Deep Dive

Key Considerations for Selecting a Next Generation Monitoring Tool

September 2015

Contributing Analyst(s): Tim O'Brien

Gleanster™

solarwinds

Executive Summary

Have you ever thought your monitoring solutions might benefit from their own monitoring solution? Ever been overwhelmed by the frequency and volume of alerts to the point that things are overlooked? With 94% adoption, performance monitoring tools and features are a staple of mid-to-large enterprise IT environments. But research suggests five out of ten IT professionals are initially unsure what causes performance problems in production, which leads to a lengthy and often manual investigation process. So it's no wonder IT professionals use an average of three separate monitoring tools; these include monitoring features in ancillary tools that lack a mature model for alerting, vendor-specific monitoring tools that focus on single subsystems, and niche application performance platforms. But are there tradeoffs between these different approaches to monitoring that leave IT vulnerable to inefficiencies and cumbersome investigation practices? Are there shortcuts to success?

This document explores best practices for maximizing investments in IT monitoring and alerting technology. Using trends and research from the Q3 2015 survey "2015 State of Application Performance Monitoring Survey" (n=262 IT professionals) on monitoring and alert initiatives, this piece provides guidance for organizations trying to identify flexible out-of-the box monitoring and alerting systems that can be configured to meet the specific needs of an organization, rather than generic alert capabilities that often fail to deliver insight or mitigate the risk of downtime.

THIS REPORT IS MADE AVAIALBLE COMPLIMENTS OF:

solarwinds

For more information on SolarWinds, click here.

ABOUT THE DATA

The primary data presented in this report was captured in the Q3 2015 Application and Network Performance Monitoring Survey (n=262) and serves as the basis for this Deep Dive, which provides analyst commentary related to a particular aspect of the topic. The objective is to provide additional perspective and illuminate certain key considerations regarding the implementation of the related technology-enabled business initiative.

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

CHALLENGES & OPPORTUNITIES WITH PERFORMANCE MONITORING

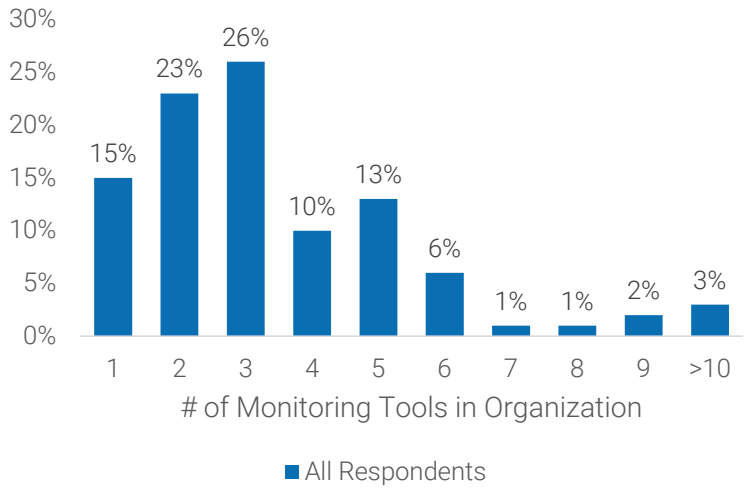
Some people love their monitoring tools. Hard as that is to believe, it is true. When this happens it's often a byproduct of a monitoring tool that is a "perfect fit" because it can be configured and tailored to the unique nuances of a specific IT environment. This Deep Dive will explore the features and practices that help transform monitoring from a stream of useless alerts into a strategic source of proactive insight for service management. As we will further explore, not all monitoring solutions are created equal, and monitoring isn't just a box to be checked for mitigating downtime.

Today's fully integrated monitoring and alerting systems make first and second generation systems look like child's play. Eighty-nine percent (89%) of Top Performers report that application performance management tools help identify the root cause of performance issues. These mature and fully configured monitoring tools are essential to maintaining service availability, but they are also a critical part of maintaining morale in IT. When an enterprise takes the time to integrate monitoring and alerting across an entire end-to-end system, it not only increases availability, it makes it easier to diagnose and troubleshoot systems without resorting to endless email threads or late-night conference calls.

Monitoring Challenges Increase with System Complexity

Systems have grown in complexity over the last few decades and organizations continue to support portfolios with an increasing number of applications. IT professionals overwhelmingly indicated that over the last 24 months the volume of applications they support has increased (on average respondents support 1.7 times more applications in 2014 than they did just three years ago—nearly double!) and 74% of IT professionals report they are under "strong" pressure to deliver software more frequently, with fewer defects. Rising customer expectations for rapid access to data anytime and anywhere don't help; these days, slow is the new broke.

FIGURE 1: HOW MANY TOOLS DOES YOUR ORGANIZATION USE TO MONITOR APPLICATION AND NETWORK PERFORMANCE ?



"Application & Network Performance Monitoring Survey", n=262, Q3 2015, Gleanster Research

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

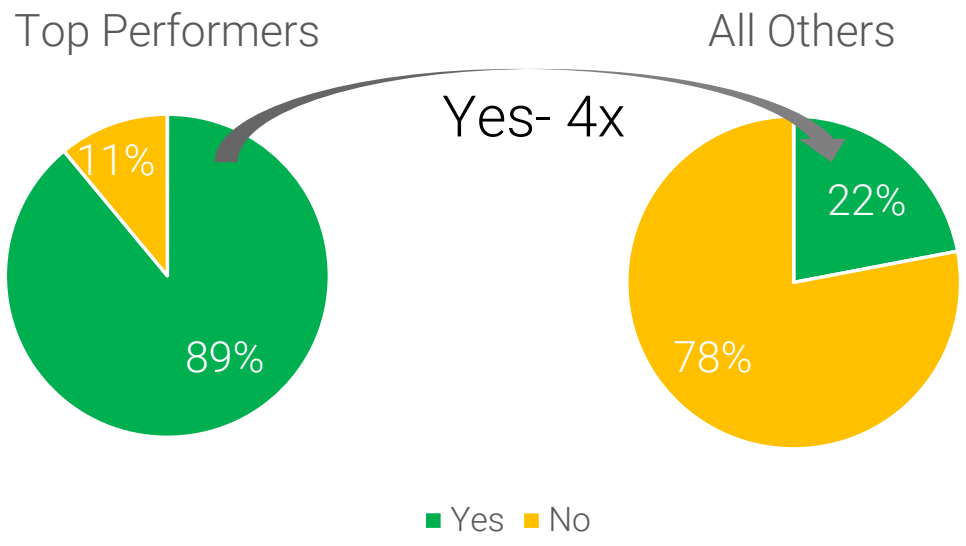
Where an enterprise could have been modeled on a single platform ten years ago, this same enterprise is now a collection of independent databases, micro-services, specialized middleware, third-party apps / services, and agile software development teams all creating interconnected systems of increasing complexity. Each of these independent subsystems is integrated into a unified whole to address customers' needs. In many cases, separate systems ship with a completely isolated monitoring and alerting solution, and it is the job of operations to create a unified view of system status to support ongoing operations.

In the past, an organization might have been served entirely by a single application sitting atop an Oracle or SQL Server database. Today there's more specialization, more systems, and a faster pace at which new systems are introduced. This explosion in complexity has created new pressures for teams responsible for end-to-end monitoring and alerting. These days it isn't as easy as just identifying which queries are taking too long to run. In a hierarchical, n-tiered, service-oriented architecture today's problems require a holistic approach to monitoring that can take into account the full scope of a system.

The Mythical Monitoring Nirvana: Automatic Everything

The ideal state for monitoring is a monitoring tool that understands all aspects of an end-to-end system. Database activity can be related to customer usage immediately, and all of the intermediate systems from a back-end database to a browser-based application (delivering a user experience) can be instrumented in such a way that administrators have a real-time view of the system status. As problems develop and as anomalous events occur, the ideal system can automatically understand expected errors versus critical errors. Unfortunately, organizations still struggle to realize this ideal state. We do know that Top Performers are 4x more likely than All Others to indicate that application performance management tools help identify the root cause of performance issues—a key source of differentiation for superior performing organizations.

FIGURE 2: OUR PERFORMANCE MONITORING TOOLS CAPTURE ALL OF THE INFORMATION WE NEED TO ADDRESS PRODUCTION ISSUES



"Application & Network Performance Monitoring Survey", n=262, Q3 2015, Gleanster Research

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

It is no longer enough just to set up alarms for disk space and CPU utilization. Today's systems demand monitors and alerts that can understand how to recognize performance anomalies and that understand the difference between a transient error that leaves the business or mission intact and an error that is impacting real customers.

The Reality of Monitoring for Most: Manual Inefficiency

The reality for many organizations is almost the opposite of this ideal end-state. For many, alerts are a very literally a joke. Systems are often configured with out-of-date thresholds, and there are communication disconnects between teams that create a sort of fog of war in the form of over-eager alerts. Anyone who has experienced this in the enterprise understands that it is easy for an entire department to let their guard down for a week when it comes to alerting, only to come into work one day and find 12,000 critical alerts filling up everyone's inbox.

Tools that require this constant, manual vigilance to set and adjust thresholds without offering any algorithmic recognition of anomalies are better than no monitoring at all, but just barely. While many organizations have invested in business intelligence tools to analyze customer activity data, we're still in the dark ages when it comes to system monitoring. Data scientists are busy analyzing customer data, but our monitoring systems are still using archaic tools that are designed to alert on static thresholds—tools that fail to understand that the database switch that happens every Sunday night, at the same time, isn't a critical alarm that should wake up the CIO.

Combatting Alert Fatigue: Have You Updated the Thresholds Recently?

In many IT departments the job of monitoring and alerting is still characterized by disparate technologies, static thresholds, and systems that need to be constantly maintained by whole departments of people paying attention to the nuances of alerts. All too often this just doesn't happen. Instead, monitoring and alerting tends to be something you'll get around to once you have time, and it is a constant unmet goal—something that becomes important only in times of extreme crisis. Indeed, just four out of ten respondents were confident their application performance management tools captured all the information they needed to address production issues (Q3 2015 Application and Network Performance Monitoring Survey, n=262). While monitoring is a box that must be checked, many organizations are simply resigned to the fact that monitoring and alerting doesn't live up to its potential. In other words, it doesn't "fit like a glove" because the monitoring tools being used require too much integration effort and manual configuration.

A large enterprise today is running a vast array of infrastructure, and the largest organizations across all industries tend to have a little of everything—Oracle databases, SQL Server databases, pre-packaged middleware application, several web sites, and a wide variety of both internal and external systems. This variety creates problems for IT professionals who are tasked with maintaining availability. The biggest challenge facing administrators is the variety and depth of the monitoring challenge they face. Many people responsible for monitoring are busy cobbling together multiple monitoring tools designed for different subsystems in a complex architecture. Some choose to standardize on monitoring solutions closely aligned with vendors, while others are busy applying duct tape between systems specialized to monitor and alert specific systems.

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

THE END RESULT OF POOR MONITORING: DISCONNECT, DELAY, AND DISTRUST

A common example in many large enterprises is the disconnect between a custom ad-hoc monitoring solution for an application developer and a specialized system for database monitoring. When problems invariably occur in production, an operations group is often at the mercy of two independent groups with separate dashboards, each trying to diagnose critical errors with only a portion of the information necessary. When an enterprise doesn't take the time to assemble monitoring and alerting into a single, unified platform they pay the price in the form of long email threads and more downtime.

When an application support team is reading one graph and a database administrator is reading another the end result is often blame and inefficiency, in addition to downtime and delay. Teams that are not supported by comprehensive monitoring struggle to communicate and struggle to understand the root cause of system problems. At an organizational level this much miscommunication leads to delay and distrust. An investment in monitoring is about more than just service availability; it's about morale and job satisfaction.

KEY AREAS TO CONSIDER WHEN CHOOSING A TOOL

While many organizations already have several systems configured to monitor complex IT infrastructure, the vendor community has only recently caught up with the rapidly developing complexity of today's modern enterprise.

Below are some key areas to consider when choosing a system for monitoring. (See Figure 3.) We will explore each of these in detail in the sections that follow.

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

Comprehensive Monitoring

- Out-of-the-Box Monitoring Best Practices
- Custom Application Monitoring
- Dynamically Adjust Metrics

Out-of-the-box Thresholds & Alerts

- Automatic Baselines
- Easy to Adjust or Customize Exceptions
- Goldilocks Alerting

Comprehensive Insight

- Showing System Data in Context
- Extensive Technology Coverage
- Regular System Updates

End-to-End Visibility

- Capable of Analyzing Dependencies
- Automated Relationship Monitoring
- Correlations Across Entire Architectures

Self-Service Capabilities

- Support Casual Configuration
- Intuitive, Easy-to-use Interface
- A Rich Community of Active Users

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

COMPREHENSIVE APPLICATION MONITORING

These are the metrics that drive application monitoring and that organizations watch constantly in the network operation centers that keep systems up and running. When evaluating monitoring tools for application monitoring the following factors should be considered:

Out-of-the-Box Monitoring Best Practices

While custom applications can be as unique as a fingerprint, most IT departments are running similar software packages and systems. Whether it is a .NET platform, a Java platform, a web server, a common CRM solution such as Peoplesoft, an Oracle database, or a SQL Server database, a monitoring tool shouldn't just ship with integration for these products; it should understand monitoring best-practices for these packages upon installation.

Monitoring solutions that "fit like a glove" don't force you to do all the tailoring. They come preconfigured with sensible defaults. Take an Oracle database running Peoplesoft as a common example: a monitoring tool should understand enough about the schema and the application to identify red flags for performance and manageability right away. Another example is an Apache httpd or an Nginx web server: a monitoring tool should understand where to look and what to measure without asking obvious questions.

Easy to Add Custom Application Monitoring

Custom application monitoring is the crown jewel of monitoring and alerting. While most of the monitoring in any IT operation is going to focus on commodity databases, networks, and middleware packages, it is the custom applications that deliver customer experiences that are the focus of high-profile monitoring initiatives.

The web site for an e-commerce company, the user interface for an online financial institution, or the online-presence of a large governmental institution—these are examples of complex applications that are often developed by hundreds of thousands of developers working on full-stack applications that are distributed across a collection of subsystems.

The ability to quickly stand up alerts for a custom application isn't just important for service availability and operations. The effort required to set-up alerts will affect time to market and agility for custom application development groups to provide client services.

Dynamically Adjust (Add or Delete) Metrics as Needed

Monitoring tools need to have a low-effort and dynamic model when it comes to adding or deleting monitors and metrics as needed. This is critical to both usability and the continued relevance of a tool, as IT systems are constantly changing both in terms of their architecture and their operational parameters.

When new applications can be stood up in days or weeks it is critical that your monitoring tool not require long lead times to add new monitors or metrics, and as the behavior of systems evolves your team must have the ability to add or delete new metrics in seconds without interrupting ongoing operations. It shouldn't have to feel like you are programming to define a new metric; it should be easy to configure new alerts, change thresholds, or delete alerts.

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

OUT-OF-THE-BOX THRESHOLDS AND ALERTS

Most monitoring solutions make an initial sale based on coverage, but the real value of a monitoring system isn't what's being monitored, it is how the monitoring system connects system behavior and statistics with actionable alerts—alerts that can trigger a meaningful response or a change in the system. When you are evaluating monitoring tools for thresholds and alerts, here are the features and characteristics you should be considering:

Automatic Baselines

Most organizations adopt a comprehensive monitoring solution for a system that is already supporting production. Modern logging solutions that have been designed to support large systems are capable of learning how to set common thresholds by watching system behavior. When you stand up monitoring for a system already in production you should have the ability to train the initial set of defaults by letting the system run for a day.

A good monitoring tool is going to be able to recognize the normal usage variations that occur over a 24-hour timeline, and it should be able to understand that a system supporting internal users or external customers will establish a set of common baselines. Thresholds for orders per minute or requests per second can be established after a baseline period. As a monitoring system watches behavior it should get more intelligent over time, adding in a model for week over week behavior that can account for year over year changes in metrics.

Easy to Adjust or Customize Exceptions

Just like adding and deleting application monitors, changing exceptions should be an easy and dynamic process. To say that a monitoring solution should have working thresholds right out of the box isn't to say that people won't be responsible for tuning thresholds. It's a fact that monitoring will require some investment of time to ensure that the right alerts are being sent to the right people at the right time.

A tailored monitoring solution understands the importance of exceptions enough to put the ability to create an exception for an alert front and center. It shouldn't take several clicks to configure an exception for an alert that will be triggered during planned downtime. Advanced and intelligent monitoring tools understand that exceptions are not exceptional—they are to be anticipated and made accessible and easy to use. As much as a system should ship with sensible defaults, it should have a concise and easy way for users to start to customize and create exceptions when they happen—because they will.

Not Too Much, Not Too Little

Goldilocks alerting refers to the Goldilocks Effect. As defined by Wikipedia,

> *The Goldilocks principle is derived from a children's story "The Three Bears" in which a little girl named Goldilocks finds a house owned by three bears. Each bear has their own preference of food and beds. After testing all three examples of both items, Goldilocks determines that one of them is always too much in one extreme (too hot or too large), one is too much in the opposite extreme (too cold or too small), and one is "just right". – (Source https://en.wikipedia.org/wiki/Goldilocks_principle)*

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

Initially users may be impressed by the detail and frequency of alerts. Most set up and configure monitoring systems because they want to be notified when problems occur. Indeed, it's better to get a notification from a monitoring tool than a call from your boss about a site outage or a problem with a customer-facing system.

But after several days or weeks receiving hundreds of unactionable alerts for expected conditions, most users will create an email filter and forget about a monitoring tool—ultimately ending up in the situation they were initially trying to avoid. In most cases over-eager alerts are as good as no alerts at all.

On the opposite end of the spectrum are monitoring tools that only send alerts after a critical error has occurred. These systems are so late and so reticent to send alerts that they merely serve as a confirmation tool that something must be seriously wrong. These tools miss the mark, as the most important alerts are rarely the alerts on critical error conditions, they are the alerts that lead up to a critical error condition. It is the warning messages that deliver the most value to organizations.

COMPREHENSIVE INSIGHT

Most organizations have a "balkanized" approach to monitoring tools—with systems dedicated to network alongside systems monitoring databases and various other monitoring tools. But blind spots occur when none of these solutions are integrated into a single view of system state. When you are looking for a comprehensive monitoring tool, here is what you need to look for in terms of how universal or comprehensive the monitoring coverage is:

Showing System Data in Context

An advanced monitoring tool can present users with an interface that can combine and relate data in new ways. System data related to CPU and memory usage should be presented alongside network graphs and application-specific metrics. Debugging performance and troubleshooting enterprise-scale systems involves investigations that span multiple data sources, and being able to combine and compare system data in context is critical to understanding the root cause of failure and performance issues.

If your monitoring tools don't allow you to quickly create a consolidated view of data across different systems and data sources, you'll be forced to take ad-hoc measures to combine this data yourself. Instead of being able to direct people at a single, consolidated view your IT professionals will become accustomed to taking screenshots of graphs from separate monitoring tools and pasting them into email threads, wikis, and Word documents. Sound familiar? Find a tool that can combine and display data in the same place.

Extensive Technology Coverage

Most IT departments are running a little bit of everything. There are back office systems running .NET and SQL Server sitting alongside systems running in Linux. A large organization may have multiple data centers while also making use of several public clouds from companies like Rackspace and Amazon. The name of the game for the enterprise is technical diversity, and your monitoring tool needs to be nimble enough to adapt to anything you can throw at it.

If your monitoring tool excels at monitoring Linux but ignores the details of a Microsoft platform

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

then you are going to experience coverage gaps in your monitors and alerts due to this weakness. Just like the previous section mentioned, your monitoring tool also needs to have a "normalizing" function for technical diversity. You should be able to compare performance data from different databases in a way that makes sense across platforms and applications. Make sure your monitoring systems don't penalize you for having a heterogeneous assembly of technology platforms.

Regular System Updates

Today's custom applications might be developed in .NET, Java, and Node.js, and the flagship databases may still be Oracle and SQL Server, but if there's one constant in IT it is the perpetually shifting sands of platforms, languages, and technologies. Look for a monitoring tool that has a record of not only adapting to new monitoring targets, but also of adapting to new monitoring mechanisms.

You'll want a monitoring solution that offers support for new and old products. Look for solutions that have adapted to the emerging prevalence of NoSQL solutions while also maintaining a solid commitment to the established relational databases that still define the market. You are going to want to measure potential monitoring solutions by the frequency with which they ship updates to monitoring "drivers." Are they shipping monitoring solutions that can be customized for new languages and server-side applications written in Node.js while also updating monitoring systems that incorporate new features of the JVM as they are released?

What you are looking for is a monitoring tool that has a rich array of plugins for monitoring targets that are constantly refreshed (either by an active user community or a dedicated vendor).

END-TO-END (E2E) VISIBILITY

It isn't enough to find a problem in an isolated subsystem and then pass the buck to another team. The new paradigm for monitoring systems is fusing data together into a single view. Instead of relying on several teams to coordinate on performance and identifying the root cause of system errors, the fully monitored organization should be using a monitoring tool that can provide a universal, end-to-end view of system state.

Capable of Analyzing Dependencies

When system failures occur or performance bottlenecks make themselves known through a monitoring system, this is an opportunity to use end-to-end monitoring to correlate how complex systems affect each other. Look for a monitoring tool that can perform analysis of alerts and system state across multiple levels of subsystems.

If an application experiences instability every time a critical index in a relational database is updated, your monitoring tool should help you identify signals that would otherwise be missed. Monitoring tools should be viewed as early warning radar systems constantly scanning alert history to identify how one system affects another. If you need to bring a critical system offline for an upgrade, your monitoring tool should be smart enough to understand how it will affect dependencies.

A monitoring tool should also be viewed as a tool that can inform decisions about architecture and risk mitigation. If you have a monitoring tool that can analyze dependencies it can also be

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

used to inform decisions about how systems can be designed to minimize operational risk and address performance bottlenecks. Don't look at monitoring tools as simply a source of alerts; the best monitoring tools provide overall intelligence that can be fed back to an ongoing process of designing architectures for maximum stability.

Automated Relationship Monitoring

A good monitoring tool with second-order intelligence about a system is able to analyze signals much like sonar or radar. By analyzing the frequency of alerts across an entire system, patterns that would otherwise go unnoticed can be uncovered automatically and users can be alerted to relationships in performance data that illustrate how systems are interdependent.

This automated relationship monitoring gives an organization the ability to avoid the "Butterfly Effect" common in complex systems where a single, seemingly insignificant data point in one system can cause unintended and catastrophic outcomes in another. A simple change to the way a table is stored on a disk in a database can lead to a small increase in latency that pushes a critical application metric to a criticality event resulting in system downtime.

Correlations Across Entire Architectures

A good monitoring tool is going to be able to recognize, forecast, and address how systems are related based on historical data using algorithms to identify correlations between metrics across the entire system.

The complexity in IT isn't present in a single subsystem. For the most part our IT systems are relatively simple. Even if you are building systems to identify subatomic particles or mange the International Space Station, most systems being managed in IT simply retrieve data from one database, process it, and deliver it to a customer or to another database.

The complexity in IT arises from the ways in which these systems are interconnected and good monitoring and alerting systems understand this challenge. The current class of monitoring systems can get by with just sending you alerts; the next generation of monitoring tools are intelligent correlation engines that can recognize patterns and make independent decisions based on input.

SELF-SERVICE CAPABILITIES

The days of having a single "monitoring" group responsible for configuring monitors and alerts across an entire enterprise are over. Most large organizations are migrating to self-service models where independent teams are responsible for operational monitoring of systems. An application team may be responsible for defining monitors and alerts for the systems they develop, and a database team is responsible for defining thresholds for storage and network performance.

Support Casual Configuration

Self-service capabilities in the enterprise cannot assume that end-users have been trained to use a complex tool. Large organizations require self-service capabilities that can be rolled out quickly and which don't require teams to hire a 747 full of consultants to configure monitoring solutions.

A large, distributed organization needs to use a monitoring tool that gives administrators an end-

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

to-end universal view while giving independent teams the ability to connect their systems to a central monitoring solution. Your monitoring tool should provide universal access to data, but it should also allow teams to manage the way their applications connect to it independently.

Intuitive, Easy-to-use Interface

Monitoring tools can't hide behind text-based interfaces that fail to understand the importance of modern UX design. A clunky interface from 1998 wasn't designed to track systems of today's complexity, and you'll want to find a tool designed by someone who understands the principles behind UX. This means that your interface can't just show you an impersonal matrix of events and call it a day.

Find monitoring tools that allow you to create experiences custom-tailored to individual roles. A high-level administrator of the entire system may need to see every metric combined into one dashboard while a DBA focused on a single database might only be interested in seeing data on a single system. An inadequate UX experience in monitoring is one that overwhelms and distracts. A good UX experience is one that minimizes distracting inputs and that supports the goals of the end-user. Find a monitoring tool created by someone who understands that less is more.

A Rich Community of Active Users

Maybe one of the most important aspects of choosing a monitoring tool: find a tool that has a large, vibrant community behind it. When you use a tool backed by a healthy community you have a more direct relationship with other users who have similar requirements. A community can more quickly identify and adapt to emerging trends and technologies, and you'll be able to share experiences and best practices across industries.

KEY TAKEAWAYS

When you are selecting your next monitoring tool, look for systems that understand the second-order benefits of monitoring. It isn't just about identifying a metric that is out of range on a database, it is about understanding the relationships among components in increasingly complex and interdependent architectures. It's about using product and UX design that create systems that allow fewer people to manage more resources efficiently.

Optimize the "Why" of Monitoring

At this stage in the evolution of monitoring no one has nothing; no one is running a complex system without a monitoring system, but the ways in which traditional monitoring systems are configured leave many systems exposed to downtime and alert fatigue. But while everyone has a monitoring system, Top Performing organizations remain 1.7 times more likely to have a consolidated end-to-end monitoring systems that can preserve uptime and help improve systems performance.

These Top Performers have recognized the value of monitoring as an essential tool to support ongoing development and service management. In optimizing the "why" of monitoring they've created systems that generate ROI in the form of stability in the face of increasing complexity. Monitoring for these organizations isn't just a boring afterthought or a tool to be used to find out why a system failed. Monitoring is the way an intelligent enterprise approaches distributed service management at a time when operational responsibilities are increasingly distributed.

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

Efficient Monitoring Is IT Management Scalability

The challenge facing IT departments isn't shrinking over time. The responsibility for maintaining uptime of systems both internal and external is becoming the primary focus for organizations serving global customers on a 24/7 schedule. More systems and more complexity mean that monitoring solutions have to be able to grow on two axes—complexity and scale.

As IT departments are being asked to support large portfolios and as systems continue to evolve to include public clouds, third-party services, and an always shifting cast of technologies, system administrators can either choose to stand up a consolidated view of infrastructure or surrender to a disconnected collection of monitoring tools.

Monitoring is service management, and without it your internal users and external customers start to lose faith in your ability to keep systems up and running. In the face of increasing complexity, an efficient monitoring tool is essential to preserving system availability and maintaining customer confidence.

Minimize Alert Fatigue with Monitoring Intelligence

The current state of most monitoring tools feels very much like the first generation of this technology. People interacting with monitoring tools are getting overwhelmed by alerts from all sides, and monitoring systems more often than not are ignored as soon as they are configured because it's just too difficult to keep up with the amount of manual effort required to tune thresholds into meaningful alerts.

What use is an alert that fires every day at the same time during anticipated outages? What's the point of configuring thresholds if those thresholds are still going to yield false positives? Most importantly, how many false positives are acceptable? When selecting a monitoring tool, end-users are advised to look for the tool that understands that false positives are often unavoidable but easily addressed via intelligent systems that can adapt to alert storms.

Without Intelligence Most Rely on Manual Intervention

When it comes to alerting and monitoring, the real logic for most organizations is a group of humans watching graphs and continuously adjusting thresholds. These groups are full of people familiar with the expected errors, and people who are being constantly told what failures to ignore.

This is the most common pattern in large enterprises, and it absolutely does not scale to meet the needs of an increasingly complex monitoring landscape. Something has to give in the enterprise monitoring space or we will continue to see organizations crippled by dramatic system outages that have just started to make headlines across multiple industries. The right monitoring tool helps ensure your employer isn't attached to those headlines in the future.

Entire content © 2015 Gleanster, LLC. All rights reserved. Unauthorized use or reproduction is prohibited.

About Gleanster Research

Gleanster Research helps business leaders uncover best practices in technology adoption by benchmarking the successes and failures of Top Performing firms. We publish the results online so you can learn from the most successful CXOs on the planet. We do this through a proprietary benchmark research methodology that informs a library of best practice market research and a comprehensive directory of vendor showcases (complete with vendor rankings based on reviews from users).

CONTACT GLEANSTER:

4695 Chabot Dr
Pleasanton, CA, 94588

www.gleanster.com

research@gleanster.com

Gleanster™