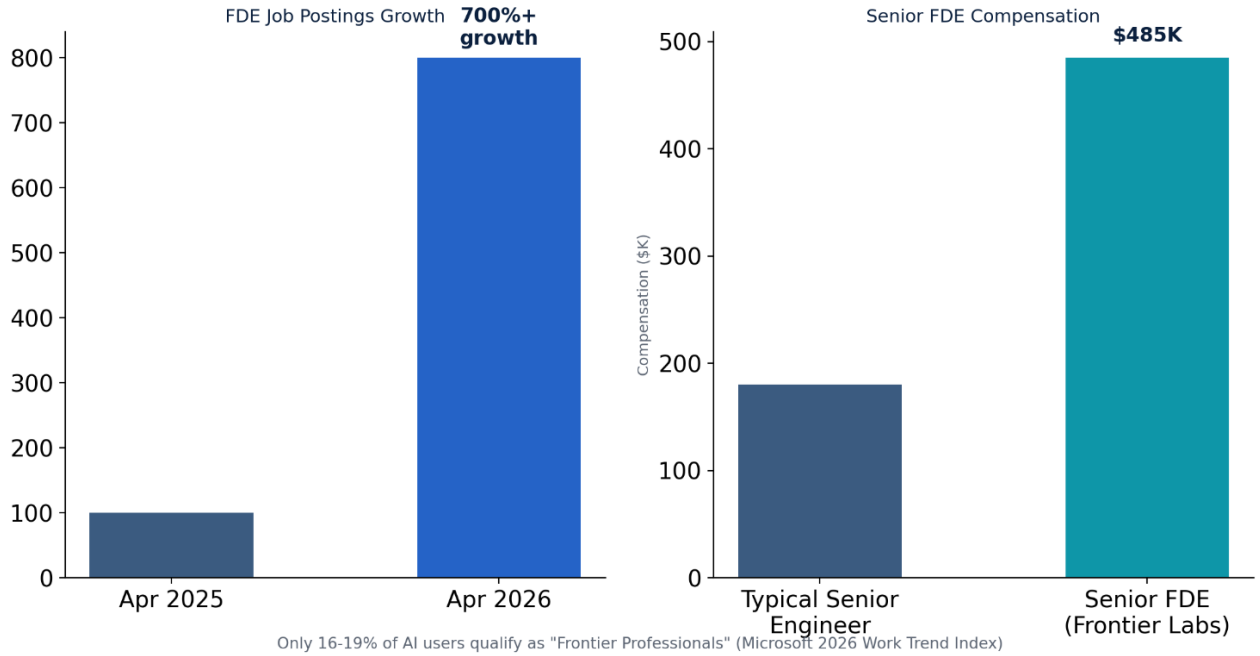
08

Talent, Skills & the New Engineering Persona

Every argument in this report eventually runs into the same wall: this talent is extraordinarily scarce, and the organizations racing to hire it are the best-funded companies in the technology industry.

The Market Has Priced This Scarcity In

The hiring numbers tell the story on their own. FDE job postings grew more than 700% between April 2025 and April 2026. Perspective AI's compensation survey of 1,500 forward-deployed engineers puts median total compensation for a senior FDE at a frontier lab at $485,000, with staff-level roles clearing $725,000.



Forward-deployed engineering has gone from an obscure Palantir job title to one of the most talked-about roles in enterprise technology in barely eighteen months. LinkedIn job-posting data, industry hiring reports, and recruiter commentary all point the same direction: demand for the role is climbing faster than almost any other engineering title tracked in 2026, and the trend is now global rather than confined to Silicon Valley.

India is a clear example of how fast this has spread beyond the US frontier labs that originated the model. A 2026 CIEL HR study found that demand for forward-deployed engineers in India surged 130% year-over-year, with 52 IT organizations actively recruiting for the role and hiring concentrated in Bengaluru, Hyderabad, and Pune. That surge mirrors the same underlying cause driving compensation up at frontier labs in the US: enterprises everywhere are discovering that AI platforms don't deploy themselves, and the engineers who can make that happen are in short supply on every continent, not just in San Francisco.

The three companies most aggressively building this capability right now confirm the pattern from three different angles, which is exactly what your manager's review flagged as worth checking directly:

- **OpenAI** built its FDE organization inside its Model Deployment for Business team, with mid-level compensation in San Francisco running $160,000 to $280,000, roughly 60% above a comparable solutions-engineering role at the same company. OpenAI didn't rely on organic hiring alone to build this bench. The company acquired Tomoro, an applied AI consulting firm bringing roughly 150 engineers with prior deployment experience at companies including Tesco, Virgin Atlantic, and Supercell, specifically to accelerate its FDE team's client-facing experience.
- **Microsoft's** contribution to this picture is less about direct FDE hiring and more about the organizational scarcity underneath it. The company's 2026 Work Trend Index, based on an analysis of Microsoft 365

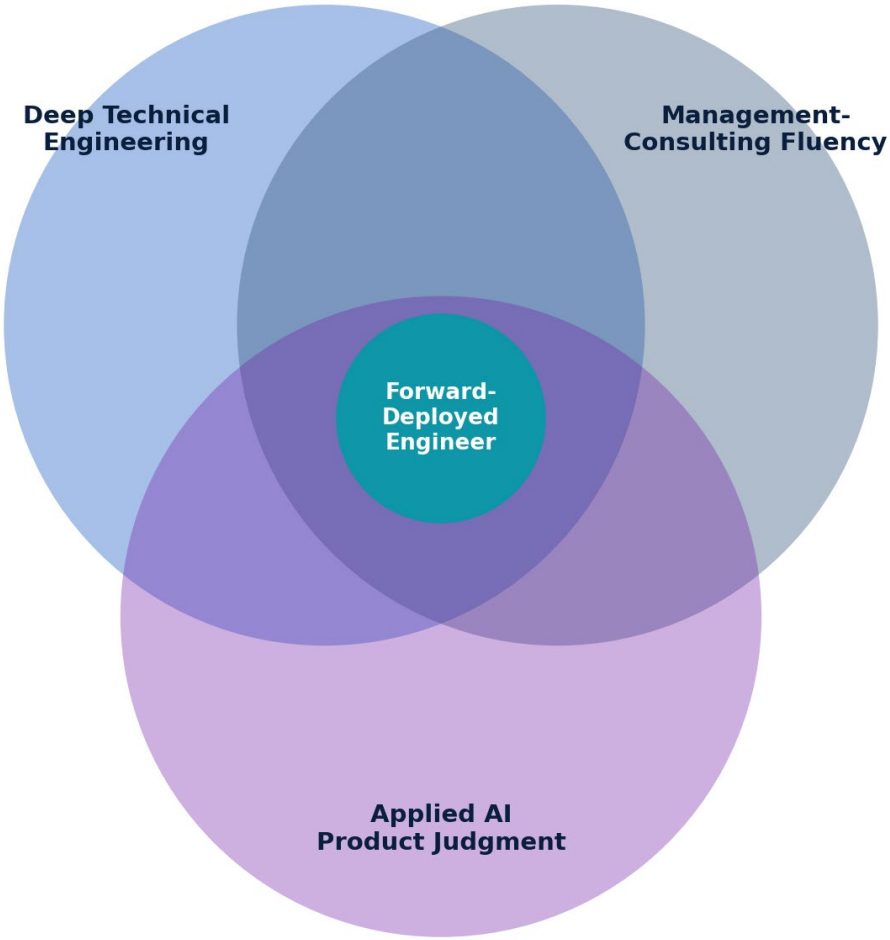
productivity signals and a survey of 20,000 workers across ten countries, found that only 16-19% of AI users qualify as what Microsoft calls "Frontier Professionals," employees advanced enough in how they apply AI to meaningfully redesign their own workflows. Even that group can't succeed alone: Microsoft's research found organizational factors, culture, manager support, and talent practices, account for more than twice the measurable AI impact of individual skill, a 67% to 32% split. That finding matters directly for FDE buyers: hiring or renting individual technical talent solves only about a third of the actual problem. The rest is organizational readiness, which is precisely what Chapter 5's framework is built to diagnose.

- **Accenture** answered the talent question by partnering rather than hiring alone, twice, within the same year. In March 2026, Accenture launched a forward-deployed engineering practice with Microsoft, bringing together what the companies describe as thousands of AI-skilled engineers to work directly with clients. Two months later, Accenture launched a second FDE program with ServiceNow, structuring every engagement around a dedicated pod combining platform-native, AI-native, and industry-specific expertise. Two separate, named FDE partnerships from the same systems integrator within ninety days is a clear signal about how scarce this talent actually is: even a firm with roughly 784,000 employees didn't believe it could build sufficient FDE capacity through internal hiring alone.

What an FDE Actually Needs to Know

Gartner's skills assessment for a successful FDE team spans seven distinct competencies (G00847066), and the breadth of the list is exactly why the role is so hard to hire for:

- **Full-stack technical depth**, moving comfortably between data pipelines, backend logic, cloud infrastructure, and generative AI development and operations, with the AI-specific skills carrying particular weight given where most FDE work is centered today.
- **Data wrangling and ETL**, since a large share of FDE work is last-mile data cleanup: architecting the retrieval or context-engineering pipelines that make an AI agent actually reliable.
- **Problem-solving**, including use-case identification and prioritization, and the ability to translate a vague customer statement like "I want an accurate claims-processing assistant" into a concrete, shippable technical roadmap.
- **Stakeholder management and business translation**, with the confidence to communicate directly with senior technology and business executives and translate business pain into technical specification.
- **An ownership mindset**, taking end-to-end responsibility for an outcome from design through production, not handing off pieces along the way.
- **Rapid learning and adaptability**, since every client environment carries its own constraints and legacy history, and there's no template that transfers cleanly from one engagement to the next.
- **Integration with product and engineering**, acting as the eyes and ears for the vendor's own product team, recognizing when a fix built for one client should become a core platform feature rather than a one-off.



One-third senior systems developer, one-third solutions architect, one-third management consultant is a reasonable way to describe the resulting profile, and it explains why this role commands compensation closer to a specialized technical leadership position than a standard engineering job.

What This Means for the Build-vs-Buy Decision

Put these three data points together and the conclusion is hard to avoid. This is not a talent pool most mid-market or even large enterprises can realistically build in-house on a two-year timeline, not when OpenAI is acquiring entire consulting firms to build the capability and Accenture is forming dual partnerships to do the same thing at its own scale. The practical choice for most organizations isn't build-versus-buy in the abstract. It's how to buy this capability, embedded, with a defined exit plan, in a way that doesn't recreate the dependency risk described in Chapter 6. That's the question the rest of this report, and Chapter 13 in particular, is built to help answer.

How Aptly Addresses the Talent Constraint

The talent scarcity documented above is precisely why Aptly's engagement model leads with knowledge transfer as a contracted deliverable rather than an optional handoff step. Rather than asking every client to independently solve the hiring problem that OpenAI addressed by acquiring an entire 150-engineer consulting firm, and that Accenture addressed by forming two separate platform partnerships within ninety days, Aptly's forward-deployed teams pair a full-stack, AI-native engineer with an internal engineer from the client's own organization for the full duration of every engagement, not just the final handoff phase.

The goal, consistent with the Chapter 8 finding that individual skill accounts for roughly a third of measurable AI impact while organizational readiness accounts for the rest, is to leave the client with both a working system and a trained internal team capable of running it, rather than a system only the vendor's engineer understands.

09

Measuring Success: Outcomes and ROI

Chapter 1 opened this report with a wall of failure statistics. This chapter closes the loop: what does the minority of organizations who avoid that outcome actually measure, and when?

Measure the Outcome, Not the Model

Gartner's guidance here is specific enough to act on directly: define and monitor success at the business-outcome level, not the model or feature level (G00847066). In a results-oriented engagement, value should be measured through concrete, observable business actions, automated discount adjustments executed, pricing interventions triggered, cycle time reduced from a defined baseline, not through an accuracy score or a token-throughput metric that means nothing to anyone outside the engineering team. If the deployed solution doesn't expose these execution-level metrics, it becomes functionally impossible to enforce an SLA, validate business impact, or tell the difference between a model performing well and an engagement being operationally effective.

The Evidence That Most Organizations Skip This Step

A 2026 Beam.ai study found that 42% of AI projects deliver zero ROI, and the underlying pattern is almost worse than the headline number: these are overwhelmingly projects that remained isolated pilots without ever being tied to a defined business KPI in the first place. Success, in the minority of cases that achieve it, correlates directly with organizations that establish clear metrics before development even begins, not after.

Gartner's own success measures for an FDE engagement specifically give a CXO three concrete things to track (G00844355):

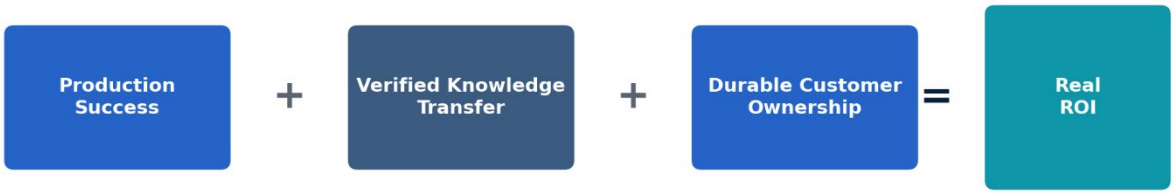
- **Time to first working increment:** A genuine end-to-end capability should appear early in the engagement, consistent with the weekly cadence described in Chapter 4. No working capability within the first half of the engagement timeline is a signal of misscoping or blocked access, and it should trigger an escalation before the remaining budget gets committed.
- **Post-transition operational independence:** The internal engineer assigned to the engagement should be required to demonstrate the solution without FDE assistance before disengagement. If that demonstration fails, the engagement hasn't actually transferred, regardless of what the closing report says.
- **Business outcome delivery:** Measurable improvement against a baseline metric, defined before kickoff, should be evident after go-live. Improvement signals the right problem was selected. No measurable change signals the wrong problem was chosen, independent of how well the technology performed.

Buying Beats Building, By a Wide Margin

The clearest single data point on the build-versus-buy question comes from MIT's Project NANDA: externally sourced AI solutions reach successful deployment roughly twice as often as internally built ones. That finding, combined with the talent scarcity documented in Chapter 8 and the governance discipline documented in Chapters 6 and 7, is the complete case for FDE done correctly: not faster for its own sake, but measurably more likely to reach a production outcome an organization can actually point to.

The Outcomes Aptly Is Built to Deliver

Aptly's own positioning compresses the three success measures in this chapter into a single formula the company uses internally to evaluate whether an engagement has actually succeeded: Production Success plus Knowledge Transfer plus Customer Ownership. Production Success maps directly onto the “time to first working increment” and “business outcome delivery” measures Gartner recommends. Knowledge Transfer maps onto “post-transition operational independence.”



Customer Ownership is the outcome that distinguishes a successful FDE engagement from one that has merely shipped a working system: the client owns the code, the documentation, and the operational knowledge required to evolve the platform without renewing a vendor contract to do it.

10

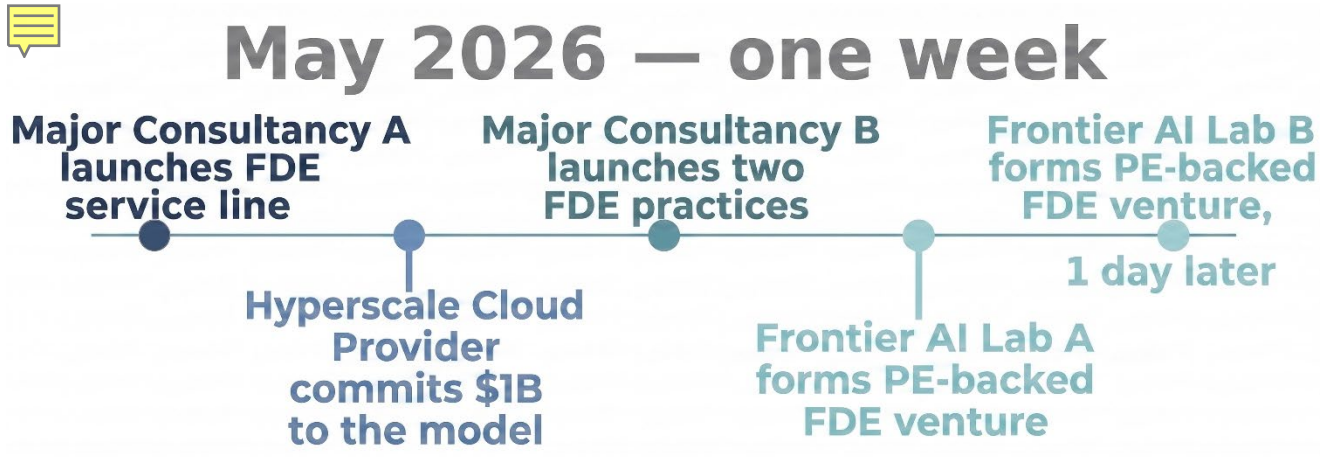
The Future of Forward-Deployed AI Engineering

Any report on a fast-moving delivery model risks feeling dated within months of publication. The developments below are already visible in the market as of mid-2026, and they point toward where the model goes next rather than where it's been.

The Billion-Dollar Validation Nobody Saw Coming a Year Ago

The single clearest signal that FDE has become the default enterprise AI go-to-market motion arrived in one extraordinary week in May 2026. On May 4, Anthropic announced a joint venture with Blackstone, Hellman & Friedman, and Goldman Sachs, backed by a wider group of investors including Apollo Global Management, General Atlantic, GIC, and Sequoia Capital, to build a new $1.5 billion enterprise services firm. Its explicit purpose: embed applied engineers alongside client teams to deploy AI directly into core operations, reaching, through the PE firms' combined portfolio companies, an instant distribution network spanning hundreds of mid-sized businesses that would otherwise take years to build through a traditional sales motion.

Within a day, reporting confirmed OpenAI was pursuing a parallel structure, a venture that press coverage nicknamed The Deployment Company, targeting a raise in the range of $4 billion from investors including TPG, Bain Capital, Brookfield, and Advent, against a reported $10 billion valuation.



Strip away the size of the numbers and the structural story is simple: the two companies that arguably need FDE least, because they own the underlying models, concluded independently and in the same week that the only credible way to win enterprise AI accounts at scale was to build (or buy) exactly the embedded-engineering capability this report describes. That's about as strong a validation signal as a delivery model can get.

Systems Integrators Are Racing to Match It

Accenture's dual FDE partnerships with Microsoft and ServiceNow, both launched within the same year and covered in Chapters 2 and 8, show the largest consulting and integration firms reaching the same conclusion from the opposite direction: rather than watching AI-native vendors disintermediate the traditional implementation business, get inside the model directly, at scale, through named partnerships rather than organic hiring alone.

Deloitte's own FDE practice fits the same pattern. Expect more of the major systems integrators to announce comparable partnerships over the next twelve to eighteen months, because the alternative, ceding the entire embedded-delivery motion to AI-native vendors, is a much larger strategic risk than the cost of standing up a new practice.

Two Structural Shifts Already Underway

Gartner's forward-looking analysis (G00847066) points to two changes in how the FDE role itself evolves, both already visible in early form:

- ✓ **The rise of the "internal FDE.":** Large enterprises, particularly in regulated or forward-leaning industries, are beginning to reduce dependency on vendor FDEs by building the function internally: an in-house forward-deployed engineering team acting as a permanent bridge between IT and specific business units, rather than a rotating cast of external consultants. Given the talent scarcity and cost figures documented in Chapters 6 and 8, this shift will be realistic for only a small subset of the largest enterprises. For most organizations, the more practical version of this trend is demanding the knowledge-transfer discipline in Chapter 7 rigorously enough that internal capability grows alongside

every external engagement, rather than choosing between full internal build and full external dependency.

- ✓ **AI agents as "junior FDEs.":** AI agents are starting to absorb the lowest-level technical work an FDE does today, mapping orphaned APIs, writing boilerplate ETL scripts, basic troubleshooting, freeing human FDEs to spend more time on stakeholder alignment and workflow judgment rather than boilerplate code. Palantir has already shipped an early version of this idea: an AI-powered FDE in beta that operates its Foundry platform through conversational commands rather than direct engineering work. Perspective AI's 2026 survey data confirms the shift is already showing up in how FDEs spend their time: 30-40% of a working week now goes to customer discovery conversations, up substantially from the mostly-coding role the title implied even two years ago.

Both shifts point toward the same underlying conclusion: as the technical grunt work compresses, organizational change management, translating business pain into a working system and getting an organization to actually adopt it, becomes the harder and more valuable half of the job. The bottleneck is already moving from "can we build this" to "can we get the organization to run it," and that's precisely the half of the problem a well-structured FDE engagement, and the governance framework in Chapter 5, is built to solve.

11

Forward Deployed Engineering Case Studies | Aptly

Chapter 1 opened this report with the numbers: 80 to 95 percent of enterprise AI initiatives never reach production, or reach it without moving the P&L. Chapter 5 introduced the Aptly FDE Readiness & Value Framework, the seven conditions that decide whether an engagement builds real capability or turns quietly into dependency. The three engagements below show what both sides of that line look like on the ground, applied to production systems rather than described in the abstract.

Each client started from a different point on the maturity curve, and none of the three arrived at Aptly with a working pilot in hand. The first had years of documentation scattered across repositories with no reliable way to search across all of them at once. The second had a support team spending real time on repetitive tickets that existing documentation already answered, if anyone had the time to find the right source mid-call. The third had gone further than most, a working AI prototype that proved the underlying idea, and stopped there: no deployment pipeline, no access controls, no monitoring, no defined cost. That third case is the specific failure mode Chapter 1 spends the most space on, a technically sound pilot that never crosses into production because nobody engineered the system around it.

None of the three needed a strategy deck. They needed engineers inside their environment, working against real data and real workflows, building toward a system the internal team could run without them. That is the model Chapter 2 defines as Forward-Deployed Engineering, and Chapter 4 lays out the five phases every Aptly engagement moves through: technical discovery, secure infrastructure setup, ingestion and integration, evaluation and user acceptance testing, and operations handover. Each case study below traces that same path,

from a fragmented or unfinished starting point to a governed production system, with the internal team demonstrated as capable of running it alone.

Chapters 6 and 7 spent real space on what happens when that last phase does not happen the way it is supposed to: vendor dependency, shadow architecture nobody at the client knows how to maintain, an engagement that never actually transfers. The Governance & Knowledge Transfer section in each case study below shows the specific mechanics Aptly used to prevent that outcome on each engagement: IP classified as commodity or differentiating work before the first sprint, an internal engineer building alongside the Aptly team from day one rather than at handover, and exit criteria agreed before anyone signed a statement of work.

The Outcomes section in each case study follows the same measurement discipline Chapter 9 recommends: time to first working increment, post-transition operational independence, and business outcome delivery against a defined baseline, the three measures that together make up what Aptly tracks internally as Production Success, Knowledge Transfer, and Customer Ownership.

Read together, the three engagements span a knowledge platform built on retrieval-augmented generation and semantic search, a customer support agent built around a controlled human-in-the-loop escalation model, and an AI prototype re-architected into a production application with CI/CD, model gateways, and full observability. Different industries and different technology choices, built on the same underlying discipline: engineers embedded in the client's actual environment, working against production data, with a defined and demonstrated path to the client running the system alone.

Case Study 01: Enterprise Knowledge Intelligence Platform

Industry: Enterprise Technology

Turning fragmented enterprise knowledge into a governed, conversational AI system employees actually trust.

Starting Point

The client, a large enterprise technology organization, had spent years accumulating documentation across internal portals, operational manuals, policy repositories, and multiple product knowledge bases, with no single system that could answer a question across all of them. Employees searched manually, often across several tools before finding an answer, and often gave up and asked a subject-matter expert directly instead, pulling that expert away from their own work to re-answer a question already documented somewhere in the organization.

This was not a case of an early pilot stalling before production. There had been no pilot. Internal teams had discussed a conversational knowledge tool for some time, but no one owned the problem end to end, and the fragmentation itself, dozens of repositories with different owners, formats, and refresh cycles, made it a poor candidate for an off-the-shelf chatbot layered on top of a single system. The organization needed retrieval that worked across genuinely disparate sources, respected existing access permissions, and grounded every answer in a traceable document rather than a model's best guess. That combination of integration depth and governance requirement, more than any single technical challenge, is what made this a Forward-Deployed Engineering problem rather than a standard software deployment.

The Engagement

Aptly's FDE team opened the engagement with a knowledge-domain assessment: identifying which repositories carried the highest query volume, who owned each source, how often content changed, and what access constraints applied to each one. That assessment defined the Minimum Viable Value for the engagement, proving that a governed retrieval system could answer real employee questions accurately across a first set of high-value sources before expanding to the full repository list.

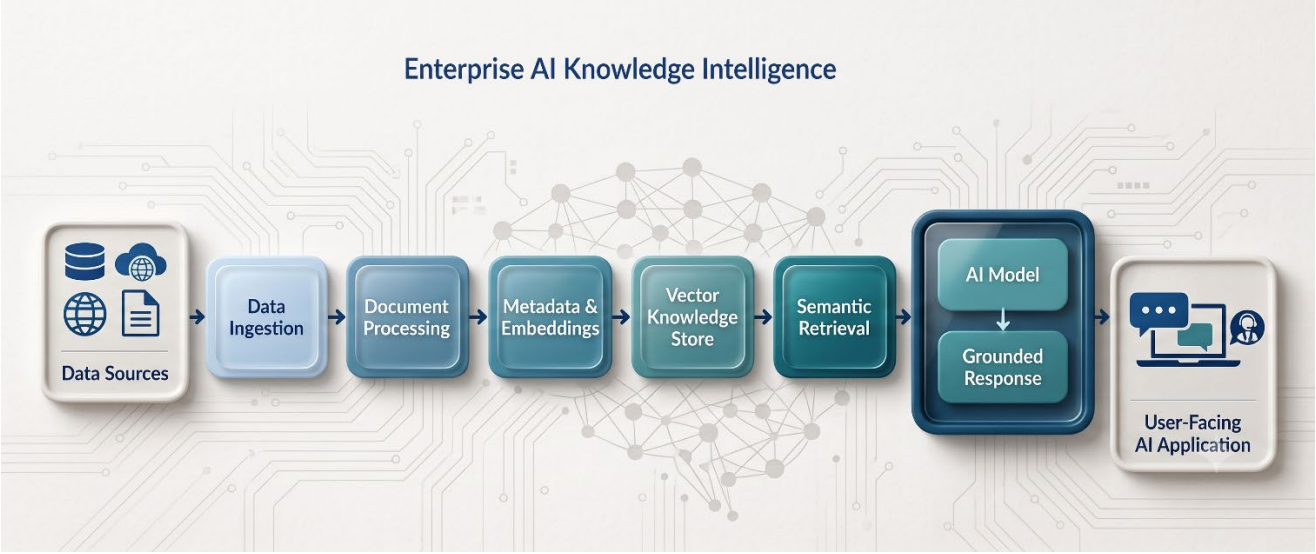


Figure: Knowledge Architecture Diagram. *Enterprise data sources flow through ingestion, document processing, content transformation, metadata classification, embedding generation, a vector knowledge store, semantic retrieval, and context construction, into an AI model that returns a grounded response through the conversational interface.*

With that scope defined, the team built a production-grade retrieval pipeline: document ingestion and normalization, metadata preservation, intelligent chunking, embedding generation, and a vector-based knowledge layer queried through a conversational interface. An AI agent layer sat on top of retrieval, handling intent understanding, checking whether a given query actually required a knowledge lookup, retrieving context when it did, and performing confidence and policy checks before returning a response, escalating to a human when confidence fell below the defined threshold.

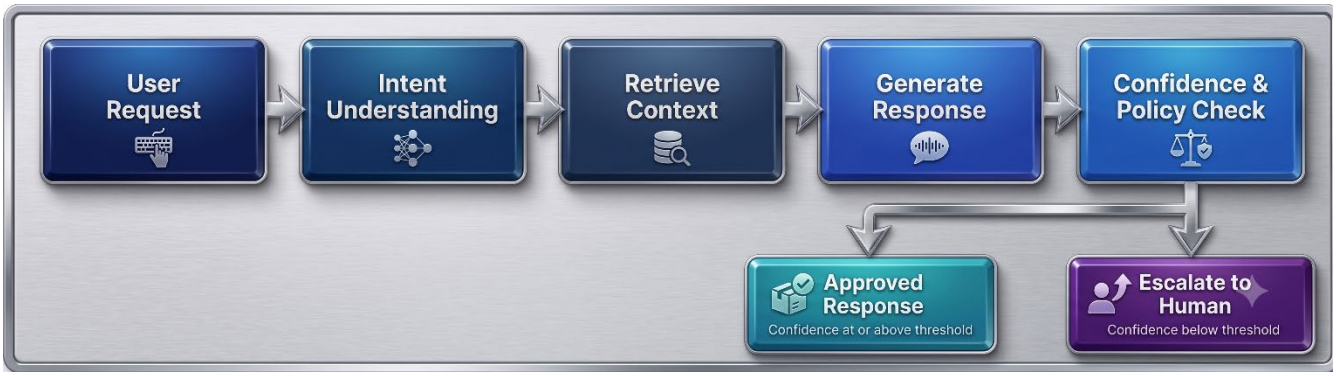


Figure: AI Agent Architecture Diagram*. A user request enters the AI agent, moves through intent understanding, checks whether knowledge is required, retrieves relevant context when needed, generates a response, performs confidence and policy checks, and either returns an approved response or escalates to a human assistant.*

Phase 3 of the engagement, ingestion and integration, took the most time on the calendar, consistent with what Chapter 4 describes as typical for FDE work: connector-based ingestion across sources with genuinely different formats and access models took longer than the platform-building work that followed it. Before production rollout, the team ran a dedicated evaluation phase against defined accuracy thresholds, testing retrieval relevance, answer grounding, and how the system behaved on questions it should decline to answer rather than guess at.

Governance & Knowledge Transfer

IP and access classification were addressed at the design stage, not after the fact. Connector logic and retrieval patterns built for this engagement were treated as commodity capability and routed into Aptly's own platform roadmap where a reusable pattern applied, while the client's specific document taxonomy, access rules, and evaluation criteria stayed the client's own IP under the engagement's standard classification terms.

An internal client engineer worked alongside the Aptly FDE team from the first sprint, not only during a final handoff, learning the ingestion pipeline, the retrieval logic, and the observability tooling as each was built rather than being briefed on a finished system. Access-aware response generation, aligned to the client's existing enterprise permissions, and AI observability covering usage, latency, retrieval quality, and model behavior were both built as standard components of the platform rather than add-ons requested after launch, giving the client's own operations team visibility into system health without depending on the FDE team to interpret it for them.

Exit criteria were defined before the engagement began: the internal engineer needed to demonstrate independent operation of the ingestion pipeline and retrieval layer, and content-freshness and retrieval-performance monitoring needed to run without FDE intervention, before disengagement.

Outcomes

- **Time to first working increment:** A working retrieval capability against the first set of high-value sources was demonstrated early in the engagement, consistent with the weekly cadence Aptly runs against production data rather than a staging environment.
- **Post-transition independence:** The client's internal engineer took over operational monitoring for content freshness and retrieval performance, and the client's own team now owns connector configuration for adding new sources going forward.
- **Business outcome delivery:** The client tracked outcomes at the business level rather than the model level, employee time spent searching for information, the volume of questions redirected to subject-matter experts, and the consistency of answers returned across repeat queries, and reported improvement across all three after go-live. The platform now functions as a reusable foundation the organization is extending into additional business domains beyond the original knowledge use case.

Case Study 02: Intelligent Customer Support Automation

Industry: Customer Service

An AI support agent built to work alongside human representatives, not replace them.

Starting Point

The client ran a high-volume customer support operation where representatives spent a significant share of their time on repetitive inquiries, questions that existing product documentation, policy references, and internal knowledge bases could already answer, if a representative had the time to search for the right source mid-call. Support volume was growing faster than headcount, and the organization had reached the point where adding more representatives to handle the same repetitive questions was no longer a sustainable answer.

The organization had not run a prior AI pilot for this specific problem. What it had was clear internal alignment that automation belonged in the routine, low-complexity share of the ticket volume, and equally clear alignment that customer trust in sensitive or ambiguous cases depended on a human handling them. That distinction, between what should be automated and what should stay human, needed to be built into the system's design from the outset rather than discovered after launch, which made confidence-based escalation the central engineering problem of the engagement rather than a secondary feature.

The Engagement

Aptly mapped the client's most common customer intents first, working directly with support leadership and representatives to understand which inquiries were genuinely repetitive versus which only looked that way on the surface. That mapping, combined with an inventory of approved knowledge sources, defined the Minimum Viable Value: an agent that could reliably resolve the highest-volume, lowest-ambiguity intent categories on its own, with everything else routed to a human by design rather than by exception.

The team built the support agent around a controlled human-in-the-loop model. Customer interactions enter through existing support channels and route to the AI agent, which detects intent, retrieves relevant knowledge, maintains conversation context across a multi-turn interaction, generates a response, and evaluates its own resolution confidence before deciding whether to answer directly or escalate.

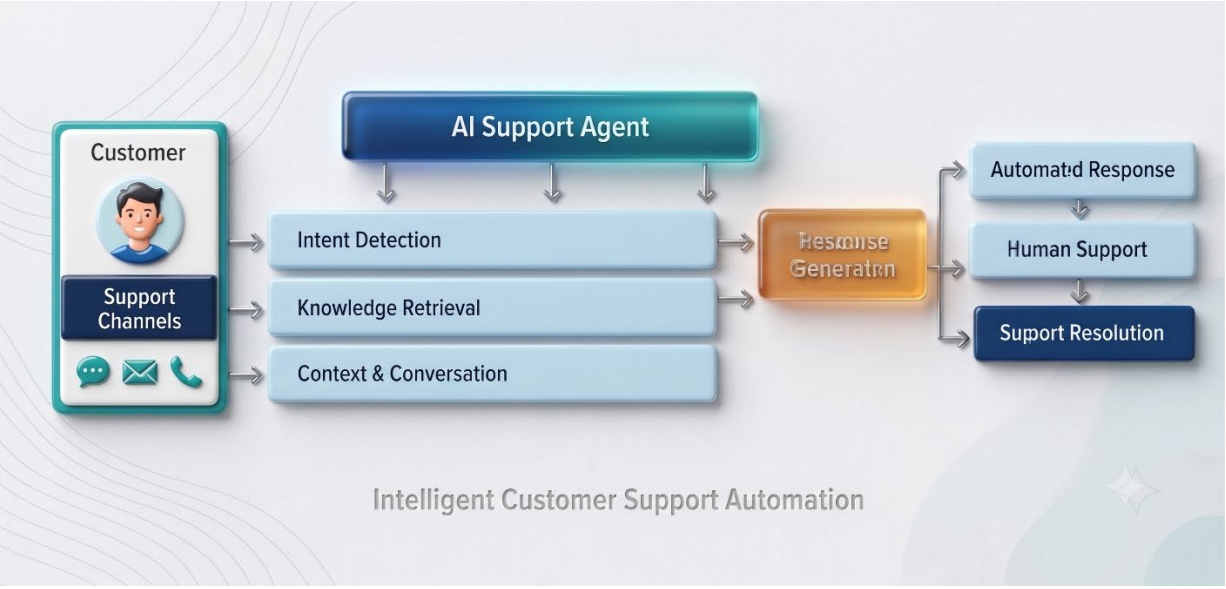


Figure: Support Automation Architecture Diagram. *Customer interactions enter through support channels and route to the AI support agent, which detects intent, retrieves relevant knowledge, maintains conversation context, generates a response, and evaluates resolution confidence. High-confidence requests receive automated responses, while low-confidence, complex, or sensitive requests escalate to human support with prepared context.*

Defining the escalation criteria and confidence thresholds was the hardest part of the engagement, harder than the retrieval or response-generation work itself, because it required translating a support team's tacit judgment about which cases need a human into an explicit, testable rule set. Aptly's team worked through this iteratively during the evaluation phase, using representative feedback from real interactions rather than synthetic test cases to calibrate where the threshold actually needed to sit. The system also captured interaction telemetry throughout, supporting response-quality review, knowledge-base improvement, and operational reporting once live.

Governance & Knowledge Transfer

Escalation logic, confidence scoring, and the underlying agent orchestration were classified as commodity capability under the engagement's IP terms and structured for integration into Aptly's own platform, while the client's specific intent taxonomy, escalation rules, and knowledge-base content remained the client's own differentiating IP.

The system captures interaction telemetry as a standard part of its architecture rather than a reporting feature added after the fact, giving the client's support-operations team the data needed for response-quality review and knowledge-base improvement without depending on the FDE team to generate those reports. That telemetry pipeline, along with the confidence-threshold logic, was built and documented alongside the client's own internal team so ownership of tuning the thresholds over time sits with the client, not with Aptly.

Exit criteria required the client's support-operations team to demonstrate independent management of escalation rules and knowledge-base updates before disengagement. The engagement was scoped explicitly to leave behind a support-agent foundation the client could extend to additional channels and support domains without renewing an Aptly engagement to do it.

Outcomes

- **Time to first working increment:** The agent handled a working set of real customer intents against production support channels early in the engagement, with escalation behavior validated against actual representative judgment rather than a synthetic accuracy benchmark.
- **Post-transition independence:** The client's support-operations team now owns tuning of escalation thresholds and knowledge-base content, using the interaction telemetry the system generates as its primary input rather than relying on Aptly to interpret system behavior for them.
- **Business outcome delivery:** The client measured response speed for repetitive inquiries, the volume of manual knowledge searches representatives performed per shift, and the consistency of customer-facing responses across representatives, and reported improvement on each after go-live, while preserving human oversight for every complex, sensitive, or low-confidence case by design.

Case Study 03: AI Productionization & Enterprise Deployment

Industry: Enterprise Software & Technology

Turning a validated prototype into a system the client's own engineering team runs in production.

Starting Point

The client had already done the hard part that most organizations in Chapter 1's failure statistics never get to: an AI proof of concept had demonstrated genuine feasibility. Internal stakeholders had seen it work. What it did not have was any of the infrastructure production software actually requires, no defined deployment pipeline, no authentication or access control model, no monitoring, no resilience patterns for handling failures gracefully, and no visibility into what running the system at scale would actually cost.

This is the specific failure mode Chapter 1 describes as the most common one: a technically sound pilot that never crosses into production, not because the underlying AI did not work, but because nobody had engineered the system around it. The organization recognized this gap correctly and brought in Aptly specifically to close it, rather than attempting to retrofit production requirements onto the prototype internally while also trying to keep the underlying product roadmap moving.

The Engagement

The Minimum Viable Value question here was different from the other two engagements: not what new capability to prove, but what minimum set of production requirements needed to be in place before the existing capability could be trusted to run unattended. Aptly's team scoped that against the client's own reliability, security, and compliance requirements before writing any production code.

Aptly converted the prototype into a production application by separating prototype logic into resilient services, defining deployment environments, and introducing CI/CD so releases became repeatable rather than manual. The team implemented authentication and authorization, evaluated model options against the client's actual workload requirements, and introduced a model-gateway pattern to manage that model traffic centrally rather than hard-coding model calls throughout the application.

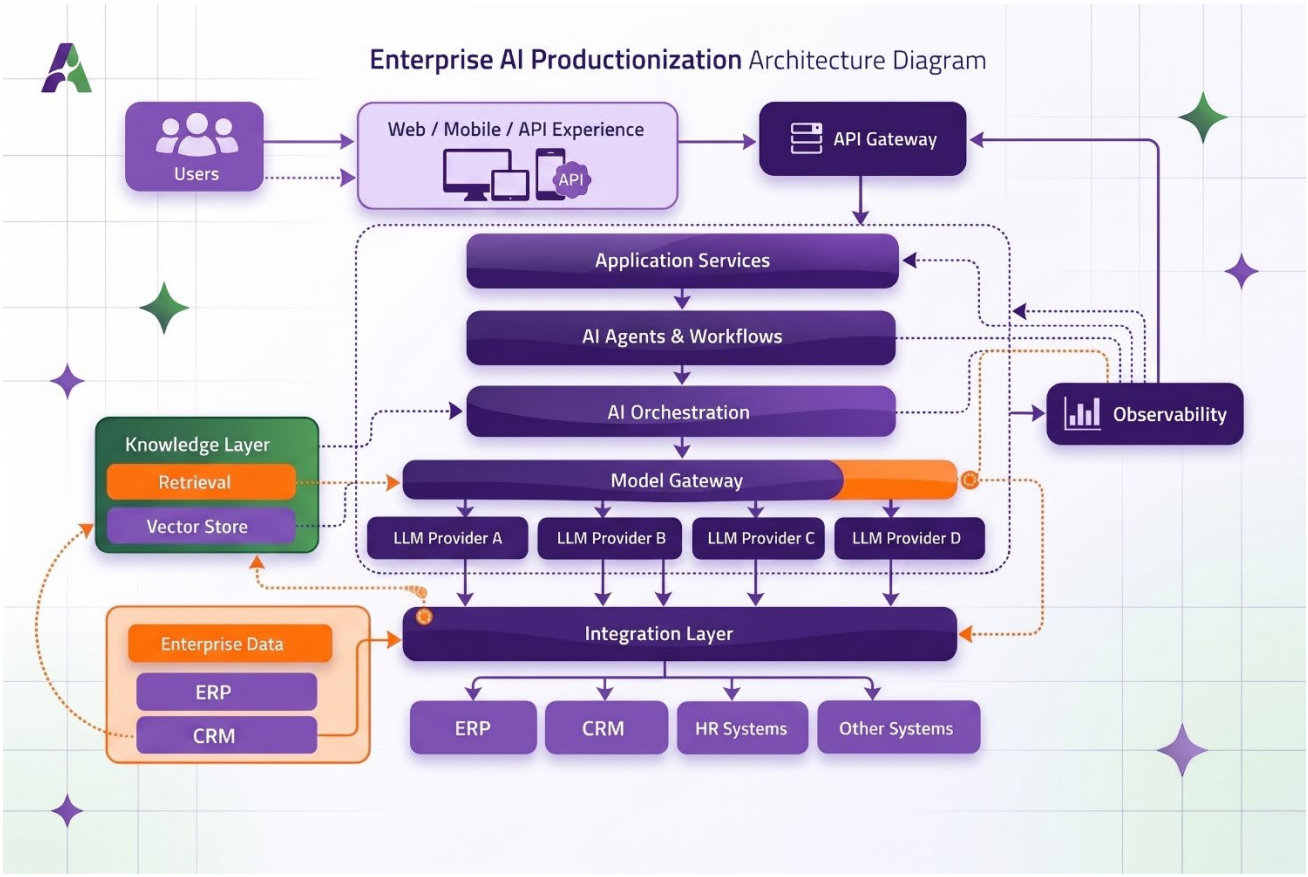


Figure: Production Architecture Diagram. *Users access the AI application through web, mobile, or API experiences. Requests pass through an API gateway into application services, AI agents, orchestration, model gateways, knowledge retrieval, and enterprise integrations. Observability spans the application, agent, orchestration, and model layers, giving the team visibility into reliability, latency, errors, and cost in production.*

Phase 4, system evaluation and user acceptance testing, carried the most weight in this engagement relative to the other two, because production readiness here meant validating reliability, security, scalability, and supportability against defined criteria, rather than only confirming the AI produced correct answers. Aptly ran structured production-readiness reviews before go-live to confirm each of those dimensions independently rather than treating a successful demo as sufficient evidence of readiness.

Governance & Knowledge Transfer

The model-gateway pattern, CI/CD pipeline design, and core observability tooling were classified as commodity capability, built to integrate into Aptly's own platform for reuse across future engagements, while the client's specific business logic, workload configuration, and cost-management rules remained the client's own IP.

Aptly added monitoring for latency, errors, token consumption, and cost as core components of the architecture, giving the client's engineering team direct visibility into production behavior and spend without needing to request that data from Aptly after the fact. The client's own engineers worked alongside the Aptly team through the CI/CD build-out and the production-readiness review process itself, rather than being introduced to the finished pipeline at handover.

Exit criteria required the client's engineering team to independently execute a full deployment cycle, including rollback, before disengagement, and required a documented architecture and operations runbook as a contracted deliverable rather than an optional artifact. That runbook, along with the observability tooling built into the platform, is what lets the client's own team maintain and extend the system going forward.

Outcomes

- **Time to first working increment:** A production-grade version of the core prototype capability was running against real workload conditions early in the engagement, ahead of the full CI/CD and observability build-out being completed.
- **Post-transition independence:** The client's engineering team executed a full deployment cycle independently, including rollback, before the engagement closed, and now owns the CI/CD pipeline and production-readiness process for future releases.
- **Business outcome delivery:** The client tracked deployment reliability, release repeatability, and cost visibility against its pre-engagement baseline, where none of the three had existed in measurable form, and reported improvement across all three after go-live, along with an architecture foundation the organization is now using for additional AI capabilities beyond the original prototype.

Executive Recommendations

Every recommendation below traces back to a specific finding earlier in this report. None of it is generic advice; all of it is what the data in Chapters 1 through 11 points to directly.

Short Term (0-6 Months)

- Run the Aptly FDE Readiness & Value Framework against any AI initiative currently stalled in pilot, before committing budget to an FDE engagement or extending an existing one. The lowest-scoring pillar, not the average, tells you where the real risk sits.
- Before signing any FDE contract, verify all four accountability terms from Chapter 7 are present: IP classification, deliverable definitions, handoff criteria, and provisioning SLAs. Treat a missing term as a reason to escalate, not a detail to fix later.
- Define business-outcome success metrics, not model-performance metrics, before any engagement kicks off. If you can't state the target metric in one sentence a business unit leader would recognize, the engagement isn't ready to start.
- If evaluating external FDE vendors, request each vendor's own score against the seven-pillar readiness framework in Chapter 5 for a comparable past engagement, not just references. A vendor unwilling to self-score, or unable to name a specific engagement where the framework flagged a risk it then addressed, is a vendor asking to be trusted rather than verified.

Mid Term (6-18 Months)

- Assign an internal engineer to shadow every FDE engagement from sprint one, not just the handoff phase, and make that assignment a contractual deliverable rather than an informal expectation.
- Build the governance guardrails from Chapter 6, context-engineering controls, shadow-AI prevention, ontology co-ownership, into every engagement regardless of vendor size or reputation. A bigger vendor name is not a substitute for these controls.
- Track the specific warning signs of dependency rather than progress: flat FDE effort across successive deployments, an FDE who can no longer name a product counterpart on the vendor side, and any handoff milestone that keeps slipping.

Long Term (18+ Months)

- Decide deliberately, and revisit annually, whether to build an internal FDE function or continue sourcing the capability externally. Given the talent economics in Chapter 8, this will be the right call for only a subset of the largest enterprises; for most organizations, the better long-term investment is demanding rigorous knowledge transfer on every external engagement so internal capability compounds over time.
- Design for the agentic future described in Chapter 10: as AI agents absorb more of the low-level technical work, invest disproportionately in the organizational change management and workflow-redesign skills that remain scarce even as the coding work commoditizes.
- Reassess vendor relationships against the accountability framework in Chapter 7 at least once a year, particularly as the vendor landscape consolidates around larger multi-partner engagements where coordination drift is hardest to catch.

12
Where Aptly Fits

Everything in this report describes what separates a Forward-Deployed AI Engineering engagement that builds lasting capability from one that builds dependency: a named executive sponsor, a disciplined weekly cadence against production data, governance guardrails defined before day one, an accountability structure with real contractual teeth, and an exit plan built in from the start rather than negotiated under pressure at the end. This closing chapter covers how Aptly's own Forward-Deployed AI Engineering Services are built around exactly those principles, and where to start if any of the gaps this report identifies sound familiar.

Who Aptly Is

Aptly Technology helps middle-market and large enterprises turn high-value AI opportunities into secure, production-ready capabilities with measurable business impact. The company's forward-deployed teams originate from the AI research and engineering group behind AptlyStar.ai, Aptly's own generative AI agent platform, which means the engineers embedding inside a client's environment are drawing on direct product-building experience, not a generic consulting playbook applied to AI for the first time. Aptly is headquartered in Bellevue, Washington, with delivery centers across North America, Europe, and Asia, including Hyderabad and Shanghai, and holds ISO/IEC 42001, ISO 27001, and SOC 2 certifications, covering AI governance, information security, and operational controls respectively.

Where Aptly Focuses First

Aptly prioritizes three industries where the pilot-to-production gap this report documents is most acute and most costly to leave unresolved: Banking & Financial Services, where large AI budgets collide with fraud and compliance exposure that demands strong security and governance from day one; Healthcare & Pharma, where clinical pilots and strict regulation create an outsized need for the governance guardrails described in Chapter 6; and Manufacturing, where legacy systems and high-value supply chain AI use cases require exactly the integration depth Chapter 4's lifecycle is built around. Defense & Government and Research & Life Sciences follow as a second wave, given the additional security review and sales-cycle length those sectors typically require.

The economics behind this focus are straightforward. IDC and Lenovo's 2025 AI CIO Playbook found that for every 33 AI proofs-of-concept an enterprise starts, only four reach production, an 88% failure rate that tracks closely with the RAND, MIT NANDA, and Gartner figures cited in Chapter 1. Against that backdrop, and against a senior AI engineering compensation range of roughly $285,000 to $473,000 in the US market, the build-versus-buy calculus in Chapter 8 resolves quickly for most mid-market and large enterprises: the cost of assembling and retaining an internal FDE-caliber team, on top of the underlying AI platform spend, routinely exceeds the cost of an embedded engagement structured with a defined exit plan.

The Full Lifecycle, Not the Last Mile

Aptly's engagement model maps directly onto the delivery lifecycle in Chapter 4, and onto the accountability structure in Chapter 7, rather than treating either as an afterthought:

- **Business case definition and use-case prioritization**, including AI readiness assessment, proof-of-concept and proof-of-value roadmapping, and success criteria and ROI targets defined from day one, addressing the outcome-measurement pillar from Chapter 5 before an engagement even begins.
- **AI and agent architecture**, covering enterprise solution design, multi-agent architecture, retrieval and knowledge-base design, human-in-the-loop approval workflows, and Model Context Protocol and tool-integration design.
- **Enterprise data and application integration**, connecting AI securely to fragmented data, legacy applications, and existing business workflows, the exact integration burden Chapter 5's readiness framework asks organizations to estimate honestly before committing to an engagement.
- **Model selection, orchestration, and evaluation**, including commercial and open-source model benchmarking, fine-tuning, and rigorous accuracy, hallucination, latency, and cost evaluation, directly addressing the evaluation discipline described in Chapter 4's Phase 4.
- **Security, privacy, governance, and Responsible AI**, built into the engagement throughout rather than added at the end: prompt-injection and data-leakage testing, red-teaming, bias and safety assessment, and governance-framework implementation that maps directly onto the guardrails in Chapter 6.
- **Production engineering, deployment, and observability**, including infrastructure-as-code, CI/CD, Kubernetes and model-serving deployment, and full-stack observability across the application, model, agent, and infrastructure layers.
- **User adoption, workflow integration, and continuous optimization**, addressing the organizational-change dimension Chapter 10 identifies as the harder half of the FDE job, including role-based enablement and human oversight and escalation workflow design.
- **Outcome measurement and ROI management**, with executive dashboards and continuous value review built in from the start, giving a client the exact business-outcome metrics Chapter 9 recommends rather than a model-performance number nobody outside engineering can interpret.



How Aptly Addresses the Risks This Report Documents

The governance and accountability principles running through Chapters 5 through 7 aren't external best practices Aptly is retrofitting onto its delivery model. They're already built into how engagements are structured:

Knowledge transfer is a stated deliverable, not an afterthought. Aptly's forward-deployed model is built around collaboration and human-in-the-loop practices specifically so a client's internal team can confidently operate, govern, and scale the resulting AI applications independently, addressing directly the talent-atrophy and dependency risks documented in Chapter 6.

Every engagement is technology-agnostic. Aptly recommends models, platforms, and architectures based on client requirements rather than steering toward a proprietary stack, working across commercial and open-source models and platforms including Microsoft Azure, Azure AI Foundry, AWS, and Kubernetes, which limits the vendor lock-in risk Chapter 6 identifies as one of the model's most persistent long-term exposures.

Security and governance are embedded from the first engagement, not bolted on before launch. ISO/IEC 42001 certification for AI governance puts Aptly ahead of where the regulatory frameworks documented in Chapter 6, Singapore's Model AI Governance Framework and NIST's AI Agent Standards Initiative among them, are heading over the next eighteen months, rather than scrambling to retrofit compliance once those frameworks mature further.

Engagements are structured around defined outcomes, not open-ended staffing. Aptly offers a range of engagement models, from fixed-scope readiness assessments and proof-of-value engagements through embedded forward-deployed teams and full production deployment programs, so a client can match engagement structure to actual need rather than defaulting to the largest, most open-ended (and most dependency-prone) option by default.

How to Start an Engagement With Aptly

Consistent with the tiered model described earlier in this chapter, Aptly offers a low-commitment entry point specifically so a prospective client isn't asked to sign a large, open-ended contract before either party has validated the opportunity. A typical starting engagement is an AI opportunity sprint, a fixed-scope, two-to-three-week engagement, priced at roughly \$15,000, that produces a prioritized use-case shortlist and an honest readiness score against the seven pillars in Chapter 5. From there, a Proof-of-Value engagement of four to eight weeks validates a single Minimum Viable Value use case against production data, consistent with the Phase 1 discovery discipline described in Chapter 4, before any decision is made to scale into a full Production or Managed Ops commitment.

This sequencing matters more than it might first appear. A vendor confident in its own delivery model has no reason to insist on a large upfront commitment before either side has evidence the engagement will work, and a buyer who insists on this kind of staged entry point is, in effect, applying the readiness framework in Chapter 5 to the vendor relationship itself, not just to the internal organization.

Why This Report Exists

Aptly built this report because the gap it documents, between how confidently vendors sell FDE and how little independent research exists to help a CXO evaluate it, is exactly the kind of information asymmetry that produces the outcomes Chapter 6 warns about. An enterprise that walks into an FDE engagement with the readiness framework in Chapter 5, the contract checklist in Chapter 7, and the outcome metrics in Chapter 9 already in hand is simply better equipped to get what it's paying for, regardless of which vendor ends up doing the work.

If any of the gaps in this report sound like where your organization sits right now, that's the right starting point for a conversation. Aptly's engagements typically begin with a focused discovery and solution-design phase: clarifying the target business outcome, prioritizing the highest-value use cases, assessing data and organizational readiness against the framework in Chapter 5, and defining measurable success criteria before a single line of production code gets written.

Schedule a Strategy Session with Aptly's FDE team →

Frequently Asked Questions

- **What is Forward-Deployed AI Engineering, in one sentence?**

    - An embedded delivery model where engineers work directly inside a client's environment, building against real data and workflows, to move an AI initiative from pilot to production, rather than handing over a strategy document and a roadmap.

- **How is FDE different from the AI consulting we've already tried?**
    - Traditional consulting typically ends at a strategy, assessment, or roadmap. FDE engagements don't end there; the team remains involved through implementation, production readiness, user adoption, and continuous improvement, and is measured on production outcomes rather than delivered recommendations.
- **Is FDE just an expensive way to get staff augmentation?**
    - It shouldn't be, and Chapter 7 exists specifically to help you verify it isn't. Staff augmentation puts an individual under your direction with no defined outcome. A properly structured FDE engagement brings a defined delivery model, contractual accountability for an agreed outcome, and an exit plan negotiated before the engagement starts.
- **How much does an FDE engagement typically cost?**
    - Anecdotal industry evidence puts FDE consulting fees in the range of $200,000 to $400,000 per quarter, per use case, before platform and integration costs (Gartner, G00844355). Actual pricing depends heavily on scope, team composition, and engagement structure, which is exactly why Chapter 7's deliverable definitions matter: vague scope is the most common way this number grows without a corresponding increase in delivered value.
- **How long does a typical engagement take?**
    - A focused proof-of-value engagement can complete in several weeks. A full production deployment spanning multiple enterprise systems, security reviews, and user groups more typically requires several months, with milestones and dependencies established at the outset.
- **Who owns the code and IP an FDE builds inside our environment?**
    - This should be defined explicitly in the contract before the engagement begins, not assumed. Chapter 7's IP classification framework, separating commodity capability that flows into the vendor's platform from differentiating capability that stays yours, is the mechanism for getting this right upfront rather than disputing it after the fact.
- **What's the single biggest predictor of whether an FDE engagement succeeds?**
    - Based on the research in this report, it's whether the client has a named executive sponsor in place before the FDE arrives, someone with real standing to accelerate provisioning and remove internal blockers. Without one, engagements consistently stall in access requests before productive discovery ever begins, regardless of how good the vendor's engineers are.
- **Will we still need this vendor two years from now?**

- That depends entirely on whether the engagement was structured around the exit criteria in Chapter 6 and the knowledge-transfer discipline in Chapter 7. Structured correctly, the goal is operational independence, demonstrated, not assumed, before disengagement. Structured poorly, the answer is usually yes, and that's the exact dependency this report is designed to help you avoid.

Appendix: Research Methodology & Sources

Methodology

This report draws on three categories of evidence. First, primary and industry research from Gartner, RAND, MIT's Project NANDA, Deloitte, Capgemini, the World Economic Forum, Microsoft, and named regulatory bodies including NIST and Singapore's Infocomm Media Development Authority. Second, verified market and corporate announcements, including press releases and financial reporting on named vendor partnerships and joint ventures, cross-checked across multiple independent outlets before inclusion. Third, Aptly Technology's own delivery methodology and engagement experience, which informs the Aptly FDE Readiness & Value Framework introduced in Chapter 5.

Two Gartner research notes, "An AI Leader's Guide to Forward Deployed Engineering to Scale AI" (G00847066, published March 2026) and "Accelerate Agentic AI Adoption With Forward-Deployed Engineers" (G00844355, published April 2026), are licensed subscription content and are cited by document ID throughout this report rather than hyperlinked, consistent with Gartner's redistribution terms. McKinsey research is deliberately excluded from this report's source base throughout.

Every external statistic in this report is hyperlinked at the point it appears, directly on the claim it supports, rather than collected into a source list at the end of each section.

Primary Sources Referenced

Source	What This Report Draws From It
Gartner	FDE definition, skills framework, benefits/risk taxonomy, vendor-accountability guidance, and the 2024 AI Mandates for the Enterprise survey (G00847066, G00844355)
RAND Corporation	AI project failure-rate research
MIT Project NANDA	GenAI Divide: State of AI in Business 2025, pilot-to-production and buy-vs-build data
Deloitte	State of AI in the Enterprise 2026 survey; agentic AI governance-maturity research
Capgemini	Rise of Agentic AI enterprise adoption survey
World Economic Forum	AI talent shortage and skills-gap research
Microsoft	2026 Work Trend Index, Frontier Firm and Frontier Professional research
NIST	AI Agent Standards Initiative (primary federal source)
Singapore IMDA	Model AI Governance Framework for Agentic AI (primary government source)

Source	What This Report Draws From It
Perspective AI	State of Forward-Deployed Engineering 2026 survey; FDE compensation and hiring-trend research
a16z, The Pragmatic Engineer, FDE Academy	Definitional and origin-story research on the FDE role and its adoption
Accenture, Deloitte, AWS, Anthropic, OpenAI (corporate announcements)	Verified vendor market activity establishing current competitive landscape

About Aptly Technology

Aptly Technology helps middle-market and large enterprises move AI from pilot to production through Forward-Deployed AI Engineering Services, GPU datacenter buildout, and AI infrastructure managed services. Aptly is ISO/IEC 42001, ISO 27001, and SOC 2 certified, with delivery centers across North America, Europe, and Asia. To discuss where your organization sits against the readiness framework in this report, contact Aptly.