Invisible

FORWARD DEPLOYED ENGINEERING

How the hottest job in software radically speeds up time to value for AI

Table of contents

Executive summary

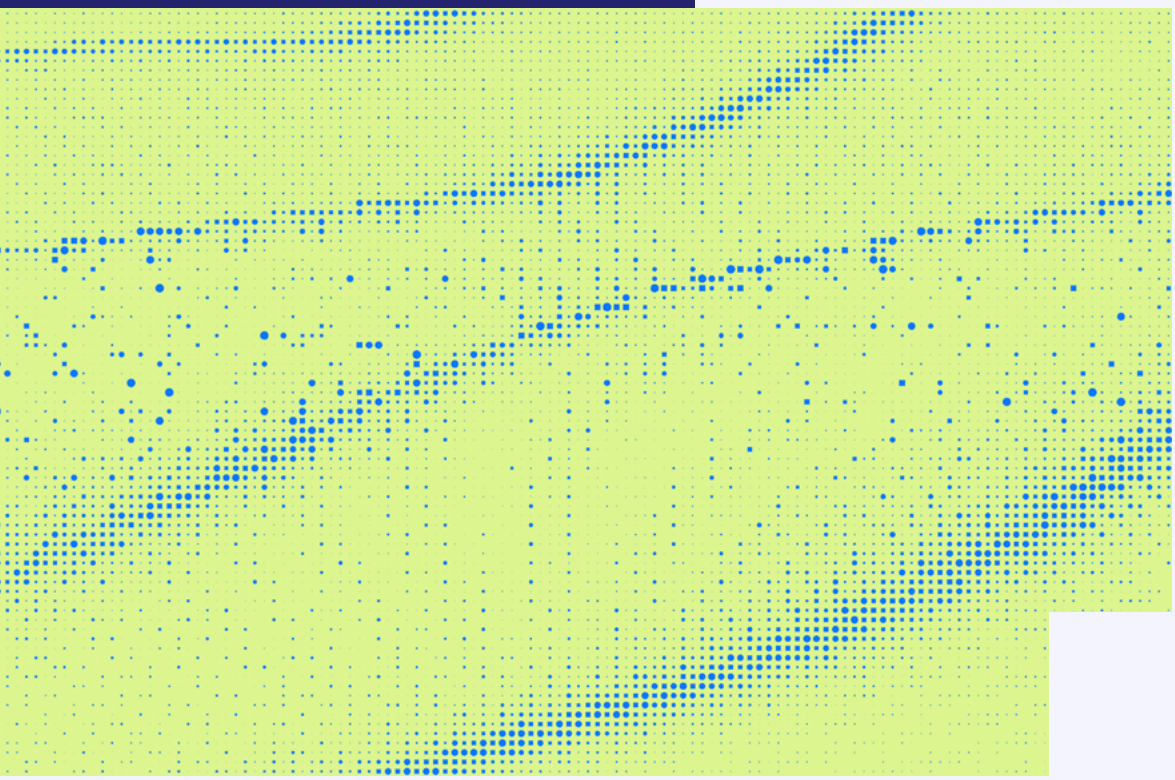
It's time to rip off the Band-Aid: MIT's finding that 95% of GenAI projects fail is too big a number to ignore. And with industry watchers like Gartner forecasting that 40% of agentic AI projects will be abandoned by 2027 over escalating costs and unclear business value, it's fair to ask: how can I stop my business from becoming another statistic?

Because AI's potential remains enormous: imagine enterprise software, customized exclusively for your business, specific to your needs and the nuances of how you operate. We're entering an era where software doesn't just serve the business – it becomes the business. Systems that learn, adapt, and reconfigure themselves around how your organization actually works are now within reach.

That vision is radically different to how the software industry used to work, which makes the old model of buying tools and bending them to fit look archaic. But if that's the dream, MIT laid bare the messy reality: AI implementations are hard. Just because an AI model performs well in tests doesn't mean the magic automatically happens for the business. And it's why an overwhelming majority of AI projects aren't living up to that promise – yet.

Because AI moves fast; the industry is changing, the capabilities of platforms and models are advancing week by week. And once you get past the generic tools for straightforward work, or pilots that only just scratch an itch to 'do' AI, things get a lot more complicated. As MIT's research found, AI projects quickly hit the buffers once you start introducing end-to-end workflows and actual business context.

Too often, AI initiatives are detached from the workflows, data, and decisions they're meant to improve. Without engineers close enough to understand how the work actually happens, even the best models remain abstract.



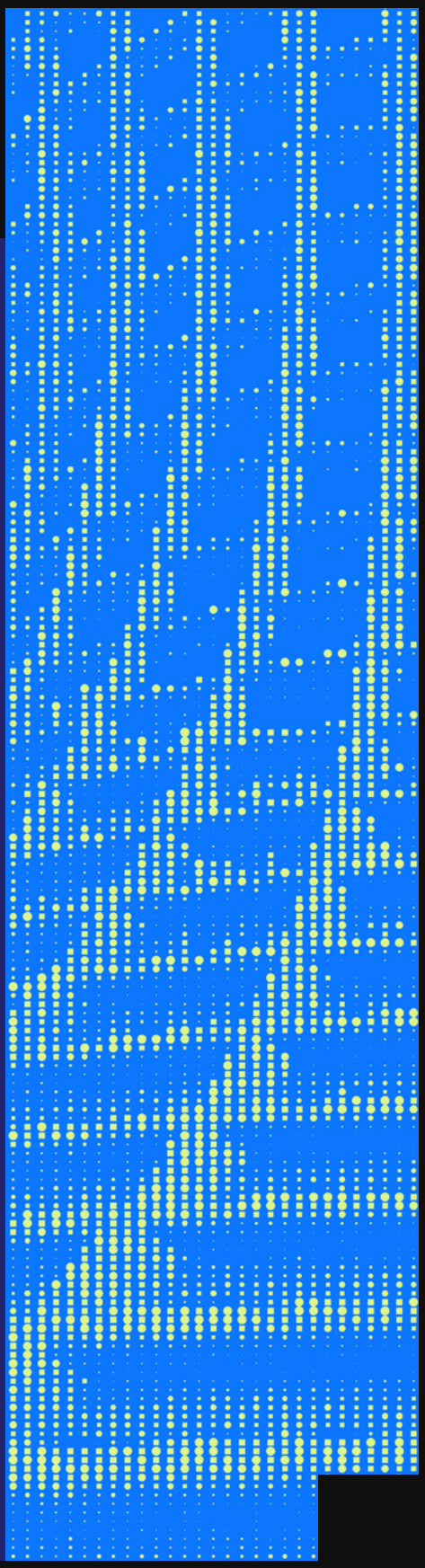
To make AI work in practice, proximity matters.

This helps explain why the concept of the forward deployed engineer is on the rise. The concept brings technical capability to the front lines of the business, where assumptions are tested, data takes shape, and theory becomes execution.

The forward deployed approach is about technical implementation that goes beyond configuration: by being close to the business problem, they're better able to understand them and build workflows on top of a next-gen AI platform to arrive at a business solution that actually delivers value.

But here's the secret: deploying FDEs isn't the end goal – it's the means to get you there.

In this paper, we'll shed light on what makes the FDE motion work, explain clearly what it does and how it differs from other approaches and engagement models. But we're also not losing sight of what matters. As far as a business is concerned, the goal is not to "deploy engineers" but to solve meaningful business challenges, quickly and sustainably, by integrating AI into their unique business logic and workflows.



The forward deployed motion

The term refers to stationing Forward Deployed Engineers (FDEs) close to the business problem to deliver solutions that are tailored yet are founded on a consistent platform. The term “forward deployed” has its origins in the military, where units would be stationed close to the front line for faster response and intelligence. The core idea is that proximity to the problem, allied to fidelity of information, leads to speed of solution.

It differs radically from the earliest days of enterprise systems and more recently with SaaS applications, where a business would acquire off-the-shelf tools and either hire a systems integrator or deploy internal teams to shape them, customize and configure them. In a world where powerful AI models hold the potential to create hyper-personalised systems, these approaches no longer work.

Palantir later popularized the Forward Deployed Software Engineer (FDSE) role. Distinct from consultants, FDEs are builders with the authority and autonomy to shape product direction. Fast forward to today, where FDEs are often found in data and AI companies, used where problems are too complex to ‘spec’ upfront and where learning with the client is the product.

The FDE model is highly collaborative, with builders on the front lines, working closely with clients in their operational and technological journey. These elite engineers combine deep technical fluency with the ability to operate inside the client’s business. They don’t hand off reports or deploy generic software. They stay close to the work, translating technology into systems that actually perform under real conditions.

The FDE model provides specialist expertise founded on these qualities:

- Builder mindset: Solves business problems by building solutions, not just strategies.
- Entrepreneurial: Can operate with autonomy, take ownership of the problem, and create solutions by connecting dots, piecing together existing solutions, and creating net new capabilities to fill any gaps.
- AI engineering: From leveraging AI coding assistants to rapidly build infrastructure and software, to a deep understanding of model behavior, prompting and context engineering.
- Collaborative: The FDE seeks to understand the client's business and technical architecture, and solve the problem together, not just deploy and leave.

> A pure engineer might not be capable of talking to the business. Or a consultant who doesn't code can't make quick changes. The success of this role is in the combination of skillsets: a very good 'jack of all trades,' who understands the client's business, our product, and can code to deploy."

Guillaume Esnou
Field CTO, Europe
Invisible Technologies

Is FDE just consulting by another name?

FDEs are the mechanism for delivering measurable business impact, not a "resource" that clients "buy". They aren't there just to execute or maintain a system. They're not contractors following a brief nor strategists handing down slide decks.

Forward deployment embeds technical builders directly into the problem space, so discovery and execution inform each other continuously. The engineers' ability to compress the "consult/translate/build" cycle, and their ability to understand both the business challenge and implement the technical solution, reduces handoffs and miscommunications.

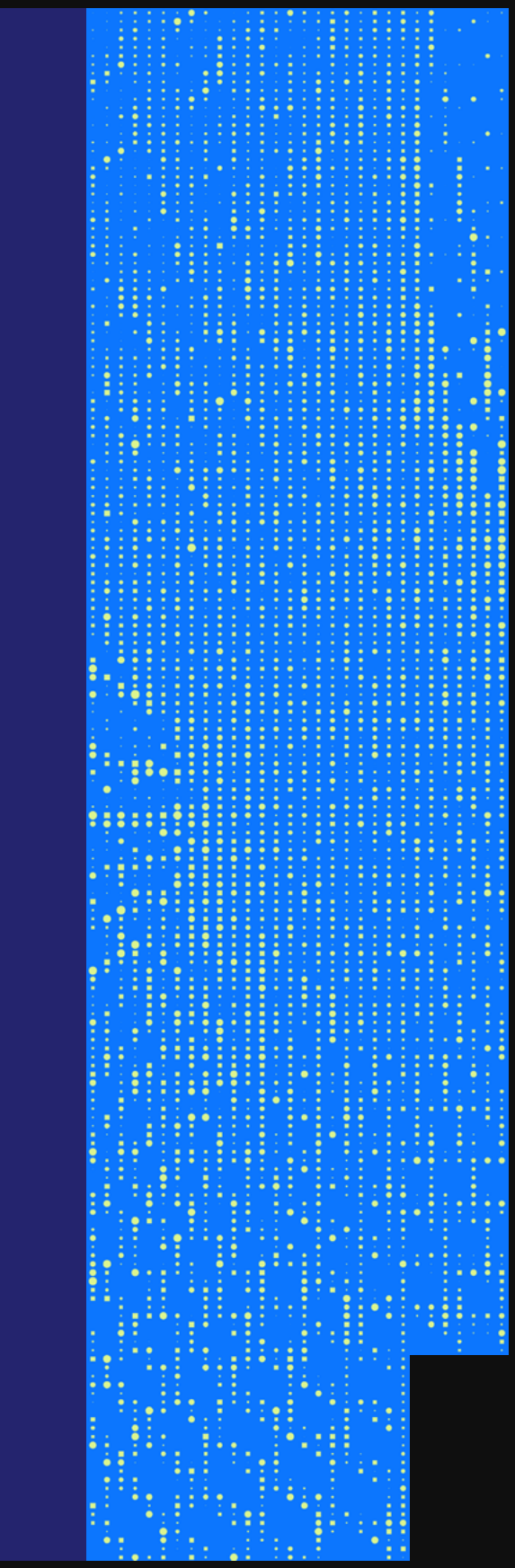
This model is not about being "closer to the customer" in a traditional service sense; in fact, the business model doesn't always involve service fees during the build phase. The cost to the client should be when value is delivered, not payment in billable hours on projects with unclear end goals.

FDEs adapt solutions to each client's context and maturity, and deliver rapid, highly tailored business value. The engagement doesn't end with a set of recommendations; FDEs leave behind an actual solution that the business can use, built on a foundational platform.

With new technologies like AI, traditional models lack the technical integration needed for modern, complex workflows, or organizations with highly customized, legacy, proprietary, or simply idiosyncratic systems. The forward deployment model addresses the challenge that many complex enterprises like this face in realizing tangible business value from advanced platforms and AI.

It does this by:

- Bridging the gap between standardized platforms and the messy, bespoke realities of each client's data, workflows, and decision processes
- Bringing expertise that is both technical and advisory, allowing for fast adaptation and value delivery
- Ensuring solutions are not just installed but actually work to meet real business needs, compressing the process from problem identification to delivered impact.



Why does this matter for AI?

Because AI isn't SaaS; you can't plug and play. The results are in: per MIT's report, The State of AI in 2025, a staggering 95% of organizations are getting zero return despite billions of dollars invested.[1] "Just 5% of integrated AI pilots are extracting millions in value, while the vast majority remain stuck with no measurable P&L impact. This divide does not seem to be driven by model quality or regulation, but seems to be determined by approach," the research concludes.

The reasons? "Most fail due to brittle workflows, lack of contextual learning, and misalignment with day-to-day operations." It's not that the models don't work in principle: it's that implementation in real environments is hard. Helpfully, the report also identifies what's working, and why: adaptation to context, and improvements over time, lead to success.

In AI projects today, client needs and workflows are often highly customized and complex, requiring solutions that can adapt to unique operational realities. Unlike legacy software where customization meant simple system integration, AI needs to be taught to understand the client's specific business logic, processes, and data flows. That's something generic, out-of-the-box platforms can't do.

[1] The GenAI Divide: State of AI in Business 2025, MIT NANDA

Forward deployment matters for AI because:

- AI requires much deeper and ongoing evaluation of how a model performs within workflows - and that needs FDEs who are deeply aware of the business goal
- Implementing AI successfully involves understanding client data, workflows, and goals at a granular level, and that's only possible with on-the-ground collaboration
- FDEs make AI "work for you where you are," adjusting to the client's maturity and context, accelerating adoption, and helping overcome high failure rates in AI projects by making AI work at scale for real use cases in the enterprise

This approach dramatically increases the success rate of AI initiatives compared to traditional models.

> The way a lot of IT enablement happened in the last 20 years is, SaaS companies used service engineers to bend the company to what the SaaS tool does. What FDEs do is bend the AI capabilities to what the customer actually needs, specific to their company and how they work and operate. The FDE model acknowledges that it's possible to create completely unique software for your problem, along with components to accelerate parts of your software creation: that's what FDEs do. They hyper-personalize a set of base capabilities very quickly using AI tooling, techniques and process to personalize the solution. I would almost call them AI engineers."
>
> **Aaron Bawcom**
> Field CTO
> Invisible Technologies

CHARLOTTE HORNETS

CUSTOMER SUCCESS STORY #1

A winning play

Like businesses, sports teams are always looking for a competitive edge – and data is often the lever. The Charlotte Hornets, an American professional basketball team, wanted to improve its analytical capabilities to more effectively evaluate talent from college and non-NBA leagues.

Historically, NBA teams have relied heavily on traditional scouting reports, subjective player interviews, and small sample sizes of performance. With the 2025 NBA draft approaching, Invisible assured the team it could deliver more detailed scouting data in a matter of weeks. Invisible's Computer Vision platform ingested raw game film and extracted detailed data about player movement to generate advanced performance metrics, giving a much richer, objective view of a player's athletic profile compared to traditional scouting.

"We were able to replicate an existing multi-camera high fidelity networked camera system that the NBA uses with single-point retail cameras and our AI analysis. We took those reusable components and used forward deployed engineers to bring in a unique custom solution in 9 days that led to a level of draft analytics that no-one's ever seen before. We integrated $5 million dollars of value within ten days," explains Aaron Bawcom, Field CTO at Invisible Technologies.

The commercial impact was almost immediate, says Patrick Harrel, VP of Basketball Insights & Analysis at the Charlotte Hornets. "Within weeks, Invisible provided us with an AI draft strategy that gifted us Kon Knueppel, the #4 overall pick by the Charlotte Hornets in the 2025 NBA draft. He was named MVP of the NBA Summer League Championship game after leading the Hornets to victory. He scored 21 points in the final game, helping the Hornets defeat the Sacramento Kings. This win also marked the Hornets' first-ever trophy and Summer League championship."

Photo via Charlotte Hornets

CUSTOMER SUCCESS STORY #2

The hidden value of consistency

A global beverage brand wanted to ensure its products appeared correctly across thousands of restaurant menus hosted on a leading point-of-sale platform. The goal seemed simple: make sure every instance on a digital menu actually represented the brand — but the reality was far messier.

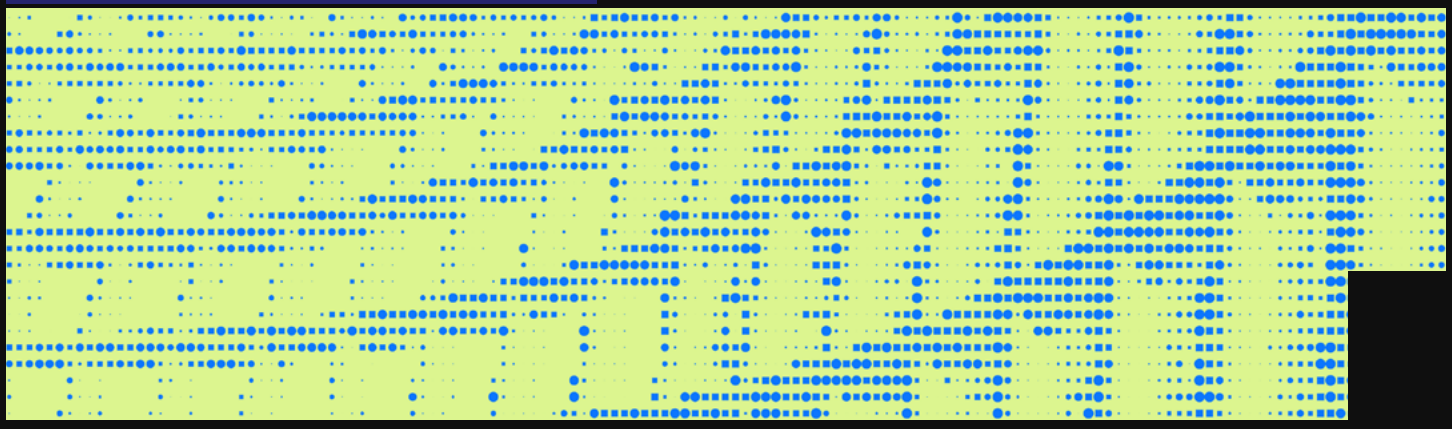
Each merchant described items in their own way, and there was no standardization or quality control. What looked like a small data problem was, in truth, a systems problem: an unstructured sprawl that diluted brand presence and left money on the table.

Invisible's forward-deployed engineers joined the platform team to help. Working deep in the data, they discovered that the brand's official catalog — the supposed "source of truth" — didn't fit the real world. It lacked contextual details that merchants used every day, like package size or format, which shaped how consumers made choices. Instead of forcing the business to conform to a static dataset, the team built something new: a dynamic taxonomy that learned from how the market actually behaved. They combined AI-based classification with human-in-the-loop validation, layering the brand's catalog with live menu data and real consumer signals.

Within weeks, the new system began standardizing product listings at scale. The beverage brand regained control of its presentation across thousands of digital menus without constraining merchant flexibility. Sales improved — a 5% lift for merchants carrying the product line — and the platform gained an enduring capability: a living, self-correcting data model that adapts as menus change.

The result wasn't just better data hygiene. A traditional consulting team might have produced a technically correct but contextually blind solution — what was asked for, not what was needed. It was a proof point for the forward-deployment model itself — embedding technical talent directly in the problem space to convert operational noise into structured advantage.

> "FDE is a way of working more than just a job title. It's a style of delivering value for our customers that combines deep technical expertise with a relentless focus on real-world business problems. We assemble diverse teams with skill sets that match each project's unique needs, but every team member brings the same client-centric mindset supercharged with the Invisible AI platform at their fingertips."
>
> **Joe Revier**
> Dir. Forward Deployed Engineering
> Invisible Technologies



The strategic payoff for enterprises

Faster time-to-value and reduced implementation risk

Using AI as an accelerator, the FDE model quickly adapts reusable components for a client's exact needs. In some cases, this delivers value in less than two weeks. The commercial model is outcome-based: clients only pay when value is produced. This aligns incentives and means enterprises aren't at risk of large, up-front spend with uncertain outcomes.
FDEs are sometimes described as "human software", and their integration only triggers payment when results materialize.

Scalable adoption across business units and functions

FDEs can rapidly personalize solutions for different problems and environments by leveraging a set of base AI-driven capabilities, supporting deployment across diverse enterprise units or markets. The analogy of not thinking small and aiming to "swing for the fences" reflects the model's suitability for widespread, high-impact transformation.

Feedback loops where client builds inform platform improvements

The model leads to tight client-FDE collaboration, since FDEs are embedded in customer environments and focused on rapid delivery and iteration. This client-centric, iterative approach provides ongoing course corrections and responsiveness to client needs.

Institutional knowledge is captured, not lost in decks

FDEs personalize and "bend" technology to customer processes, rather than forcing process changes to fit the software.
This approach ensures enterprise-specific knowledge, workflows, best practices – and, yes, nuances – are retained and enhanced, not lost or overridden, improving adoption and long-term value.

Forward deployment in practice: what to know

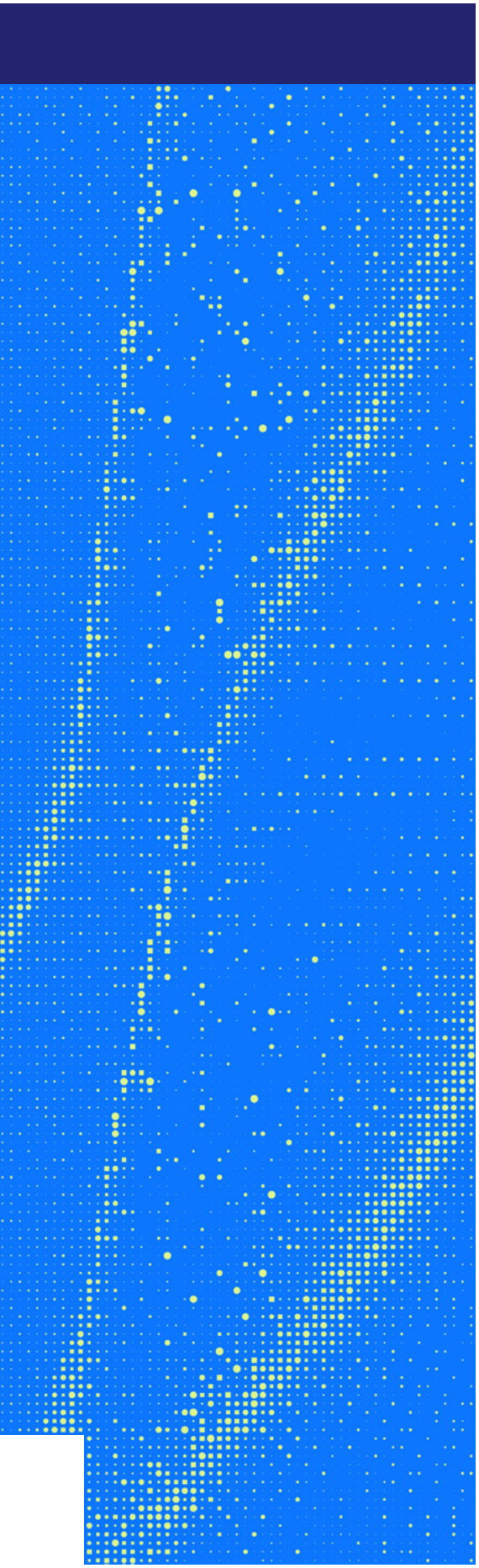
In the FDE model, success is defined by moving clients from confusion and chaos towards delivered results – turning the "95% failure rate" statistic in AI to success. And by success, we mean outcomes clients care about: solving real, high-value business challenges. But there are a few things worth knowing about how the model works in practice.

- The first solution (v1) is not meant to be perfect; it's a working prototype that will evolve quickly based on client feedback.
- This way, clients see visible results early, can rapidly shape the solution to their unique needs, and are embedded in a collaborative process where their insights drive outcomes.
- Unlike large IT projects of old, where results only came at the end, FDE encourages incremental improvement and learning at every step, thus reducing risk and misalignment.
- Each iteration builds on previous work, so the end solution is fine-tuned through real usage and direct input, leading to a more tailored, effective outcome.

A new default

Forward-deployment is no longer niche – it's becoming table stakes in AI-heavy domains. It's the new operating system for how modern enterprises should build. There are several benefits and advantages of the FDE motion:

- Rapid delivery: faster proofs of concept with less upfront resource investment – often producing a version 1 within weeks after initial client meetings. This speed reduces both financial and human resource investments needed to demonstrate value and tailor solutions
- Highly iterative: solutions are developed through rapid cycles of client feedback and continual refinement, always in the knowledge that the first versions won't be perfect
- Embedded nature: FDEs work closely with client operations teams to understand nuanced business requirements, which leads to more tailored and valuable solutions
- Goodbye to drawn-out IT projects: Compared with more traditional models, the FDE motion emphasizes delivering quick, incremental improvements rather than long, unproductive engagements
- Nuance and knowledge capture: Proximity to the client allows FDEs to capture tacit knowledge and institutional expertise essential for building effective models and processes
- True tailoring: Solutions are customized rather than off-the-shelf, making FDEs truly 'embedded' partners rather than distant consultants.



Reverse gear: when not to go forward

There might be growing industry buzz about the forward deployed model, but as this paper has hopefully made clear, it's best suited to complex organizations and business challenges where systems, processes, or data are so idiosyncratic that generic software can't capture them.
In other words, it's not made for every situation, every time.

Here are **five scenarios** when forward deployed isn't the right way to go.

When the problem doesn't require deep integration

Forward deployment creates value in complex, idiosyncratic environments. If your systems are standard, API-ready, or easily configured, embedded engineers add unnecessary cost. In these cases, investing in product enablement or self-service tooling delivers faster and cheaper outcomes than field deployment.

When business conditions shift too quickly

FDEs work best in stable operational contexts. If your data, processes, or priorities change constantly, the code and integrations FDEs build lose relevance quickly. For fast-evolving startups or dynamic enterprises, flexible platform tools or internal development capacity are a better fit.

When you can't absorb or maintain what's built

FDEs should leave behind capability, not dependency. If your organization lacks internal teams to maintain and extend deployed solutions, the investment fades as soon as engineers leave. Without operational ownership or data governance, embedded work becomes shelfware instead of leverage.

When economics don't justify high-touch work

FDE engagements make sense only when the problem is mission-critical and the contract value is high. For smaller budgets or non-core use cases, the cost of embedding engineers outweighs the benefit. Standard SaaS offerings or light professional services are more efficient alternatives.

When the goal is bespoke consulting, not product evolution

As you'll have seen so far, forward deployed is very different from a classic consulting engagement. But organizations expecting tailor-made systems rather than contributing to a scalable product will find the relationship drift into consulting. FDEs are valuable only when their learnings feed back into the core platform. When that feedback loop is missing, both sides lose scalability and long-term value.

Invisible

The model for making enterprise AI work

In the era of AI-enabled enterprise systems, forward deployed engineers are more than a niche deployment tactic: they're a structural lever. Where the core challenge is not building a model, but integrating and operating it in real, messy organizations, embedding engineering capabilities with business problems delivers a strategic advantage.

One final thought: when it comes to implementing the FDE model, businesses need strategic clarity and a focus on high-value problems – that's where the greatest potential for value, and significant market differentiation, lies.

If you're ready to bet big on AI, contact us today to learn how our FDE model can help deliver measurable business value.

GET IN TOUCH