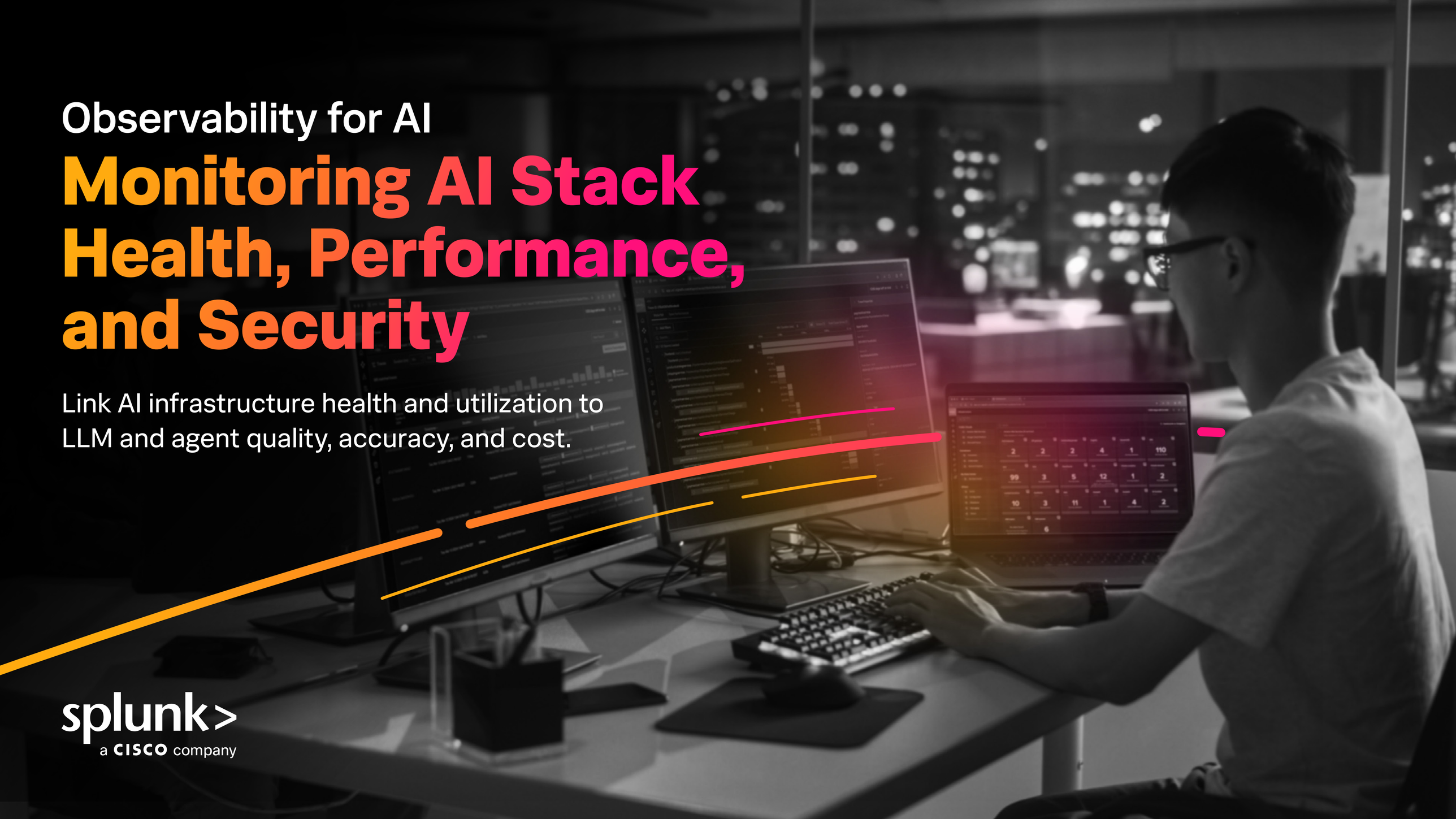

Observability for AI

Monitoring AI Stack Health, Performance, and Security

Link AI infrastructure health and utilization to LLM and agent quality, accuracy, and cost.

splunk>
a CISCO company

The new rules of observability in the AI era

Today, AI is rewriting the rules for observability. As agentic AI emerges, it is even coding software — making it critical for teams to have the visibility needed to ensure applications perform as expected.

Metrics, events, logs, and traces (MELT) from AI environments behave differently than in traditional and even modern application environments. GPU utilization, model latency, and data pipeline throughput matter as much as CPU or uptime, and they rarely move in predictable patterns. In isolation, these signals are noise. Pulled into a single view, they reveal the full picture of reliability, accuracy, quality, and security issues before they disrupt performance and undermine the user experience.

AI agents and LLMs place heavy, shifting demands on infrastructure. Monitoring alone can't capture those dynamics. Teams benefit from measuring them against performance, reliability, and cost targets to confirm they're delivering value without exhausting resources.

For operations and development teams, a **shared view** that not only shows something is broken but also explains how the failure started and which layers it affects. Even a small infrastructure shift, a spike in queries, a mis-timed retraining job, or a surge of data across networks, can cascade into customer frustration, unexpected costs, or lost trust.

Why observability matters for AI infrastructure

AI infrastructure is different from traditional IT. Models run on GPUs, workloads spike unpredictably, and outcomes are judged on accuracy and quality, not just availability. Traditional monitoring was not designed for this landscape.

The gaps show quickly. A dashboard may show GPU utilization, but not tie back to the data pipeline causing the slowdown. It can flag a surge in traffic but not explain why APIs are overwhelmed or other jobs are delayed. Without correlation across layers, root causes remain buried.

AI workloads also demand continuous validation. Teams need to know that models are performing as expected, costs are under control, and resources aren't being drained by hidden bottlenecks.

The two primary use cases we'll explore

Within this e-book, we'll look at two domains that together form AI observability:

- **AI Infrastructure Monitoring** — Tracking system health, performance, and consumption of the AI infrastructure stack to manage AI workloads and resource utilization.
- **AI Agent Monitoring** — Tracking the performance, quality, cost, and security of LLMs and the AI agents and applications they power.

Both lenses are needed to see the full AI stack and give teams real oversight, from the new infrastructure it runs on, to the LLMs generating outputs, to the AI agents and applications built on top of them.

Reflection Questions

1. Which AI services in your organization rely most on infrastructure performance?
2. Where do you face the most alert fatigue or false positives, and how does it affect response time?
3. How do you connect application issues to infrastructure bottlenecks, and how effective is it?
4. If you had a unified view of infrastructure and application data, which KPIs or outcomes could you improve first?

A new kind of infrastructure stack, a new kind of challenge

Specialized infrastructure components are needed to support generative AI applications. What began as simple, deterministic compute tasks has matured into complex, adaptive, specialized, and interdependent workflows.

A few key elements in a modern components include:

- **Specialized hardware** such as Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs) for accelerated computation.
- **Orchestration frameworks** to manage multiple models and workflows.
- **Vector databases** for context-aware search and retrieval.
- **Large language models and serving layers** that process prompts and produce outputs.
- **Support services** including API gateways, data preparation pipelines, and observability tools.

These components form tightly integrated chains of dependency. A single user request can trigger orchestration logic, query a vector database, engage a model-serving endpoint, and return a result. Each step depends on the prior one meeting performance and quality targets. If one link falters, the process slows or fails completely.

Conventional monitoring tools often lack visibility into these new, unique behaviors. AI requires new forms of telemetry like token usage, hallucination frequency, prompt success rates, or token and GPU utilization that go beyond traditional service level objectives (SLOs) like service uptime, latency, availability, and incident response time. While traditional SLOs focus on static, predictable service metrics, new AI SLOs emphasize dynamic, evolving performance and governance metrics tailored to the complexities of AI systems and their operational environments.

What Observability for AI requires

With an observability solution focused on AI, teams can resolve AI stack issues much faster and prevent misleading, erroneous, or undesirable responses or outputs by LLMs (and the agents they power).

Observability tools designed for the AI stack should:

- Give you comprehensive, correlated visibility from specialized hardware through orchestration layers to application behavior and business impact.
- Support end-to-end tracing to follow interactions across models, pipelines, and frameworks.
- Capture metrics unique to AI such as prompt success rates, quality and data drift indicators, or token costs tied to specific jobs to optimize resources and reduce security risks.
- Correlate technical performance with business impact, allowing teams to view any observed change in terms of user experience, service reliability, or operational cost efficiency.

An enterprise AI stack demands visibility across hardware, pipelines, models, and agents, linking signals into a clear view of how system health affects business results.

Reflection Questions

1. Which AI components are most critical to performance?
2. How do you track AI-specific costs like tokens and GPU time, and use this to optimize workloads?
3. Which AI performance indicators do you use to tie back to business results, and how often are they reviewed?
4. How well do your tools connect infrastructure conditions to application behavior, and where are the gaps?

Monitoring the infrastructure powering AI and LLMs

AI workloads stack on more layers and push infrastructure limits across compute, storage, networking, power, and security. Training and running large models puts stress on systems in ways conventional IT never did. Think massive data moving at low latency, GPUs working at full tilt, infrastructure that scales fast, and racks that run hotter and draw more power. Networks need to stay reliable, security has to extend deeper, and operations teams face new layers of complexity.

LLMs, in particular, can generate unpredictable traffic patterns based on prompt complexity, dataset size, and user concurrency, producing usage spikes that many standard monitoring tools aren't designed to anticipate or correlate. The underlying systems must handle everything from short-burst inference workloads to long-running training jobs, both of which can spike demand unexpectedly.

Because AI infrastructure resources are costly and always under pressure, teams tune and monitor them to keep workloads efficient and prevent wasted capacity. In shared environments, one misconfigured workload can eat up more than its share and slow everything else down, the classic "noisy neighbor" effect.

Key telemetry to track

AI-aware observability must go beyond traditional "is the service down?" checks and use rich metrics showing real-time performance and historical trends. Key telemetry categories to measure include:

- **Compute** — CPU, GPU, and TPU utilization, queue depths, and memory usage trends, mapped to workload type and business priority.
- **Networking** — latency, throughput, and error rates, especially between tightly coupled services.
- **Storage** — I/O rates, capacity thresholds, read/write latency, and retrieval times.
- **Component health** — orchestration frameworks, model-serving layers, and vector database status.
- **Cost signals** — GPU/TPU runtime, idle capacity, and over-provisioning trends.

This data helps teams compare workloads to SLOs, track usage patterns, and spot performance spikes. But greater value emerges when correlated across all components of AI infrastructure and viewed in context across AI apps and system logs.

Multicloud and hybrid visibility

Many AI environments run across a mix of public clouds, on-premises data centers, and private or hybrid architectures. Each environment may use different monitoring systems, naming conventions, and security models, making it difficult to form a complete picture. Without a unified view, troubleshooting becomes fragmented, and teams must hunt for clues across multiple consoles.

With observability that spans hybrid and multicloud environments, teams can see metrics from every system in one place and eliminate silos between infrastructure and application performance. This unified view speeds root cause identification and helps determine the best placement for workloads, whether that's a low-latency on-prem location or a cost-optimized cloud region.

Visibility across environments also enables compliance and governance. In regulated industries, understanding where AI workloads run and proving they adhere to policy are critical to reducing operational risk.

Consequences of poor infrastructure visibility

Without deep visibility, complex and dynamic AI infrastructure issues can go undetected until the latency, throughput, network, and, ultimately, the user experiences are impacted.

For example:

- Resource exhaustion can cause active inference jobs (using a trained model to generate outputs) to fail mid-process.
- Scaling inefficiencies may lead to expensive GPUs sitting idle while important workloads wait in queues.
- Persistent degradation can cause intermittent errors that erode trust in AI outputs.

Security is also a concern. Spikes in GPU usage, unusual access patterns, or repeated failed login attempts could indicate misuse or an attack. Teams miss early warning signs when monitoring isn't fine-tuned for AI infrastructure. Over time, this lack of insight not only slows operations but can also result in tangible financial loss, missed deadlines, and reputational damage.

These AI infrastructure-specific issues become increasingly untenable as AI deployments scale.

Connecting infrastructure health to business outcomes

Teams should ensure that their observability solution provides clear business context and insights. In AI environments, this connection carries even greater weight, as the risks and costs of poor performance multiply quickly.

Some examples include:

- GPU allocation delays slowing model responses to time-critical queries and frustrated customers.
- Storage bottlenecks blocking data ingestion and undermining analytics for strategic decisions.
- Network congestion degrading real-time AI services such as supply chain routing or personalized recommendations.

By tying technical metrics to business KPIs, from transaction success rates to user retention, teams can see instantly which infrastructure events put key outcomes at risk and prioritize issues accordingly. Fixes that restore a major business process move to the front of the queue, while less impactful issues are scheduled strategically.

Reliable infrastructure visibility is the operational anchor for enterprise AI.

Reflection Questions

1. Which AI infrastructure components are hardest to monitor, and how do these blind spots affect performance or cost?
2. How do you track GPU, TPU, and other specialized hardware and cloud usage, and what trends guide planning?
3. Are your AI performance metrics tied to business outcomes, and which KPIs are most influenced?
4. What challenges arise in standardizing infrastructure data across environments, and how does this affect troubleshooting?

Monitoring LLMs and agentic AI applications

LLMs and agentic applications differ from traditional software in both behavior and risk profile. They're probabilistic (the same input can produce different outputs), which can excite users with creativity or frustrate them when reliability or quality falters.

To correct for inconsistencies and inaccuracies, agentic applications often involve multiple LLM calls chained together, sometimes using retrieval-augmented generation (RAG) to pull context from a data source or from other external tools before generating answers, multiplying points where failures, slowdowns, or quality drift can occur, making end-to-end visibility crucial.

AI systems introduce challenges that monitoring tools designed for deterministic applications were not built to handle. Even when uptime and infrastructure metrics look solid, AI models can still introduce risks like hallucinations, bias, toxicity, or prompt-injection attacks that quietly bleed into user experience and decision-making.

While traditional infrastructure performance can degrade, AI systems add layers of new complexity. Performance is shaped not only by compute and networking, but also by latency, relevance, factual accuracy, model behavior, and resource use.

For practitioners, this means observability must assess output quality alongside security and operational health, with clear linkage to the infrastructure and data pipelines that influence those results.

Key metrics for LLM and agentic application monitoring

Getting observability right means tracking more than uptime. Teams need a mix of performance, cost, and quality indicators that reveal how models behave in the moment and how those behaviors shift over time.

These fall into four categories:

- **Objective accuracy** — measures of correctness such as F1 score (a balance of precision and recall) or task-specific benchmarks.
- **Human-centric quality** — semantic quality, relevance, informativeness, or appropriateness, often evaluated through automated scoring or human review.
- **Operational efficiency** — latency, throughput, error rates, and token usage/cost, including detection of spikes from inefficient prompts or premium model overuse.
- **Bias and fairness** — indicators of toxicity, bias, safety, or helpfulness, showing where model behavior drifts into unwanted patterns.

The value grows when these signals are correlated with infrastructure telemetry that explains why behavior changes. A sudden rise in latency, for example, might trace back to GPU contention, orchestration delays, or inefficient prompt design. Together, these measures give teams a full picture of how well AI applications are performing, where they are drifting, and how those shifts affect business outcomes.

Safeguarding trust and alignment in AI environments

AI systems introduce risks that extend beyond traditional infrastructure. Hallucinations erode confidence, bias or toxicity can damage reputation, and prompt-injection and distributed denial of service (DDoS) attacks may expose sensitive data, enable unauthorized actions, or disrupt services. Left unmonitored, these issues undermine not only performance but also security, compliance, and customer trust.

The right level of visibility helps catch issues before they escalate by:

- **Confirming accuracy** — ensuring AI agents and applications behave as intended and deliver reliable outputs.
- **Spotting drift** — detecting when model behavior shifts, data use strays into unsafe territory, or responses begin to put decisions at risk.
- **Surfacing security signals** — revealing possible data leaks, privacy breaches, prompt injections or misuse that might otherwise slip through the cracks.

Governance also benefits. Trust and safety teams gain proof that personal data is protected, while compliance teams can verify that models follow legal, regulatory, and internal standards. Audit trails, traceable interactions, and monitored guardrails provide the evidence they need.

Most importantly, observability connects these safeguards back to business impact. It shows how reliability, accuracy, and safety shape user experience and enterprise value. By preserving trust and validating compliance, organizations can scale AI services with confidence, laying the foundation for what comes next.

Reflection Questions

1. Which application-level metrics are most critical across those four categories, and how do you validate?
2. How do you assess AI output quality, and what thresholds or patterns trigger intervention?
3. Where are risks like hallucinations or bias most likely, and how are they detected or mitigated?
4. Do you detect prompt misuse or injection attacks or data leakage and impact on AI agents in real time, and link them to response plans?

Scaling for the future: building a unified observability strategy

LLMs and agentic applications behave in dynamic, sometimes unpredictable ways that call for monitoring approaches unlike those used for standard environments. Observability unifies those layers into a single view. Instead of blind spots that slow detection, inflate costs, and erode trust, teams gain a connected picture they can act on, accelerating troubleshooting, optimizing resources, and keeping AI systems performing as intended.

Fostering collaboration across teams

Effective observability for AI depends on cross-team collaboration. Infrastructure teams, data scientists, AI platform engineers, and trust and safety specialists need to work from a shared context grounded in key data, aligned timelines, and common goals to make well-informed decisions in real time.

The most successful organizations break down silos by giving every team access to correlated telemetry and tying it directly to business objectives. In this model, the same incident report that shows GPU usage spikes also highlights the customer-facing latency impact and the cost implications. That shared view speeds resolution and helps prevent repeat issues.

The need for unified visibility isn't limited to AI environments. The right observability approach supports everything from traditional applications to modern agentic systems, helping teams stay aligned and act quickly.

The future of AI observability: Proactive, unified, and trustworthy

Observability solutions are evolving fast to meet the demands of AI. They now spot problems early and launch automated fixes before users ever feel the impact.

Meeting that bar requires broader telemetry, sharper detections, and response workflows that span both the infrastructure powering AI and the applications built on top of it. Guardrails for security, bias, and compliance are woven into the same fabric that tracks uptime, performance, and cost.

A unified strategy is what makes scaling possible. Small AI deployments may scrape by with fragmented views, but as environments grow, every blind spot grows with them. Observability closes those gaps so teams can scale confidently without sacrificing performance, efficiency, or trust.

Reflection Questions

1. Where are the gaps in linking infrastructure and application monitoring, and how do they impact troubleshooting or service quality?
2. How well do teams share and act on the same operational data, and what barriers limit collaboration?
3. Which business indicators should tie into your observability platform, and how would they influence decisions or investments?
4. How could a unified strategy improve performance, trust, compliance, or cost control in 12-24 months?

Want to keep exploring?

Learn more about **how to monitor the performance and security** of your AI application stack, and **other observability in AI innovations**.

splunk> a CISCO company

25_CMP_ebook_Observability-for-AI_v6