Omdia
by informa techtarget

JANUARY 2026

When Observability Economics Break: The Case for Bring Your Own Cloud

Torsten Volk, Principal Analyst, Application Modernization

Abstract: Modern enterprises often generate more operational data than they can effectively observe in practice, streaming from production application code, third-party services, and an expanding fleet of AI agents. This data should form the foundation for data-driven decision-making. Increasingly, it doesn't because the volume, velocity, and cost of observing modern systems now outpace most organizations' ability to collect and analyze data economically.

As agentic AI, distributed microservices, and federated data architectures become foundational to automation and resiliency, organizations need full visibility into system behavior without letting observability costs spiral out of control. Running these environments safely and efficiently requires deep, continuous insight into application behavior, infrastructure performance, and model outcomes. The problem is scale: The telemetry produced by modern workloads, especially AI workloads, grows faster than traditional observability economics can keep up with.

AI inference pipelines, GPU utilization patterns, token usage, prompt and response pairs, retrieval quality signals, and model drift indicators all demand fine-grained, high cardinality instrumentation. Development teams need this context-rich observability data to improve code, anticipate failures, and debug issues before users feel the impact. Yet this is where the system breaks down. Faced with unpredictable costs, organizations throttle data collection. They invest in sophisticated observability platforms and then starve them of the data required to deliver value.

Fixing this requires a simple but fundamental shift: Make instrumentation effortless and eliminate the fear of runaway observability costs. Only then can teams instrument freely, observe deeply, and operate modern systems with confidence.

Figure 1. High Observability Costs Hinder Data-driven Decisions

Source: Omdia

The Cost of Incomplete Visibility

Recent research from Enterprise Strategy Group (now Omdia) quantifies the tension between observability requirements and economic constraints.[1] The data reveals a market where organizations systematically under-instrument their systems, not because they lack the technical capability, but because the dominant pricing models penalize comprehensive telemetry collection.

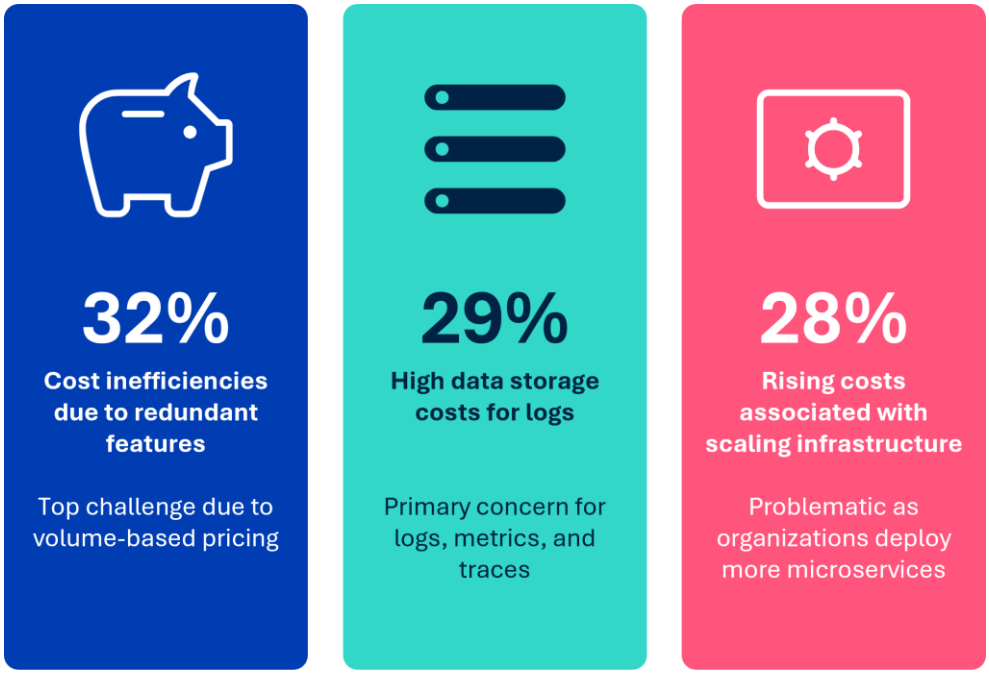
- **32% of organizations cited cost inefficiencies due to redundant or underutilized features** as one of their top observability challenges. This paradox—organizations paying for capabilities they cannot afford to use fully—results directly from volume-based pricing. Organizations purchase platforms with sophisticated analytics, then throttle their data collection to stay within budget, leaving those analytics starved of the inputs they need to deliver value.
- **29% reported high data storage costs for logs, metrics, and traces** as a primary concern. When every byte of telemetry translates to incremental cost, development teams face impossible choices: Instrument the new AI inference pipeline comprehensively, or stay within the quarterly budget. The result is selective blindness—gaps in visibility that often correspond precisely to the novel, complex systems that most need observation.
- **28% identified rising costs associated with scaling infrastructure** as problematic. As organizations deploy more microservices, more AI models, and more distributed data pipelines, their observability costs grow faster than their infrastructure—precisely when comprehensive visibility becomes most critical.

To understand the impact at scale, consider a single LLM-backed service handling a few million user requests per day. Each request can generate prompts, responses, token counts, retrieval traces, and latency metrics. At production scale, this quickly becomes billions of high cardinality telemetry events per month. Under ingestion-based pricing, fully instrumenting this data can cost more than the cloud compute resources the

[1] Source: Enterprise Strategy Group (now Omdia) Research Report, *Transforming Observability and Monitoring Through AI*, April 2025. All Enterprise Strategy Group research references in this showcase are from this report.

application runs on. Teams are forced to trade visibility for budget discipline, even when that visibility directly affects reliability and model quality. This tension is structural, not the result of over-instrumentation.

Figure 2. Top Observability Challenges in 2025



Cost inefficiencies and high data storage are the leading observability challenges, forcing organizations to compromise on comprehensive telemetry collection.

Source: Omdia

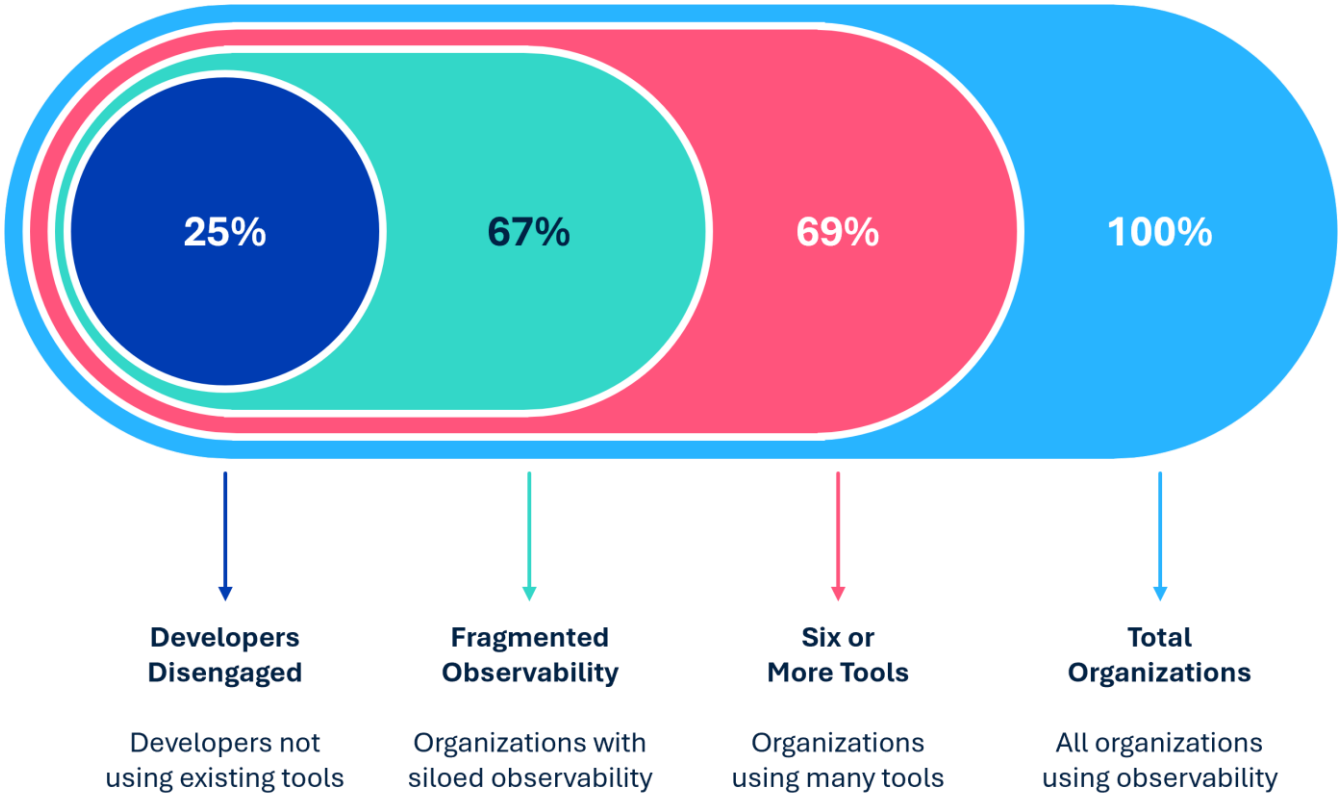
Fragmentation as a Symptom

When comprehensive platforms become too expensive to use comprehensively, organizations fragment. The research confirms this: **69% of organizations use six or more separate observability tools and platforms.** This is not architectural preference—it's economic adaptation.

The consequences extend beyond tool sprawl, and **67% of organizations reported fragmented and siloed observability** across tools and teams. When an AI-powered recommendation service degrades, the investigation spans multiple systems: application traces in one tool, infrastructure metrics in another, model performance data in a third. Each context switch extends resolution time and increases the probability of missing the actual root cause.

For development teams, this fragmentation translates to friction: **25% of organizations indicated that developers cannot or do not want to use existing observability tools**. When the feedback loop between production behavior and code changes requires navigating multiple platforms and justifying each instrumentation decision against budget constraints, developers disengage. They ship code without adequate telemetry, and the visibility gap widens.

Figure 3. Observability Tool Fragmentation



Source: Omdia

AI Workloads Expose the Breaking Point

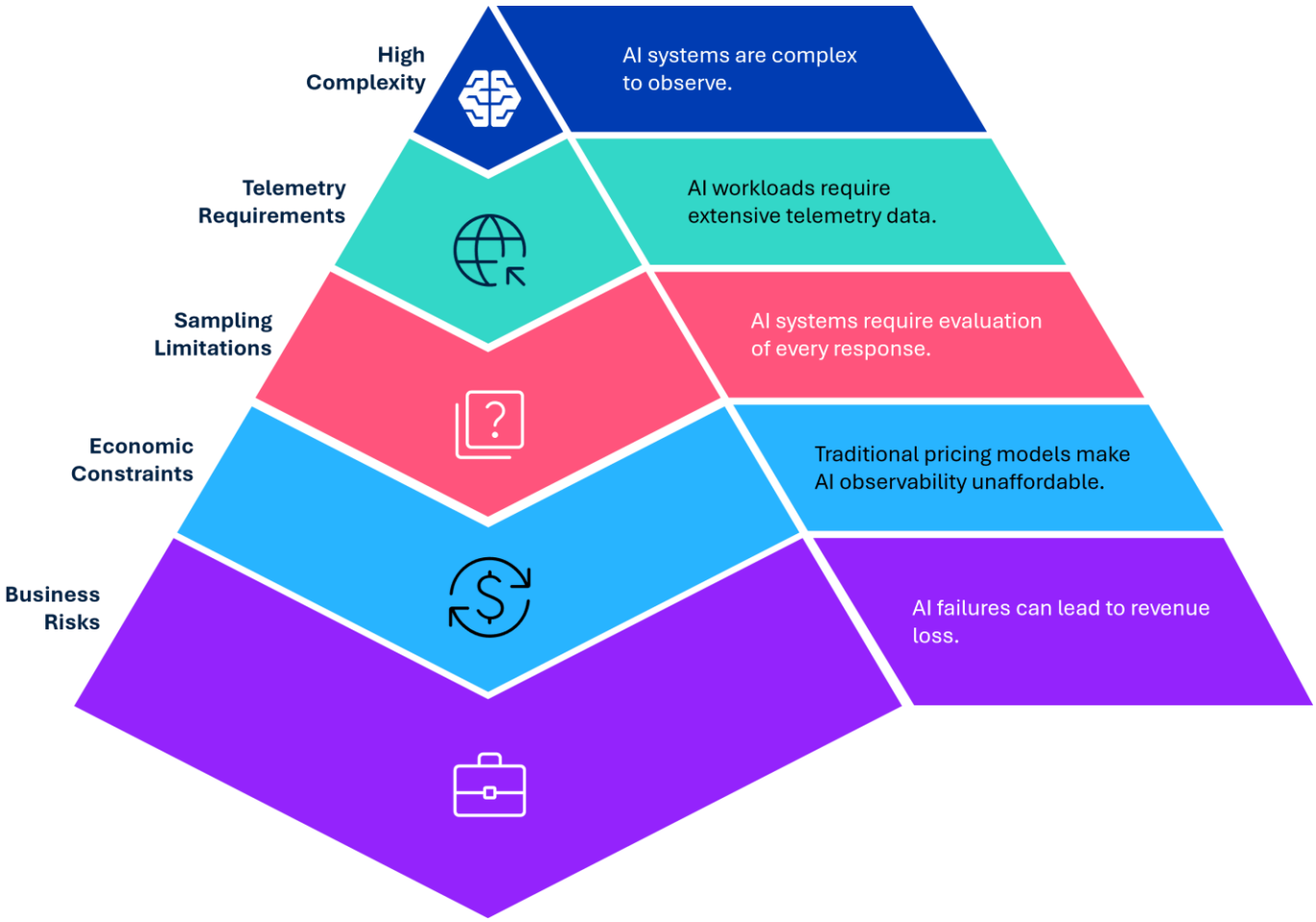
Traditional applications stress observability economics. AI workloads break them entirely. Consider the telemetry requirements of a production LLM deployment: inference latency distributions, token throughput, GPU memory utilization, prompt/completion token ratios, embedding quality scores, retrieval precision for RAG pipelines, response quality metrics, cost-per-query calculations, and drift detection across all of these dimensions. This is not optional instrumentation—it's the minimum viable visibility for operating AI systems responsibly.

While many organizations use sampling to manage economics in SaaS vendors that charge by volume, AI workloads fundamentally break this approach. Unlike traditional applications, where sampling a percentage of requests provides representative performance data, AI systems require evaluation of every response to assess quality. It is not sufficient to test some responses; organizations must test all responses to detect hallucinations, measure prompt effectiveness, identify drift patterns, and maintain quality thresholds. Some workloads, especially in their inception, do not allow organizations to sample heavily from day one and perhaps not at all. A modern observability platform must accommodate these concerns and provide tools to support the ongoing refinement process as workloads evolve.

The research reflects this reality: 69% have deployed dedicated observability for AI and ML models. Yet 44% cited high complexity of AI systems as one of their top challenges in achieving comprehensive AI observability. The complexity isn't primarily technical—it's economic. Organizations know what they need to instrument, and they cannot afford to instrument it under traditional pricing.

A drifting recommendation model, a hallucinating language model, or a degraded retrieval pipeline can translate directly to revenue loss, customer churn, or regulatory exposure. Yet the cost of comprehensive AI observability under volume-based pricing can exceed the operational budget of entire platform teams.

Figure 4. AI Workloads Expose the Breaking Point of Observability Economics



Source: Omdia

Bring Your Own Cloud: Decoupling Data from Cost

The bring-your-own-cloud (BYOC) model addresses this structural problem by relocating the observability platform to the customer's infrastructure. Data stays within the customer's environment, and storage and compute costs follow cloud provider pricing. The observability vendor charges a license fee for platform use, not data volume.

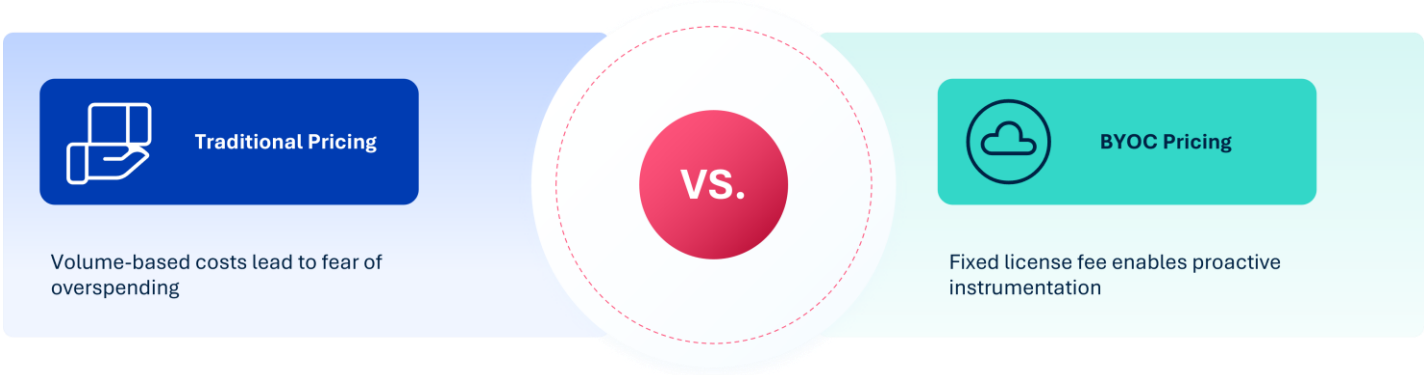
This architectural shift fundamentally changes the instrumentation calculus and eliminates the fear of overspending on observability data ingestion, processing, or storage. When adding a new metric, trace, or log stream does not increase the bill, development teams can instrument proactively rather than reactively. The question shifts from “Can we afford to observe this?” to “What visibility do we need?”

Particularly for AI workloads, BYOC takes away the fear of high cardinality, high volume telemetry that optimal AI operations require. This includes collecting inference metrics across millions of daily predictions to detect model drift, measure prompt effectiveness, optimize token costs, and catch quality degradation before users notice. These are insights that volume-based pricing makes prohibitively expensive to obtain.

Incident response also changes. Under traditional pricing, bills often spike during incidents as teams scramble to increase log verbosity to capture error details, error conditions automatically generate stack traces with large diagnostic data dumps, and cascading failures spread across dependent services. BYOC eliminates this volatility as organizations pay for infrastructure that is typically already in use, letting finance teams budget with confidence and developers instrument without fear.

For regulated industries such as healthcare, financial services, and government, BYOC addresses data sovereignty requirements by keeping telemetry within the customer’s cloud perimeter. Existing security controls, IAM policies, VPC configurations, and encryption standards apply to observability data automatically.

Figure 5. Choose the Best Observability Pricing Model for AI Workloads



Source: Omdia

eBPF and OpenTelemetry: Zero-friction Instrumentation

Removing economic barriers to instrumentation is critical but not sufficient. The complexity of code instrumentation constitutes a significant challenge on the path toward optimal observability. This is where eBPF and OpenTelemetry come in.

eBPF enables kernel-level observability without any application code modifications. By automatically instrumenting at the kernel layer, eBPF captures network traffic, including HTTP requests, database queries, DNS calls, and TCP connections, while also collecting system-level metrics such as per-process CPU and memory utilization, disk latency, and scheduling delays. This provides developers with immediate

observability for new deployments without needing to add instrumentation code or modify existing applications.

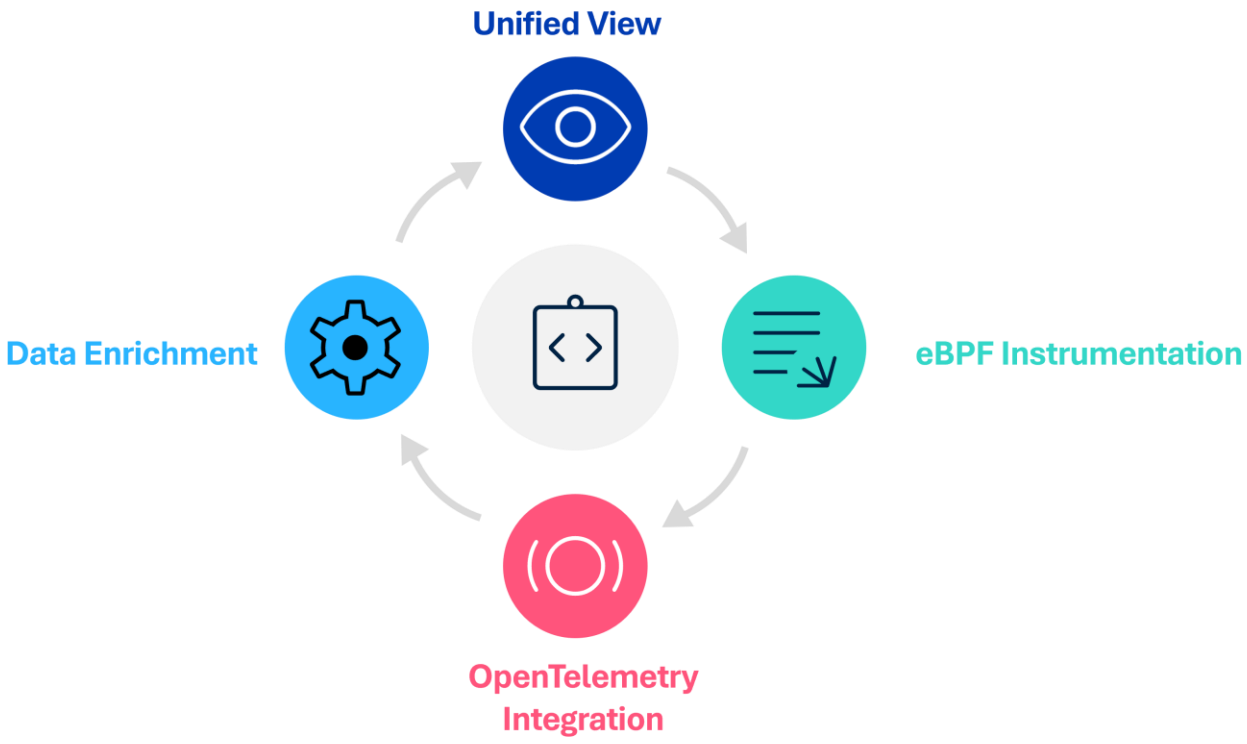
eBPF identifies the what but not the why. It can capture that a request took 250ms, but it is not able to connect this kernel-level metric with the user ID, transaction ID, or feature flag. This requires an observability platform such as groundcover to correlate kernel-level telemetry with application context through a combination of user-space probes and OpenTelemetry integration. groundcover deploys two types of probes:

- **Kernel-space probes** to capture system calls, network traffic, and resource utilization at the OS level
- **User-space probes** that observe application behavior before encryption occurs in the application stack

This dual-probe architecture enables groundcover to capture full request and response payloads. The captured telemetry flows through an OpenTelemetry collector into the dataplane, where groundcover automatically enriches standard OpenTelemetry-distributed traces with deep context from eBPF.

For example, when a RAG pipeline returns a slow response, OpenTelemetry provides the distributed trace structure showing the request flow across services—from API gateway to vector database to LLM—while eBPF automatically captures the payload details: which embedding model was invoked, how many documents were retrieved, the actual prompt sent to the LLM, and the token count in the response. Engineers see both the architectural path (OpenTelemetry) and the operational substance (eBPF) in a single view, without having manually instrumented these components.

Figure 6. eBPF and OpenTelemetry Integration Cycle



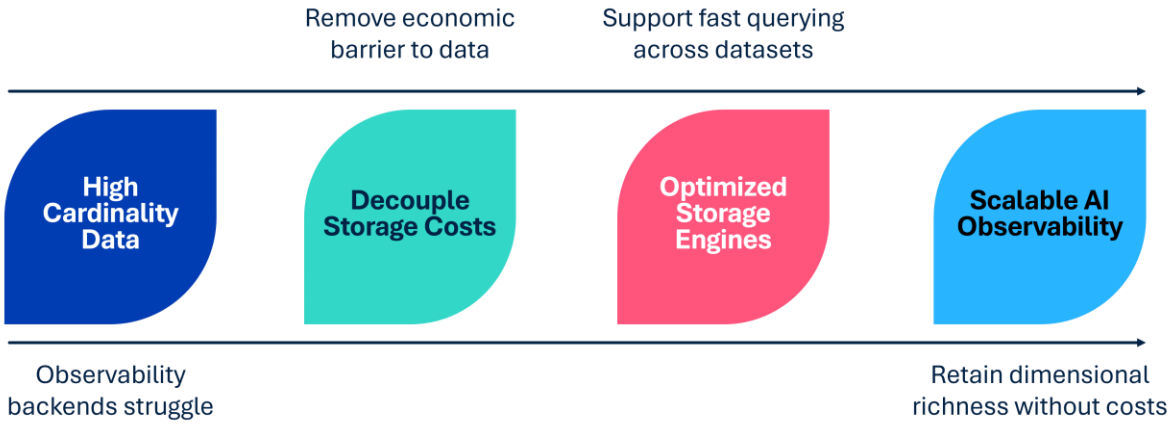
Source: Omdia

Handling High Cardinality at Scale

Comprehensive instrumentation of AI workloads generates high cardinality data, metrics with millions of unique label combinations representing per-user, per-request, per-model dimensions. Traditional observability backends struggle with this cardinality, either rejecting high cardinality metrics or degrading query performance as the number of dimensions increases. Traditional pricing models compound the problem: High cardinality means high data volume, which means high cost.

BYOC architectures remove the economic barrier to high cardinality data by decoupling storage costs from observability vendor pricing. When telemetry stays in the customer's cloud at standard storage rates, organizations can retain the dimensional richness that AI observability requires—per-customer latency tracking, per-prompt effectiveness scoring, and per-inference cost allocation—without incurring additional costs. Implementations such as groundcover leverage optimized storage engines like ClickHouse to support fast querying across massive datasets, demonstrating that high cardinality observability is technically and economically feasible.

Figure 7. Scalable Observability for AI Workloads



Source: Omdia

Self-Service BYOC: Enabling Team-level Adoption

Traditional enterprise observability deployments require centralized procurement, lengthy implementation cycles, and organization-wide rollouts. Self-service BYOC deployment changes this dynamic by enabling team-level adoption. Individual development teams can deploy observability infrastructure within their own cloud accounts, gaining immediate access to comprehensive instrumentation without waiting for enterprise procurement cycles.

This addresses a specific pain point from the Enterprise Strategy Group (now Omdia) research survey: 25% of developers cannot or do not want to use existing observability tools. Self-service BYOC enables development teams to own their observability experience, configured for their workflows. Because BYOC costs don't scale with data volume, organizations can extend observability to multiple teams without creating budget allocation conflicts.

Conclusion: The Complete Observability Stack

The observability market's fundamental tension between the visibility requirements of modern systems and the economics of ingestion-based pricing will intensify as AI workloads proliferate. Organizations that continue accepting volume-based pricing will continue accepting visibility gaps, with corresponding impacts on reliability, development velocity, and AI operations quality.

Addressing this tension requires more than a pricing model change. Comprehensive observability demands three capabilities working together:

- Economic models that remove the cost penalty for complete instrumentation.
- Technical approaches, such as eBPF, that eliminate instrumentation friction.
- Storage architectures that handle the high cardinality data that comprehensive instrumentation produces.

BYOC provides the economic foundation, eBPF provides zero-friction instrumentation at the kernel level, and OpenTelemetry provides an industry-standard technical compliance posture, while optimized backends provide the storage performance, and self-service deployment brings observability directly to development teams.

The differentiation of modern observability platforms lies in their ability to enable enterprises to adopt telemetry data at scale without worrying about instrumentation from both the technical and financial perspectives. AI adoption is forcing organizations into experimentation across all verticals, and a modern observability solution must support this without penalizing organizations as they adopt it, while keeping compliance and security posture fully aligned with organizational needs. Ultimately, AI cannot compensate for data that is never collected.

Get in touch: www.omdia.com askananalyst@omdia.com