

dotcom-monitor



Shift Left

Why Your Performance Testing Strategy Needs to Shift Left

You have likely uttered the phrases “test early and often” and “shift left,” but do you always remind yourself of the importance of that phrase from the end user’s perspective? Or has it become white noise? While the agile approach is focused on implementing small, iterative tests throughout the development and testing phases, the primary goal should be to deliver a world-class, consistent user experience. Your customers demand optimal performance, whether they’re the only user or one of a thousand.

Today’s applications are built on multiple technologies, relying on vast networks of third-party providers and CDN’s. Additionally, end users can access your site or applications from anywhere in the world using different browsers, operating systems, and mobile devices, all with varying connection speeds. Response times, quality, and availability are critical factors that need to be assessed before pushing applications into production.

Knowing exactly how much a site, application, or API can handle during the development and testing phase can help identify and uncover various issues, such as bugs, hardware bottlenecks, and database slowdowns, well before your application is pushed into your production environment. These performance metrics are necessary to assist with capacity planning to meet the demands of your customers. Poor performing sites and applications impact conversions, and ultimately, revenue.

LoadView - Best Practices and Process

LoadView, our on-demand, cloud-based load and stress testing platform, takes an outside-in approach to performance testing. It helps assess how a site, web application, or API will respond to various traffic without adding any additional infrastructure. Executing an internal test can tell you how well your application or site handles an increase in traffic, but it will never reflect real-world conditions. LoadView can simulate thousands of concurrent users during a test and allows you to create scenarios based on a user’s physical location. The results from LoadView tests can also assist in these critical areas:

- Establishing response time baselines under specific user load numbers
- Identifying performance bottlenecks
- Finding upper limits of your current systems for capacity planning
- Analyzing server performance (CPU, memory, bandwidth, disk I/O) and database response times

Additionally, along with the EveryStep Web Recorder, you can create advanced scripting actions that mimic real user actions with your application, simulating over 40 desktop/mobile browsers and devices. The scripts can be uploaded into the LoadView platform and replayed by a virtually unlimited number of simultaneous users, giving you actual performance from real browsers.

Load and stress tests, in addition to other performance tests, should be conducted as early as possible during the development phase of a release or update. However, before heavy traffic days, such as Black Friday or Cyber Monday, it's vital that testing be completed months prior so issues can be identified and resolved before they potentially impact customers – and your reputation.



- **Plan.** Design your test without the hassle of managing hardware, giving you the ability to identify objectives and define a scenario by setting up a number of users and test duration. Distribute user load by geographical testing locations, simulating your customer base.
- **Execute.** As your test begins, the service instantly allocates the required capacity to execute your scenario. All resources are dedicated to this test – no shared servers. Occasionally, a load test may become a stress test if it hits certain thresholds that cause the site's performance to degrade or fail.
- **Visualize.** Use baseline metrics to identify performance degradation by reviewing elements provided by waterfall charts. You can also export all HTTP related data in CSV format for analysis using any tool you choose.
- **Monitor.** Continue to use your load test scripts to create continuous monitoring to check for performance, functionality, and accessibility of your services with UserView monitoring.

Shift Right - Monitoring Performance after Deployment

After your application goes into production, the scripts that are created during load testing can be uploaded into our UserView monitoring platform. UserView replicates a user's browser interactions with a website or an application. You can simulate specific user actions, such as mouse clicks, page navigation, shopping cart, and text inputs. These actions are recorded using our EveryStep Web Recorder, allowing you to record and playback virtually any set of critical transaction steps for monitoring. EveryStep is one of the few tools on the market today that allows you to interact with Rich Internet Applications (RIAs), such as AJAX, Flash, HTML5, PHP, Ruby, etc.

The ability to monitor and test applications by mimicking a user's path - and being alerted if something goes wrong – gives you and your team time to troubleshoot performance issues before they affect other users, thus reducing downtime and ultimately improving the overall user experience. So, even though “test early and often” might be cliché to say, as your development processes and technologies shift left, not only will your users and customers be happy, but you'll also gain efficiencies, achieve better results, and give you and your organization peace of mind.

Think Big, Act Small

