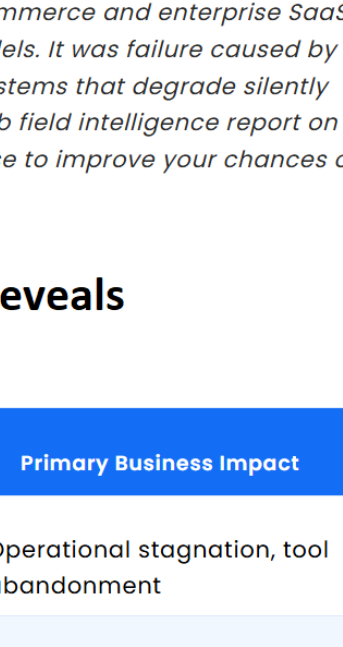


# Radixweb's Field Intelligence Report on AI Failure [2026]: Where AI Fails in Real-World Systems

Unlock real insights on the five failure zones where AI projects tank in production, the real-world outcomes behind them, and the structural fixes that prevent them.



## What's Inside: We surveyed 75+ practitioners actively running AI in production across industries, from fintech and healthcare to eCommerce and enterprise SaaS. What they reported wasn't failure caused by bad models. It was failure caused by workflow gaps, dirty data, absent governance, and systems that degrade silently while everyone assumes they're working. The Radixweb field intelligence report on AI Failure surfaces actionable insights that you can use to improve your chances of successful AI deployment.

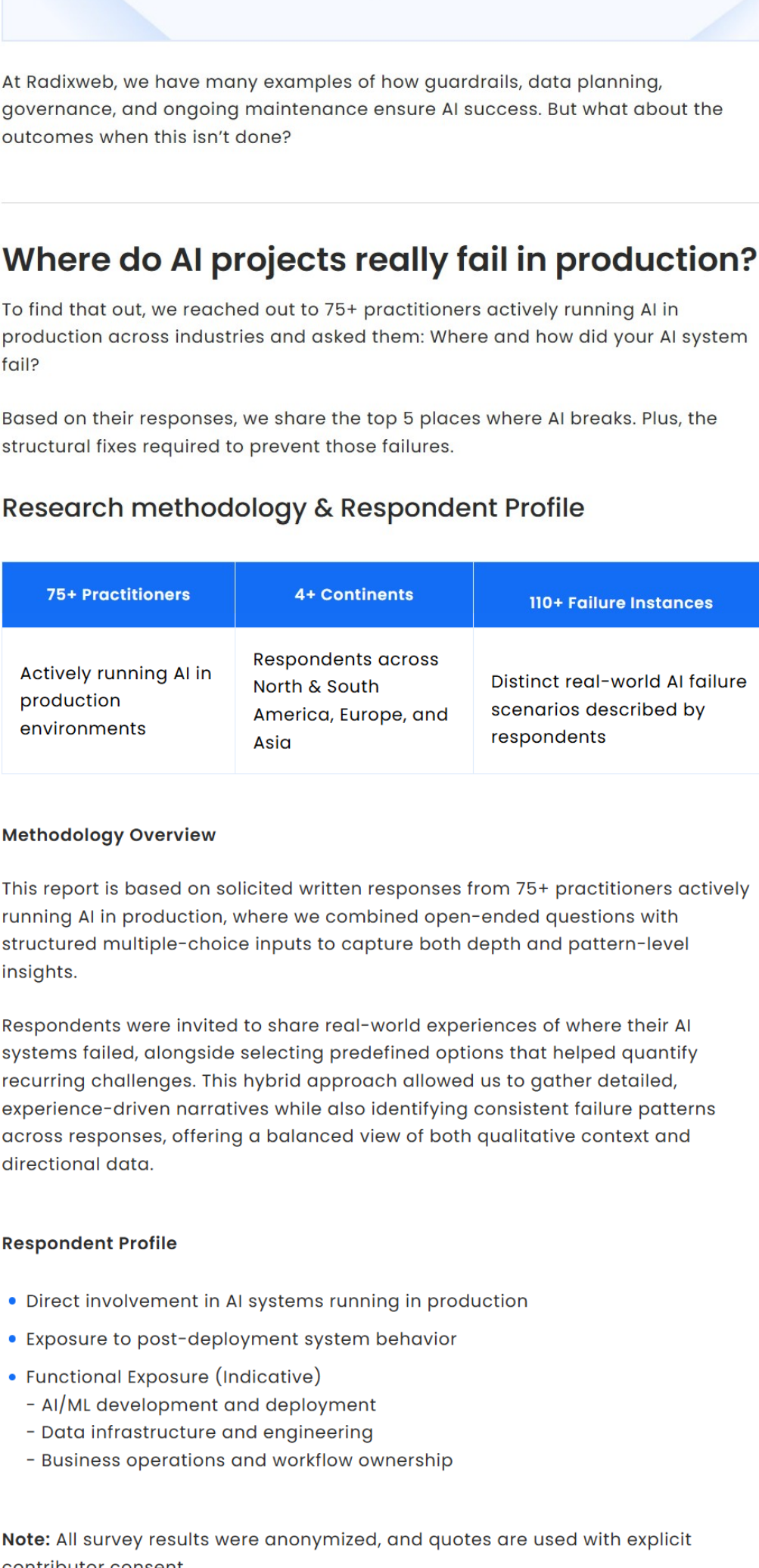
## TL;DR: What The Data Actually Reveals

A snapshot of where AI systems actually break

Failure Zone	Share of Total Citations	Primary Business Impact
Workflow Integration Failure	31.2%	Operational stagnation, tool abandonment
Data Quality & Cleanliness	22.1%	Output errors, cascading inaccuracies
Model Inaccuracy & Trust Deficit	15.6%	Project abandonment, shadow workflows
AI as One-Time Project (No Maintenance)	14.3%	Silent model decay, stale predictions
Silent Failure / No Monitoring	10.4%	Undetected risk, customer-facing errors
Technical Model Drift	6.5%	Secondary contributor, rarely root cause

- Other key findings of our field intelligence report:
- 31% of AI failures come from workflow integration gaps—not model issues
  - Just 6% of failures are caused by technical model drift
  - Over 50% of teams rely on ad-hoc or no AI maintenance at all
  - Nearly 1 in 3 AI systems are used optionally, not operationally
  - 35% of teams experience parallel workflows due to AI distrust
  - Only 18% have continuous monitoring + retraining in place
  - Up to 20% of failures reach customers before being detected
  - Organizations overestimate AI adoption vs real usage by a significant margin (~25–30%)
  - Data quality issues contribute to over 1 in 5 AI failures
  - Lack of lifecycle planning affects ~15% of systems directly
  - Monitoring gaps account for ~10–12% of critical failures

Let's dive into these metrics in detail now.



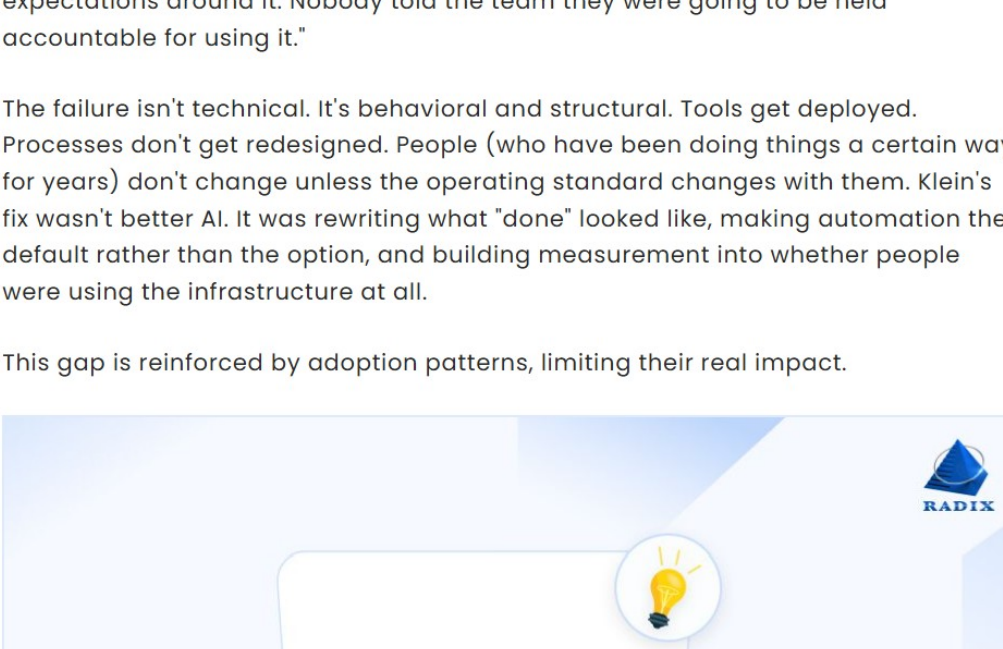
## Why Does AI Failure Need to Be Understood?

92% of companies are planning to increase their AI budgets in 2026. Yet only 8% of AI implementors see tangible ROI within the first year.

Why? Because the last mile of AI implementation is filled with invisible tripwires, the things that don't show up in demos, but break systems in production. But most people aren't talking about the reasons behind AI failure in public.

Pratik Mistry, EVP of Technology Consulting at Radixweb, explains, "We mostly hear success stories. And that success-washing shows clearly when I speak with teams evaluating or scaling AI and they ask me: What difference will AI guardrails actually make or is continuous model monitoring really necessary."

He further explains....



At Radixweb, we have many examples of how guardrails, data planning, governance, and ongoing maintenance ensure AI success. But what about the outcomes when this isn't done?

## Where do AI projects really fail in production?

To find that out, we reached out to 75+ practitioners actively running AI in production across industries and asked them: Where and how did your AI system fail?

Based on their responses, we share the top 5 pieces where AI breaks. Plus, the structural fixes required to prevent future failures.

### Research methodology & Respondent Profile

75+ Practitioners	4+ Continents	10+ Failure Instances
Actively running AI in production environments	Respondents across North & South America, Europe, and Asia	Distinct real-world AI failure scenarios described by respondents

### Methodology Overview

This report is based on solicited written responses from 75+ practitioners actively running AI in production, where we combined open-ended questions with structured multiple-choice inputs to capture both depth and pattern-level insights.

Respondents were invited to share real-world experiences of where their AI systems failed, alongside selecting predefined options that helped quantify recurring challenges. This hybrid approach allowed us to gather detailed, experience-driven narratives while also identifying consistent failure patterns across responses, offering a balanced view of both qualitative context and directional data.

### Respondent Profile

- Direct involvement in AI systems running in production
- Exposure to post-deployment system behavior
- Functional Exposure (Indicative)
  - AI/ML development and deployment
  - Data infrastructure and engineering
  - Business operations and workflow ownership

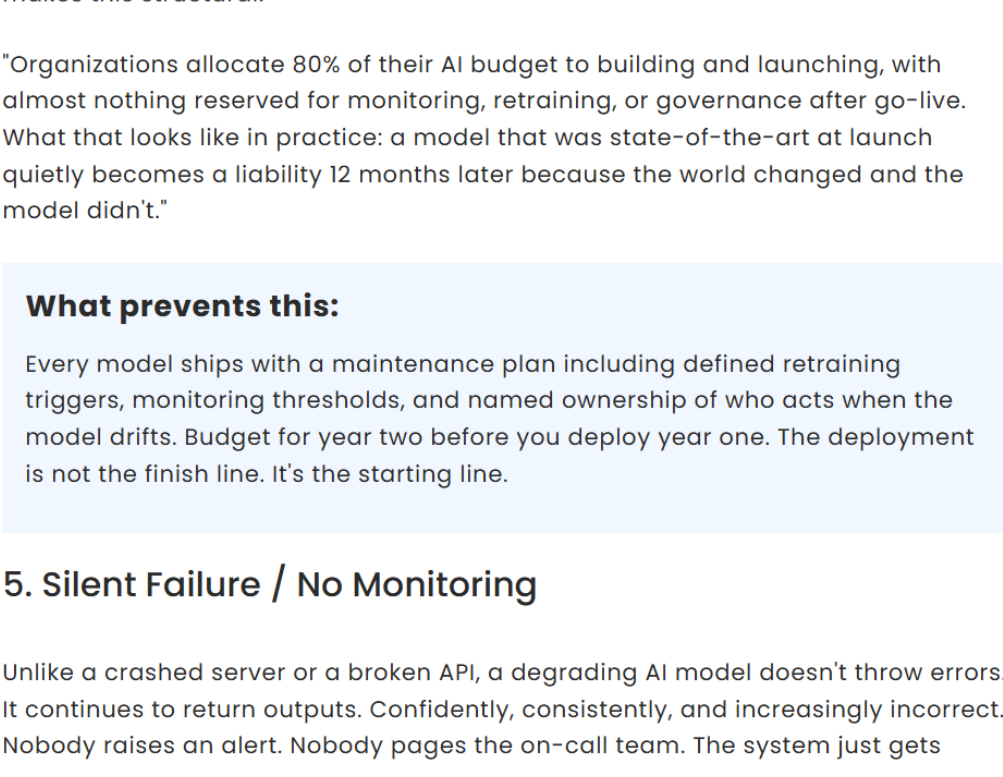
Note: All survey results were anonymized, and quotes are used with explicit contributor consent.

With this dataset in place, the next step was to identify where failures were most consistently occurring across systems.

## The Landscape of AI Failure: The 5 Areas Where AI Fails

Certain AI failure zones showed up far more consistently in the responses we received. What follows is an explanation of where experts are actually encountering breakdowns in production-grade AI systems.

Across the responses analyzed, these five zones account for over 95% of all observed AI failures in production environments.



### 1. Workflow Integration Failure

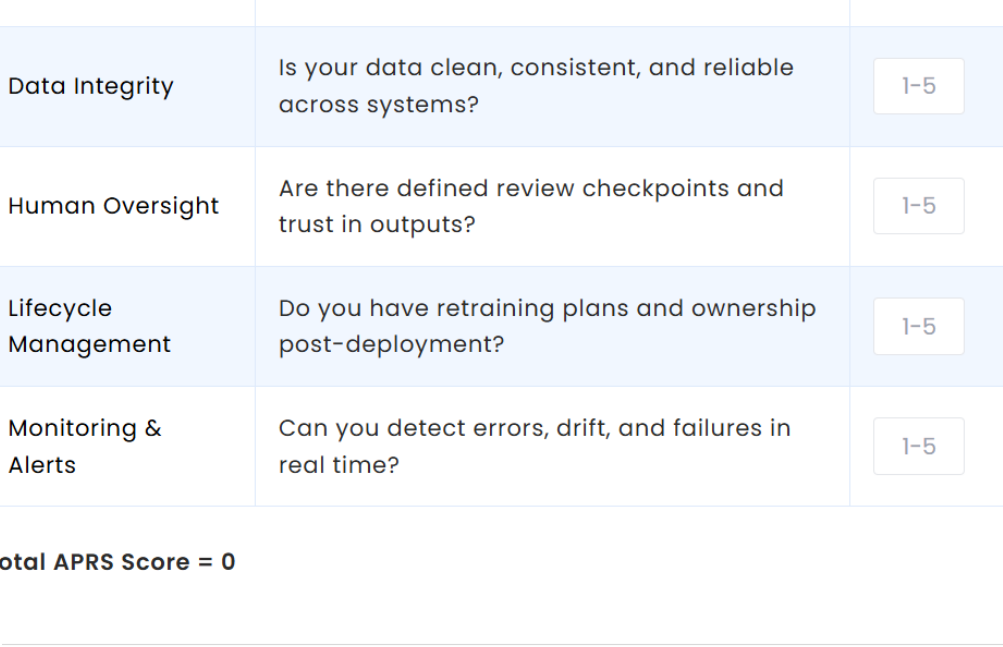
Accounting for 31% of all reported failures, workflow integration is the single largest breakdown point in production AI systems. And it's the one that surprises leadership the most. The model works, but the difference happens. The ROI projections look reasonable. And then the team keeps doing the work by hand.

Ryan Klein, CEO of a digital marketing firm that deployed AI across 222 client websites, describes it with unusual precision:

"We built a fully operational AI agent that could execute sitewide SEO fundamentals across 222 client websites. Then we watched the team do the same work by hand, because nobody integrated the tool into their workflows. Nobody set expectations around it. Nobody told the team they were going to be held accountable for using it."

The failure isn't technical. It's behavioral and structural. Tools get deployed. Processes don't get redesigned. People (who have been doing things a certain way for years) don't change unless the operating standard changes with them. Their key fix wasn't better AI. It was rewriting what "done" looked like, making automation the default rather than the option, and building measurement into whether people were using the infrastructure at all.

This gap is reinforced by adoption patterns, limiting their real impact.



Antonella Serino, CEO of KLA Digital with a decade of AI governance experience, frames the systemic version of this problem:

"What works in a clean demo breaks when it hits live operations. AI does not fail first because of benchmark quality. It fails because real business systems are full of permissions, handoffs, exceptions, audit requirements, and ambiguous ownership."

In high-separation-of-duties environments, this manifests as an ownership gap: the data science team builds and validates the model, then hands it off to an operations team that didn't build it and can't diagnose it when something downstream shifts. When that happens and when a policy changes, a data source moves, a workflow evolves, the model starts failing. Nobody knows why, nobody owns the answer.

Alix Gallardo, CPO at Invent (working in fintech and asset/gary operations), surfaces the version of this that hits regulated industries especially hard:

"In a review process we automated, our AI vs. manual comparison showed surprising variance. After getting on calls with reviewers, we discovered each person was applying their own formulas and judgment, so there wasn't a single ground truth for the AI to learn or replicate."

There was no workflow to integrate into. The workflow itself was inconsistent. The AI exposed that inconsistency at scale. The result is duplication of effort—35% of teams report running parallel workflows due to lack of trust or integration gaps.

#### What prevents this:

Map the actual process the AI will live inside before writing a line of code. Who triggers it? Who reviews the output? What happens when it gets something wrong? That mapping work isn't overhead but the difference between a model that sits unused and one that changes how a team operates.

### 2. Data Quality & Cleanliness

Data-related issues account for 22% of all AI failures, making this the second most common breakdown point.

Real business data is a mess. Invoices have inconsistent formats. CRM fields are filled in differently by different team members. The same product has four names across four legacy databases. Notes fields are used religiously by some and never by others.

Production AI has to work with those conditions. Not the clean, structured dataset prepared for the demo.

Kevin A. Thomas, a CFA who founded an AI-powered bookkeeping service, encountered the gap directly when his tools met real client data:

"Getting 98% accuracy in a controlled setting is easy. But the moment you incorporate it into your company's ledger systems, accuracy could fall to 15–20%, requiring far more human intervention than anticipated."

That gap (from 98% in testing to 15% in production) is not a model problem. It's a data problem. And it compounds. Wrong outputs erode trust, triggering a manual parallel processes, which eventually causes tool abandonment.

Ryan Klein's team at Market My Market manages over 75 million records across a dozen connected systems. Their experience with data quality failure is worth reading in full:

"We've had integrations that silently stored the wrong identifier in the wrong database, which meant every downstream report was matching with the wrong data. We've had jobs scheduled weekly for months that marked themselves successful while creating every day errors. The team mentally filed the AI as 'unreliable,' builds parallel processes, and now the organization is paying for both."

The constraint isn't compute. It's the integrity of the data layer and every inconsistency in that layer gets exposed by AI faster and at larger scale than manual processes ever revealed.

But the data quality problem isn't just about cleaning data before deployment. It's about controlling how data structures change over time, which is something most teams dramatically underinvest in.

#### What prevents this:

Don't start with the model. Start with a data audit and see where your data lives, how it flows, and where it breaks. Data architecture is the first deliverable in any production-grade AI engagement. The model comes second.

### 3. Model Inaccuracy & Trust Deficit

Imagine this: the AI tool is deployed, usage is reported, leadership is satisfied. But one layer deeper, the people actually doing the work stopped trusting the outputs weeks ago.

That's not just imagination, but actually what is happening. Our field intelligence report on AI failures shows that model-related issues account for ~16% of failures. But these are often a downstream effect of data or workflow gaps rather than standalone technical problems.

Why? Usually because of something small. One wrong output that a team member caught. One recommendation that didn't match their domain knowledge. One edge case the model handled badly. That's enough. The team mentally files the AI as "unreliable," builds parallel processes, and now the organization is paying for both.

Alli Hayat, CEO of Axipiro Technology, names the version of this that's nearly invisible at the leadership level:

"When staff don't trust AI outputs, they quietly work around them while leadership believes automation is running smoothly. The gap between reported AI adoption and actual AI usage inside organizations is significant and mostly invisible."

The more dangerous version plays out in customer-facing environments. Faster responses that are lower quality don't retain customers — they erode trust. And eroded trust is worse than no tool at all.

Tanner Crow, who designs AP Computer Science exam prep materials and understands exactly where AI-generated content falls in high-precision domains, makes a counterintuitive point that applies for beyond education:

"Companies often think AI reduces the need for expert review, but in practice it increases the need for it when the content must be accurate, challenging, and trustworthy."

More AI capability doesn't reduce the human oversight requirement. In domains where accuracy matters, it raises it. The companies that don't account for that end up with a false confidence problem: AI outputs that look correct but aren't, reviewed by people who were told they wouldn't need to review as much.

#### What prevents this:

Build review mechanisms into the workflow and include defined checkpoints where humans see the AI's reasoning, not just its answer. Transparency in how a decision was reached is what converts skeptics into users. Adoption reviews alongside technical reviews, before deployment, not after the first month of disappointing usage numbers.

### 4. AI as a One-Time Project

Lifecycle neglect contributes to 14% of AI failures, often leading to silent degradation over time.

The pressure to adopt AI is at an all-time high. The incentive, in many organizations, is to ship the model and move on to the next initiative. Six months later, the data pipeline has changed, the model is making stale predictions, and nobody notices until a business stakeholder asks why the numbers feel off.

Pragati Awasthi, an applied ML practitioner and Drexel University professor teaching production AI systems, calls this the "launch and abandon" pattern:

"There's no alert, no owner, no retraining schedule. The model didn't fail — the process around it did."

This is reinforced by the operational reality that over 50% of teams rely on ad-hoc or no AI maintenance at all. Pragati further talks about the budget reality that makes this structural:

"Organizations allocate 80% of their AI budget to building and launching, with almost nothing reserved for monitoring, retraining, or governance after go-live. What that looks like in practice, a model that was state-of-the-art at launch quietly becomes a liability 12 months later because the world changed and the model didn't."

#### What prevents this:

Every model ships with a maintenance plan including defined retraining triggers, monitoring thresholds, and named ownership of who acts when the model drifts. Budget for year two before you deploy year one. The deployment is not the finish line. It's the starting line.

### 5. Silent Failure / No Monitoring

Unlike a crashed server or a broken API, a degrading AI model doesn't throw errors. It continues to return outputs. Confidently, consistently, and increasingly incorrect. Nobody raises an alert. Nobody pages the on-call team. The system just gets worse.

So, while monitoring gaps account for just 10.4% of failures, they represent some of the highest-risk breakdowns in production AI systems.

Christopher Coussons, Director at Visionary Marketing, describes how this failure landed for a client's product categorization system:

"Accuracy dropped to roughly 68% over three weeks, but because there was no monitoring layer, nobody noticed until an customer reported finding hiking boots in the formal shoes category."

This isn't a one-off example either. In practice, lack of monitoring leads to delayed detection with up to 20% practitioners mentioning that AI failures reached customers before being identified internally.

Algars Pilmanis, whose options analytics platform at VolRoad runs AI-powered signal generation against live market data daily, experienced the high-stakes version of this:

"A model trained on four years of options data showed 91% accuracy in backtesting. In production, during the 2022 rate-shock regime, it failed precisely when traders needed it most — generating high-confidence signals on patterns that no longer existed."

The world changed faster than the training distribution. The model didn't drift because of bad engineering. It drifted because reality moved and nothing in the system caught it. Despite this risk, only ~18% of surveyed teams had continuous monitoring and retraining systems in place.

#### What prevents this:

Instrument every model with confidence scoring and drift detection from day one. Build statistical checks that compare live data distributions against training baselines. When outputs fall below defined thresholds, the system flags them for human review, so you find out before your customers do.

Taken together, these five failure zones explain over 94% of observed AI breakdowns, with technical issues contributing just ~6%, the lowest among all categories. This counters the common "better model = more successful AI" hypothesis that's floating in the market, which raises a more practical question: if the model isn't the problem, how do you know if your own system is at risk?

To make these insights actionable, we developed a simple diagnostic model.

## The Radixweb AI Production Readiness Score (APRS)

The Radixweb APRS is a simple diagnostic tool designed to help you assess how prepared your AI system is for real-world performance. Based on the most common failure zones identified, it translates common breakdown risks into a structured scoring model, so you can identify weaknesses early, prioritize fixes, and improve your chances of successful AI deployment.

### Score Your System

Rate each area on a scale of 1 (weak) to 5 (strong) based on your current setup:

Areas	What You're Evaluating	Score (1–5)
Workflow Integration	Is AI embedded into real workflows with clear ownership?	1–5
Data Integrity	Is your data clean, consistent, and reliable across systems?	1–5
Human Oversight	Are there defined review checkpoints and trust in outputs?	1–5
Lifecycle Management	Do you have retraining plans and ownership post-deployment?	1–5
Monitoring & Alerts	Can you detect errors, drift, and failures in real time?	1–5

**Total APRS Score = 0**

### Interpret Your Readiness

Score Range	Readiness Level	What It Means
20 – 25	Production-Ready	Strong system, ready to scale
15 – 19	Fragile	Works now, but risks under scale
10 – 14	High Risk	Structural gaps likely to cause failure
Below 10	Failure Likely	System is unreliable or unsustainable



## What Happens Next: Questions Teams Ask Before Scaling AI

Understanding where AI fails is only part of the equation. The next step is knowing how to make the right decisions before those failures occur.

In conversations with teams evaluating or scaling AI, a consistent set of questions comes up. Questions that reflect the exact risks highlighted in this report.

Here are the five most common ones we hear, along with our direct responses.

### 1. Is it worth starting an AI project when internal data is fragmented and messy?

It is absolutely worth starting, but you must start with the data architecture, not the model. Most "AI failures" are actually "Data Failures" in disguise. We help organizations build a "Data Strategy" that cleans, siloes, and structures information before the AI is even turned on. This ensures that when you build and deploy your enterprise AI project, the AI has a reliable foundation to learn from, preventing the "garbage in, garbage out" cycle that kills pilots.

### 2. How do we know Radixweb isn't just giving us a generic 'wrapper' around a public API like ChatGPT?

A "wrapper" is a surface-level interface, while a Radixweb solution is a foundational integration. We offer pure gen AI solutions like custom RAG (Retrieval-Augmented Generation) pipelines and secure API connectors that ensure your AI is grounded in your specific business logic and your private data, not generic internet data. Our work focuses on the security layers, the monitoring frameworks, and the custom reasoning engines that make the AI a proprietary asset for your company, rather than a generic utility.

### 3. Won't implementing all these 'guardrails' and 'monitoring layers' significantly slow down our time-to-market?

It might feel that way in week one, but it saves months of "re-work" in year one. Building without guardrails inevitably leads to a point where your pilot fails to scale and you have to rewrite the entire codebase to handle real-world complexity. Our Reliability Framework is modular though and we "strip" those guardrails into your development process, allowing you to build fast and not safe, rather than having to choose between the two.

### 4. Does your team actually understand our specific industry logic, or do you just offer one-size-fits-all solution?

We never build in a vacuum. Every Radixweb engagement starts with a domain immersion phase. We don't use cookie-cutter AI modules, but help you map and map AI application across your specific business processes, regulatory requirements, and tacit knowledge (the rules that live in your experts' heads). We understand that the implementation of artificial intelligence in healthcare requires a completely different governance structure than AI in eCommerce, and we architect accordingly.

### 5. How do I justify the ongoing maintenance and retraining costs of AI to a board that expects a 'one-time' software cost?

The best way to frame this is as "Model Insurance." You wouldn't buy a fleet of delivery trucks and expect them to run for five years without an oil change, right? AI maintenance is the "oil change" that prevents your \$500k investment from becoming a \$0 asset due to data drift and market shifts. We help you build a TCO (Total Cost of Ownership) model that proves ongoing maintenance is significantly cheaper than the cost of a failed, "stale" system.

These questions reflect a broader shift from experimenting with AI to operationalizing it reliably. And that shift requires a different way of thinking about AI systems.

## Building Failproof AI in 2026 and Beyond

Across all responses analyzed, one pattern becomes clear: AI failure is largely driven by model limitations. It is driven by systemic gaps in how AI is integrated, governed, monitored, and maintained in production environments.

These gaps are not always visible during early deployment. But over time, they compound, impacting output quality, slowing adoption, and ultimately eroding ROI.

As Dharmesh Acharya, COO of Radixweb, puts it...



For real AI success that goes beyond performative demos, the focus needs to be on what we call the boring infrastructure like:

- Data cleanliness
- Observability pipelines
- Workflow integration
- Organizational change management
- Post deployment system support

These elements are often overlooked in early-stage AI initiatives. Yet they are the primary determinants of whether an AI system sustains performance under real-world conditions.

As AI adoption accelerates into 2026 and beyond, the competitive advantage will not come from access to better models. It will come from the ability to operationalize those models reliably over time.

Organizations that invest in building complete AI systems (not just deploying models) will be better positioned to scale, adapt, and realize long-term value from their AI investments.