

Mastering application performance monitoring: A comprehensive guide to APM best practices and strategies



What are we going to cover?

Introduction	3
What is APM?	4
Why do organizations need APM?	10
Challenges in modern applications	11
APM best practices	18
Which metrics should you capture?	20
Why Site24x7?	22
Conclusion	23

Introduction

In the days before digital transformation swept through companies and software was relatively straightforward, IT admins employed basic monitoring tools to oversee server and infrastructure performance. The only way to troubleshoot or validate issues and fixes was through logging, debugging, and quality assurance testing.

However, with the advent of microservices and cloud-native technologies, the scope of monitoring expanded beyond the server and infrastructure layer. This development brought with it the ability to trace the execution of requests without having to write more code. The spotlight shifted from fixing an application's developmental issues to fixing operational issues, with an emphasis on minimal disruption to the end-user experience. In addition to monitoring the traditional metrics like server response times and network latency, a consensus emerged on how to measure factors such as user satisfaction scores, load times, and transaction monitoring.

It was during this transformative period that the concept of application performance monitoring (APM) was born. APM addresses the evolving need for a more comprehensive and modern approach to monitoring.

As technology continues to advance and applications become more complex, APM is instrumental in providing organizations with the tools and insights necessary to optimize performance, troubleshoot issues, and ensure a seamless UX.

However, not all APM solutions are built the same and vary in quality, with some failing to provide a holistic view of all layers of the application stack, making observability an impossibility. In this guide, we'll define APM and discuss its fundamentals: why organizations need it, how it helps solves various hurdles an organization may face, and what metrics to capture to make observability a reality.

What is APM?

APM is the process of monitoring and managing the performance and availability of software applications to ensure they consistently operate at their highest potential. This goes beyond merely tracking response times and server metrics; APM offers crucial visibility into complex systems without disrupting availability and enables real-time monitoring, transaction tracing, code-level insights, alerting, and troubleshooting. APM empowers teams to swiftly identify and resolve issues, optimize performance, and ensure seamless scalability.

The fundamentals of an APM solution

Every APM solution contains a crucial set of fundamentals at its core to deliver effective monitoring. Delving into these critical aspects is essential for not only grasping the intricacies of APM but also for unlocking its full potential in optimizing an application's overall performance.

A proficient APM solution captures diverse aspects of an application to deliver thorough analysis, including:

- ✓ Key performance indicators (KPIs) such as Apdex score, throughput, and response time.
- ✓ User experience (UX), spanning from initial interaction to task completion.
- ✓ The interconnectedness and communication between different application components.
- ✓ Potential performance bottlenecks and anomalies.

A holistic approach to monitoring

APM transcends mere code checks, casting a watchful eye on the core functionalities while delving deep into the infrastructure's intricate network, meticulously dissecting its dynamics, and vigilantly monitoring external dependencies. This panoramic view empowers the proactive identification and resolution of performance issues, ensuring every facet of the application functions at its peak. To accomplish holistic monitoring, an APM solution should monitor the entire application ecosystem. This begins with monitoring UX.

Monitoring the **front end** of your application shows how your end users experience its performance in real time. This is primarily accomplished by capturing metrics like page load times, user sessions, and the response time and throughput of individual web transactions. Real user monitoring, an integral component of APM, facilitates the collection of these metrics. To make the most of your front-end monitoring efforts, it's important to start looking for potential issues in the test environment. This is facilitated through synthetic monitoring, a process that helps simulate potential problems within your application.

At the core of APM lies the crucial task of meticulously and comprehensively monitoring various **backend components** of your application in real time. The essence of a robust APM solution lies in its ability to offer code-level monitoring, allowing organizations to gain granular insights into the performance of applications constructed with diverse programming languages. These languages may include Java, .NET, Python, Node.js, PHP, and Ruby. By delving into the intricate layers of code execution across a spectrum of languages, APM empowers you to proactively identify bottlenecks, optimize performance, and ensure a seamless and responsive experience for users interacting with your applications.

Integrating your APM solution with **infrastructure components** also helps you monitor critical server metrics such as memory usage, CPU usage, server response times, network latency, database capacity, and resource utilization in containerized cloud environments. This scale of monitoring provides early warning signs of an unhealthy application by unmasking the root cause of issues and helps it to utilize resources appropriately, reducing the risk of downtime.

Actionable insights

Comprehensive monitoring helps with capturing metrics from the entire application ecosystem. The next natural step after collecting metrics is to analyze them, providing actionable insights that enable informed decision-making.

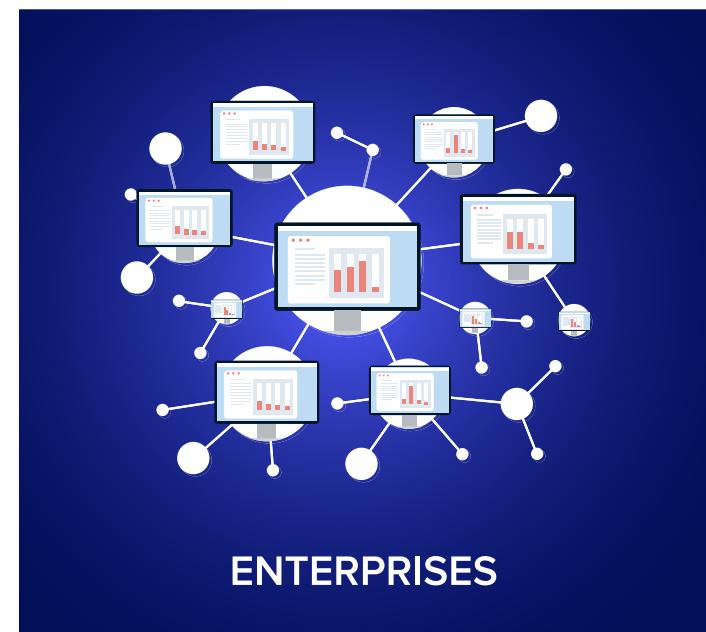
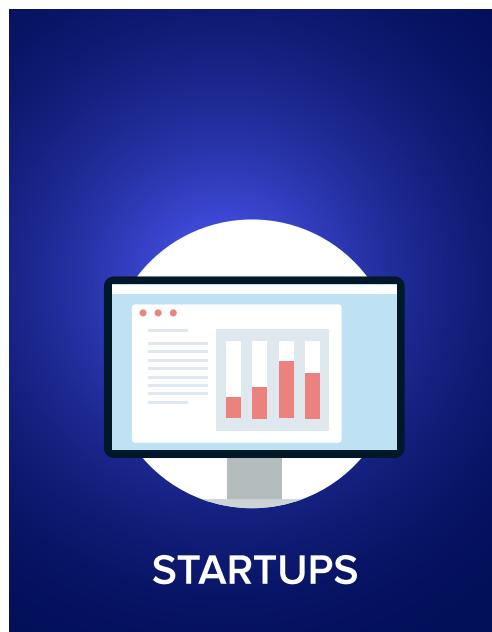
By distilling huge volumes of complex performance data into actionable recommendations, APM empowers teams to swiftly address performance issues, optimize resource utilization, and enhance overall user satisfaction. Choosing an APM solution with in-built AI and machine learning capabilities helps you stay ahead of the curve.

An APM tool that provides deep analysis of your application ecosystem gives you:

- ➊ **AI-powered anomaly detection** that establishes connections between diverse data points, discerns patterns and trends, and swiftly detects issues as they occur in real time.
- ➋ **Personalized recommendations** and predictive insights to proactively manage downtime and achieve your desired SLAs.
- ➌ **Automated remediation** that helps make your application more available and efficient. APM does this by using a more proactive approach to monitoring and setting up automation profiles that act on known responses triggered in a timely fashion from an application's infrastructure whenever a known pattern of failure is detected.

Scalability and adaptability

Scalability and adaptability is crucial for accommodating the dynamic and ever-changing nature of modern applications. Applications have become complex and now exist in different states, from the traditional monolith to more contemporary microservices-based or cloud-native formats. An effective APM tool should possess the capability to accommodate the growing workloads and infrastructures of all these applications, scaling up or down to ensure uninterrupted monitoring and performance optimization, and seamlessly integrate with various development and operations platforms to help streamline the monitoring process. By prioritizing scalability and adaptability, organizations can effectively future-proof their APM strategies, ensuring they remain agile and responsive in the ever-changing digital landscape.



Why do organizations need APM?

In today's digital landscape, ensuring top-notch digital experiences for customers is make or break for a business' success. Yet, modern applications, with their intricate complexities, are prone to failure in nuanced ways, highlighting the importance of swift issue detection and resolution.

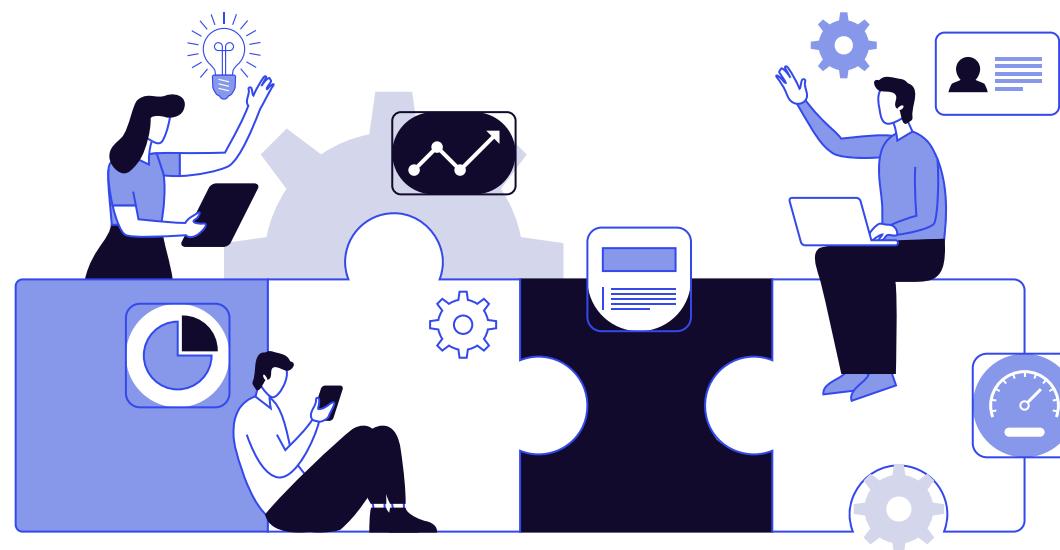
By harnessing the power of an APM tool, businesses can gain swift access to the underlying causes of issues within their applications by pinpointing the exact line of code or component causing the issue, effectively minimizing downtime and bolstering application performance. This means organizations can pinpoint crucial areas for enhancement and tailor their marketing strategies more precisely, enhancing customer engagement and loyalty.

Additionally, APM tools serve as cost-saving mechanisms by facilitating the rapid identification and resolution of issues, thereby mitigating the necessity for additional resources. By leveraging APM, businesses can position themselves ahead of the competition by delivering seamless digital experiences, ultimately fostering higher levels of satisfaction and bolstering customer loyalty.

It's common for companies to question whether APM is necessary for ensuring glitch-free applications and delivering top-tier digital services to their customers. The unequivocal response? Yes—continuous application monitoring is essential to optimize performance and adeptly tackle any challenges that will arise. In the following section, we'll delve into potential challenges your application might encounter and explore how APM offers solutions to overcome them.

Challenges in modern applications

Navigating the intricacies of modern applications presents a myriad of challenges, from scalability and reliability to performance optimization and security. Let's delve into the multifaceted landscape of modern application challenges and strategies for overcoming them.



Increased application complexity

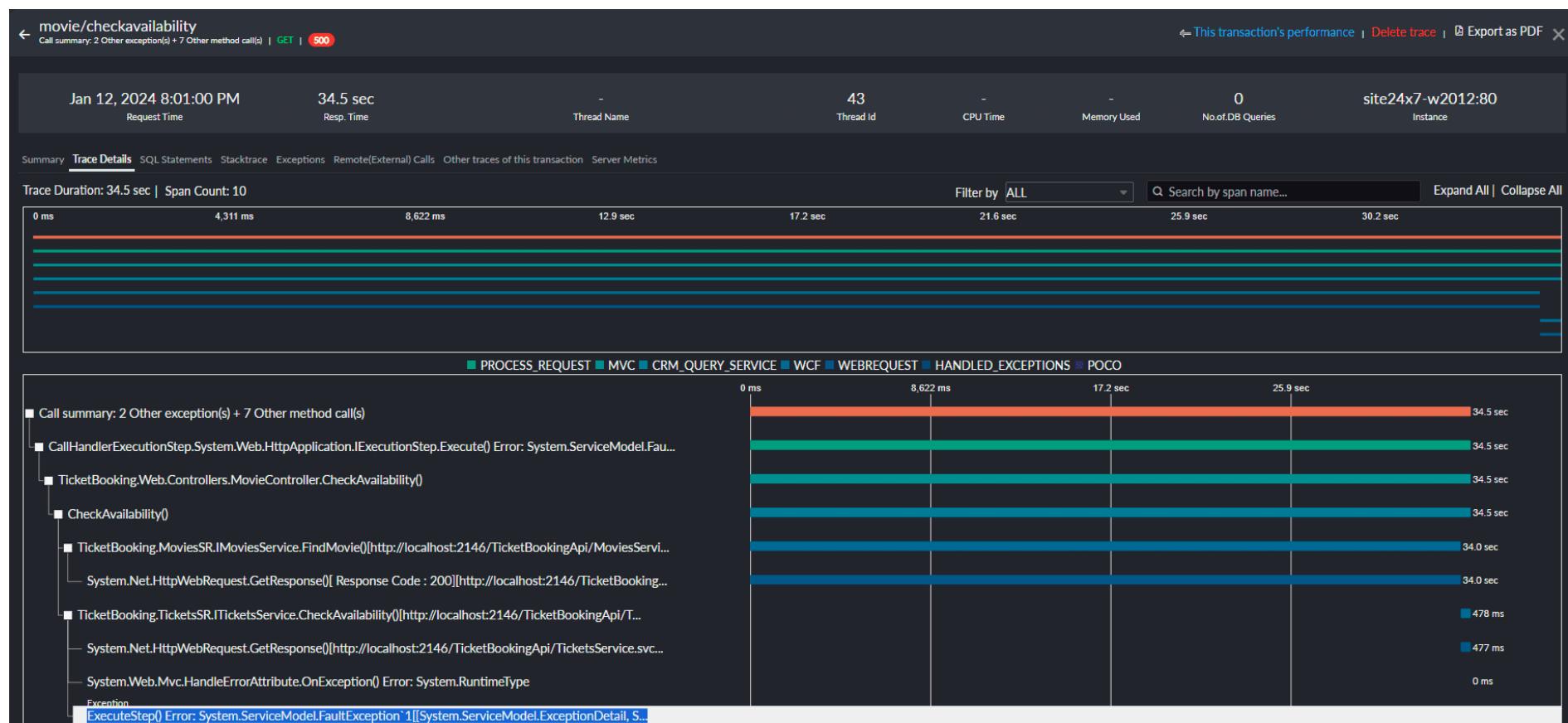
In the increasingly complex environments of modern applications, identifying the specific component responsible for a failure or issue is close to impossible. Without this crucial insight, mitigating the problem becomes a daunting task, often resulting in prolonged downtime, frustrated users, and potential revenue loss.

Modern applications are distributed in nature and have multiple interconnected components that interact with each other across diverse environments. This complexity makes it challenging to trace the root cause of an issue when it occurs, in turn delaying the troubleshooting process. In microservices architectures, latencies can further exacerbate the problem, as a delay in one service can cascade across the entire system, impacting overall performance and UX.

Distributed tracing to the rescue

To analyze issues in highly complex environments, you must correlate data from multiple services spread across the system—a painstaking process. Distributed tracing is a sophisticated tracing mechanism that helps to diagnose and troubleshoot issues within complex application environments. With distributed tracing, you can track the entire path of a web transaction across applications or service boundaries and drill down to the exact line of code or service responsible for the problem.

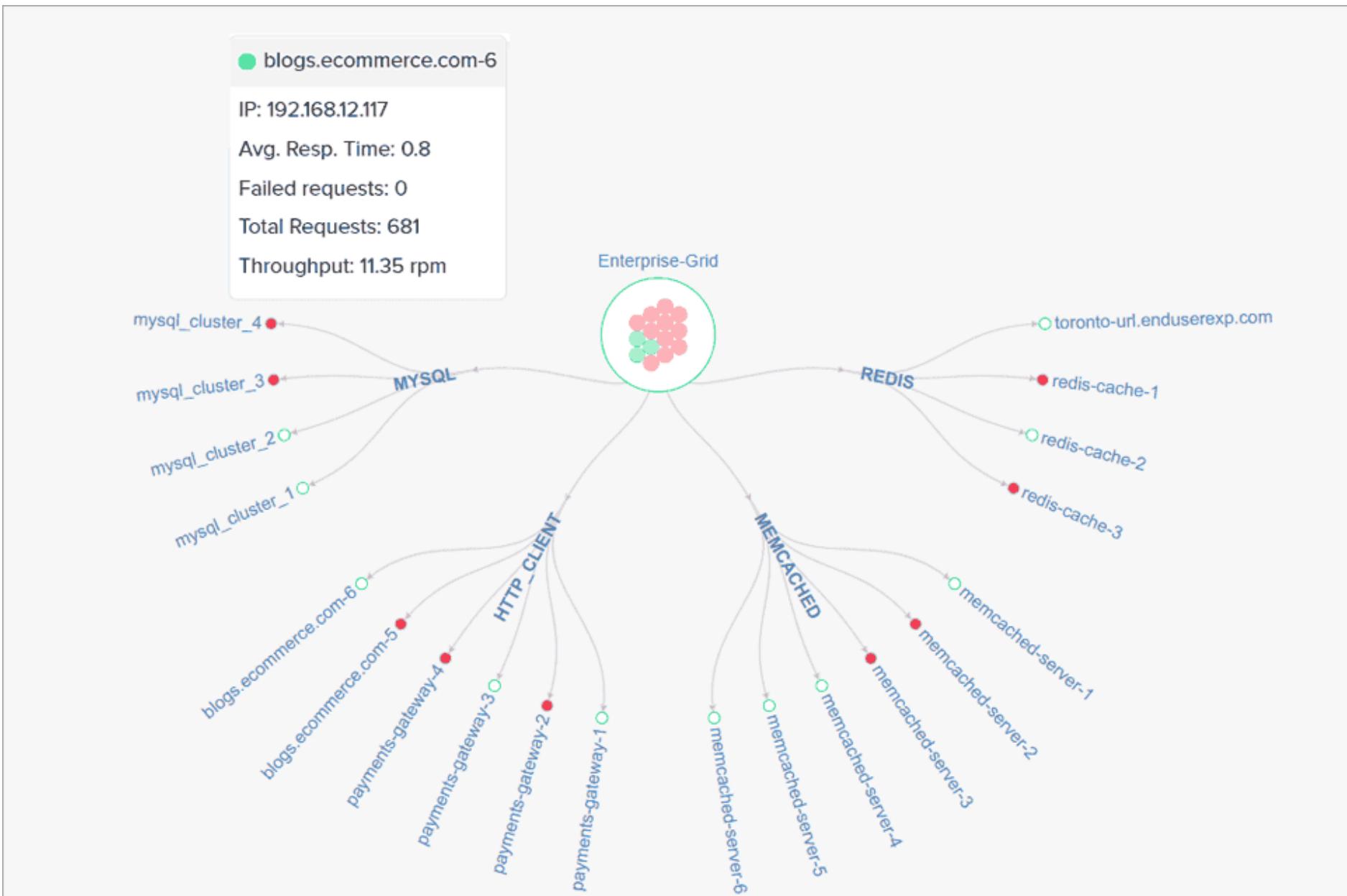
The Site24x7 APM tool correlates data from distributed traces with other monitoring metrics, logs, and events, providing holistic insights into application performance and behavior. Using this actionable intelligence, you can identify patterns, trends, and anomalies and address critical issues proactively, minimizing downtime and ensuring optimal application performance.



Performance degradation caused by external components

Applications communicate with external components through various protocols and mechanisms to complete actions such as communicating with external APIs or web services using HTTP or HTTPS protocols to fetch customer or product data, sending or retrieving data using RESTful APIs, or storing data in a cache framework like Redis or Memcached to optimize performance.

All these processes have some external dependencies that may cause performance issues in your application from time to time. Without an APM solution, it's virtually impossible to catch any anomalies in these workflows. Application dependency mapping and service maps in APM solutions help in spotting any bottlenecks caused by external components with ease. With these service maps, you can easily drill down to the root cause of an issue and source it back to the exact point or component that is causing a transaction failure in your system. With options to view these dependencies in a map view, graph view, or tabular view, it can't get any easier to spot and eliminate issues in your application ecosystem before your customers even catch a whiff of it.

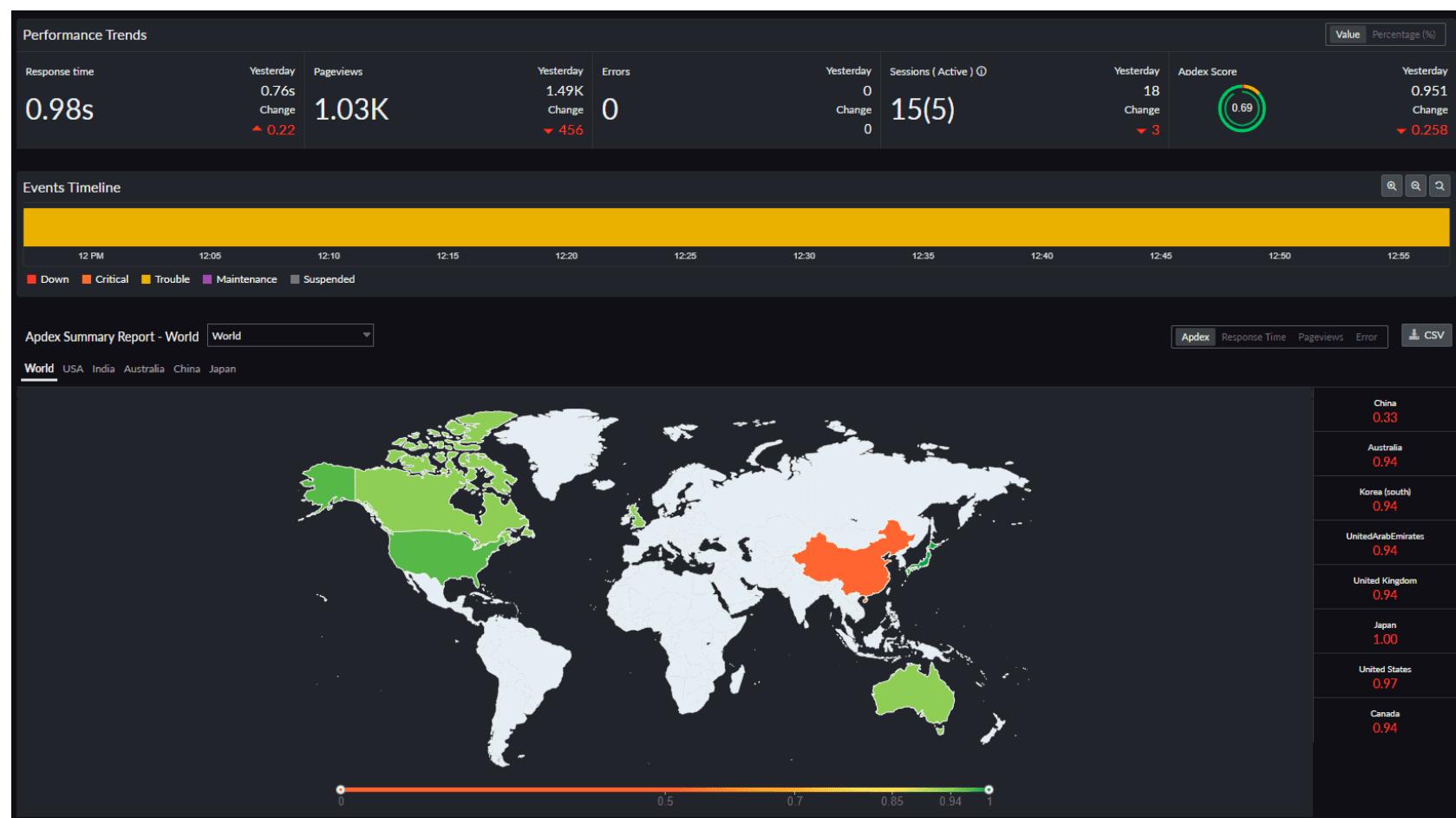


Your application ecosystem is not observable

Observability has become a widely embraced concept in recent years, emphasizing the need for holistic monitoring solutions. While deploying an APM tool to monitor backend operations is a crucial step towards achieving observability, true comprehensive observability extends beyond backend monitoring. It's equally imperative to monitor frontend applications and infrastructure components to attain a holistic view of system performance. This entails tracking work metrics, tracing transactions across services, and analyzing logs to gain deep insights into system behavior, performance bottlenecks, and user experiences. By integrating frontend, backend, and infrastructure monitoring, organizations can enhance their ability to detect, diagnose, and resolve issues proactively, thus achieving true observability across their entire application stack.

Similarly, an APM tool should offer comprehensive insights into the performance of applications deployed across diverse cloud environments like AWS, Azure, GCP, and Kubernetes. It should provide tailored insights for specific requirements, such as offering deep visibility into containerized applications, monitoring resource utilization, and ensuring the health of Kubernetes clusters. In AWS, it should be capable of monitoring EC2 instances and Lambda functions, while in Azure and GCP, it should extend its monitoring capabilities to Azure Virtual Machines, Google Compute Engine instances, and other relevant services. Only by capturing all this data in real time can your organization truly achieve comprehensive observability across your entire application ecosystem.

Issues can stem from various points within an application's system, potentially causing downtime. In such scenarios, an APM solution that seamlessly integrates with real user monitoring (RUM) and infrastructure monitoring tools becomes indispensable. By capturing metrics such as response times and CPU usage from both frontend user interactions and backend infrastructure, you can achieve a comprehensive understanding of your application's performance and ensure optimal user experiences.



APM best practices

Monitoring your application is necessary to maintain its overall health and continuously enhance its performance. Although this may seem like a daunting task, it is actually easy if a few best practices are followed.

- ➊ **Define clear objectives:** Clearly define the goals and objectives of your APM strategy, aligning them with the overall business objectives and user expectations.
- ➋ **Monitor key metrics:** Identify and monitor KPIs relevant to your application, such as response time, throughput, error rates, and user satisfaction metrics like Apdex score.
- ➌ **Practice comprehensive monitoring:** Implement a holistic approach to monitoring that covers all components of your application stack, including infrastructure, network, databases, and third-party services.
- ➍ **Track baseline performance:** Establish baseline performance metrics to understand normal behavior and deviations, enabling proactive identification of performance anomalies.
- ➎ **Set up alerting to get notified instantly:** Configure alerting mechanisms to notify stakeholders of performance issues or deviations from predefined thresholds, ensuring timely intervention and resolution.

- 🕒 **Use correlation to identify root causes of issues:** Utilize correlation techniques and root cause analysis to identify the underlying causes of performance issues, enabling targeted remediation efforts.
- 🕒 **Always prepare to scale up or down:** Plan for scalability by monitoring resource utilization and performance under varying load conditions, ensuring the application can handle increased traffic or workload.
- 🕒 **Implement continuous testing and continuous improvement:** Employ continuous testing and continuous improvement practices to validate performance changes and new features before deployment, ensuring that performance improvements are achieved without introducing regressions.
- 🕒 **Establish a feedback loop:** Set up a feedback loop to gather insights from monitoring data and user feedback.
- 🕒 **Foster collaboration:** Create a collaborative synergy between development, operations, and business teams to ensure that APM initiatives are aligned with business priorities and requirements.

Which metrics should you capture?

It is important to measure what matters to your application. There is no one-size-fits-all solution when it comes to monitoring performance. However, here are some of the crucial metrics for APM that provide insights into the health, performance, and UX of an application:

- ➊ **Response time:** The time taken for the application to respond to user requests.
- ➋ **Throughput:** The rate at which the application processes transactions or requests. A higher throughput indicates greater efficiency in handling user traffic.
- ➌ **Error rate:** The frequency of errors or failures occurring within the application. High error rates can indicate potential issues impacting UX and application functionality.
- ➍ **Apdex score:** A user satisfaction score based on response time thresholds, Apdex scores provide a standardized measure of user satisfaction.
- ➎ **Availability:** The uptime and downtime of an application that ensures reliable access for users. High availability is critical for maintaining user trust and preventing revenue loss due to outages.

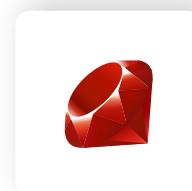
- 🕒 **Resource utilization:** The utilization of resources such as CPU, memory, disk I/O, and network bandwidth. Optimizing resource utilization helps ensure efficient use of infrastructure resources and prevents performance bottlenecks.
- 🕒 **Transaction performance:** The performance of critical business transactions within the application, such as checkout processes or database queries. Monitoring transaction performance helps ensure the efficient operation of key workflows.
- 🕒 **User session metrics:** User engagement metrics such as session duration, bounce rate, and active users. Understanding user behavior and engagement patterns helps optimize UX and identify areas for improvement.
- 🕒 **Dependency health:** The health and performance of dependencies such as third-party APIs, data bases, and external services. Identifying issues with dependencies helps prevent cascading failures and ensures the reliability of the application ecosystem.

Why Site24x7?

The Site24x7 APM tool plays a crucial role in optimizing the performance of applications in real time. With Site24x7 APM Insight, you can continuously monitor application performance and promptly detect and address any real-time issues that may occur. The tool delivers essential metrics like response time, server health, and UX and aids in the rapid identification and resolution of performance bottlenecks. By combining our APM tool with our log management tool, you get access to data that is essential for troubleshooting, performance monitoring, anomaly detection, root cause analysis, and compliance. Site24x7 APM Insight harnesses log data effectively and empowers your organization to optimize application performance, enhance UX, and ensure the reliability and security of its applications.

Equipped with rich features like customizable dashboards, application dependency mapping, thread profiling, milestone markers, and comprehensive reports and analytics, the Site24x7 APM tool efficiently bridges the gap between end-user experience and application performance with Site24x7 APM Insight, resulting in a seamless digital experience for your consumers.

The Site24x7 APM Insight tool helps you monitor applications built using:



Conclusion

APM is a multifaceted process that involves measuring application performance, capturing performance metrics from various systems, and correlating them to provide a comprehensive view.

Leveraging machine learning, APM solutions automate anomaly detection and reduce mean time to resolution (MTTR), enabling instant root cause analysis and proactive performance optimization. Additionally, an ideal APM solution integrates with your AIOps ecosystem, offering a framework for seamless integration with complementary technologies. Given the potential impact of application performance issues on business outcomes, you should evaluate APM solutions based on these capabilities and select the most suitable option for your environment.

About ManageEngine Site24x7

ManageEngine Site24x7 is an AI-powered observability platform for DevOps and IT operations. The cloud-based platform's broad capabilities help predict, analyze, and troubleshoot problems with end-user experience, applications, microservices, servers, containers, multi-cloud, and network infrastructure, all from a single console. For more information about Site24x7, please visit www.site24x7.com.

[Get Quote](#)[Request Demo](#)