

Vamsi Krishna Reddy Munnangi Senior Software Engineer Infrastructure Engineering Walmart

Cloud-Native Observability: Building Visible, Resilient, and Scalable Enterprise Systems

Transforming enterprise systems management through advanced monitoring and analytics in modern cloud-native architectures.



From Monitoring to Observability

Traditional Monitoring

Focuses on collecting pre-defined metrics with limited context and scope.

Creates siloed visibility across disconnected system components and teams.

Enables primarily reactive problem identification after issues impact users.

Cloud-Native Observability

Provides comprehensive visibility into system state through metrics, logs, and traces.

Delivers a unified, correlated view across distributed microservices architectures.

Enables proactive issue detection, faster root cause analysis, and automated resolution.

Real-World Implementation Metrics

40%

25%

Faster Resolution

Resource Efficiency

Reduced incident response time

Improved infrastructure utilization

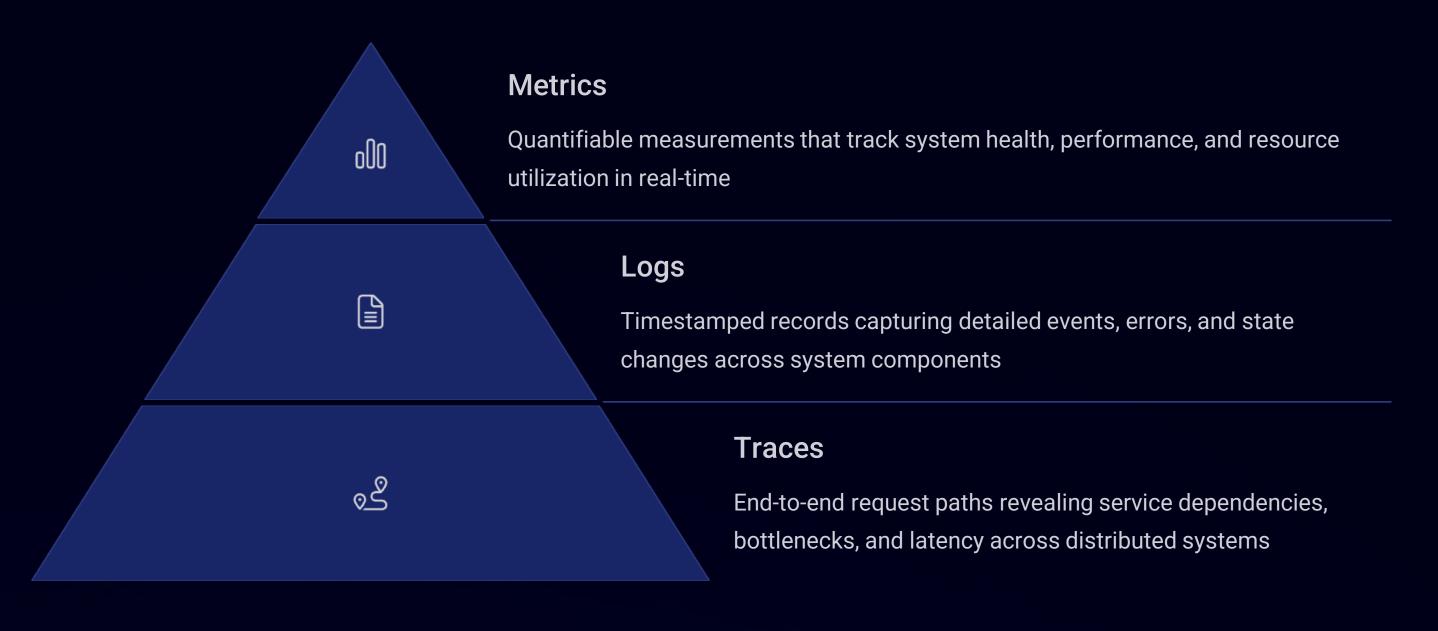
200ms

Detection Speed

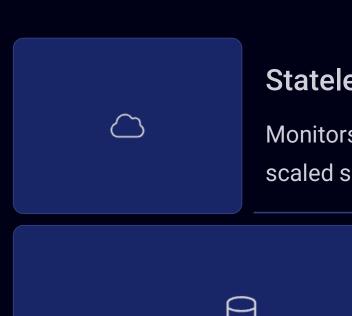
Near real-time anomaly identification



Observability Pillars



Stateless vs. Stateful Services



Stateless Observability

Monitors request throughput, latency distributions, and ephemeral resource utilization across horizontally scaled services



Stateful Observability

Tracks data persistence metrics, replication health, consistency levels, and storage performance across distributed datastores



Shared Patterns

Implements comprehensive health probes, dependency visualization, and automated anomaly detection for both service types

Distributed Tracing Implementation



Instrument Service Code

Integrate trace context propagation across all service-to-service communications



Configure Intelligent Sampling

Implement strategic sampling to capture both normal and exceptional request patterns



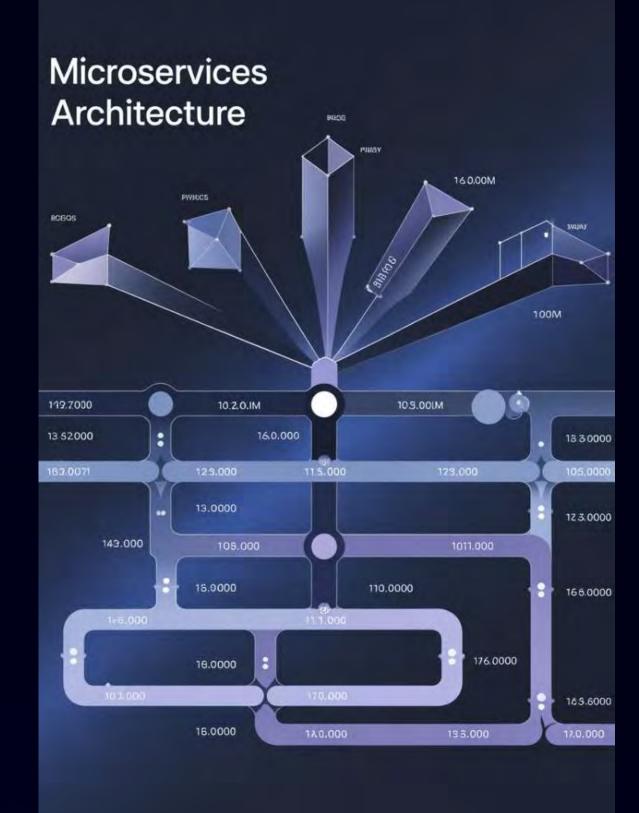
Centralize Collection Pipeline

Establish scalable infrastructure to ingest, process and persist distributed trace data



Perform Root Cause Analysis

Leverage trace visualization to identify performance bottlenecks and error propagation paths





API Gateway Observability

Traffic Insights

- Request volumes and patterns
- Service dependencies
- Rate limiting effectiveness

Security Monitoring

- Authentication events
- Authorization failures
- Attack pattern detection

Performance Metrics

- Latency measurements
- Throughput analytics
- Error rate tracking

Observability Pipeline Design

Collection

Capture comprehensive telemetry data from services, infrastructure, and applications across your distributed ecosystem

Visualization

Transform complex data into intuitive dashboards that enable rapid identification of patterns and anomalies



Processing

Normalize, enrich, and intelligently filter data streams to reduce noise while preserving critical signals

Storage

Implement tiered storage strategies with optimized retention policies balancing performance and cost-efficiency

High Availability Strategies



Redundant Observability Infrastructure

Deploy collectors and storage in multiple availability zones.



Graceful Degradation

Design systems to function with reduced telemetry.



Local Buffering

Temporarily store metrics when backends are unavailable.



Multi-channel Alerting

Ensure notifications reach teams through diverse paths.





Capacity Planning & Optimization



Collect Historical Patterns

Aggregate comprehensive utilization metrics and performance data across all distributed services over extended time periods.



Forecast Future Demand

Leverage machine learning and statistical modeling to identify seasonal patterns, growth trajectories, and potential resource bottlenecks.



Optimize Resource Allocation

Implement dynamic scaling policies and infrastructure provisioning based on empirical usage data to balance cost efficiency with performance requirements.



Continuous Improvement

Establish feedback loops to regularly compare forecasted projections against actual consumption patterns, refining prediction models and allocation strategies.

Enterprise Observability Implementation Planning 0 Design Design

Implementation Roadmap

Foundation Building

Implement basic metrics collection and centralized logging.

Establish meaningful SLIs and SLOs for core services.

Advanced Capabilities

Deploy distributed tracing across service boundaries.

Develop custom dashboards for specific team needs.

Optimization Phase

Implement anomaly detection and predictive analytics.

Create automated remediation for common issues.

Thank you