

AKAMAI WHITE PAPER

**The Complete Guide to Performance
Testing Your Retail Websites and Apps**



Table of Contents

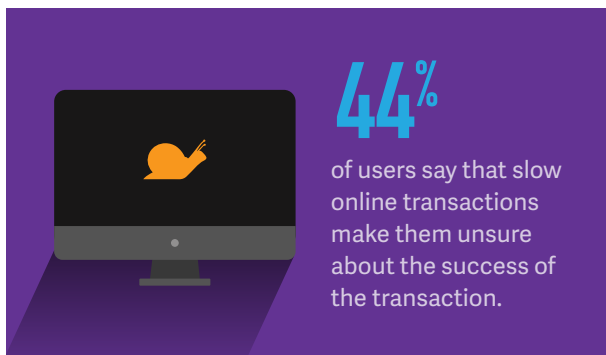
Getting started with testing retail web and mobile properties	1
What are some of the common performance challenges faced by online retailers?	2
Getting started: Performance test checklist	2
What are the top business metrics for online retail?	3
Who should be on your testing team?	4
What to watch for when testing	5
Types of performance tests	6
Testing in production vs. testing in the lab	6
Testing in the performance lab	7
Testing in production (Yes, it can be done)	8
Security and performance testing	10
Appendix A: The Akamai Performance Test Methodology	11
Appendix B: CloudTest Performance Data Q&A	12

Getting Started with Testing Retail Web and Mobile Properties

Retailers competing in the high-pressure world of online retail have many challenges. Not only do you need to provide an interesting and compelling shopping experience, you also need to make that experience fast and reliable.

Almost half of all online shoppers say they will abandon a page that takes more than 2 seconds to load. In a traditional brick-and-mortar scenario, abandoning one store for another requires leaving the outlet and physically traveling to the waiting arms of the competition. On the web, competitors are just a couple of clicks away.

MAKING PAGES FASTER IS A PROVEN STRATEGY FOR INCREASING CONVERSIONS AND REVENUE.



At Walmart.com, every 1 second of load time improvement equals a 2% conversion rate increase. Staples.com shaved 1 second from its median load time and improved its conversion rate by 10%.

SITES THAT ARE SLOW OR UNAVAILABLE CAN HAVE A LASTING IMPACT ON CUSTOMER RETENTION.

Revenue isn't the only metric that is affected by outages and slowdowns. Akamai studied the impact of outages versus slowdowns on abandonment rates for online retail sites. Their findings were eye opening: Sites that

went down experienced, on average, a permanent abandonment rate of 9%. Sites that suffered from slow performance experienced a 28% permanent abandonment rate.

TESTING IN PRODUCTION IS THE ONLY WAY TO ENSURE THAT ONLINE APPLICATIONS WILL PERFORM AS EXPECTED.

External testing against the production infrastructure is the best way to get a true picture of capacity and performance in the real world. (Of course, this doesn't obviate the need for — or eliminate the benefits of — testing in a lab environment as well. It's important to have continuity between the two.)

So, how can you build confidence that your online shop will provide a great shopping experience and withstand the traffic spikes caused by seasonal upswings, events, and promotions? And how should you measure success?

This guide is designed to help you get started by answering questions like:

- What are some of the common performance challenges faced by online retailers?
- What should be on your performance testing checklist?
- What are the top business metrics for online retail?
- Who should be on your testing team?
- What types of tests should you run?
- What should you watch for when testing?
- Why should you test in the lab? And why should you also test in production?

Let's get started.

What are some of the common performance challenges faced by online retailers?

Online shops work hard to mirror their bricks-and-mortar counterparts. That's because shoppers are accustomed to going to physical stores and being able to see, touch, and engage with products before they make a decision.

The web can't offer the same level of interactivity as a store, but thanks to a plethora of features —videos, recommendation engines, user reviews, high-resolution images showing products from multiple angles, tools that let you “try on” clothing — retail websites offer an unprecedented shopping experience.

There's also the challenge of conveying your brand online. Stylesheets and custom fonts let control how you present your brand, but these features can come at a cost to your site's performance and availability.

Balancing performance with the rich content used to capture a specific design goal can be a struggle.

Behind the scenes, back-end systems for payment processing, authorization, inventory checking, and more can also add performance challenges.

And finally, many of these features — from user reviews to custom fonts to payment processing — are created and hosted by third parties. This content can add functionality or information to your site, but it also adds a layer of traffic and complexity that is extremely difficult to control.

Understanding the impact of all of this content is critical to managing the overall performance of your site.

Getting Started: Performance Test Checklist

Creating a successful test strategy is key to a successful performance engagement. It begins with understanding how your users interact with your online retail site.

For example, the following information is crucial to defining the test engagement strategy:

- Main business processes (flows) followed by site visitors
- Average time a user stays on the site
- Average percentage of users that complete a purchase vs. users who browse only
- Abandonment ratios and where in the process users abandon
- Average and peak number of concurrent users per hour
- Average and peak page views per minute/hour
- Average and peak orders placed per minute/hour
- Traffic pattern differences for the above metrics for specific events such as Black Friday or Cyber Monday
- Geographic regions of the site traffic's source
- Percentage of traffic that comes from mobile devices, the types of devices, and the differences in user flows and the impact on the above metrics
- If a CDN is used, percentage of content that is served from the CDN

This information helps to define a successful performance testing strategy and drives how your test will be architected and executed.

What are the top business metrics for online retail?

Online retail sites have specific key performance indicators (KPIs) that help you measure your site's performance:

- **Orders per minute** — For an online retail site, orders completed in a period of time is the gold standard KPI. Orders directly translate to revenue.
- **Page views per hour or minute** — Customers can only complete orders if they can effectively use the site. Ensuring that the online retail site can serve web content as users search, compare, and interact is critical.
- **Sessions per hour** — Sessions are individual users or systems interacting with the site. Ensuring that those users can establish and maintain a session for the duration of their interaction is very important.
- **Errors** — It may seem obvious, but keeping track of both the overall error rate and the type of errors you are receiving is critical. Not all errors are created equally.
- **Average response time** — Understanding how long, on average, pages and page assets take to be served is important for uncovering potential bottlenecks. It's also the basis for many customers measuring the utility of an online retail site.
- **90th percentile response time** — This provides an even finer level of granularity when it comes to response time. 90th percentile removes the slowest 10% response times. This eliminates timeouts (which can average 120 seconds) while giving a true indication of the response for 90% of the users.

How fast is fast enough?

A retailer striving for good site performance has an average response time of less than 3 seconds, with a 90th percentile response time of less than 2.75 seconds.

A retailer wanting to ensure their site is among the leaders in performance needs to know how fast is fast enough.

It's worth noting there are different user expectations for different types of pages. For example, a user browsing the site will expect lightning-fast response times, while the final step in a checkout process is generally expected to take a bit longer.

If an online retail site can maintain a target level of performance while testing with 150–200% of expected peak load, a retailer can go into any marketing event or holiday season with increased confidence.

Who should be on your testing team?

It is vital to have the right people involved in the test execution.

When the right mix of people is brought together, tests are much more productive. When issues arise, having the right team available enables those issues to be quickly taken care of and decisions to be made when actionable intelligence is provided.

The size of your company and staff, the scale of the site, the complexity of the testing, whether or not you've outsourced development or deployed third-party applications within your site, the use of a managed service provider for infrastructure, and more all influence how many individuals are involved in a test.

In some cases it's one or two people who do everything. In other cases, such as the complete reconstruction of a site or testing a popular site for seasonal readiness, there will be multiple people for many of the responsibilities, from architects to individual contributors responsible for specific elements of the infrastructure.

There are a number of responsibilities to address as part of a testing strategy:

- **Coordination** — Coordination of testing activities with all key stakeholders. This includes application engineering, operations, vendors, and third-party connected services, as well as business and senior leadership.
- **Communication** — Communication of test results, issues, issue resolution plans, and progress against overall strategy to business and engineering leadership.
- **Strategy** — Definition of an overall strategy that includes establishing business processes for applications, key performance indicators, capacity plans, monitoring coverage, and individual test plans that roll up into a performance management aspect of the software development life cycle.
- **Architecture** — Delivering best practices on creating high-performing application and infrastructure architectures for new projects or for improving existing applications.
- **Test creation** — Turning business process definitions into executable tests, creating individual performance test workloads from capacity plans, and maintaining the current library of test cases on an ongoing basis.
- **Test execution** — Executing ongoing performance tests in the performance lab and production environment with delivery of test results.
- **Analysis** — Reviewing test results from all environments and analyzing the performance with respect to previous tests. Responsibility for calling out any violations of established success criteria, thresholds that may be violated on key performance indicators, or deviation from previous baseline tests.
- **Diagnosis** — Root cause analysis of bottlenecks or performance problems that might arise. Using domain knowledge and specialized tools such as profilers, memory leak detectors, and monitoring tools to pinpoint problem areas.
- **Tuning** — Tuning involves applying best practices, tuning recommendations, and isolating parts of the application or infrastructure that could be optimized to give better capacity or performance.
- **Measurement** — Responsibility for analyzing activities and their progress against the overall strategy, in addition to making process improvement recommendations for optimizing all strategic or tactical activities.

In small companies, one person may take on all of these responsibilities, often assisted by Akamai in test strategy, creation, execution, and analysis.

Some or all of the roles may be outsourced, typically to take advantage of expertise, create focus on areas lacking attention and/or to lower costs.

Normally there is a blend of individuals and roles. For example, the presence of a Project Manager, performance engineers and/or specialists would either remove the need for a tech lead, or allow that person to fill other gaps in the process.

Below are common roles/titles and how they might map to the various responsibilities:

- **Project Manager** — Coordination, communication
- **Technical Lead** — Coordination, communication, strategy, architecture, analysis, diagnosis, tuning, measurement
- **Architect** — Infrastructure architecture, app architecture
- **Performance Engineer** — Develop strategy, architecture, analysis, diagnosis, tuning, measurement
- **Test Engineer** — Test creation, execution, analysis
- **Specialist** — Diagnosis, tuning

What to Watch for When Testing

Understanding what makes an online retail site fast or slow helps to focus your testing effort.

Many online retail sites are quite complex and made up of several different components and application tiers. It's important to understand each of these components and how they interact with each other.

Some of the common areas to focus on while testing are:

- **Application issues** — There is no such thing as perfect code. Watch for inefficient code, synchronization issues, garbage collection, memory leaks, and application dead locks.
- **Database performance** — This is the core of performance. Watch for locking and contention, missing indexes, inefficient queries, memory management, connection management, and unmanaged growth of data.
- **Configuration settings** — The default settings are rarely optimal. Check for differences between environments and understand the tuning options and best practices for the various devices in the architecture.
- **Load balancing** — Use hardware efficiently. Look for algorithms that are not optimized and underutilized features and capabilities.
- **Connectivity** — Communication is vital. Ensure that systems can communicate with a minimum of latency, that the firewall has sufficient capacity, that the system is optimized for mobile networks, that the DNS routing is correct, and that CDN caching is optimized.
- **Bandwidth** — Can you be reached? Ensure that the bandwidth pipe is big enough for the traffic. Review the content making up the pages. Rich content can mean big data and bandwidth requirements. Verify that the site can support different connection types/speeds including mobile devices.
- **Architecture** — Match the car to the engine. Watch for unbalanced tiers, mismatched technology choices, or a dead-end scalability path.
- **Third-party services** — You are only as fast as your slowest page resource. Ensure that analytics and tracking, payment systems, aggregated content, social networks, or CDNs are not slowing down your site.

Types of Performance Tests

A question that often comes up is “What kind of tests need to be run?”

There are several test options — baseline, stress, spike, endurance, and failover. For a retailer, all of them have value.

Baseline

Retailers need to establish a baseline of what is acceptable performance for their site when under “average load.” We recommend taking the last six months of analytics and the busiest hour from each day and using that as your average load values for page views/hour and orders/minute.

Stress

Retailers need to run stress tests to ensure that site performance does not degrade under heavy load for an extended period of time. It is not unusual for memory leaks or faulty garbage collection to cause performance issues that are not seen until stress tests are conducted. Akamai recommends that stress tests be executed at 150–200% of peak expected load.

Spike

A spike test is when a very large number of users ramp up very quickly or almost simultaneously. For retailers, spike tests are vital. Many retailers have spike events — flash promotions, Black Friday, Cyber Monday, Valentine’s Day, etc. — and these events can bring a site to its knees. Retailers have to ensure that during such events, orders are not lost or users turned away. Akamai recommends that spike tests be run at 200% of peak expected load.

Endurance

Endurance tests differ from stress tests in magnitude. While a stress test will have a very high number of page views and orders, an endurance test will simulate slightly less load but for a longer period of time. These tests have great value as they can uncover additional areas where performance can degrade over a prolonged period of time. There are batch jobs, database backups, and cache refreshes that can be scheduled and run on a recurring basis. Running an endurance test over several hours may show that these types of events affect performance.

Failover

Most retailers have disaster recovery or failover servers that should spring into action in case of a failure. If a hardware failure occurs during a period of heavy load, can the backup hardware adequately step into the void? There is no way to know unless a failover test is executed.

In early testing, the goals are often simply to understand the performance characteristics of the site. For those tests, the ramp-up rates may be slower and more methodical.

Once these baselines are established, and you have a good sense of what the site can handle, it’s critical that the tests be executed as realistically as possible. This includes having realistic user levels and ramp-up rates. Reviewing the analytics from previous years will help you build user ramp-up levels.

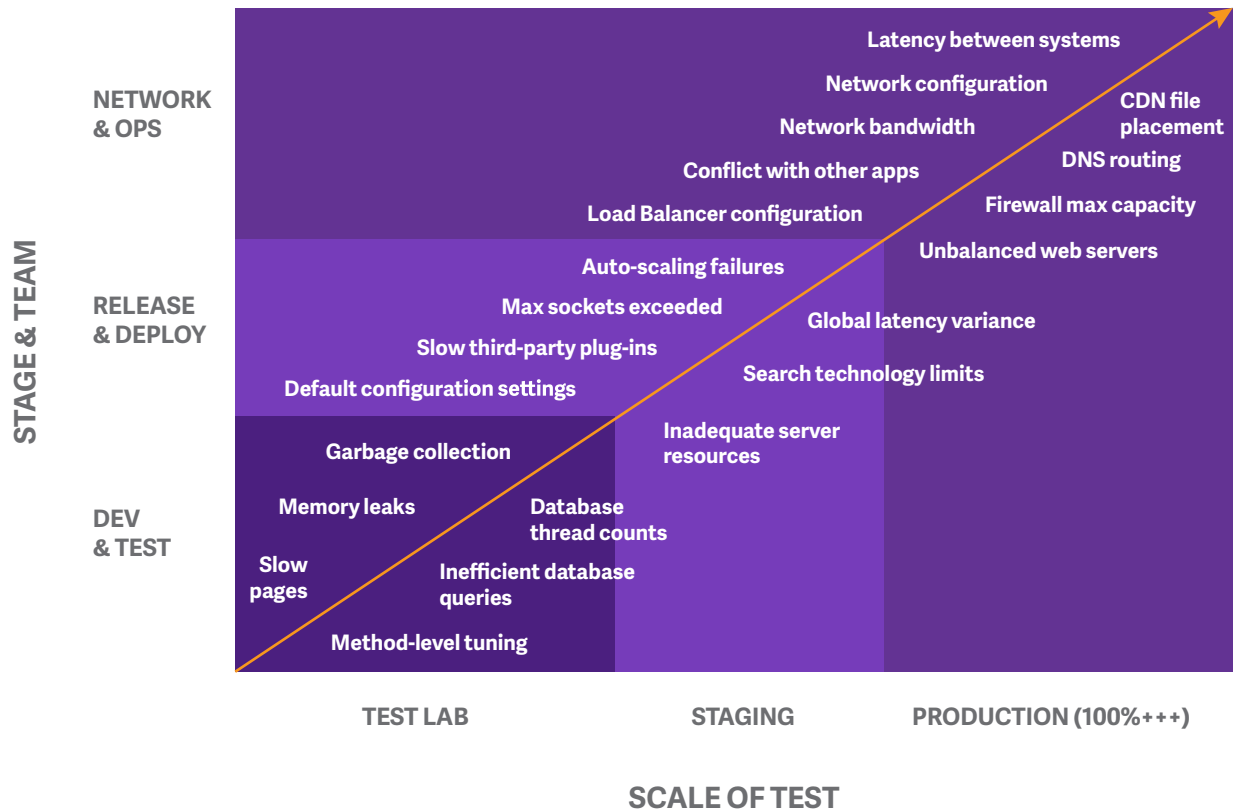
It is imperative that you do not cheat the test. Do not alter think times or remove critical components of the test. Removing or altering them will only skew the results and lessen the value of the tests.

Testing in Production vs. Testing in the Lab

To test for success, you need an effective online retail test strategy. Having the right team in place to implement the strategy will then enable the tests to be executed successfully. Testing for success means understanding the value of testing in production and in the performance lab.

Executing different types of tests in different environments is a key element in finding and resolving bottlenecks in architectures that make up an online retail site, which are often quite complex.

Building an End-to-End Test Plan where production testing is an essential component



The graphic above illustrates how different types of tests in the lab and in production can help pinpoint specific issues as you work toward optimal performance.

Testing in the Performance Lab

Ongoing performance testing in a lab allows application engineering teams to assess performance over time, and helps catch any showstopping performance bugs before they reach production.

In addition, the lab provides a place to performance-test code and configuration changes for performance regression before releasing changes to production and outside of the normal build cycle. This could include things like a quick bug fix in a page, or a seemingly minor configuration change that could have a performance impact. Often, these kinds of changes are deployed with little to no testing and come back later as a cause of performance issues.

Testing in the lab also gives you the ability to break the system without having to worry about the impact on users.

It is important to understand how the online retail site responds when taken to failure and how it recovers. No one wants to bring a live production site to failure.

The Akamai CloudTest Methodology is a software QA performance testing strategy with a set of lightweight processes that ensure that online applications are ready for peak usage

Akamai CloudTest Methodology			
Strategy	Implementation	Execution	Measurement
Define	People	Define	Analyze
Integrate	Processes	Design	Adjust
		Test	
		Assess	

Testing in Production (Yes, it can be Done)

The only way to truly build confidence in a site is to test that site in real-world situations.

Due to the sheer number of users accessing their site at any given time, very few companies can replicate a production environment in a staging environment or in a performance lab. Testing in production is the best way to get a true picture of capacity and performance in the real world. And it is the only way to ensure that online applications will perform as expected.

Traditionally, tests have been limited to making extrapolations that apps tested on a few servers in a lab can support exponentially higher amounts of load in production. Without proper testing, these types of assumptions always result in unexpected barriers.

SOME CATCHES CAN ONLY BE FOUND BY TESTING IN PRODUCTION

Here's a shortlist of issues that can't be found in lab tests:

- Batch jobs that are not present in the lab (log rotations, backups, etc.) or the impact of other online systems affecting performance
- Load balancer performance issues, such as misconfigured algorithm settings
- Network configuration problems such as 100 MB settings instead of 1 GB on switches, and routing problems
- Bandwidth constraints
- Data pipe not configured as "burstable" to handle spikes in traffic
- Latency between systems inside and outside of application bubbles
- Misconfigured application servers or web servers
- CDN not configured to serve up new content
- Radically different performance depending on database size

CONTENT DELIVERY NETWORK (CDN)

The vast majority of test labs do not have a CDN as part of the infrastructure. Therefore, it's impossible to test how the CDN affects performance without testing in production.

When a retail company tests in production, it can fully test the caching/offloading capabilities of its CDN. This is vital to understanding the true performance of a production environment.

When preparing to test in production it is imperative to engage with the CDN provider early in the process to ensure that a production test has the support of the CDN provider. (Akamai has developed a set of best practices for tests that include CDN assets.)

One Akamai customer, a performance engineer for a multi-billion-dollar retailer with more than two thousand storefronts, considers the inclusion of a CDN a vital part of production testing:

“When we test in production, we generally have Akamai caching enabled. This definitely influences the testing in terms of number of requests that reach origin (main site) and the response times depending on from where the page is served — Akamai or origin. In the lab it is most likely that we won't include Akamai. This means we have to include the CDN in our production tests or we have no true idea of our production performance.”

THIRD-PARTY CONTENT

Many online retail sites use third-party providers to enhance their overall site content. It's vital to involve those third-party providers that might have an impact on performance when the strategy is being formulated.

On the other hand, you would not typically include domains such as Google Analytics or Omniture metrics as part of your test. They do not want to be surprised by a test or have it bring down their service or site with fake transactions.

Involving third-party providers early in the process helps to ensure their support. They may even want to be measured in the process. After all, they want to have a well performing site, too.

POTENTIAL IMPACT TO LIVE CUSTOMERS

Testing in production can be done with or without live users on the environment. (Note: The majority of customers testing in production with Akamai CloudTest are doing so on a live environment.)

It's possible, but not always feasible, to put up a maintenance or turn-away page and wait until all existing user sessions have finished. In reality, this method is actually rarely used because the right tool and methodology working together can allow testing to take place with active users on the site in all but the most extreme cases.

To use one Akamai customer's exact words, “The cost of not testing is far greater than the potential impact on live customers.”

You can make revenue at the same time you're testing — with the right approach. It's also possible to segment a portion of the live environment during a low-traffic period and allow for testing on this sub-environment. Or test against one data center while live users are all routed to another data center.

Typically a separate IP address is configured on a load balancer and servers are moved out of the live pool and placed into the test pool. Sometimes configuration changes need to be made to other shared components. This is a more costly approach due to the need for extra hardware and the maintenance overhead. It's also somewhat less reliable because you start to deviate from the actual production configuration, and you cannot test to true scale. It is, however, a more realistic test than simply testing in a lab.

THREE REQUIREMENTS FOR SUCCESSFUL LIVE TESTING

1. Real-time analytics

The first requirement for being able to do a large-scale test with live customers is having real-time analytics in your test tool. With up-to-the-second information about site performance, you can quickly tell if a site has started to experience poor performance or become completely unresponsive.

2. A good “kill switch”

You must be able to press “stop” or “abort” to stop load immediately in a running performance test. Bandwidth, concurrent connections, threads in use, and other typical pinch points will all drop back to normal.

3. Solid internal monitoring practices

Finally, having good monitoring practices internally, preferably integrated with your testing solution, can prevent you from ever needing to abort a test because of live user impact. Real-time visibility into metrics like CPU utilization, heap usage, garbage collection, and connection counts on load balancers or firewalls can help prevent those thresholds from ever being violated by routine testing.

Security and performance testing

Security is one of those things that no one notices until something goes wrong. You rarely see a story on the news about how a company is doing a great job of keeping their customer data secure. But you will see a story about a company letting customer data slip into the wrong hands.

Data security is one of the key concerns when it comes to performance testing. Companies do not want their corporate data outside corporate firewalls.

Your performance testing solution should:

- typically generate synthetic data or use scrubbed data depending on your business and security requirements;
- only capture key HTTP/HTTPS metrics and statistical data for reporting, analytic, and diagnostic purposes; and
- have a stringent security policy to ensure that your data remains safe.

And it should not:

- ask you to export data or your applications outside your firewalls; and
- save any response content on the test servers or in the result database.

Appendix A: The Akamai Performance Test Methodology

Akamai's methodology emphasizes various tests in different places to find what is often a needle in a haystack.

AKAMAI'S CLOUDTEST METHODOLOGY

The Akamai Performance Test Methodology is a software QA performance testing strategy with a set of lightweight processes that fit into the software development life cycle and ensure that online applications are ready for peak usage. The methodology includes testing applications in a lab environment inside of the firewall and processes for testing the production infrastructure used by customers.

Cloud testing leverages the efficiencies of the cloud to test web applications and sites. By using the infrastructure and services provided by companies like Amazon Web Services and Rackspace, Akamai customers can reduce the cost of load and performance testing while increasing the accuracy of representing the actual traffic to the site. CloudTest is a distributed architecture that can also be deployed completely behind the firewall, or in a combination of on-premise and cloud-based configurations.

Based on years of experience testing from the cloud and leveraging existing best practices and proven performance testing methodologies, the Akamai methodology extends traditional approaches to address new opportunities and challenges. These include:

- Testing both in the lab and with live web-based applications in production
- Leveraging the cloud to test at both typical and peak traffic levels, from hundreds of users to millions
- Responding to the acceleration of development cycle times by making agile performance testing a realistic alternative
- Generating geographically dispersed load to most accurately represent real-world traffic
- Generating both internal and external load and against both the lab and production environments for the most efficient and effective results
- Analyzing performance intelligence in real-time to speed problem resolution

STRATEGY AND PLANNING

Akamai's approach to performance engineering calls for an umbrella strategy with associated individual test plans.

Test plans roll up into an overall view that ensures confidence in the ability of key revenue generating applications to perform as expected.

The result is an ongoing performance engineering strategy throughout a site's evolution.

It includes a number of test plans centered on individual objectives such as holiday readiness, a major architectural change, or the release of a major version of code.

Having a well-defined strategy, with explicit test plans, provides business and engineering leaders with a high degree of confidence in operational readiness. Using this approach gives greater insight into an application's performance.

Using an iterative process within test plans to achieve defined goals allows for a stream of continuous improvement in the applications being tested.

It starts with the test definition and ends with obtaining actionable intelligence results.

The process of creating a test plan starts with the Define Phase. During this phase, the flows to be tested throughout the site are defined, metrics to be monitored are established, and success criteria for the tests are agreed upon. In the Design Phase, the user scenarios are written and test parameters are set up. Things such as the mix of users executing different parts of the application, the virtual user targets, and the ramp time are modeled.

The Test Phase is where the execution of tests takes place, and where data is collected for assessment.

Finally, the Assess Phase, parts of which may occur during the test execution, is when the data collected throughout test execution is used to provide actionable intelligence.

Appendix B: CloudTest Performance Data Q&A

WHAT IS PERFORMANCE DATA?

There are many different types of performance data that are leveraged as part of performance testing. Performance data can be broken up into three major categories: master data, user-generated data, and external data. Master data typically exists in the customer's database(s) and is required for conducting business (usernames, passwords, etc.). User-generated data is anything that the user inputs in editable fields on the application (new email address, new address, etc.). External data is provided upon execution of the application (confirmation numbers, session IDs, etc.).

WHAT DATA IS USED DURING TESTING?

Data requirements are dependent on the application and the business processes under test. For example, a static site might not require any performance data, just access to the site to make the requests, while a more complex application might require all three types of performance data.

WHAT HAPPENS TO THE RESPONSE DATA RECEIVED?

All customer-related response data is discarded from CloudTest servers during load testing. Only performance data related to that response data is retained (response times, errors, etc.). In addition, as noted above, CloudTest servers are temporary instances discarded after each test, with only performance metrics retained in the results database. However, all customer and external data is stored on the CloudTest server during script creation and debugging. The data used to create scripts can be deleted after script creation is completed to ensure no customer data is stored on Akamai servers.

HOW ARE LOG FILES OR OTHER METRICS THAT CAPTURE INFORMATION ABOUT DATA HANDLED?

Akamai does not keep log files or other metrics that capture information about customer data during load tests. But as previously noted, we do store this type of data during script creation.

WHAT DATA IS SENT BACK TO THE CLOUDTEST SERVERS?

During load testing, only key external data is kept, such as HTTP response codes, cookies, session IDs, etc. All data is sent back to the CloudTest servers to parse, at which point it is fully discarded from any Akamai servers.

WHAT IS THE DIFFERENCE BETWEEN SCRUBBED AND SYNTHETIC DATA?

Scrubbed data resides in a customer database and has gone through a process so it no longer includes any real customer data; it is, in essence, data that has been transformed and/or created from actual customer data. Synthetic data is generated from scratch and is designed to be an accurate representation of the customer's production database.

HOW DO WE CREATE SYNTHETIC DATA?

Akamai works closely with your application team to ensure a rich spread of representative data is created. Data is also created to target the business process under test and can expand as testing expands.

WHAT HAPPENS TO THE DATA ONCE THE TESTING IS OVER?

When testing is complete, the CloudTest servers only store performance metric data relevant to the test runs. No corporate data is retained on the CloudTest servers.

HOW IS THE TEST DATA STORED IN THE CLOUD?

For tests executed by Akamai, the test results are stored in the cloud. The results are on Linux servers that are taken down at the conclusion of the testing event. These servers are only available during testing sessions or when results are being analyzed. In most cases, the result data is stored in a relational database in EC2 in an EBS (elastic block store). This data is only available to Akamai employees.

CAN TEST RESULT DATA BE REMOVED FROM THE CLOUD?

Yes, results can be deleted and removed from the cloud at the request of the customer. When requested, the results are not deleted until after the result report is completed. Once the results are deleted, they are permanently deleted; there are no backups of deleted results. Note that deleting results limits the ability to compare results from prior and future tests. The only way to compare results once a result is deleted is by looking at the result report.

CAN TEST RESULT DATA BE EXPORTED FROM THE CLOUD FOR OFFSITE STORAGE?

Yes. All results are exportable in XML format. Results can be exported from the cloud and moved to a customer-defined location for safekeeping. Depending on the length of the test and amount of performance data related to the test, it can take up to 24-48 hours to export the data from the cloud and move to a secure location. Optionally, the results can then be deleted from the cloud as well.

We help companies like Apple, Target, Etsy, and Microsoft continuously test, monitor, and optimize their web and mobile applications.

We can help you, too.

Get Started with CloudTest Today!



As the world's largest and most trusted cloud delivery platform, Akamai makes it easier for its customers to provide the best and most secure digital experiences on any device, anytime, anywhere. Akamai's massively distributed platform is unparalleled in scale with over 200,000 servers across 130 countries, giving customers superior performance and threat protection. Akamai's portfolio of web and mobile performance, cloud security, enterprise access, and video delivery solutions are supported by exceptional customer service and 24/7 monitoring. To learn why the top financial institutions, e-commerce leaders, media & entertainment providers, and government organizations trust Akamai please visit www.akamai.com, blogs.akamai.com, or [@Akamai](https://twitter.com/Akamai) on Twitter. You can find our global contact information at www.akamai.com/locations. Published 07/17.