



Living Up to the Web 2.0 Load-Testing Challenge

A wave of delay would sweep through the application, then dissipate.

Sean Molloy and his colleagues were struggling to make sense of this troubling trait. They were performance-testing part of the compliance management suite offered by ControlPath, where Molloy is director of software engineering.

Such buggy behavior would have been no big deal, except for two facts. First, even after exhaustive testing, Molloy's engineers had never seen an exception quite like it. Second, the hiccup was occurring in front of a team from a potential suitor.

"The tech guys across the table wanted to see a live performance-test 'off the cuff' [of] a test that we could perform in a pinch," says Molloy, a 15-year software industry veteran. "We were on edge."

ControlPath's application is Web-based and makes heavy use of AJAX. And so, at least broadly speaking, it can be tagged with the Web 2.0 label. Yes, we all agree that Web 2.0 is already well worn and impossibly over-hyped. But anyone who roots around the Web for even a few hours each day will invariably see several examples of browser-as-platform, Web site-as-application approaches to computing.

It's not at all hard to find analysts to provide context for these anecdotal observations. In an Oct. 3, 2006, report, Evans Data said that 1.7 million developers are already using AJAX while some three million more are evaluating the methodology. A Gartner research publication released the same month declared there to be an 80 percent probability that "by 2008, the Web 2.0 vision will be adopted as the mainstream Web and will disappear as a separate category."



Geoff Koch

The idea of Web load-testing itself is undergoing something of a shift. For now, the term still evokes the image of running tests to make sure that a hot new application, cobbled together in the wee hours of the morning by a would-be entrepreneur, can at least theoretically scale to support a worldwide audience.

Scaling to support the masses remains important, a point made vividly by Facebook's chief operating officer Owen Van Natta in a May article in *Fast Company* magazine. Here's an excerpt of the article, written by reporter Ellen McGirt:

"We were trying to predict how many new users we'd get, how they would use the site, and what we'd need to serve that," [Van Natta] says. There weren't enough people to do all the analysis. "We were just trying to keep the wheels on the wagon." When he went to check the data center, he was horrified. "There were little fans like this big" holding up his hands to indicate the size of a grapefruit "tucked between the servers. It was over 110 degrees in some aisles." And the data-center guys were plugging in more servers and screwing them into racks, trying to keep up with the rapidly scaling site. The Plexiglas sides of the server racks were warping from the heat. "I was, like, Mayday!" he recalls. "We need to get on top of this!"

In reality, far fewer sites will ever deal with explosive, Facebook-type growth than will start to experiment with AJAX, Adobe Flex, RSS feeds or blogs. For these and other technologies and features, it's less about user clicks and page refreshes, and more about an explosion of browser-to-server HTTP calls made in the background, of which few but the savviest users are aware. Subtle changes

in such traffic can swamp servers and networks, which is why it's increasingly important for vendors of hosted applications to be able to troubleshoot load issues at runtime.

In other words, it was perfectly reasonable for Molloy and his engineers to suffer through their own Mayday! the moment, however minor.

"You can't go running to the code to performance-tune an already deployed service," acknowledges Molloy, whose developers use a load-testing tool provided by France-based Neotys. "If you find that there are aspects of the application that can affect performance and can be changed like block sizes on streams, SQL query text or numeric values, these should be abstracted to configurations so that server administrators can work to squeeze the max performance out of the app on the running hardware."

Web 2.0 is bringing other changes to load testing, which until recently was the specialty of a select few testers and developers, most of whom were usually kept far away from actual customers or users. Now, however, everything is loosely coupled and asynchronous, from applications and services to workflows and e-commerce transactions. This means everyone from customer service reps to coders has to pitch in when it comes to understanding performance.

Improvements On Board

Molloy says that this all-hands-on-deck approach is made easier by improvements in Web load-testing tools during the last several years. Today, many load-testing products automatically create complex usage patterns and scenarios.

At Centric Software in New Hampshire, for instance, QA engineers use a load-testing tool not only to gauge

Best Practices columnist Geoff Koch can be reached at koch.geoff@gmail.com.



Best Practices

performance when a new implementation is complete, but also to reproduce and better understand specific customer use cases.

“The reports and results we receive at the end of each test yield precious information such as time to run every request, as well as what type of error resulted from the server,” says Jean-Paul Durand, the QA manager at Centric, which also uses Neotys tools.

Features like this, increasingly available in even moderately priced load-test tools, address one of the more inalienable truths in all of software, best summarized by Molloy: “The human user base is often not 100 percent predictable,” he says.

Of course, architects, developers and testers are human, too. So all the nuanced load and performance information provided by the newest tools occasionally gets ignored.

“From a personal point of view, load test analysis and software support is as important as load testing itself,” says Porter,

whose firm uses Borland’s SilkPerformer. “If your Web site is an important part of your company’s revenue stream, then load testing and analysis should be a part of your deployment strategy.”

Don’t Forget the Database

It’s not enough to stop analyzing once you reach the edge of your Web-based application, especially if it’ll be regularly querying a database. Indeed, as the compute model evolves away from client/server, writing code that can scale well with massive amounts of data may be equally important as scaling to support massive numbers of users.

Not that this is easy. Molloy points out that understanding database loads means getting a handle on both the queries and the execution plans.

“It’s more of a style choice than a best practice, but we believe that the data from the database should be sorted, paginated and have only the columns required before it hits the application’s processing,” he says. “This

helps ensure that, even if the data stored is immense, the network need only scale to handle the data required by end users.”

There’s no word on the potential suitor’s identity or whether its representatives ever made a proposal, indecent or otherwise, to ControlPath. But Molloy is happy to report that his team did figure out that wave-of-delay problem in time to salvage the demo.

“Since the ‘wave’ pattern is often associated in Java with garbage collection or memory usage, we looked at the memory paging on the three machines being tested,” Molloy recalls. “Sure enough, the Tomcat configuration had the memory set to what it is on a development laptop, not a production server. One change to a config file, and the load test line resolved to a nice, flat response time given static load.”

Problem solved, for now. But expect waves of new load-test issues as Web 2.0 rolls toward the mainstream in the months and years ahead. ☒

Index to Advertisers

Software Test & Performance

Advertiser	URL	Page Number
EclipseWorld	www.eclipseworld.net	30-31
Empirix	www.empirix.com/freedom	6
Hewlett-Packard	www.hp.com/go/software	40
IBM	www.ibm.com/takebackcontrol/open	20-21
iTKO	www.itko.com/lisa	8
PNSQC	www.pnsqc.org	39
Seapine Software	www.seapine.com/stptcm	4
Software Test & Performance Conference	www.stpcon.com	2-3
Test & QA Report	www.stpmag.com/tqa	25