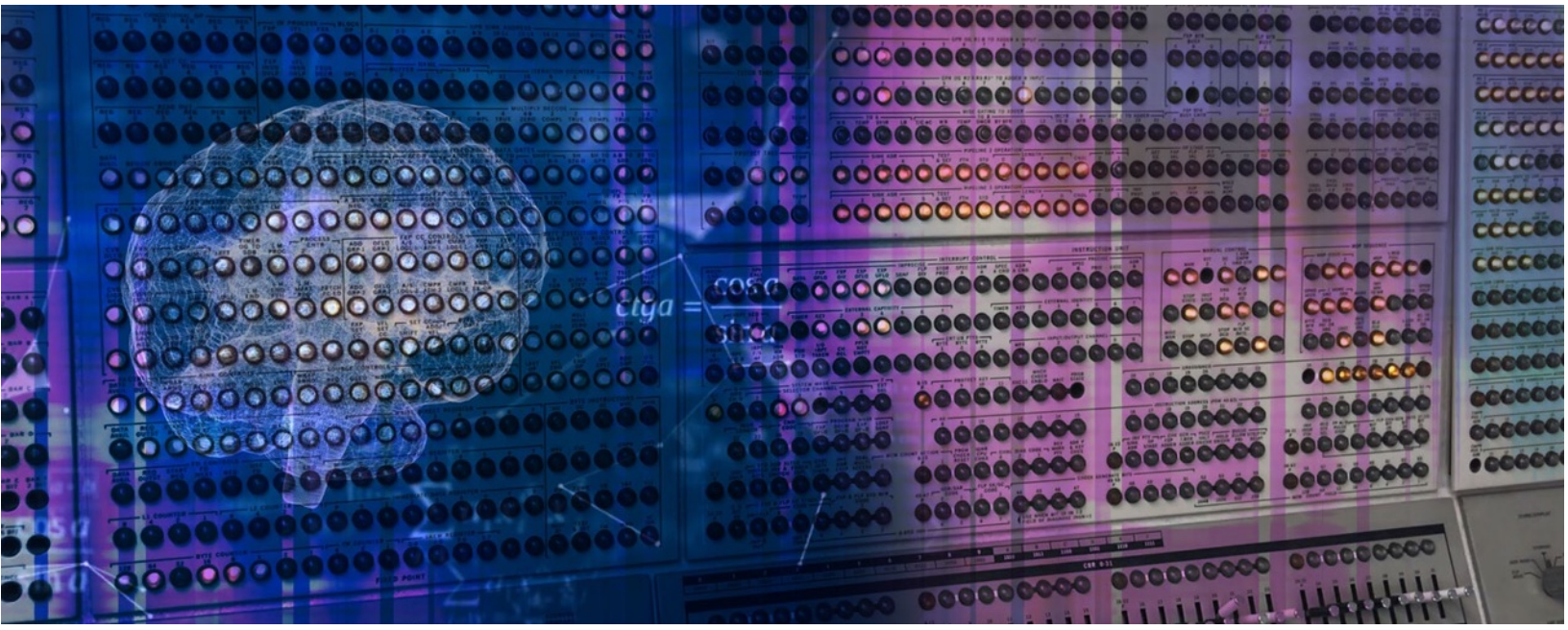

# Modern Incident Management: Evolving Toward a Positive Incident Culture

*An Intellyx Buyer's Guide*

**Jason English**
Principal Analyst, Intellyx

**September 2021**

## MTTD. MTTI. MTTA. MTTR.

No matter what corner of IT we happen to work in, some part of our job performance is likely tied to our ability to quickly identify software and system-level flaws and resolve problems.

For the longest time, it seemed like the primary metric of incident management was measuring the MTTR (the mean time to resolution) it took to find and fix something.

Now the market is evolving, and perhaps it's time the 'mean-time' metrics are replaced by a kinder, more positive incident culture.

We must lean more than ever on digital collaboration with remote co-workers and flexible team structures to support fast-changing software and cloud infrastructures. Success in this new environment will be measured by improvement in our overall organizational resilience -- the ability to learn from mistakes, and to bounce back faster and better over time.

This guide will provide an understanding of the ideal criteria for modern incident management as a bridge between engineering, IT operations and management stakeholders. The fulcrum is shifting toward a positive incident culture by changing the way organizations think about incidents.

# How did we get here?

Before incident management solutions existed, system issues naturally emerged from user help lines and sysadmin support desks which once relied on trouble tickets, binders of runbooks, and of course, spreadsheets.

## Pushing issues and tickets

As the IT function started to become core to the operations of many businesses, the first differentiated ITSM and SecOps solutions started appearing.

Remedy and JIRA became mainstays of sysadmins and support desk ticketing. A differentiated security team also arose, with its own SoC desk management systems. In some ways, at the lower usage velocity and relative lack of connections of many systems at the turn of the century, these early ticket-based issue response systems were adequate.

Capturing and routing issues became more complex. The first ITSM (IT service management) and support platforms such as ServiceNow, Zendesk, and an expanded Atlassian suite joined rising SOAR (security orchestration automation & response) and SIEM (security incident event management) tools from vendors like Splunk, Elastic and Palo Alto Networks to keep teams aligned on multiple tasks.

At the same time, Ops teams increased focus on secure application delivery: monitored gateways and firewalls--accompanied by developers instrumenting software with vastly improved error reporting and telemetry, which led to the presence of today's observability tools for pinpointing failures and performance problems.

## From issues to incidents

As service desk tools evolved, companies started to collect and triage issues, promoting the most severe ones as ***incidents*** requiring paging an individual or team response.

A great bifurcation happened here as systems became infinitely more complex, heterogeneous, and service-oriented. Incidents could no longer be contained within any team, nor any company's four walls.

Existing ticketing programs just didn't have enough flexibility to take inputs from dozens or thousands of event sources, nor coordinate responses from different groups, third-

party service providers and others. High-performing organizations began measuring their performance using incident response times.

PagerDuty entered the IM space with a big splash, and Atlassian, Splunk and even some of the observability/monitoring vendors built platforms to help companies rally resolution teams in moments that matter.

## Trying DIY task management tools

Many companies have team members that will attempt to go 'rogue IT' and manually manipulate spreadsheets and other productivity tools to fill holes in incident coverage left behind by their officially approved support platforms.

This makes sense, as Microsoft Office, Google Apps, Slack, Asana, Basecamp and Trello represent common shared communication and task frameworks that are readily available on employee desktops, and easy and intuitive for users to get started with right away.

Incident management inherits many of its requirements from productivity and task management tools as vital sources of input--and obviously, these primary communication channels aren't going away anytime soon. Group video calls and screen shares in Zoom, Teams, Skype and many more also became a part of this toolset.

However, attempting to manage a complex array of inputs, meetings, and resolution steps with a task-list view--or via shared spreadsheets and chats--results in a completely ungovernable mess with very limited visibility.

# Key criteria for Modern Incident Management

Fortunately, ***Modern Incident Management (MIM)*** solutions are arising to specifically address the gaps in existing toolsets, as a layering on of new functionality and services that can string together all the various productivity and support tools with a collaborative sense of empathy.

## Definition of buyer is changing

Since management of the entire IT estate is considered a core function, incident management would usually roll up to a PMO reporting to a CIO, who would make a major investment decision for the whole enterprise. In today's MIM space, software dev managers, SREs, service delivery orgs AND the C-Suite are taking more interest in company-wide improvements.

Much of this change in buying patterns stems from startups that can instantly have the need to support very sophisticated, high-volume software and hybrid IT infrastructures, but don't have the tolerance for high cost of entry. As these startups grow, talented leaders will naturally partner with and migrate to other firms with their own rigid issue resolution processes in need of modernization.

## To modernize, work positively

A new breed of purpose-built vendors born for cloud-based software delivery arose with modern approaches for continuous integration, delivery automation, observability and tracking customer satisfaction and service level objectives (SLOs). 'Everything-as-code' automation and open telemetry instrumentation make it possible to feed incident management with better signals.

Since we now have all of this rich input to draw upon, which activities need more structure to achieve a modern incident management process, and which do not? This prioritization leads us to a positive incident culture, rather than negative reactions to failures.

| Activity | Legacy culture | Current standard | Positive culture |
|---|---|---|---|
| **Issue Capture** | Phone and support email. Ticketing systems (Remedy/JIRA) | Automated alerts, customer issue reports, SLA violations | Filtered alerts from observability, trending smaller problems |
| **Collaboration** | Pagers, group calls, group emails. War rooms for Severe issues.. | ITSM, SOAR, task mgmt dashboards<br>Group email, group Slack | Just in time team organization, less duplication of effort, some automation |
| **Resolution success** | Closed issue rate. Fewer reported issues. | Time to resolution metrics (MTTR, etc.), reduced incidents, knowledge library | More L3-L2 incidents, SLOs optimized, captured resolution and response scripts |
| **Feedback** | Bring up in next ITIL review, documentation work | Scheduled postmortems, incentivize closed incidents | Fast, frequent retrospectives, customer & employee experience SLOs |

*Figure 1: Positive incident culture behaviors. A well-balanced team process and modern incident management toolset incentivizes beneficial customer and employee outcomes.*

We can see that success in MIM means far more than encouraging managers handing out Lucite awards for faster MTTRs. A positive incident culture gives extended teams a shared sense of responsibility for continually improving customer and employee experiences.

From initial ideation and design of a new service or product, through every step of its delivery to customers and support in the field, to feedback and retrospectives on anything that went wrong (or *almost* went wrong), the defining characteristic of a positive incident culture is the celebration of finding more incidents, not less, and dealing with them earlier, before they become severe.

What levers do we have for selecting tools that help us reach this positive state for incidents?

## From granular tasks to enterprise-wide responsiveness

"Doesn't play well with others" is a common reason for adoption failure of any new incident management solution. If the new incident system is a walled garden without simple event-driven APIs, it will generate alert fatigue among response teams as it fails to integrate with existing systems of record and productivity tools.

A modern incident management solution should be able to accept messages, documents, metrics and alerts from a wide variety of sources and services, and then associate them with at the individual task or team member level, or across much broader federated response teams and incident categories that represent the overall performance of the enterprise across longer time horizons.

**Gusto** is a SaaS leader in the payroll, employee benefits and HR automation space that built up its own incident reporting tools atop its systems over the past few years. But as the company started maturing an SRE (site reliability engineering) practice and expanding its remote workforce, their definition of incident management grew beyond IT.

When a massive power grid failure and freeze in Texas threatened their remote employees' safety, Gusto was able to spring into action and coordinate on-call teams to work to ensure employee access to internal resources despite the outage, much like any real-world relief organization would coordinate to deliver services around a natural disaster.

## Purpose-built for open, positive incident visibility

Fine-tuning a system to address the specific needs of modern incident management requires getting past the existing IT support desk definitions of reduced downtime, fewer errors, lower support labor costs, faster MTTR and fewer SLA violations.

A positive incident culture demands an open view into incident detection, triage, and resolution in flight, with work rails for spotting, reporting, and contributing to the resolution of more incidents, not less. Everyone with a stake in the success of an

application should be able to see and help resolve more Sev3/Sev2 incidents, thereby reducing the risk of Sev1 catastrophic failures in production.

Even in scenarios where troublesome failures occur, end customers are much more likely to respond favorably to a company that offers transparency into what went wrong. The responding team's positive incident response and detailed documentation assures them their partners are on the job and continually improving. In fact, regularly report incidents can pay dividends in customer engagement and satisfaction.

**Vercel**, a leading front-end delivery platform, started out using a basic instance of Kintaba as a back-end system for critical task management among remote developers resolving high priority product issues.

As the company's customer base and usage levels scaled dramatically, their teams continued to embrace a positive incident culture to the insights generated by observability tooling, essentially flagging and coordinating real-time responses to functional and performance problems as soon as they were trending in the wrong direction.

# The Intellyx Take

Once, teams were conditioned to think that IT incidents were bad things. Teams were incentivized to close support tickets as fast as possible, in hopes they wouldn't reappear as catastrophes to be resolved through late-night debugging sessions and all-hands war rooms.

Now we're chasing a new goal for modern incident management that highlights everything going on to improve responsiveness and resiliency within the enterprise, while dealing with the inevitable externalities of the outside world that can impact our systems.

> *What could possibly go wrong?*
> *When you find an incident, don't hide it.*
> *Celebrate it. Learn from it.*

A positive incident culture moves organizations beyond blame and punitive measures for failure and generates process-led improvements that intentionally improve resiliency over time.

# About the Author

Jason "JE" English (@bluefug) 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.

In addition to several leadership roles in supply chain, interactive and cloud computing companies, Jason 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.