

Software Test & Performance Collaborative

Community, Resources & Knowledge Sharing for Test & QA Professionals

Article: CapCal's Randy Hayes Sees Clouds in the Forecast

By [Andrew Muns](#) on Oct 1st, 2009 | In [Interviews](#), [Performance Testing](#) | [Share](#) |

Randy Hayes is co-founder and CEO of Capacity Calibration, a software company that specializes in cloud-based Web load testing.

Incorporated in 2008, Capacity Calibration develops and markets test automation solutions for medium and large companies. Its flagship product is CapCal, a Web performance testing tool intended to help reduce the effort of script creation and maintenance. CapCal is designed for developers, QA and IT teams and integrates with functional testing tools.



Regular readers of Software Test & Performance might recognize the name of Randy's sister Linda Hayes, who's a frequent contributor to the magazine. Their career paths have crossed on multiple occasions.

Andrew Muns: Let's start by talking about your background. How did you get involved in the software testing industry?

Randy Hayes: Well, I am from Albuquerque, New Mexico, and during the 1970's there were some exciting things happening there.

One of them was a little company called MITS [Micro Instrumentation and Telemetry Systems] which had a computer called the Altair and a pair of guys named Bill Gates and Paul Allen were living there at the time and working with this company.

It was shortly after they left and moved to Seattle that I got a job at MITS writing diagnostic test programs. That was my beginning in testing.

I began as an entrepreneur in 1985 when my sister [Worksoft CTO] Linda [Hayes] and I started a company called Autotester, which introduced the first automated testing tool for the PC.

At the time, I assume there were automated testing tools for mainframe environments?

Yes, there were, but they were all script-based, nothing like what we had. We used a tape recorder analogy so you record “tapes” instead of programming.

People without any programming experience could build them all day long, it was great. But we raised a lot of venture capital to fund that company and they had the majority interest. At some point we had a falling out with them so we chose to move on. In the subsequent years it finally tanked, I’m sorry to say.

So you learned about venture capital the hard way I guess.

Yes indeed, although it doesn’t always go that way. After I left Autotester, I joined Linda and a couple of others to do a project for Fidelity Investments in Boston that evolved into a company called Worksoft, which is now focused in the SAP market and doing quite well.

Finally, CapCal was started [as Distributed Computing] in March of 2000, when I raised a couple of million dollars, hired a team of really good developers and spent almost two years creating the first version of it.

What was it that led you to the business idea behind CapCal?

I was hired as a consultant by a dotcom company to evaluate load testing tools and tell them which one they should buy. I evaluated all of them and came back to say I wouldn’t buy any of them. That was a bad idea from a consulting point of view.

What was lacking in the tools that you evaluated at the time that led you to believe that you could do a better job?

Well, the Internet. Period. These tools were all designed for client-server. They were antiques. CapCal was designed from the ground up for the Internet, to be distributed and service-oriented. And in fact, we started it with a program that people could download, and when their computers were used in a test, we would pay them.

What has been the evolution of the company?

When we talk about performance testing at CapCal, we are really talking about three different offerings. One of them is what we call CloudBurst. This is a service on the Amazon cloud that offers on-demand, self-service, pay-as-you-go load testing.

The second are integrated performance testing services in which we capture what functional test tools do and use these inputs for performance testing.

Lastly, we have an agile testing tool that conducts nightly performance tests to allow testers to compare their application’s performance after each iteration in the development lab. But CloudBurst is the way that the majority of the people will purchase and use the product.

What has been the biggest growth driver so far?

The most significant drivers have been small- to medium-sized businesses that need these services but have been priced out of the market. There are many companies that will pay for

performance testing, but it has to be in their budget and the ability to use the Amazon cloud puts this in the price range of smaller shops by allowing pay-as-you-go pricing.

How are your customers able to simulate various types of users interacting in different ways at different times?

They can record different profiles, or sessions, and those can be mixed and matched so you can construct a test that has a variety of different behaviors.

How do you think about the competitive landscape of this business and how is it changing?

There has been an onslaught of entrants to the space and there will be more coming, because it is such a natural use of cloud computing. What we want to do is to bring together all forms of test automation and make them available as a service on the cloud. So if someone wants to run their functional tests in parallel, we can do that. Nobody is quite there yet, but I think that if you can deliver all of it as a service on the cloud people will eventually rush for it. It is so easy...and a lot cheaper.

For load testing, the advantage of using of the cloud seems obvious, but with regard to functional testing, what is the primary benefit?

Well, let's say that you have a functional regression test and let's say it runs for four hours. If you could run it in 10 minutes, then you just shaved off a lot of time from your development cycle.

Do you feel like the popularity of agile methods makes a lot of these tools even more important as teams work on the basis of short iterations?

Definitely. Performance and scalability testing need to be part of that cycle. They aren't in most cases, but they need to be. We have a way to do that and we are hoping that people will get on board with this. I think it will happen when the tools become easy and affordable enough.

Do you think the efficiency of the agile methodology itself has increased as a result of some of the just-in-time tools that now exist? Is technology an enabler of agile just like agile is to technology?

Oh wow, that is a good question. The way I think of it is that test automation in the beginning meant getting rid of all this mind numbing repetitive work. We were replacing the person and doing it much, much faster. But now, with cloud computing, we don't just replace the person, we get rid of the labs that don't need to be taking up space because they are just doing things like testing. We think that people will gravitate quickly to the cloud for test automation of all kinds.

When people do Web load testing with CapCal, do applications typically outperform or underperform expectations?

They are typically surprised that it was not as good as they expected.

Do you guys load test your own site with your own product?

Yes, we do and it is yielding some very interesting results. We have come up with a way to deploy CapCal on the cloud and scale automatically when it needs to. We only have one instance running most of the time but when it starts to get weighed down at all, it spins up another one, and so on.

Looking ahead, what do you think are the next big shift will be that testers will have to adapt to?

As people start to have test tools that are as sophisticated as the apps they are testing and as easy to use, very large numbers of people who don't currently use tools will start to do so, especially as these become more affordable.

This in itself is a sea change, so I am not worried about what happens after that, because it is going to take a while for testers to catch up with tools already being introduced.

Where does Web load testing stand on the spectrum from being an essential expense to a luxury expense?

Determining the value of performance testing is just a matter of thinking what would it cost per hour of downtime. Customers should ask what this cost is relative to the cost of the tool, and the higher that number is, the more negligent you are being if you don't do performance testing. So I encourage people to do this math.

About Authors



Andrew Muns

Andrew is CEO & Founder of Redwood Collaborative Media and the Publisher of Software Test & Performance Magazine.