

Incident.MOOG Use Cases



Discover how Incident.MOOG is solving Operations and DevOps problems and providing tangible benefits to customers in a variety of industries. With Incident.MOOG, you can:

- Scale up to handle projected growth in event volume driven by changing business needs
- Enable Ops and DevOps teams to support continuous delivery and adapt to dynamic, complex environments
- Reduce or eliminate dependence on costly legacy systems

Read these use cases to learn more.



Online Web Commerce:

PROBLEM: Service degradation of customer-facing applications.

SOLUTION: Incident.MOOG correlates customer Tweets and IT telemetry to detect problems that were “undetectable” by the legacy system.

BENEFIT: Incident.MOOG is preserving the revenue stream by identifying problems as they occur and minimizing the impact on customers.



DevOps:

PROBLEM: DevOps efforts to accelerate application roll out are stymied by legacy processes that require writing complex rules for problem detection.

SOLUTION: Incident.MOOG immediately ingests application log files and other unstructured data to spot anomalies in apps and infrastructure during QA/Testing – without requiring rules at the application or device level.

BENEFIT: New applications roll out faster to support business requirements; using Incident.MOOG in QA/Testing helps ensure uptime and successful production deployment.



Data Analytics/IoT as a Service Provider:

PROBLEM: Support anticipated 2X to 5X increase in data produced by an “Internet of Things” infrastructure that underpins a rapidly growing retail analytics service.

SOLUTION: Incident.MOOG correlates events and alarms from video cameras, sensors, appliances, PoS terminals and inventory management systems, analyzing the data in real time to identify potential service-affecting situations.

BENEFIT: The retail analytics service can cost-effectively scale operations and enhance the customer experience even as their underlying IT infrastructure grows more complex and dynamic.



X as a Service (XaaS) Provider:

PROBLEM: Simplify the process of segregating IT operations data by client and by service to support multi-tenancy without maintaining rules at the device level.

SOLUTION: Incident.MOOG segregates data at higher level in the overall fabric. Each customer sees only their data while the service provider’s central NOC gains a holistic view across all clients and technologies.

BENEFIT: Incident.MOOG is enabling faster on boarding of new clients and services, easier segregation of data by client and service, and compliance with regulations requiring data segregation by country.

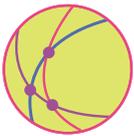


Cloud Service Provider:

PROBLEM: Lack of visibility into anomalies and behavior of leading-edge cloud software and infrastructure technologies were impacting the customer experience in beta testing, delaying production launch of new services.

SOLUTION: Incident.MOOG ingests event streams across each of the cloud software and infrastructure components, providing cross-domain visibility and early detection of service-affecting incidents.

BENEFIT: Ops teams now have a head start on seeing and resolving problems before customers notice, increasing early customer satisfaction and accelerating production service launch by six months.



Internet Consumer Portal Provider:

PROBLEM: Manage hypergrowth without adds to staff.

SOLUTION: Incident.MOOG reduced 300 diagnostic threads into 1 during a major outage.

BENEFIT: Incident.MOOG is enabling a 10X improvement in throughput of the IT operations support staff, especially during outages when it matters most.



Mobile Service Provider:

PROBLEM: Fine-tuned NOC filtered out “non-critical” failures – thousands of transient outages – even though they were affecting quality of service and increasing customer churn.

SOLUTION: Incident.MOOG clustered 36 alerts from several devices to reveal cause of transient outages affecting several thousand customers.

BENEFIT: Incident.MOOG is having direct and measurable impact on reducing customer churn.



Financial Services Provider:

PROBLEM: Trades failed to settle – application SMEs couldn’t diagnose the problem.

SOLUTION: Incident.MOOG clustered events from multiple silos to identify that the cause was database, not the application.

BENEFIT: Incident.MOOG identified the cause and impacted parties 45 minutes faster than the legacy system.

Contact Moogsoft today to learn more. Visit www.moogsoft.com or email info@moogsoft.com

For more information, visit www.moogsoft.com.

U.S. 140 Geary Street
Office 1000
San Francisco, CA 94108
+1 415 738 2299

U.K. The Sanctuary
23 Oakhill Grove
Surbiton KT6 6DU
+44 208 399 8266

NY +1 646 843 0455

Singapore +65 3158 4393