avpcap.com

The models are ready. Now comes the hard part.

Ethan Volk

Bringing perspectives and investing in tech
from both sides of the Atlantic

Everyone has an AI agent in 2026. Twelve months ago, most enterprises were still running experimental pilots. Today, pilots have become roadmaps, roadmaps have become budgets, and a growing share of those budgets is now funding production deployments. Agentic AI is showing up in customer-facing products, internal tooling, and critical business workflows. The teams building and deploying agents, however, are finding that a working demo and a production system are vastly different things.

The Production Problem

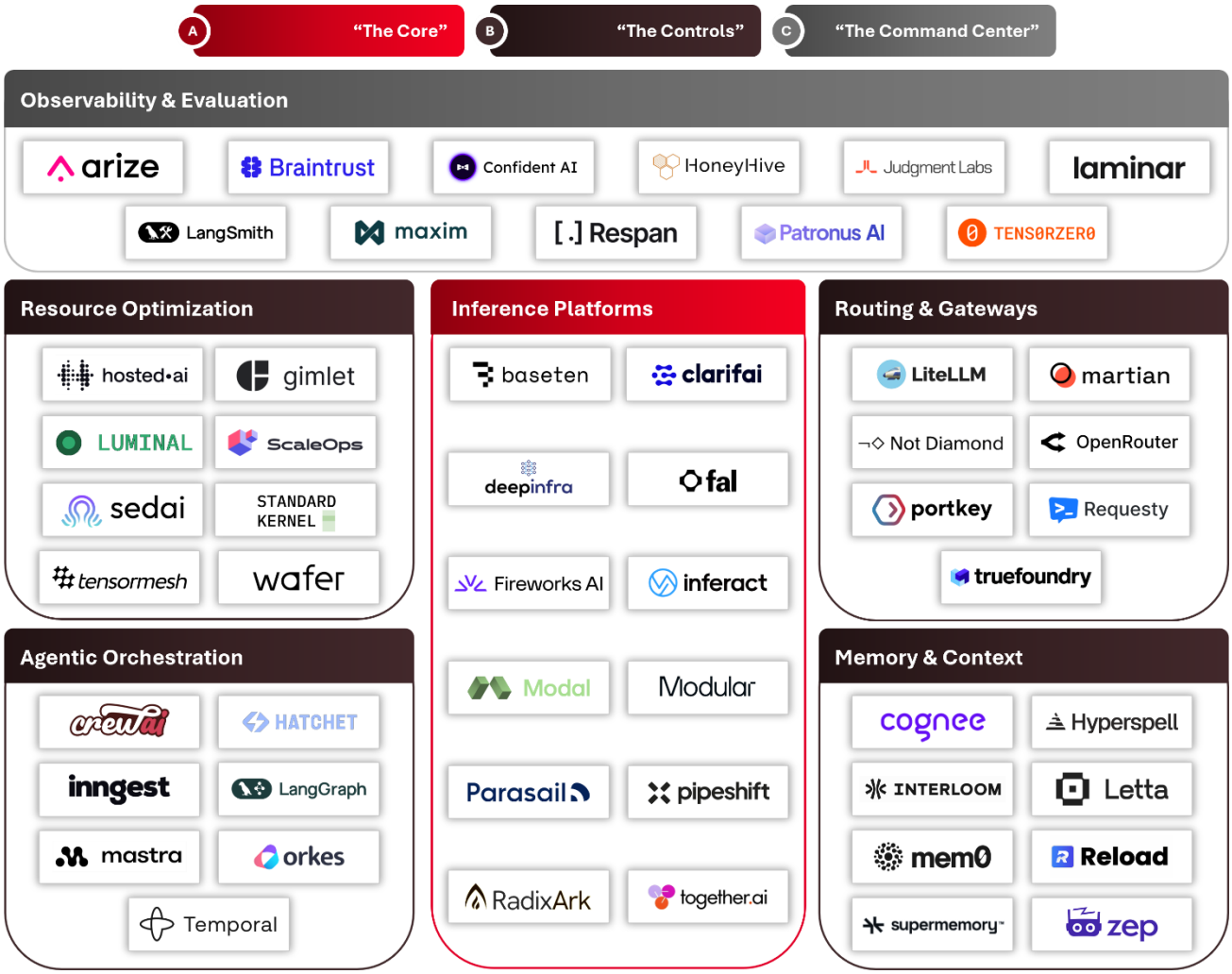
With the continued advancements in model quality and capability, AI adoption is becoming less of a technology problem and more of an operational one. The failure and abandonment rates tell the story. 42% of enterprises killed most of their AI initiatives last year before they reached production, up from 17% in 2024[1]; meanwhile, estimates suggest that more than 40% of agentic AI projects could be canceled by 2027, citing escalating costs and inadequate operational controls.[2] Previously, enterprises were asking whether AI could solve their problems at all, whether models performed well enough, and whether specific use cases justified the investment. To move AI into production, enterprises are now asking a more operational set of questions:

1. **Can we run this reliably at scale?** Pilots run in controlled settings against clean data. Production means data drift, variable usage patterns, and workloads that behave differently.
2. **Can we control the cost?** Enterprise budgets are often strict and set well in advance. Unpredictable token consumption means some teams are burning through their annual allocation by Q2, with overages directly hitting margins.
3. **Can we diagnose and understand what's happening?** Agents don't follow predefined paths across tools, models, and data sources. Any one of them can fail without surfacing an alert. Teams need visibility into why an agent failed, where it failed, and how to fix it.

The answer to these questions is a function of infrastructure. The largest hyperscalers are on pace to spend more than $700 billion on infrastructure in 2026, up 60% from 2025, with roughly three quarters of that tied directly to AI.[3] The bet is that more hardware, data centers, and energy mean more capability. But despite the massive investment, GPUs deployed by organizations often run below 30% utilization.[4] Getting more out of existing compute is a software problem.

As agents enter production, the center of gravity is shifting from training models to running them. More advanced and specialized hardware is important but only goes so far. **The bigger lever is software: purpose-built infrastructure that serves and runs models, improves GPU utilization, directs each task to the right model, orchestrates agentic workflows, maintains memory and context, and makes the entire system observable, measurable, and governable.** That's where we believe the most important infrastructure businesses of this cycle are being built.

The Hard Part



Source: AVP research. Representative sample, non-exhaustive; companies may operate across multiple categories.

A. "The Core"

Enterprises running AI in production are often dealing with inference that's slower and less reliable than it needs to be. That performance profile is a function of how inference is served and managed. A single inference request is trivial, but delivering millions reliably requires infrastructure that's purpose-built for scale. Production inference systems manage burst traffic, variable workload patterns, model updates, and concurrent requests from thousands of users without dropping requests, degrading response times, or requiring manual intervention. Getting this right is what inference platforms are built for.

Open-source engines have seen strong adoption, with **Inferact (vLLM)** and **RadixArk (SGLang)** commercializing the two most widely deployed open source serving engines, each built for the reliability and throughput demands of production workloads. Teams that don't want to run the infrastructure themselves can turn to managed providers like **Baseten** and **Fireworks AI** which abstract away provisioning, autoscaling, and uptime management. These are among the solutions that represent The Core: inference platforms that determine whether a model call happens at all, whether it returns on time, and whether the system holds up as demand scales.

B. “The Controls”

This is where the economics of running AI in production get decided. The Controls are the software levers that shape inference costs at enterprise scale: how efficiently GPUs run, which model handles each request, how agentic workflows execute, and what context persists between sessions.

Even a well-architected serving layer leaves performance on the table if the GPUs underneath are underutilized. Companies are approaching this problem from several angles. **Luminal** compiles AI models ahead of time into optimized code for each specific GPU, resulting in 2-3x higher throughput on the same hardware. **Tensormesh**, founded by the team behind LMCache, the leading open-source KV cache project, eliminates redundant GPU compute by storing and reusing the intermediate calculations models generate with every request. **Gimlet Labs** routes each stage of a workload to the chip best suited for it, spreading inference across multi-silicon fleets and delivering 3-10x inference speedups. Across all three, the lever is the same: software extracting more from the infrastructure already in place.

Better GPU utilization doesn't fix whether a task went to the right model in the first place. A portion of inference spend is wasted on frontier models doing work that smaller, cheaper models handle just as well. Routing platforms and gateways address this from two angles. Routing platforms like **Not Diamond** and **Martian** decide which model should answer each request based on the complexity and cost profile of the task. Gateways like **OpenRouter**, **LiteLLM** and **Portkey** standardize access across providers, manage fallbacks, enforce rate limits, and maintain audit logs. The difference between routing a task to the right model and defaulting to a frontier model is small per request but significant across millions of them.

Routing and gateways operate at the level of a single request; orchestration is what holds a distributed agentic system together across dozens of them. The challenges distributed systems present are familiar in an agentic world: retrying failures, maintaining state, and debugging where workflows broke. On top of that, agentic workflows run non-deterministically, chaining calls across tools, data sources, and multiple models without a predefined path from input to output. An agent performing at 99% accuracy per step succeeds only 82% of the time across a 20-step workflow. Orchestration platforms ensure that the system can recover when something breaks. Companies that solved durable workflow orchestration and execution for microservices architectures, like **Orkes**, founded by the creators of Netflix's open source Conductor project, are well positioned to extend that infrastructure to agents. Newer entrants like **Mastra** and **Hatchet** are building orchestration natively for agentic workflows, rather than adapting infrastructure that predates them.

An agent that can't remember previous interactions starts from zero every time. Larger context windows can partly compensate, but packing a prompt with more tokens adds latency and degrades output quality as the model processes irrelevant context. Memory platforms solve this with a better approach: managing what gets retained, how conflicting facts get reconciled, and how context moves between working memory and long-term storage. **Letta** (from UC Berkeley's MemGPT research) gives agents OS-style memory

management where the agent itself controls that movement. **Mem0** acts as a persistent memory layer, bolting onto existing agent frameworks and extracting facts from interactions automatically. Without memory infrastructure, every session is a cold start, and cold starts are expensive. Each of these levers compounds at scale. A task running on the wrong model, with redundant compute, and across agents that don't retain state or context costs more and delivers less than one where every layer is tuned.

C. “The Command Center”

The Core and The Controls determine how a production AI system runs; The Command Center makes it observable. Unlike traditional software, AI systems fail silently. An agent can hallucinate, skip a step, or return a plausible-but-wrong output without triggering any alert. Today, only 13% of engineering teams report feeling very confident in their ability to observe and debug production AI workflows.[5] Observability and evaluation platforms are how enterprises close that gap, covering tracing (what the agent did), evaluation (whether it did it correctly), and production monitoring (whether performance holds at scale).

The need to close that gap is driving demand for platforms like **Arize AI**, which brings enterprise-scale production monitoring to LLM workloads, **HoneyHive**, which focuses specifically on observability for agentic systems and the feedback loop between production and development, and **Respan**, which turns agent observability into action by automatically surfacing regressions and telling teams exactly what to fix. Observability has traditionally been a reactive tool, something teams reach for after something breaks. With agents increasingly running in production, it's becoming a proactive one, shaping deployment decisions before failures surface.

Taken together, the Core, the Controls, and the Command Center describe where the production AI stack is being built, and where the most important infrastructure businesses of this cycle will emerge.

What We Believe

The frontier has moved. Over the past few years, the race was driven by model capability: bigger weights, better benchmarks, more parameters. Today, the race is about running those models in production. As that shift plays out, we believe a few themes will hold true.

- **The bottleneck is coordination**
  Single model calls are largely solved. The hard problem is managing how models, tools, and memory interact across multi-step agentic workflows, where a failure or inefficiency at one step compounds through every step that follows. Latency, cost, and reliability are all coordination problems.
- **Rising inference costs are a systems problem, not a pricing problem**
  Per-token costs are falling, but token consumption volumes are growing faster. The systems layer is where that spend gets controlled. Directing tasks to the right models, caching repeated computations, and cutting redundant steps can meaningfully reduce cost per outcome even as overall usage grows.

- **Observability becomes the control plane**
  As evaluation tooling matures, the output of observability systems will increasingly influence infrastructure and deployment decisions, making observability less of a reactive debugging tool and more of a central piece of proactive runtime infrastructure.
- **The AI production stack will consolidate**
  Enterprises running multiple solutions across the AI production stack will look to reduce that surface area into an end-to-end platform over time. M&A activity has accelerated in the past 6 months: Cloudflare acquired Replicate (Inference Platform), Mintlify acquired Helicone (AI Gateway), and Cisco and ClickHouse acquired Galileo and Langfuse (Observability & Evaluation), respectively. The winners will be the vendors who can credibly own multiple layers without losing depth in any of them.

The gap between demo and deployment is an infrastructure problem, and infrastructure is software. The companies that will deliver the most value won't be defined by the models they use, but by how well they enable enterprises to run them at scale. If you're building the infrastructure that makes AI production-ready, we'd love to talk.

Footnotes:

1. *S&P Global*
2. *Gartner*
3. *S&P Global*
4. *Mirantis*
5. *CIO.com*

avpcap.com

Bringing perspectives and investing in tech
from both sides of the Atlantic

New York Paris London