whitepaper

Selecting the Right Server & Application Monitoring Tool

Ethan Banks
Packet Pushers host and guest author

solarwinds

Selecting the Right Server & Application Monitoring Tool

whitepaper

Troubleshooting Tough IT Problems

One of the most difficult challenges an IT infrastructure team faces is that of pinning down what exactly is broken when the technology a business depends on starts to go awry. A day that begins smoothly can descend quickly into chaos as reports from various departments and sundry locations tell a tale of an application that isn't behaving well.

Underperforming applications are a classic complaint. Email is slow. A report took too long to generate. CRM page loads are taking too long. The intranet site is throwing sporadic errors.

The help desk team escalates to the sysadmin team. The sysadmin team escalates to the application engineering team. The engineering team points fingers at each other. Management hauls everyone into a conference room to facilitate resolution. All the while, the badly behaving application continues its miscreant behavior, angering and alienating the customer base while the IT engineering team sits around the conference room table, poking furiously at laptops in search of a cause.

There's Got to be a Better Way

Of course, there is a better way. SolarWinds Server & Application Monitor (SAM) is a readily available toolset that IT shops leverage to become proactive instead reactive. We're not just talking about red/green or up/down status messages, either. There are a number of sophisticated ways to monitor within the Application Performance Monitoring (APM) realm, falling roughly into one of the following categories:

- Measuring what a user experiences when they interface with an application at each step of the transaction process to more accurately pinpoint where bottlenecks are occurring – before your end-users notice
- SQL query response time monitoring
- Modeling an application so that each underlying component is tracked as a logical whole
- Applying intelligent performance, fault, and log alerts, based off of industry best practices
- Discovery of out-of-compliance systems and applying patches automatically
- Proactive problem analysis using historical performance and capacity data to predict performance issues proactively

Applications are usually not stand-alone. Instead, applications form an integrated, and often interdependent, system. Therefore, as implied by "system," we're talking about SolarWinds performing the job of monitoring each of the integrated components that make up an application. For instance, consider the key elements that make up a typical enterprise email system:

- At least one Internet-facing gateway
- The domain name system
- Anti-virus and malware deep message inspection
- Blacklists and whitelists
- Authentication mechanisms
- Inbound and outbound queues
- One or more data stores with the database and disk storage requirements this implies
- Backup and restore capability
- Support for a variety of mail clients leveraging both proprietary and standard protocols
- Calendaring integration
- Free/busy publication

And let's not forget the additional layers of complexity introduced when the messaging system uses clustering and other high-availability technologies!

Let's take another example. If you are an e-tailer, your payment gateway is your lifeblood. People purchasing from your online storefront expect to be able to browse your catalog, select products, review their cart, and complete their order. In this world of instant gratification, waiting is not an option. When someone clicks the magic button to submit their order, the last thing you want is an abandoned cart because your payment gateway timed out while authorizing the purchase. When your payment gateway is underperforming, how do you know where the problem is? Was the transaction post to the order database too slow? Did name resolution fail? Did the approve/decline request to your payment processor time out? Is a load balancer acting strangely? Is a firewall or intrusion prevention system introducing unusual latency? How do you find out, shy of getting all your IT heavy hitters together to troubleshoot?

A properly installed system and application monitoring solution is an excellent answer. Practically speaking, only a very small number of people in your IT organization will know enough about every piece of critical application infrastructure such that they can diagnose problems on the fly. Beyond that, people come and go. Even the most loyal of employees can be drawn away from your organization, taking their knowledgebase with them. And even if those loyal, knowledgeable people are around, they can still only be reactive and not proactive. To be proactive, an IT infrastructure needs an active, constant, and automated monitoring tool that raises problem alerts as soon as an issue is detected, minimizing or eliminating the amount of the time that the user community is impacted.

Understanding Application Performance Monitoring Solutions

Proactively monitoring a complex application is a tall order, and over the years many tools have set out to do the job. Nearly all seasoned IT professionals cringe at the memory of a beastly framework APM system foisted upon them sometime in their pasts. The promise of a framework-monitoring system as described during the sales process fills one with a starry-eyed eagerness. One place to go for alerts. One system to generate reports. One entity that sees all, knows all, and correlates all the data into a unified presentation of the IT infrastructure. A single pane of glass. One monitoring tool to rule them all. As if. The sad reality of these huge APM frameworks is that they are too much of an animal to manage in their own right. Managing a framework APM becomes a full-time job for at least one, if not more than one IT team member, and organizations are usually loathe to commit a resource like this. This style of APM requires multiple servers to run, a variety of modules and plug-ins to accomplish what you need, and usually come bundled with professional services to get up and running as well as an inside sales rep to help you interpret the licensing scheme. There is no question that you're going to need help, and lots of it. And let's not forget the cost. You have a budget line item for wheelbarrows, right? Because that's what you'll need to carry the required cash over to the framework APM vendor's bank.

Of course, the open-source community has its share of offerings in the APM space. Nagios leaps to mind for many, and a quick Google search turns up several other projects. I've never been entirely thrilled with open-source offerings for production environments. Open-source has a forgiving licensing structure that means your company typically pays little to nothing for use of the software, but there are always rough edges to contend with. Often, open-source user interfaces are clunky. Or the configuration process is a bear. Or they are fragile due to library dependencies. Open-source customizability (since the code is accessible) is both a blessing and a curse. It's true that you can customize the application to do whatever you like if you're a code monkey, but that also means you're stuck maintaining your own customizations with each upgrade of the open-source package. That implies that upgrades are available as often they are not. Open-source projects are notoriously abandoned, as the primary coders are pressed into duty doing work that actually pays some bills, moving former pet projects to the backburner. Even if you do find a project with longevity that appears to be committed to by the open-source community, there are still challenges of support. Who do you call when your open-source APM tool isn't doing what you need for it to do? Or when it won't stay up and running, providing only an inscrutable error message by way of explanation? Or in a worst case scenario, what legal recourse do you have if you discover the open-source package you downloaded came bundled with malware? This isn't to say that all open-source projects are bad, but my take is that a business that cares about its customers deserves a commercial tool to keep tabs on its IT infrastructure.

The sweet spot between framework and open-source APM tools is what I term the "value APM tools," such as SolarWinds Server & Application Monitor (SAM). A valuable APM offering has the backing of a real company you can do business with, while not being so overly cumbersome or complex that you need a small army of consultants to get the platform going. For the majority of companies, the value APM tool is the "best fit" for their monitoring needs. Giving up only a bit of the capability that might be offered by the large framework APM tools, the value APM systems of today offer a large array of application layer monitoring capabilities. They also run well on a single server out of the box, while still allowing a company to scale them up as their monitoring needs increase. Therefore, a company can invest in a value APM platform today, deploy it quickly with existing IT staff, and then grow it over time as needed.

	SolarWinds	Framework	Opensource
Cost	• Budget-friendly	• Expensive software cost + professional services for deployment & customization	• Free, but costly to customize
Installation/config time	• DIY tool, designed to be easy to deploy without outside assistance • Data is presented immediately without costly customization	• 1/2 day to days or weeks	• Hours
Logical, usable, customizable, intuitive, drill-down UI	• Integrated UI across SAM, virutalization, network and storage products without the costly footprint	• Integrated portfolios with a heavy footprint	• Poor UI
Support	24/7	24/7	
Cost to maintain	• 10% of one person's time	• At least one full time person	• Upgrades negate customization
OOTB features & platform coverage	✓	✓	
Community content	• SolarWinds has hundreds of community contributed application templates, scripts, and reports • Community integration is available directly from within the product	• Very limited	• Built on community
Scale	✓	✓	

Selecting the Right APM Tool

What features should you look for when shopping for an APM? In part, this depends on what sort of applications you're trying to monitor. Generally speaking, you're looking for a mix of generic monitors and specific monitors. If you consider that most vendors expose interesting statistics about their hardware or software via Simple Network Management Protocol (SNMP), then clearly your APM system should be able to poll any sort of SNMP object identifier that a vendor might have provided. In this way, you can monitor elements of your application that you find compelling. For instance, you might want the APM system to monitor the incoming queue depth of your Internet mail gateway. If the queue is beyond acceptable thresholds, various alerts could be triggered, allowing IT staff to investigate and resolve before the queue gets too deep. The same line of thinking goes for Windows Management Instrumentation (WMI) if you're a Windows shop and Java Management eXtensions (JMX) if you run Java applications.

There's more to choosing an APM system than just the generic monitors, though. You should also consider the specific applications your business runs where it might be nice to have pre-built monitors included for you. For instance, many IT environments use Microsoft Active Directory, Exchange, and SQL. Do you really want to build a complex application-monitoring infrastructure by hand to keep tabs on Active Directory? I don't. I'd much rather point my shiny new APM tool at a subnet or three, have it auto-discover my AD domain controllers for me, and then offer me a list of application monitors that I can enable via a template. That auto-discovery and auto-monitoring process will reduce the chance that I'll forget something, and it guarantees that all the critical services that every AD domain controller runs are indeed being monitored.

That said, it's not as if Windows is the only common platform to be found. While you do want an APM tool that copes extremely well with Windows and Microsoft back office applications, you don't want a solution that's Windows-centric. If your infrastructure is like most I've seen recently, the data center is a mix of bare-metal Windows and Linux as well as virtual machines sitting on top of a hypervisor. Therefore, the APM system should be able to monitor intelligently UNIX systems of various stripes, as well as interface meaningfully virtual infrastructures like VMware. Monitoring virtualized environments is tricky. To get down to what a performance problem really is can be convoluted in that the statistics about memory and CPU utilization reported by a virtual machine don't necessarily reflect the bare-metal reality abstracted by the hypervisor. To really know the score, the APM system needs to be able to talk to the hypervisor directly.

Another key element of an APM system is that of the interface. I've used some terrible interfaces, and I'll wager that you have, too. A bad user interface makes for an undesirable experience. In an APM system, you should be able to group all the key elements of a particular application into a single, cohesive group, and then organize those groups into a hierarchy that reflects how your IT functions. I like to be able to drill down into a hierarchy to pinpoint an issue quickly. For example, if the email application has an alarm drill into that application container. If the Internet gateway component is red, drill down further. The transaction monitor indicates that the SMTP gateway is not accepting inbound messages, and now the root cause of the issue is known. A monitoring interface should be simple, intuitive, and obvious to help you navigate from a general issue to a root cause in a straightforward way. You don't want to manage the interface. You want the interface to help you manage your applications. Don't assume the interface will be wonderful, because many know from experience that a good user interface is far from the norm.

It's been said that the whole is greater than the sum of its parts, and sometimes, monitoring the individual elements of an application obscure how that application is experienced by the end-user. Therefore, another piece of the APM puzzle that should be sought is that of synthetic transaction monitoring. In other words, you want your APM tool to pretend that it's a user and periodically push data through an application. When done properly, this will validate key application functions such as name resolution, authentication, payload processing, and overall responsiveness. Transaction monitoring can reveal an overall application issue when individual elements might be testing within parameters. For example, the disk volume an SQL database lives on might have lots of available space, but if the table privileges have changed so that users can no longer commit a transaction to the database, the database might as well be living on a storage volume that's full. An APM transaction level test should reveal this kind of an issue, while merely monitoring drive space would not.

When an APM system does its job well, the natural progression is for the system to be called upon to monitor more and more elements. Therefore, the APM tool must be able to scale. I have been in the situation where an application as originally perceived became hopelessly outmatched by the demands placed upon it, but there was nowhere to go but a forklift upgrade. While that's not always a huge concern depending on the application, with an APM platform, there's a good bit of time that's been involved in tailoring it to your environment, as well as a reliance placed upon it by the IT teams. Therefore, you want an APM solution that scales up as demands on it grow without a service interruption or loss of time investment. That scalability should be in two forms. One is in licensing. While this point seems obvious, it's an easily overlooked detail. I've seen a number of applications where a special entry-level version is sold at a discounted price, but is only suitable for the smallest of installations and cannot be upgraded. For cost reasons, this version is chosen instead of the enterprise version that will be able to grow. It happens. The other scalability concern is that of modularity. For instance, can you split up some or all of the polling, reporting, interface, and database engines? If you can, you'll be able to scale the application monitoring systems as large as you're likely to need them. If not, you're at risk of outgrowing the APM tool- and faster than you might think.

One APM tool that matches up extraordinarily well with these requirements is the SolarWinds Server & Application Monitor (SAM). One of the things I like about SolarWinds products is that they are created by people who "get it." While so many of the GUIs I've used seem to be written by someone whose sole purpose in life is to make important information difficult to find, SolarWinds understands the perspective of an infrastructure engineer in the trenches who need to keep a data center running. Therefore, SolarWinds SAM does as much of the work for you as possible, without limiting your ability to customize and scale the applications you wish to monitor. The price is comfortable enough so that you don't have to break the budget to get started with it, plus you can show everyone on the team how to get information out of it without sending them to a week of training. SolarWinds Server & Application Monitor just works; it's an obvious fit for most of the environments I've worked with during the course of my career.

About the Author

Ethan Banks is CCIE #20655 and a 16+ year IT veteran. He has designed, implemented, and supported networks for government, banking institutions, higher education, and various corporations. He is a host of the Packet Pushers podcast and an independent blogger covering the data networking industry.

About SolarWinds for Sysadmins

SolarWinds, a leading provider of IT management software to more than 100,000 customers worldwide, now offers a more comprehensive suite of highly regarded, highly effective products for sysadmins in organizations of all sizes.

SolarWinds systems management portfolio includes simple and affordable solutions for sysadmins, including:

- Patch Manager automates patching applications across tens of thousands of servers and workstations; notifies sysadmins of Microsoft Windows and third-party patches from Adobe, Apple, and Google.
- DameWare NT Utilities (DNTU) provides an integrated collection of Microsoft Windows administration utilities within a centralized interface for remote management of Windows servers, workstations, desktops, and laptops.
- DameWare Mini Remote Control (MRC) delivers one of the most comprehensive feature sets for Windows remote management in the industry with a price point at $99, that makes it one of the most affordable as well.
- Server & Application Monitor (SAM), a comprehensive server and application management product that enables sysadmins to monitor Windows, Unix, and Linux servers with visibility into the performance of critical IT services, underlying application components, and operating system and server resources on which they run.
- Synthetic End User Monitor (SeUM) delivers affordable, cloud-based, and internal web application monitoring, as well as external website monitoring, allowing users to respond to problems proactively.

SolarWinds strives to provide sysadmins with the tools they need to get their jobs done faster and easier, at a price they can afford.

For additional information, please contact SolarWinds at 866.530.8100 or email sales@solarwinds.com.
To locate an international reseller near you, visit http://www.solarwinds.com/partners/reseller_locator.aspx

Did you like this white paper? Tweet about it. http://www.twitter.com

solarwinds