

Business white paper

# Short test cycles

For performance testing with HP TruClient Technology



# Table of contents

- 3** Keeping up in a complex era
- 3** Riding the Web 2.0 wave
- 4** Adjusting for modern methods
- 4** Out with the old: The trouble with traditional testing tools
- 5** An up-to-date solution for testing Web 2.0 apps
- 5** Benefits of HP TruClient
- 7** Inside HP TruClient technology
- 8** Answering the Web 2.0 challenge

## Keeping up in a complex era

Many of today's Web applications are developed using Web 2.0 technologies. These powerful new tools offer advanced features for building dynamic, interactive, user-friendly applications that help companies stay ahead of competition. But at the same time, they also present considerable new risks and challenges. The changes in the way rich Internet applications (RIAs) work and behave make it even more critical to adequately test the performance of the applications. However, the sheer complexity of the new technologies, the ever-growing and changing number of toolkits, and the absence of industry standards make it difficult to find an accurate solution for performance testing.

Traditional testing tools can't keep up with the continual new levels of complexity of RIA. HP TruClient technology, available in HP LoadRunner and HP Performance Center, is an answer to this challenge—an innovative, browser-based virtual user generator that supports simple Web applications, as well as modern, JavaScript-based applications built with virtually any available Asynchronous JavaScript and XML (Ajax) toolkit. By using the UI-level recording based on the browser's own Document Object Model (DOM), the HP TruClient technology makes scripting much easier, so that companies can use nontechnical or novice testers for scripting. This frees up valuable technical resources to focus on other critical tasks such as the actual test execution and analysis. In addition, TruClient also makes scripting much faster, allowing companies to significantly reduce test cycle time.

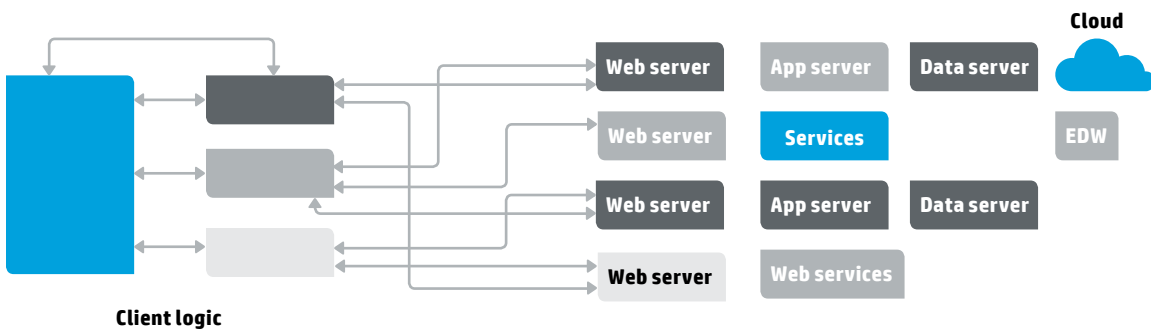
This paper explores the challenges of testing Web 2.0 applications and looks into reasons why traditional testing solutions aren't a good fit for testing applications built with Ajax and other Web 2.0 technologies. It introduces the HP TruClient technology, and discusses how this revolutionary technology helps advance, accelerate, and simplify performance testing for Web and Web 2.0 applications.

## Riding the Web 2.0 wave

Web 2.0 has taken the IT world by storm. It has provided a new way of developing and using Web applications: taking a leap from simple, static methods of retrieving and delivering information to highly complex, interactive, user-centric, collaborative Web applications. A growing number of companies are taking advantage of the new techniques and concepts of Web 2.0, using Web 2.0 technologies to help with their application transformation and modernization initiatives. Organizations are building new, modern, interactive front ends for their core business applications while preserving the underlying business logic and database structure.

Technologies such as Ajax give companies a great way to add a fresh, contemporary look and advanced features to their applications without having to completely rewrite them. Almost all new Web applications today are developed using technologies such as Ajax, Adobe® Flash, Adobe Flex, or interactive Web application frameworks such as Microsoft® Silverlight. The unique advantage of such technologies is that they can transport data from the Web server without having to reload the entire page, making websites function more like desktop applications and creating a highly interactive user experience.

**Figure 1.** Innovative technologies can create unforeseen challenges if not properly tested



**Figure 2.** Scripting for Web 2.0 applications such as Ajax can be extremely complicated

But, as always, with power comes complexity, and innovative technologies can create unforeseen challenges if not properly tested. In Web 2.0 applications, the clients become “thicker,” richer in features and functionality, and more resource intensive. Similarly, features like prefetching—an interactive way of returning a query with every keystroke—can significantly increase the roundtrip traffic to the database and greatly impact performance. This can potentially create a considerable amount of performance issues, violate SLAs, and negatively impact the end-user experience.

## Adjusting for modern methods

Rich Internet applications allow for dynamic, asynchronous data transfer, using multiple protocols and a variety of servers. They gather data from distributed, heterogeneous sources, including cloud-based and external data storage options. Thick clients with widgets and client-side functionality often have server-side components, which may need additional processing before the server sends the data back to the client. Developers who build these widgets—often adding them from available toolkits—do it on their development machines and don’t realize that once separated across the network, the server component may cause latency and affect the overall system performance.

New technologies such as Ajax enable prefetching, where every new letter that a user enters into a search engine suggests a new set of results that are dynamically delivered from the server. All this activity generates a lot of network traffic and can significantly impact performance. Network latency and bandwidth constraints can also create performance bottlenecks. To accurately predict the performance of an application, it is necessary to test individual components and services, but equally critical are server monitoring and end-to-end performance testing, along with accurate WAN emulation.

Testing Web 2.0 applications presents its own set of challenges. The complexity of new technologies, the lack of commonly recognized and accepted standards, and the sheer multitude of emerging frameworks and toolkits make it difficult for companies to build Web 2.0 testing strategies and select appropriate automation solutions. Traditional testing tools focus on protocol-level verification, offering no framework-level support or ability to accurately recognize objects in these new, rich clients, making it virtually impossible to effectively validate the performance of Web 2.0 applications. Script creation, which has always been a lengthy, time-consuming process that requires domain and application expertise, becomes even more complex in Web 2.0 applications.

## Out with the old: The trouble with traditional testing tools

The emergence of new Web 2.0 technologies has also transformed the world of testing. In the early Web days, testing tools were based on protocol-level recording. They recorded the http requests from the browser to the server and back. Dynamic values that the server sent back, such as session IDs, had to be manually correlated. As applications became more complex, so did the scripting. Correlations started to require advanced scripting and application expertise, and IT scripting became a complex and time-consuming process.

QA organizations then started shifting to UI-level recording, which focuses on verifying specific objects in a browser. Testing tools didn't need to work on the lower transport level layer; they could instead focus on the objects in the DOM. However, Ajax introduced a new set of complexities: client-side processing and asynchronous communication. The UI-level testing tools that focused on the DOM also no longer worked. Record and replay tools needed to add JavaScript-rendering agents on top of the DOM to be able to support the multitude of different toolkits used to build Web 2.0 applications.

This approach in turn presented its own insurmountable challenge: with new toolkits becoming available every month and old toolkits being constantly updated and revamped, no vendor could keep up and provide a reasonable level of support for the new functionality. Additionally, conventional GUI automation tools were simply too "heavy" and could typically automate only a single user session per operating system session. A successful performance testing solution needs to have a concurrent multiuser, multisession driver, simultaneously automating multiple sessions.

## **An up-to-date solution for testing Web 2.0 apps...Web 2.0: HP TruClient patented technology**

HP has been in the forefront of performance testing. With the growing popularity of Web 2.0 technologies, HP set out to find a toolkit-agnostic solution that would support all modern applications, as well as make scripting a faster and easier process. The result was a completely revolutionary, patented solution that works with the events from the end-user's interactions with the Web application.

HP TruClient, which works with HP LoadRunner as well as HP Performance Center, is an innovative, browser-based virtual user generator (VUGen) that supports simple Web as well as modern JavaScript-based applications. The scripting engine is embedded within the browser, and behaves like a true browser client. It utilizes a unique, patented approach to object recognition, which makes it an extremely flexible and extensible solution for testing Web 2.0 applications.

It provides interactive recording and scripting, which makes scripting faster. It significantly reduces script maintenance. UI-level recording removes the need for programming and correlations, enabling even the nontechnical business users to participate in the testing process. HP TruClient is completely agnostic to any specific framework or toolkit being used in the application and supports nearly all available Ajax toolkits, helping make testing of Web and Web 2.0 applications faster, easier, and more comprehensive.

## **Benefits of HP TruClient**

Arguably the most time-consuming and labor-intensive process in performance testing is developing the test scripts. Although modern testing tools have significantly simplified the process of capturing the test script by recording the user's interaction with an application, the fundamental problems with scripting remain the same:

- Traditional scripting tools require a sequential approach to scripting: first, the basic steps are recorded, creating a "shell" script. This script is then taken offline, and undergoes further manual steps such as data parameterization and correlations, followed by additional steps of adding transactions and any other required logic in order to make the scripts perform as desired.
- To create a well-designed, reusable, working performance testing script, a tester requires a certain amount of expertise and programming knowledge. This is especially true for complex, interconnected applications with new technologies that often support critical business transactions.
- Scripts created at the protocol level are vulnerable to the slightest application changes. If changes occur to the application, often the scripts have to be discarded and redone.
- Scripting for Web 2.0 applications has a unique set of challenges due to the complexity of applications, client-side logic, asynchronous communication, and multiple, ever-evolving frameworks and toolkits.

**1. HP TruClient simplifies scripting for Web applications**

By moving the scripting engine into the browser in an updated and greatly simplified form, the HP TruClient technology completely revolutionizes the scripting process. The UI-level recording, based on the browser’s own DOM, makes it easier for nontechnical users to participate in script creation. The user simply interacts with an application, and the powerful HP TruClient engine captures every step in the form of a script. No correlation, coding, or programming skills are required for capturing and replaying the HP TruClient script. For power users who are familiar with C, JavaScript, and other programming languages, HP TruClient also offers full access to work with the script and enhance it on the code level. By combining the ease of use and the breadth of code-level functionality, HP TruClient becomes a versatile tool for both business users and developers alike. HP TruClient, available in HP LoadRunner and HP Performance Center, enables novice and nontechnical testers to create performance testing scripts, without requiring technical knowledge or in-depth understanding of the scripting process. Now scripting can be done faster and easier, saving companies time and freeing technical resources to focus on more advanced tasks.

**2. HP TruClient accelerates scripting and shortens test cycles**

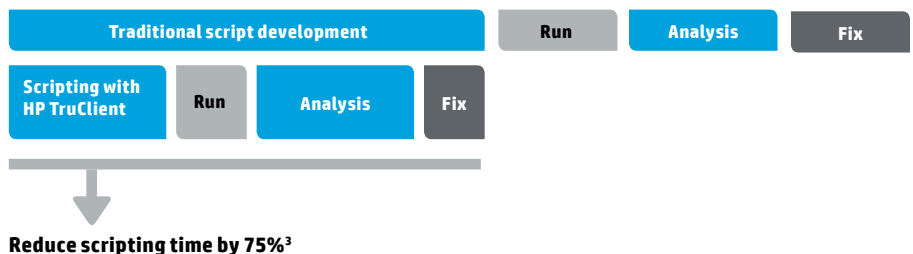
The HP TruClient technology makes scripting faster, reducing scripting time by as much as 75 percent,<sup>1</sup> by offering dynamic, interactive functionality that lets users customize scripts during the recording. By removing the need to perform multiple sequential steps and time-consuming functions such as correlations, and by combining the recording and editing process into one, the HP TruClient technology helps significantly reduce test cycle time.

This innovation is especially welcome in organizations that use Agile methods for delivering applications faster and with higher quality. By shortening the script creation time, HP TruClient allows more load tests to be performed earlier in the development process, identifying problems earlier in the lifecycle where they are easier and cheaper to fix. HP TruClient scripts are also more robust and intuitive, making them easier to maintain as the application changes. This also leads to greater reuse of the testing scripts, increasing the ROI on automated load testing and improving application reliability and stability. By reducing the amount of time required for test script creation, HP TruClient helps accelerate applications’ time to market. In today’s highly competitive business environment, this can easily translate into a significant competitive advantage, increased revenues, and improved customer loyalty.

**3. HP Ajax TruClient leverages the HP TruClient technology and supports all Ajax applications**

HP Ajax TruClient can precisely and efficiently test all types of Ajax applications. It easily handles a variety of Ajax objects such as slider bars. By looking at the “true” DOM, HP TruClient automatically inherits compatibility with all supported toolkits in the browser. Rather than going through a DOM or a DOM emulator, the new HP TruClient is based on an event-level record and replay. Instead of recording the state changes to the objects in the DOM, HP TruClient looks at the event handlers that are causing the JavaScript to render in particular ways, and as a result, HP TruClient can achieve a very high level of Ajax object recognition. Ajax applications represent business-critical, end-user-facing functionality. Until now, it has been practically impossible to effectively and efficiently test these applications. The innovative technology of HP Ajax TruClient makes Ajax testing simpler, faster, and more reliable than ever before.

**Figure 3.** HP TruClient simplifies scripting for Web 2.0 applications and reduces scripting time by as much as 75%<sup>2</sup>



<sup>1,2,3</sup>Based on data from customer surveys.

**Figure 4.** HP TruClient is the revolutionary new technology for testing modern applications



## Inside HP TruClient technology

The recording mechanism in HP TruClient technology is fully interactive. When users interact with an application, they can see the script being developed on the screen. Using a unique, patented approach to object recognition, the script automatically recognizes and inserts the names of widgets in the application. The HP TruClient recording engine can automatically suggest alternative automation steps, saving you time on script debugging and troubleshooting.

To enable the script to more accurately represent the production environment, you can enhance it with parameters, replacing static data values with variables.

To enable acceptable performance levels, you can add a transaction that measures the amount of time it takes to perform a certain function. To check the accuracy of the transaction, you can add a verification step. In HP TruClient, you can simply select the object and define the value that you need to verify. For example, you may want to validate that the amount of money that's been added to the account matches the number that you asked to transfer.

You can add synchronization functions to match the execution of the HP TruClient script with the output from your application. Synchronization points allow the script to pause in the replay while waiting for an object, a window, a dialog, or another type of control to appear based on a specified condition. By adding a JavaScript synchronization code logic, you can accurately capture the response times based on asynchronous events.

Object identification is a major component of the HP TruClient technology. Various functions are available to enable correct object identification, such as the ability for the client to relearn the objects you are using in the step, or the fact that you can “teach” the system about the objects that are related to the one that you are using in the step. You can also use manual options for object identification, such as XPath and JavaScript code.

Once the script has been recorded, the user can replay it to see if there are any errors. During replay, you can see the recorded actions being replayed in the browser. If an error is discovered, the user can highlight the object that failed, and HP TruClient automatically recognizes the selected step. Replay logs are also available through the VUGen.

The HP TruClient technology works with HP LoadRunner and HP Performance Center—the enterprise-level performance testing solution—as well as with HP Application Performance Management (APM), for creating synthetic transactions to monitor applications in production.

## Answering the Web 2.0 challenge

With more companies moving toward modernizing their applications and adding new Web 2.0 functionality to their core business systems, the HP Software engineering team has taken on the challenge of completely rethinking the way events and objects are being recorded. The result is the HP TruClient technology—an innovative, browser-based virtual user generator that supports simple Web applications as well as modern JavaScript-based applications, including Ajax. Using a unique approach to event-level recording, HP TruClient is able to support nearly all available Ajax toolkits, helping make the process of performance testing Web 2.0 applications more reliable, efficient, and scalable.

### Support for mobile applications

HP LoadRunner is the most complete solution for mobile application performance testing for both browser-based, as well as native applications. HP TruClient extends as HP Mobile TruClient to support testing for browser-based mobile applications. A mobile applications protocol provides support for native applications. HP solutions for mobile testing can be used to test mobile applications against any platform and OS.

HP LoadRunner seamlessly integrates with the Shunra Network Virtualization solution, enabling WAN simulation conditions.

To learn more about testing mobile applications visit [hp.com/go/mobiletesting](http://hp.com/go/mobiletesting).

HP TruClient makes performance test scripting faster and easier. It helps shorten the test cycles by integrating the script recording and editing processes. Testers can customize tests as they record their interactions with an application, removing the need to return to the same script multiple times to add parameters, data, and correlation. This streamlined scripting process helps save valuable testing time and shortens the release cycles, enabling organizations to gain a competitive advantage by bringing applications to market faster and with better quality. Faster, easier scripting also makes HP TruClient scripts truly efficient in organizations that use Agile methods. Faster, more accurate performance tests earlier in the lifecycle mean fewer problems downstream.

The intuitive, interactive nature of HP TruClient technology and its complete lack of correlation and programming enable even novice testers to create working load testing scripts. At the same time, HP TruClient provides full access to “behind-the-scenes” technology for power users with advanced scripting requirements. Involving nontechnical resources into the scripting process translates into significant savings for organizations that are now able to use their highly skilled programmers for more advanced and technically demanding tasks.

**Learn more at**  
[hp.com/go/performancevalidation](http://hp.com/go/performancevalidation)

**Sign up for updates**  
[hp.com/go/getupdated](http://hp.com/go/getupdated)



Share with colleagues



Rate this document

© Copyright 2010, 2012–2013 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Adobe is a trademark of Adobe Systems Incorporated. Microsoft is a U.S. registered trademark of Microsoft Corporation. Java is a registered trademark of Oracle and/or its affiliates.

