Whitepaper

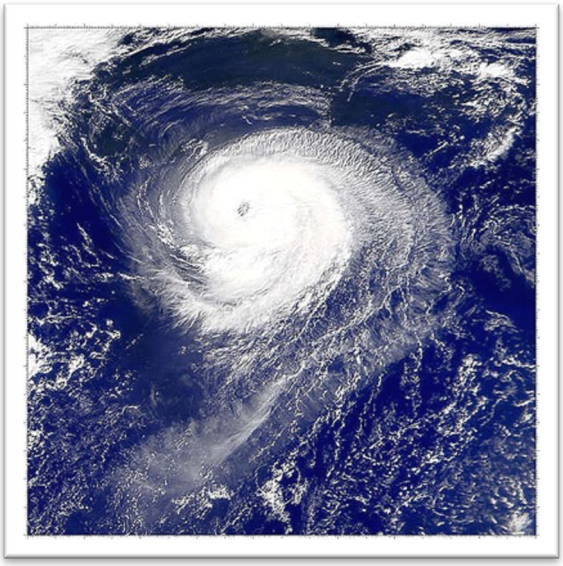
Escaping the Alert Vortex with AIOps

Maintaining perspective at the eye of the storm and taking action

Jason English

Principal Analyst, Intellyx

May 2020



Like a hurricane, the operational alert fatigue generated by today's fast-changing complex hybrid IT systems can draw development and IT Ops teams into a confusing maelstrom of events and notifications.

How can a modern software-driven business achieve escape velocity against the pull of this storm, and reach a point of clarity?

This paper will explore how AIOps can help IT teams navigate through the debris of events, combining observability and intelligent filtering to not only reduce application risk and resolve incidents faster, but deliver a meaningful competitive advantage that satisfies both end customers and valued employees.

Challenges: Warning signs of complex applications

Since software defines much of every modern enterprise, warnings issuing forth from business applications are no longer the exclusive problem domain of IT Ops teams. An overabundance of alerts will pull everyone into the fray: development, InfoSec, support, customer service, management, and so on.

What are the contributing factors to such an alert storm?

Hybrid IT complexity. Today's software is deployed into highly distributed hybrid IT environments, usually made up of a mix of hardware, VMs, and containers, running in data centers, private, or public cloud services. Workhorse enterprise applications can still operate alongside workloads running in containers as microservices or functions. An ideal hybrid architecture keeps what works, and either abstracts away or replaces what isn't as flexible as needed.

With millions of combinations of conventional services and ephemeral microservices running at any given time behind every touchpoint of the business, even the most well-

architected application infrastructure is bound to generate plenty of interoperability and timing conflicts.

Hyperaccelerated delivery. Real continuous delivery was once reserved for a very few 'born in the cloud' startup leaders -- Netflix, Facebook, Uber, Google, and the like. But now, even once-considered-stodgy companies may be pushing hundreds or thousands of releases a day.

DevOps practices are now entering the mainstream, with more than 20 percent of companies now considered 'Elite performers' who on average conduct 208 times more code deployments and recover from issues 2,604 times faster than their low-performing peers, according to the *DORA 2019 State of DevOps* report. This kind of DevOps velocity is a nice problem to have, of course, but it has the potential to generate alerts at an equally high velocity.

Automation and monitoring. How can human IT teams possibly keep up with so much complexity and continuous change in the wind? DevOps teams lean heavily on the equalizing capabilities of automation and monitoring.

Every once-manual process of testing, continuous integration, build, deploy, and operation gets its digital equivalent as coded automation. And for each automated script or procedure that deploys a new server or node in any form, there needs to be an associated way to measure and report back its status and outcomes, so that success or failure can be confirmed by the automation.

Wait, aren't these all good things? Very much so! But think about the byproduct or exhaust adding up from this DevOps energy.

Each individual instance of a deployment, and all of its accompanying status notifications: successes, logging, metrics, security issues, failures – generate events that could potentially be reported back to IT teams as issues worth paying attention to.

The storm of events, state data, and alerts coming back from a complex, constantly changing system causes critical signals to get overlooked by the team. There can be so

many notifications, in fact, that the alerts themselves can become as dangerous as the potential errors they might be reporting.

Ops managers, SREs, core developers, and business managers are getting paged in the middle of the night, and toiling away on firefights and in all-hands war rooms. Customers are dissatisfied by poorly performing applications.

Alert fatigue sets in, and teams lack the relevance to prioritize and pinpoint an appropriate response. Is there an escape from this vortex?

Approach: Rethinking the requirements for AIOps

"I'm sorry Dave, I can't do that."
-- HAL9000, in 2001: A Space Odyssey

It would be nice if an omniscient artificial intelligence superhero could focus a laser eye on the heart of the alert storm and save the day here. While that may be too futuristic to ask for, we can still look to AIOps strategies that exist here and now.

What is the 'AI' in AIOps?

A big analyst firm defined AIOps as the combination of *artificial intelligence* (as the 'AI') atop IT Operations monitoring and automation technology for finding root causes of issues and prescribing solutions.

If you look back 10-15 years ago, when AIOps was emerging from Ops monitoring and reporting tools that were event data into early ITSM and security dashboards, that definition would have been rather accurate.

These early forms of AIOps might have focused on better visibility of logs, or code-level searches for anomalies, with some visualization of incoming traffic and intelligent spotting of trends and exceptions, so teams could optimize or remediate problems.

Augmented or applied intelligence? Newer definitions of AIOps called it *augmented* intelligence, where there is a machine learning filter that can act as the digital expert, curating important trends by observing operational trends and then assisting humans with recommendations when they are on the task of resolving problems.

Let's further refine AIOps as *applied intelligence* for ops, a category of solutions that learns from deep observability, bringing only the most relevant issues and associated alert data to bear exactly where humans need it the most.

Observability, or external visibility into the internal workings of operations infrastructure and the application workloads and code running on it, is also a requirement for AIOps solutions to exist.

Without observability instrumentation feeding millions or billions of logs, traces, and metric data points into the AIOps solution, there would be no alert storm for IT teams to deal with in the first place -- but that's a different problem entirely, because an unmaintained business system would be doomed to eventually fail in front of customers.

Integration to plan, collaborate, and execute. Lastly, an AIOps solution requires a broad set of integrations with upstream sources of alerts, whether coming from an observability platform like New Relic One; ingested from ITSM, SIEM, and incident management vendors like PagerDuty, ServiceNow, or Splunk; or fed from a heterogeneous mix of installed enterprise data and open source tracing and metrics tools like Prometheus, Grafana, Nagios, etc.

Then to complete the other bookend, AIOps feeds back into the downstream planning, collaboration and execution destinations. That may include confirmations back to the observability platforms, as well as any other solutions that feed alerts to AIOps, like OpsGenie, Jira, and Zendesk. AIOps-filtered alerts can be accepted as tasks in project management tools, runbooks and Slack channels.

In general, AIOps solutions should be able to filter and route problem-solving work with all relevant issue data directly to responsible teams, through whatever tools they are using.

Solution: Alert gateways for AIOps

In a real storm, the wind or water itself doesn't always cause the most damage; it's the debris that the storm carries on the wind. In an active application environment, the role of AIOps is to act as a series of buffers for alerts in flight, so the team can avoid alert fatigue and toil, and focus on resolving only the important issues.

The evolution toward AIOps could be represented as a series of five buffers, or gateways that alerts must pass through to be resolved and put in the rear-view mirror.

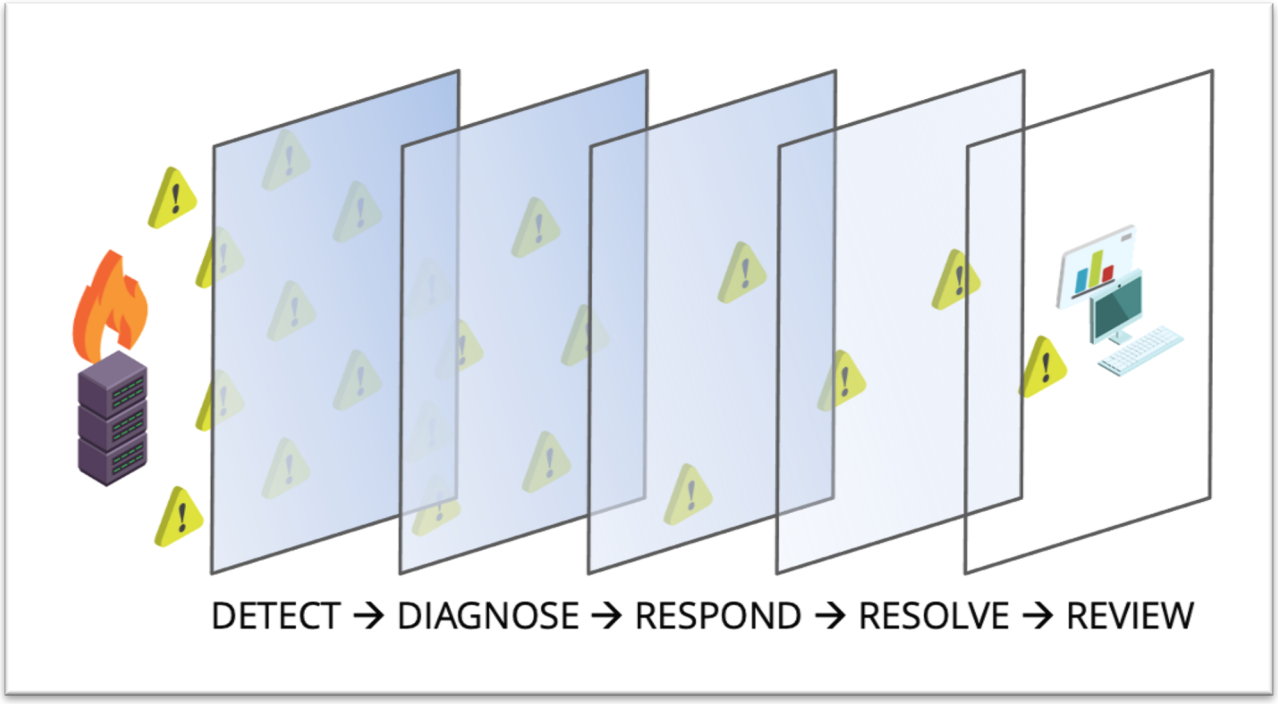


Figure 1: Five gates of AIOps solutions, designed to buffer alert noise and decrease time to resolution.

The first gate: Anomaly detection

At this level, AIOps is taking in an unfiltered, continuously increasing wave of incoming events and alert notifications as application use scales. Here's where teams and data feedback inform the AI model with thresholds that describe what alerts might simply be meaningless notifications or flapping issues that don't impact production, and what issues exceed boundary conditions, and constitute a potential anomaly.

The second gate: Diagnosis

Here's where the applied intelligence of AIOps really goes to work, as the number of detected anomalies in the alert stream is still very high. Unlike a typical monitoring solution that would just pass out-of-bounds alerts onward, AIOps applies discretion based on human-generated or machine-learned guidelines.

By correlating alerts against issue patterns to determine which ones may present a risk to customers or the overall health of the system, AIOps can collect relevant issues into incidents worth responding to.

The third gate: Response

Now that alerts have been reduced to a smaller number of incidents, it's time for AIOps to triage them and prioritize by severity, and schedule based on resource requirements -- exactly where human teams often encounter 'decision paralysis' delays or mistakes in assignment. The definition of 'who is responsible' isn't as simple as pointing to an org chart, or putting a name on a trouble ticket -- it's targeting the appropriate system of record, so the responders, whether human or computer, are notified depending on which components or systems are being affected.

The fourth gate: Resolution

Now that a high-priority problem has been identified and responders have been selected, the AIOps solution should automate delivery of all relevant incident data to the tools the response teams are working in, enriched with its full context and guidance for root cause analysis.

Whether issues are routed to PagerDuty, Slack, Jira, ServiceNow, VictorOps, or any other ITSM, Ops, or support tools of choice, it's critical that responsible parties can get right to work on the problem in the tools they are already using every day.

The fifth gate: Review

Now that we're out of the storm and in the clear with perspective on the outcomes of our issue resolution process, it's important to reflect on what worked well, and what didn't. Issue resolution data gets fed back as tuning data for the machine learning model of AIOps, so responses are optimized for future cycles.

The organization should benefit as well, with business systems of record getting value data to feed into KPIs. IT and DevOps teams should definitely record learnings, benchmarks and discovered procedures into knowledge bases or runbooks regularly as well.

What's cool about all of the above gates? While these buffers soak up and prevent alerts from unnecessarily hitting teams, they don't slow them down. AIOps needs to act almost instantaneously to allow optimization at web velocity.

Value: Issues generate improvement

AIOps helps organizations deliver improvement across the primary SLOs for application reliability and resiliency: MTTR (mean time to resolution) of issues, less system downtime and more time between failures, and faster application response time because of better maintenance.

All of these metrics can be translated into better customer experience, resulting in financially recognizable values of less order abandonment, lower customer churn rates, more sales, and fewer chargebacks or other penalties.

More importantly, AIOps extends the value of the entire digital delivery function of the enterprise because of the drastic reduction in alert noise it creates.

Vendor surveys estimate 80 percent or more of the alerts presented to NOC, IT ops, and SRE teams are deemed irrelevant in mid-to-large sized enterprises. A great deal of toil can be saved, freeing up valuable employees to work on the hard, important problems.

When you multiply that reduction in fruitless labor cost by the number of applications and infrastructure assets that could generate alerts, multiplied by the amount of times groups in DevOps and IT were handing off issues to each other, a significant labor savings is at stake, as well as a higher rate of employee retention.

Lastly, the AIOps solution shouldn't require teams to undergo massive retraining, nor learn new languages and issue resolution tool chains. Leveraging existing skills and tool investments wherever possible lowers the adoption cost and accelerates the time-to-value of AIOps.

Signify Health has been employing combined APM observability and AIOps solutions from New Relic across 16 delivery teams to proactively detect and filter action items, as they managed many of their systems in a hybrid IT on-premises and cloud architecture over the last two years.

"New Relic AI's proactive detection capability was very easy to set up and use. There were zero agent configuration changes or deployments needed," said Jeffrey Hines, Senior Site Reliability Engineer, Signify Health. "Specifically, it helped my team achieve speed, agility and provided operational visibility which ultimately helps us reduce incidents, integrate machine learning and analytics into operations and improve overall customer experience."

The Intellyx Take

Escaping the alert vortex caused by ever-increasing complexity and change requires a clear perspective and the wherewithal to act decisively on only the most important issues.

Maybe your company is already doing AIOps. But chances are, your company hasn't adopted a mature form of it yet. *(Gartner predicts as many as 20% of companies will have adopted AIOps technologies by 2023.)*

AIOps augments human capabilities with applied intelligence and observability as a fulcrum, presenting relevant data to resolve production issues within familiar tools.

About the Author

Jason "JE" English is Principal Analyst and CMO at Intellyx., a boutique analyst firm covering digital transformation. His writing is focused on how agile collaboration between customers, partners and employees can accelerate innovation.

He led marketing efforts for the development, testing and virtualization software company ITKO, from its bootstrap startup days, through a successful acquisition by CA in 2011. JE co-authored the book Service Virtualization: Reality is Overrated to capture the then-novel practice of test environment simulation for Agile development, and more than 60 thousand copies are in circulation today.

About New Relic

New Relic is the industry's largest and most comprehensive cloud-based observability platform built to help customers create more perfect software. The world's best software and DevOps teams rely on New Relic to move faster, make better decisions and create best-in-class digital experiences. If you run software, you need to run New Relic. Learn why more than 50% of the Fortune 100 trust New Relic to make the world's software run at newrelic.com.