



**GIGAOM** RESEARCH

# Performance testing: balancing agility and quality

---

Ryan Shriver

October 17, 2014

*This report is underwritten by Apica Systems.*

## **TABLE OF CONTENTS**

---

Executive summary .....	3
Today’s challenges .....	4
What leaders are doing .....	6
New methods for success .....	8
New tools for success .....	11
Key takeaways .....	15
About Ryan Shriver .....	16
About Gigaom Research .....	16

# Executive summary

Today, application performance is more important than ever. Numerous studies have shown that even small degradations in application performance can have severe impacts on revenue and customer experience. As organizations continue adopting agile methods for delivery, verifying application performance must not get bypassed in the pursuit of faster time to market. Yet it must also not stand in the way of getting enhancements to customers quickly, before the competition does.

Simply stated: organizations must balance agility, quality, and performance. Leaders today are learning to be more efficient *and* effective at integrating performance testing and monitoring with rapid Agile delivery cycles. Those that get it right can consistently and quickly deliver new features to their customers, with higher quality and with confidence that their solution scales to meet performance expectations.

IT and business leaders who are responsible for business apps can learn from leaders who are balancing agility, quality, and performance. They should introduce new methods and tools while elevating the perception and authority of performance testing to ensure consistent quality and rapid delivery.

Key findings in this report include:

- App performance directly impacts customer experience, which correlates to customer satisfaction and business outcome.
- Ensuring consistent app performance for every release of every app requires an integrated approach to performance testing and monitoring.
- Leaders should use Agile and DevOps to manage app performance proactively across all the teams involved with planning, designing, developing, testing, and operating apps.
- By automating the gathering, analysis, and dissemination of key performance data to the right decision makers, organizations free their teams to spend more time on value-added practices related to proactive management.
- Next-generation app performance testing and monitoring tools form essential building blocks for high-quality rapid releases.

# Today's challenges

## Performance directly impacts customer experience and business results

In today's competitive markets, success and failure are directly influenced by how well organizations manage their customers' digital experiences. These experiences directly affect business performance and are greatly influenced by the performance of apps – both customer-facing and internal-facing. These customers are not just online at home or in the office, but are increasingly mobile and global. They expect app performance to be consistent across a range of devices from laptops to tablets and smartphones. They also expect app performance to be consistent across a range of networks, from broadband to wireless and cellular around the world.

Proactively managing user expectations is an enormous challenge for organizations of all sizes. But there are upsides for those organizations that get it right. Walmart has done significant research on the impact of performance on conversion rates and has found a sharp decline in conversion rates as site response time increased from 1 second to 4 seconds. For every one-second improvement, conversion increased by 2 percent. For every 100ms improvement, incremental revenue grew by up to **1 percent**. Amazon's results were nearly identical, as the company found every 100ms delay costs it **1 percent in sales**.

Organizations that don't focus on performance have downsides. In addition to poorer customer experience and less revenue, a company might see an impact on its brand. [SingleStone's Customer Research Lab](#) recently reported that 66 percent of customers say website performance influences their impressions of the company, while 35 percent of customers are less likely to purchase a product if the website performance is slow.

The picture is clear: in the age of the digital customer, organizations must take app performance seriously if they want to improve those customer experiences that lead directly to better business results. Ensuring apps are properly performance tested and monitored is key to achieving these desired results.

## Organizational structures are not aligned around performance

Traditionally, many organizations have separate functional departments for designing, developing, testing, and operating apps in production. When waterfall methods were prevalent for app development, the collaboration and communication between these departments was best described as “throw it over the wall.” As each group passed its work to the next department, it also passed hidden assumptions and risks – including technical debt and design flaws. Unfortunately, this often became apparent very late in performance testing or, even worse, in production when users are impacted.

Today, an Agile approach has replaced department silos with cross-functional teams working together, iteratively. Although this has dramatically improved collaboration and communication, most teams still don't performance test each iteration of an app as they deliver new features. More often, a centralized group tests performance toward the end of a development cycle. These shared resources are typically scheduled far in advance and can become a bottleneck for overall business agility, especially if performance problems are found late and can't be fixed quickly.

### Belief that performance testing is expensive and time consuming

Previously, performance testing was expensive, time consuming, and cumbersome. In fact, this remains true for many organizations. The list of expenses is long: a scaled-down version of the production environment, complex performance-testing tools, sufficient hardware and software to generate traffic, complex monitoring tools to measure the app under test, performance test scripts that simulate transactions, etc. On top of those problems, a team of very talented (and expensive) engineers and analysts is necessary for testing an app under real-world conditions – and additional engineers to diagnose and fix performance problems when they occur.

Given this reality and the rush to improve time-to-market for new features, organizations logically want to avoid the cost and time delays associated with performance testing if they can. As organizations transition to Agile and release new features more quickly, performance testing must be able to keep up. Given the current expense and complexity of performance testing in many organizations, it is typically only done once – right before the release – introducing huge risks to schedule and expectations should the app not perform well.

### Assumption that auto scaling ensures application performance

Auto scaling – a cloud platform's ability to automatically allocate more compute, storage, and network infrastructure to apps on-demand – is a key capability of all leading cloud platforms. Organizations migrating apps to the cloud can have a false sense of security that the cloud's elasticity with auto-scaling features will ensure app performance and scalability.

While auto scaling is a valuable feature to mitigate the performance impact of dynamic workloads, the app still must scale *well*. Auto scaling is only as good as the app's ability to take advantage of the additional resources. In other words, unless an app is architected to run in the cloud and take advantage of elastic infrastructure, throwing more resources at an app isn't a panacea for better performance.

# What leaders are doing

How are performance-sensitive industries such as gaming and financial services approaching performance testing and monitoring? Perhaps not surprisingly, they are integrating performance testing and monitoring into their development lifecycles from the start.

## Integrating performance testing and monitoring

Mitigating the challenges described in the previous section involves closer collaboration and communication among the people involved in app development, performance testing, and monitoring. Organizations doing this are seeing benefits including:

- Faster time to market for new features, especially when performance testing is automated and performed regularly as a part of a continuous deployment pipeline
- Reduced time and costs for performance testing apps when leveraging modern tools and using the cloud for performance test environments
- Tighter feedback loops that catch and fix performance problems earlier in the development lifecycle, mitigating schedule risks
- Cross-population of app performance knowledge across the teams responsible for designing, building, testing, and running apps in production

## Elevating the importance of performance

While app performance is important in every industry, it is especially important in the financial services industry. In 2010, one Fortune 500 financial services firm ranked near the bottom of the industry in application performance for both its public website and its secure customer portal. Realizing the impact to its brand and customer experience, the company set about dramatically improving its app performance as well as the culture of the teams that built the apps.

The company started by hiring a lead performance engineer and assembling a team. To elevate the importance of performance, large plasma TVs with real-time app performance dashboards were installed in the C-suites. Next, the new performance team went to work testing and monitoring the key customer-facing apps. The team worked directly with other teams who were building and running them to improve performance incrementally. Some improvements were done with the operations teams, such as adding a

CDN and SSL offloading. Others, like minification and sprites, were done with the development teams. Also during this time a cross-functional governance process was established that set standards for image sizes and the use of third-party analytics tools and other performance-related guidelines.

Within two years, the team had helped reduce app response time from 7 seconds to less than 1 second. The longest running customer transaction went from 16 to 3 seconds. While these results required many new methods and tools, it was leadership's commitment to elevating the importance of performance and a team focused on the problem that were the biggest contributors to success.

## Ensuring new product launches are a hit

Activision Publishing, Inc. is a leading worldwide developer, publisher, and distributor of interactive entertainment. It provides top performance for tens of millions of game players during the launch window of popular titles such as “Call of Duty.”

For the “Call of Duty” companion app, which released in conjunction with “Call of Duty,” Ghosts, Activision understood that quality and scalability were an essential part of the game launch. Keith Miller, Activision's Director of Technology, explained that mobile users are performance sensitive and that if performance suffers, “most players will never come back again.”

Activision worked with a third-party testing provider to supplement its in-house experience. Together, they performed regular load tests and performance reviews throughout development, incorporating their insights into Activision's development and testing methods.

Ultimately, the testing ramp-up paid off. Working this way allowed Activision to hit its release date while scaling to meet the user demands of launch day. This approach also enabled Activision to reduce troubleshooting time, while providing crucial insight into bugs and defects caught before release.

# New methods for success

## Agile

Agile development – first introduced in 2001 – has today all but supplanted the traditional “waterfall” method of software development that had prevailed since the 1960s. With short iterative cycles, a focus on working software or comprehensive documentation, and a collaborative approach to building apps, Agile methods such as Scrum have become mainstream today. Business of all sizes, from small start-ups to Fortune 500 organizations, are turning towards Agile methods to reduce time to market for new capabilities and increase their ability to change based on market demands.

Agile’s impact on testing has been enormous. Functional testing, traditionally done after all development is complete, is now done alongside development in order to deliver working, tested software at the end of each iteration (typically two-to-three weeks). Cross-functional teams composed of developers and testers work side-by-side to develop and test code quickly, increasing knowledge and reducing defects found late in the release process. But even for teams using Agile, performance testing and monitoring (unlike functional testing) is often not performed within each iteration. Mature Agile teams run nightly performance baselines and the entire performance regression tests each iteration, but this is not the norm for most Agile teams.

An emerging best practice for Agile teams is taking more responsibility for ensuring app performance with each iteration, rather than waiting until right before a major release to verify that performance goals have been met. This starts with involving members of the performance testing and operations teams as a part of a team’s Agile planning and delivery process. It also includes adding technical stories or tasks to the backlog to cross-train on knowledge while setting aside team capacity to focus on performance in addition to new features.

## DevOps

In recent years DevOps, an extension of Agile, has emerged to help break down the silos between app development and IT operations. With a strong focus on communication and collaboration and a healthy dose of automation, DevOps is helping automate most aspects of building, deploying, and testing the app along with provisioning environments, configuring, and monitoring the apps once they are live.



DevOps is impacting app performance testing and monitoring in many ways. A few of the more important ones include:

- Cross-pollinating skills related to performance testing and monitoring across development and operations team members, including shared processes and tools
- Integrating performance regression testing into the continuous-delivery pipeline so tests are run for each release candidate, and deployed only if a candidate meets all performance criteria
- Integrating monitoring consistently throughout dev/test and production environments so that performance issues can be identified, diagnosed, and fixed more quickly and without negative impacts to the user experience
- Creating more self-service capabilities for app dev teams so they are empowered to do their own performance testing and rely less on a single centralized performance-testing team for feedback on their apps

DevOps and Agile are helping organizations take a holistic, end-to-end approach to ensuring app performance is verified before every release to customers.

## Proactive performance management

Given the importance of app performance to customer experience and business results, organizations must move beyond reactive response to performance problems and must proactively manage performance all the time. Organizations that are in perpetual fire-fighting mode are reactive in responding to their customer's requests, their competition's next move, and the ever-changing market. Even when they have the right tools installed, reactive organizations often have ad-hoc, manual processes in place that greatly increase the cost and effort of root-cause analysis on performance problems. This is not sustainable.

By contrast, organizations that proactively manage app performance continually gather real-time data from their apps and quickly connect the dots to provide new insights. By automating the gathering, analysis, and dissemination of key performance data to the right decision makers, proactive organizations free their valued associates to spend more time on value-added practices. Additionally, there is less guesswork and more real-time feedback to teams on whether their latest improvements are having a positive (or negative) impact on app performance.

Using a proactive approach, organizations can better understand the correlation between investments they make to improve performance and the business value associated with the results. Regular app performance benchmarking and trending over time provides concrete data available for analysis, not only on the value of previous investments, but also as a basis for the business cases required for future investment requests.

# New tools for success

## Application-performance testing

Application-performance testing tools have come a long way in the last decade. In the pre-cloud era, organizations didn't have many choices if they wanted to performance test their apps. Enterprises could easily spend hundreds of thousands of dollars to purchase new tools and create new performance testing environments. Even after these initial investments, the additional "virtual user" licenses required for some of the leading tools made running performance tests frequently cost prohibitive. Organizations not wanting to make big investments were left with open-source options that could generate traffic, but lacked essential features for performance testing apps under realistic scenarios. These solutions often required organizations to focus their best software engineers on building custom testing frameworks around these tools – a costly and timely effort.

Filling this void between "too expensive" and "not enough" is a new generation of performance-testing tools. The most popular ones share common capabilities essential for balancing speed with quality and agility:

- **Software as a Service (SaaS) model.** Most of these new performance tools use the SaaS model and run on an underlying cloud infrastructure. Users can quickly register and get started right away without painful installs. For performance testing, these tools can quickly provision resources running datacenters around the globe to generate traffic and measure the results in one solution.
- **Record and playback with real browsers.** Just like popular acceptance-testing frameworks, new performance-testing tools provide simple record and playback capabilities for web and mobile apps that can be parameterized for use in performance testing. Some of the better tools can even take existing acceptance tests recorded in tools like Selenium and quickly convert them to performance tests as well as run the tests in real browsers instead of simple HTTP GET and POST commands.
- **Performance scripts used for monitoring.** The same performance scripts used for testing are also used for ongoing app-performance monitoring once they are released into production. These new tools are increasingly incorporating monitoring as a core part of the testing process to provide a full lifecycle support for performance testing before and after production releases.

- **Not just web but mobile too.** In addition to testing web apps, the next generation of tools is also focused on performance-testing apps designed to run on smartphones and tablets. Testing app performance across the myriad of mobile devices and carrier networks used by customers is a daunting challenge and these newer tools are trying to bring simplicity to the inherent complexity.

While these new tools are dramatically reducing the time, cost, and complexity of performance testing, there are some important differentiators:

- **Public only.** A big limitation with many public SaaS-based tools is that the app under test must have a publically available URL. With current security policies and practices, many organizations may not be able to make their app under test available via a public URL.
- **Cloud-based.** Many of the new tools are built using test agents running from cloud environments. While great for elasticity and auto-scaling of simulated users, the cloud is opaque, so individual test agents may report very different test results based on the shifting performance underlying multi-tenant cloud platforms rather than on the app's performance itself. These inconsistent results can greatly affect the accuracy of information that the test agents are reporting and potentially lead to time wasted while chasing phantom issues.

The best of these new-generation solutions mitigate these challenges by offering both in-cloud and on-premise solutions and by running their test agents directly from their own data centers – not on shared public cloud infrastructure.

## Application performance monitoring

The next-generation of app performance-monitoring tools offers a common set of capabilities that are essential for ensuring performance in production. These include:

- Proactively monitoring the availability, performance, and quality of business services and transactions enabled by one or more apps running across a distributed virtual and cloud ecosystem from the user's perspective
- Simple, intuitive screens and custom dashboards that bring together the most relevant information in a single pane of glass for clearly understanding the issues impacting performance – and then taking action

- Defined performance thresholds including service level agreements (SLAs) with proactive notification when performance deviates from expectations
- Storage and analysis of historical app-performance data with tools that can correlate this data with milestone events and be used as a basis for building cases for performance improvements
- Tight integration with other tools in the DevOps tool chain via plugins and APIs to be an integral part of an organization's overall ecosystem of development, testing, and operational tools

Traditionally, these performance-testing and performance-monitoring tools have been created by different vendors and used by different teams (quality assurance and operations). With the rise of Agile and DevOps methods, vendors are increasingly folding these distinct capabilities into tightly integrated solutions that appeal to the cross-functional teams responsible for ensuring performance.

## Virtualization and cloud

Customer-facing and business-critical apps are increasingly running on virtualized infrastructure. One of the biggest benefits of virtualization is the ability to create a layer of abstraction between the underlying hardware and the operating system that runs an app. This layer, called a hypervisor, allows more effective management of the hundreds or thousands of distributed virtual servers found in today's enterprises.

Building on virtualization, the cloud further abstracts the underlying hardware by making compute, storage, and network resources elastic and scalable on demand. With a usage-based business model and the ability for users to provision entire ecosystems of virtual servers quickly, the cloud is at the core of today's software-defined enterprise. The cloud is rapidly becoming the preferred platform for running critical web and mobile apps. While public clouds such as AWS and Azure get the most attention, private clouds from vendors behind the OpenStack standards are helping organizations build their own private clouds on their existing infrastructures.

## DevOps automation

As described, DevOps helps break down the silos between app development and IT operations. While focused around improving culture through better collaboration and communication, tools with a heavy dose of automation are what make DevOps work in practice. DevOps tools enable organizations to release new software quickly and safely, thus improving their overall business agility.

Some of the popular tools in the DevOps tool chain are:

- **Development:** version-control repository, automated build-and-deploy scripts, automated unit, regression, and performance test scripts, app configuration code and tests, static analysis, continuous integration
- **Collaboration:** wikis, instant messaging, ticketing/task management, knowledge bases, mailing lists, physical and virtual team rooms
- **Operations:** monitoring, service management, log file analysis, virtual or cloud management, virtual resource provisioning code and tests, run book automation

App-performance testing and monitoring tools are a part of the DevOps tool chain because they appeal to both IT operations, who are responsible for managing and troubleshooting performance problems in production, and app developers, who must be able to diagnose and fix any performance issues impacting the business and its customers.

## Key takeaways

IT and business leaders responsible for balancing agility and quality with app performance face many daunting challenges. Those leaders who focus on performance and overcome the challenges are experiencing success while those who do not are negatively impacting their brand and likely losing market share. Effective app performance testing and monitoring requires integrating the methods and tools described in this report into existing development and operations cultures, managing change along the way.

Here are some concrete steps leaders searching for the balance among performance, agility, and quality can take:

- **Be a change agent in your organization around improving performance – it makes great business sense for your organization and your career.** Numerous studies point to a fact: app performance directly impacts customer experience. Organizations that focus on improving performance are seeing the benefits of happier customers and improved business results.
- **Seek out opportunities to share knowledge, methods, and tools across your QA, operations, and development teams. Start with simple lunch-and-learns and go from there.** Ensuring consistent app performance for every release of every app is an enormous challenge that requires an integrated approach to performance testing and monitoring. It starts with having the all the people responsible for performance knowing each other and a common way of understanding end-to-end processes.
- **Make performance a key part of your product planning and Agile delivery.** Leaders should use Agile and DevOps to manage app performance proactively across all the teams involved with planning, designing, developing, testing, and operating apps.
- **Identify your key criteria and start evaluating solutions. Most tool vendors offer free trials to minimize up-front investment.** By automating the gathering, analysis, and dissemination of key performance data to the right decision makers, organizations free their teams to spend more time on value-added practices related to proactive management. Next-generation app performance testing and monitoring tools are a natural place to start. They form essential building blocks for high-quality releases – quickly and every time.

## About Ryan Shriver

Ryan Shriver is the Director of Technology with SingleStone, a company focused on helping our customers improve their customer's experiences. There he leads a team of talented individuals who engineer for the human experience using methods and tools including Agile, DevOps and the Cloud. He works with clients ranging from start-up to Fortune 500 across a range of industries..

## About Gigaom Research

Gigaom Research gives you insider access to expert industry insights on emerging markets. Focused on delivering highly relevant and timely research to the people who need it most, our analysis, reports, and original research come from the most respected voices in the industry. Whether you're beginning to learn about a new market or are an industry insider, Gigaom Research addresses the need for relevant, illuminating insights into the industry's most dynamic markets.

Visit us at: [research.gigaom.com](http://research.gigaom.com).

© 2014 Giga Omni Media, Inc. All Rights Reserved.

This publication may be used only as expressly permitted by license from Gigaom and may not be accessed, used, copied, distributed, published, sold, publicly displayed, or otherwise exploited without the express prior written permission of Gigaom. For licensing information, please [contact us](#).