From Chaos to Clarity

AI-Powered Observability for G-Store Systems

Tom Oberkfell
Joel Miller
Josh Carr

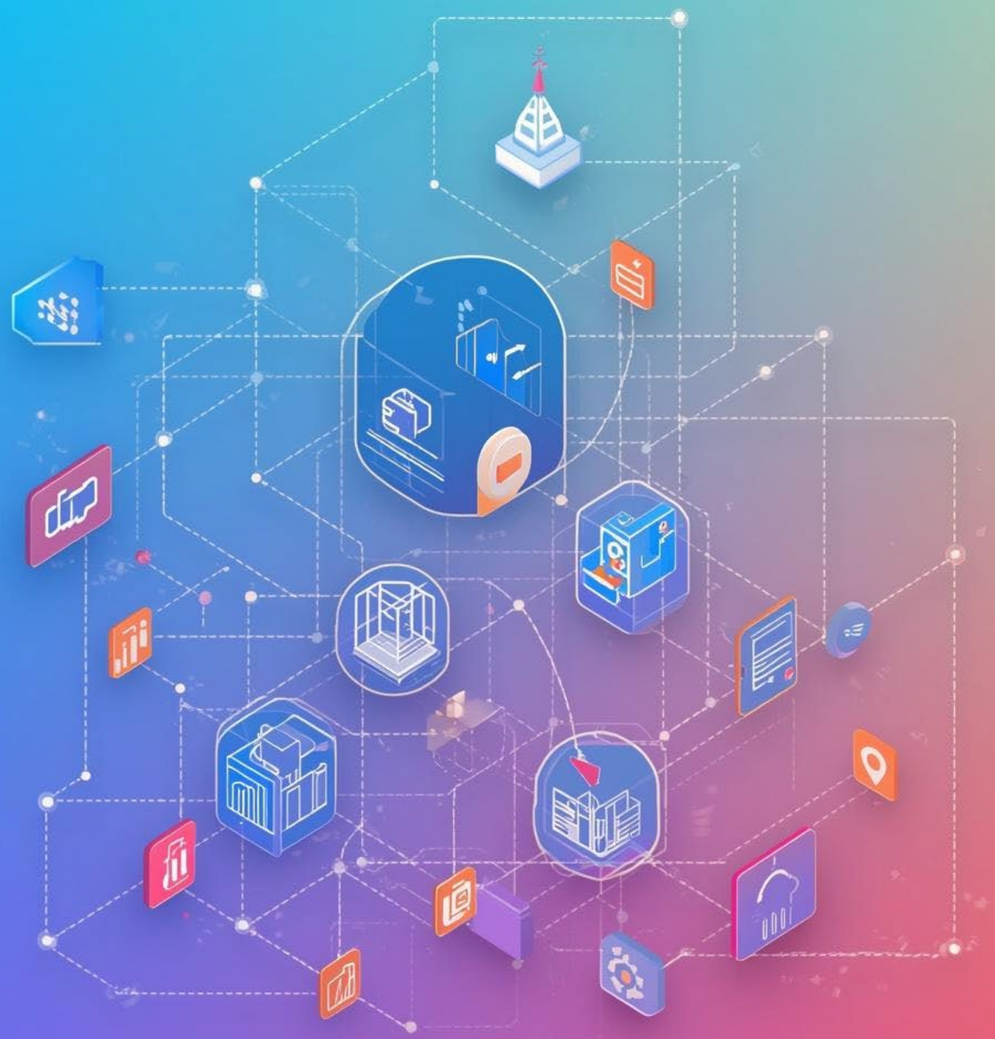
Agenda

- Intro - What is Observability
- Getting Started: Observability in G-Stores
- Best Practices
- Use Cases
- The Transformative Power of AI

Understanding Observability: A Key to IT Success

UNIFYING METRICS, LOGS, TRACES, AND EVENTS FOR COMPREHENSIVE INSIGHTS

- Metrics, Events, Logs and Traces
- Shifting from *reactive* monitoring to *proactive* insights
- End-to-end visibility across applications, infrastructure, networks, and user experiences
- Goal is to enhance operational effectiveness.



Full-Stack
Observability

The Goal of Observability

Unified Insight into IT Systems

Provides a comprehensive view of interconnected IT components.

Proactive Problem Identification

Shifting from reactive monitoring to proactive insights enhances operational efficiency.

Holistic Coverage Across Environments

Observability encompasses applications, infrastructure, networks, and user experiences seamlessly.

The Shift from Traditional Monitoring to Modern Observability

UNDERSTANDING THE PARADIGM SHIFT IN IT MONITORING AND INSIGHTS

Reactive Monitoring Approach

- Traditional monitoring focuses on whether the system is down
- Emphasizes immediate fixes
- Lacks insight

Proactive Observability Perspective

- Modern observability shifts the focus to understanding system behaviors and patterns
- Enables teams to identify root causes and prevent issues before they impact users

The Four Pillars of Observability: MELT

Metrics

- Provide quantitative data on system performance (e.g. benchmarks, KPIs)
- By analyzing metrics, organizations can identify trends and anomalies efficiently.

Events

- Represent significant occurrences within the system, such as deployments or failures.
- Provides context that enhances overall observability and incident response.

Logs

- Contain detailed records of system activities
- Offers insights into transactions and user interactions
- Analyzing logs can aid in troubleshooting and uncover the root causes of issues

Traces

- Follow the path of requests through complex systems, illuminating the interactions between services
- Correlating traces with metrics and logs, teams gain a comprehensive view of their systems

Observability Examples for C-Stores

Equipment Issues

- Pump Offline
- Flow Rate
- Printer Paper Out
- Excessive canceled sales on Pump X
- IoT sensor monitoring (cooler temps, energy consumption, etc.)

System Issues

- POS Offline
- Peripheral Connectivity Lost
- High CPU utilitization
- Credit or Loyalty Offline
- High Latency
- Network Issues

Operational Issues

- Critical Item Out of Stock
- Too many / Too few Open Shifts
- Speed of Service
- Suspicious No Sale Usage
- Over/Short Trends by Employee

Getting Started - Observability in Convenience

Start with What Matters Most

Build Foundational Visibility

Select the Right Platform

Define Success Criteria Early

Scale Through Automation

Data Sources

Best Practices: Align Observability with Business Outcomes

- Measure KPIs tied to revenue & customer experience
- Prioritize alerts for customer -facing failures
- Connect system performance to store KPIs
- Visualize: Sales impact from downtime (lost throughput)

Best Practices: Standardize & Automate Everywhere

- Consistent tagging structure
- Automate
- Continuous Integration/ Continuous Deployment

Best Practices: Implementation

- Identify Stakeholders and Critical Areas that Need Observability
- Create Usable Alerts
- Implementing Effective Dashboards
- People and Process
- Measure Impact
- Continuous Improvement

Use Case: POS & SCO Health Monitoring

Problem:

- Registers and self - checkout lanes intermittently go offline, causing lost transactions and longer queues.
- Root-cause analysis is slow due to limited visibility into local device and service performance.

Solution:

- Implement host and log monitoring for registers, kiosks, and SCO terminals.
- Track service uptime, transaction latency, and device error logs in a unified dashboard.
- Set alerts on payment failures, pin pad disconnects, and POS freezes.

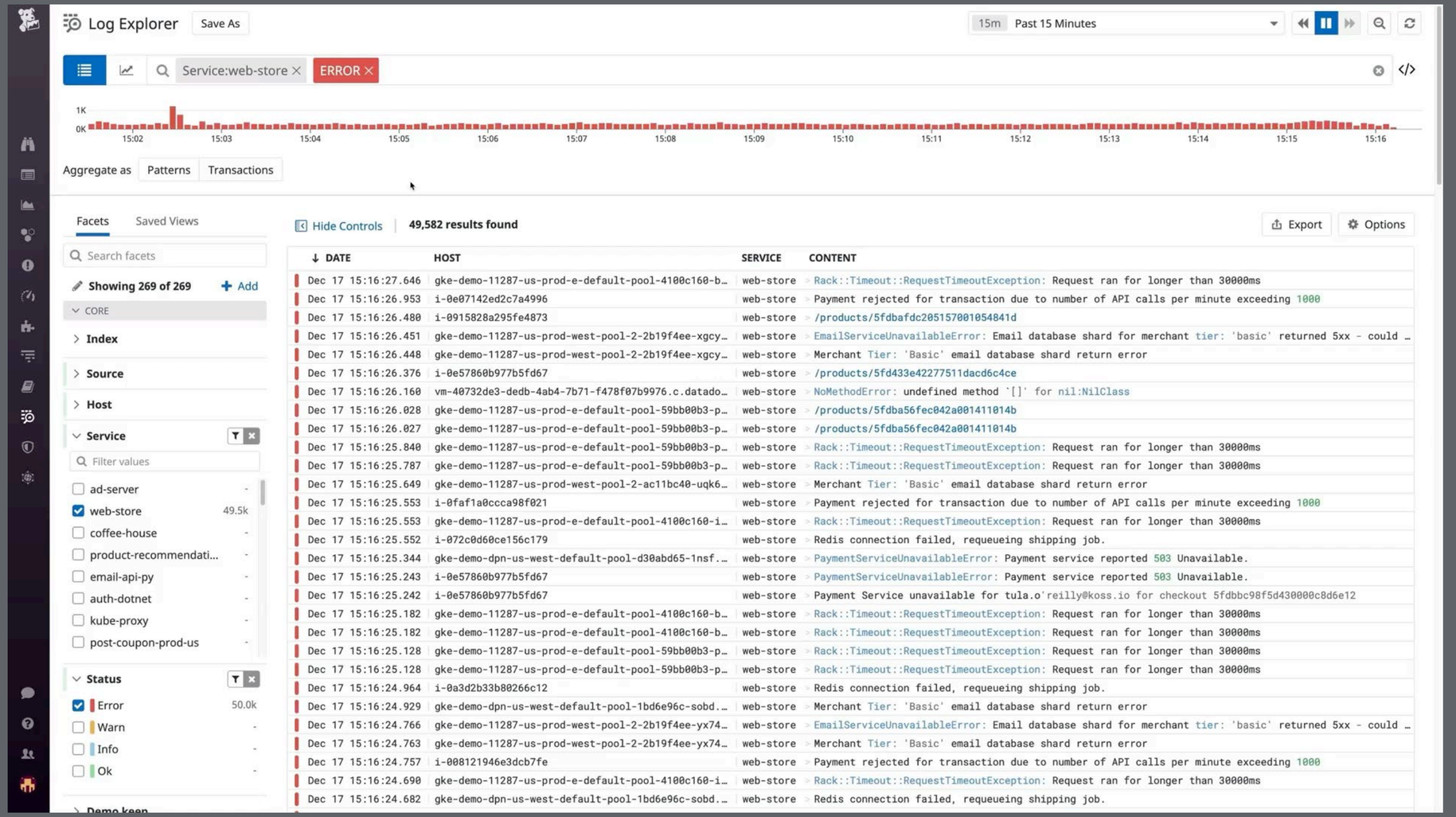
Challenges:

- High store count means scaling agent deployment and tagging consistency across environments.
- Varying hardware and network configurations at the store edge.

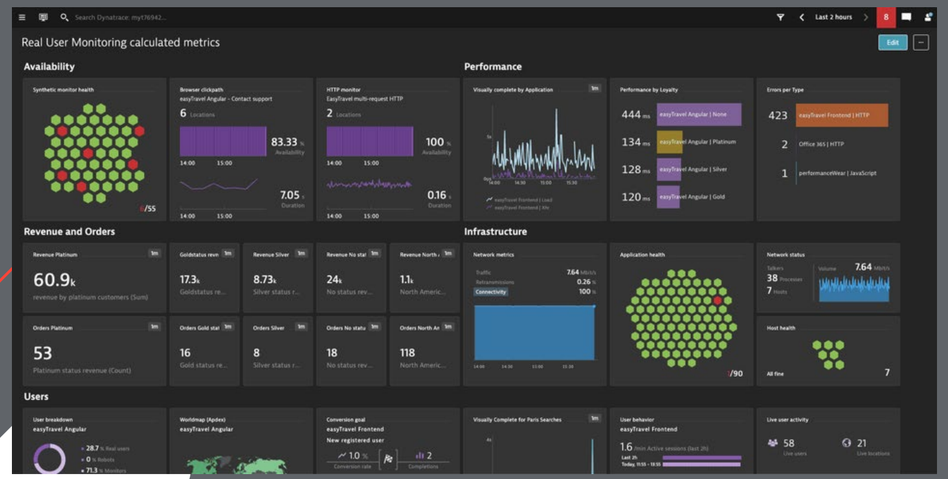
Results:

- Faster incident resolution through centralized alerting.
- Reduced downtime → more completed sales and higher customer throughput.

Use Case: POS & SCO Health Monitoring



Use Case: Real-Time Fuel Metrics Dashboard



Problem:

- Store managers lack visibility into fuel volume flow, pump throughput, and tank inventory.
- Slow or failed pumps impact both sales and customer satisfaction.

Solution:

- Use metrics pipelines to capture dispenser throughput, wet stock trends, and delivery performance.
- Create real-time dashboards for fuel volume by grade and pump health.

Challenges:

- Integrating legacy forecourt co ntrollers with modern telemetry.
- Ensuring reliable data buffering and upload during network drops.

Results:

- Early detection of low - volume pumps reduces fuel downtime.
- Optimized replenishment scheduling via visibility into tank performance.

Use Case: Loyalty & Customer Engagement Monitoring

Problem:

- Loyalty sign -ups and redemptions intermittently fail due to API latency or service timeouts.
- Failures go unnoticed until customers complain or revenue declines.

Solution:

- Instrument loyalty APIs and POS integration points using APM traces and custom metrics
- Track response times, error rates, and customer engagement patterns.
- Visualize loyalty activity per store and promotion in real time.

Challenges:

- Correlating POS logs, cloud API telemetry, and mobile app activity.
- Maintaining data privacy and secure observability across systems.

Results:

- Reduction in loyalty API errors through proactive monitoring.
- Enhanced promotion performance tracking and customer retention.

Use Case: Loyalty & Customer Engagement Monitoring

Business Impact & ROI from Observability

Measured ROI

Reduced downtime = higher sales throughput

Operational Efficiency Gains

Improved customer experience

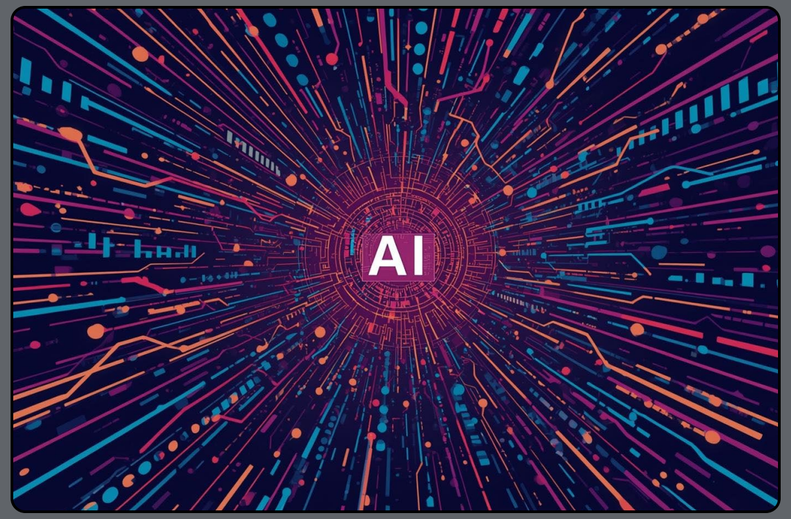
Data-driven Decisions

The Transformative Power of AI in Observability

ENHANCING VISIBILITY AND DECISIONMAKING ACROSS COMPLEX IT LANDSCAPES

- Revolutionizes observability by enabling **real-time insights** from vast data sets
- Addresses the challenges of alert fatigue and data overload through advanced anomaly detection and event correlation
- Empowers organizations to move to proactive problem resolution and maximize operations and uptime

The Transformative Role of AI in Observability



Overwhelming Data Volume

- AI helps manage vast data inflows, turning chaos into actionable insights for users.

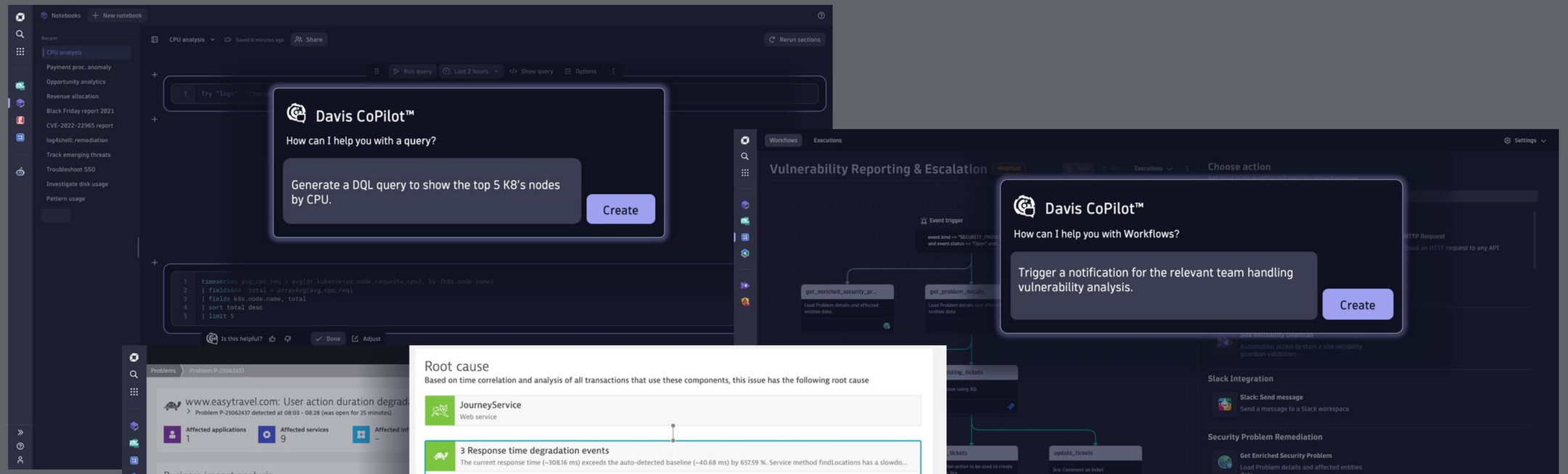
Anomaly Detection

Utilizing AI, organizations can identify unusual patterns swiftly, enhancing operational responsiveness.

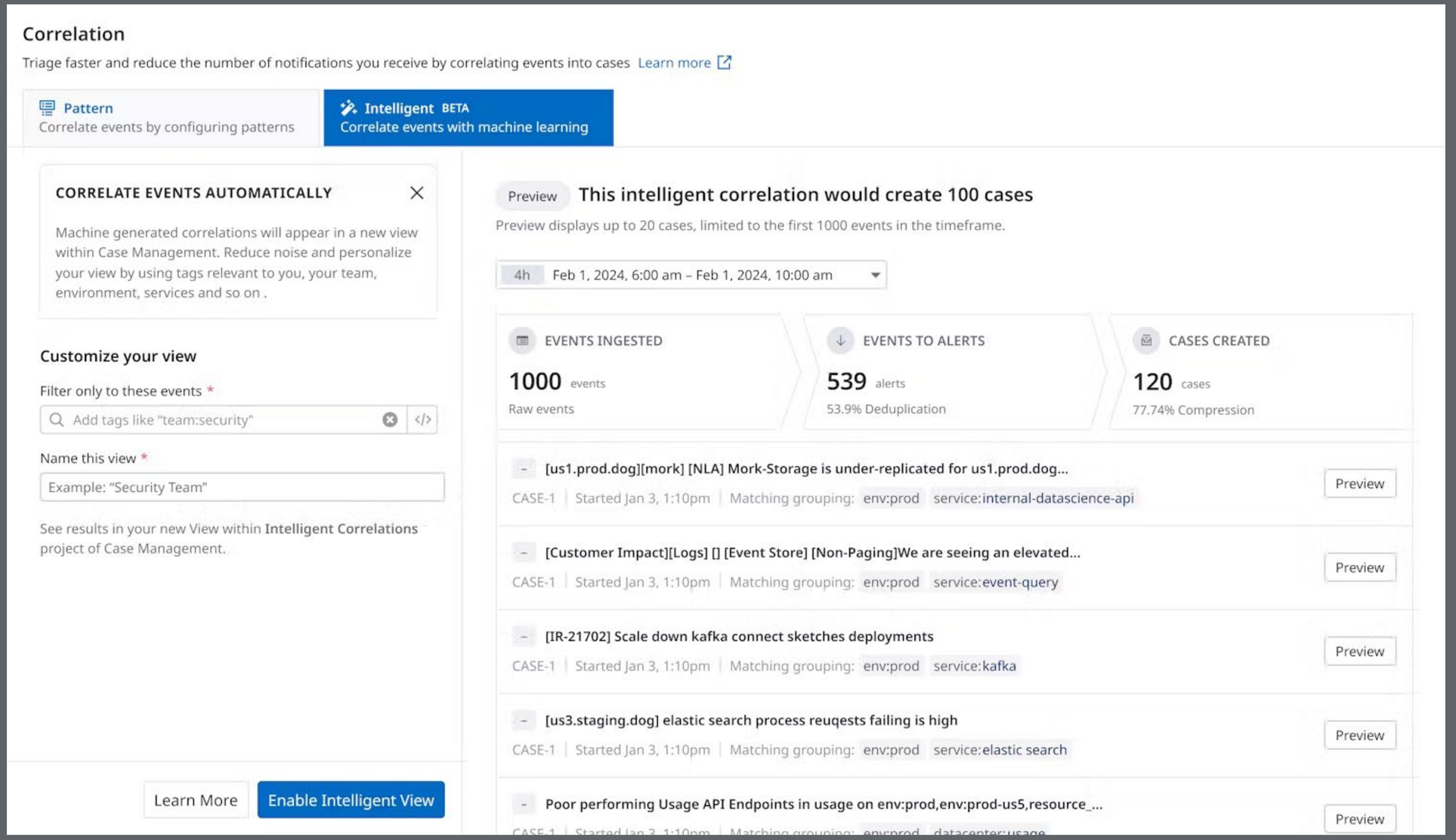
Event Correlation

AI correlates events across systems, enabling faster understanding of complex interactions and dependencies.

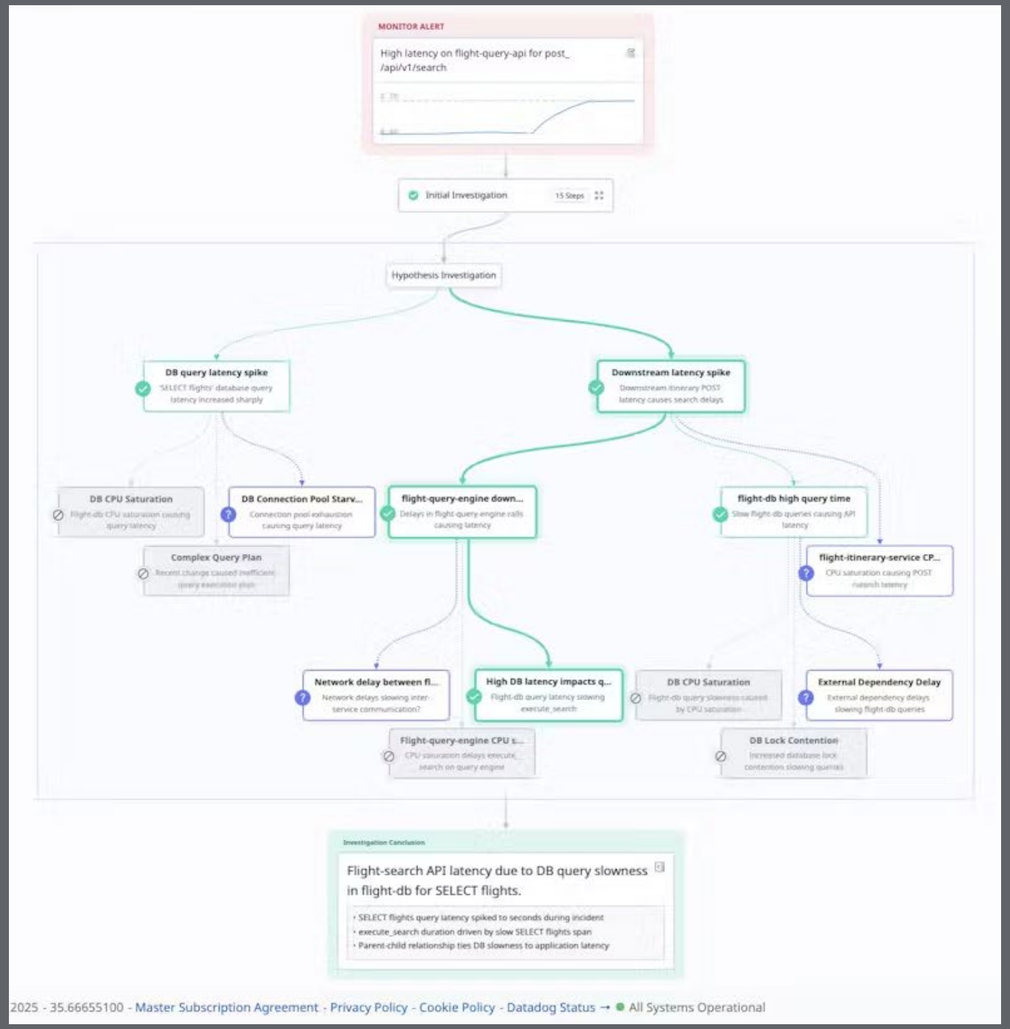
Natural Language Queries

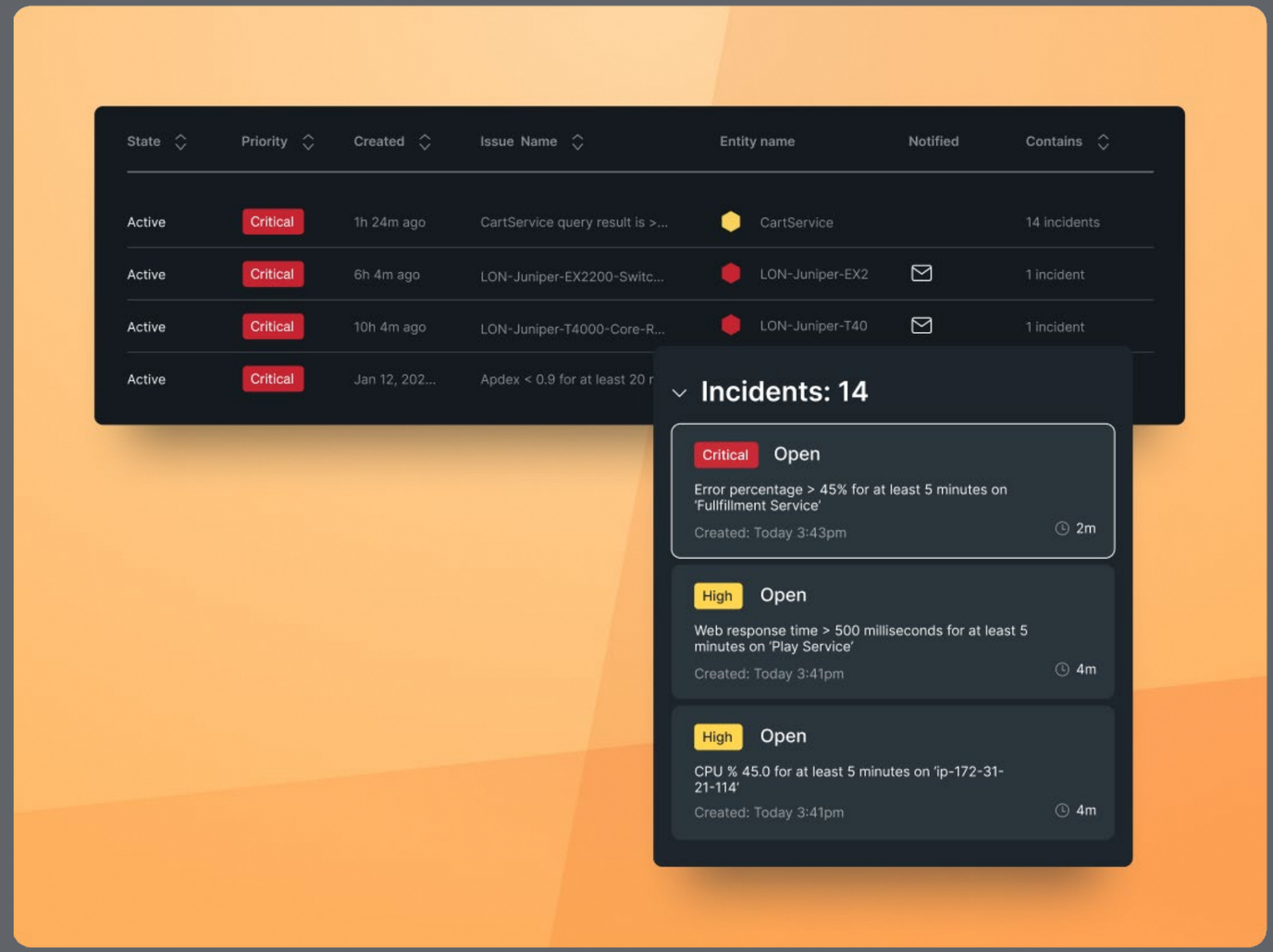


Intelligent Alerting



Event Correlation





Thank You for Your Attention!

Reach out if you have any questions!

Josh Carr

(678) 612 - 8168

Email Address

josh.carr@rocassociates.com