Elastic Calibration

- Cloud Application Migration Service

- Get the empirical data you need before you launch applications in the cloud

CapCal
CAPACITY CALIBRATION

A Service Offering Of

Grid Robotics

CapCal Web Performance Testing is a new paradigm in cloud testing that leverages today's distributed, service-enabled technologies to simplify and accelerate performance testing and monitoring of web and mobile web applications.

Elastic Calibration is a service offering from Grid Robotics who are experts in cloud computing and the difficulties of moving applications to a cloud based environment.

We partner with the leading cloud services companies as well as companies who specialize in scaling, cloud application monitoring and optimization.

You have identified the applications that would benefit from running in the cloud. But how do make the final decision to move your application to the cloud? How many servers will you need and how much will they cost? Grid Robotics can help. We have worked with many different cloud ser-vices providers and with ISVs who specialize in cloud application deployment. The missing component that is key to decision making though is always "does all this new technology work together effectively, efficiently and meet the technical and economic objectives set by upper management

Benefits:

- Thoroughly test your application in both on-premise and cloud configurations with CapCal

- Have the security and assurance of knowing that the cloud does deliver on the promise of elastic, on-demand resources without sacrificing performance or security for your applications and business

- Demonstrable reduction in cost/time/effort to deploy your target applications

Calculate your ROI based on empirical data and not guesstimates.

Delivery Methodology

Benchmarking Step 1 – Capturing a Baseline on Bare Metal

Take the mission-critical transaction in your application (e.g. "place order") and create an end-to-end recording of it. Then run a load test using XXX users against an in-house test environment (QA, staging, etc). Bear in mind that this is not done to stress the system, just to get a good statistical cross-sampling of user response times from the cloud using local "bare metal" of a known configuration.

This baseline configuration will be the one duplicated in the cloud to serve as a foundation, first of all for building out a staging environment for testing the scalability of the application. A staging environment that mirrors production, whether production is hosted on bare metal or in the cloud, is necessary in any case. Fortunately this same staging environment can be instantly duplicated or promoted to production if you're moving <u>to</u> the cloud!
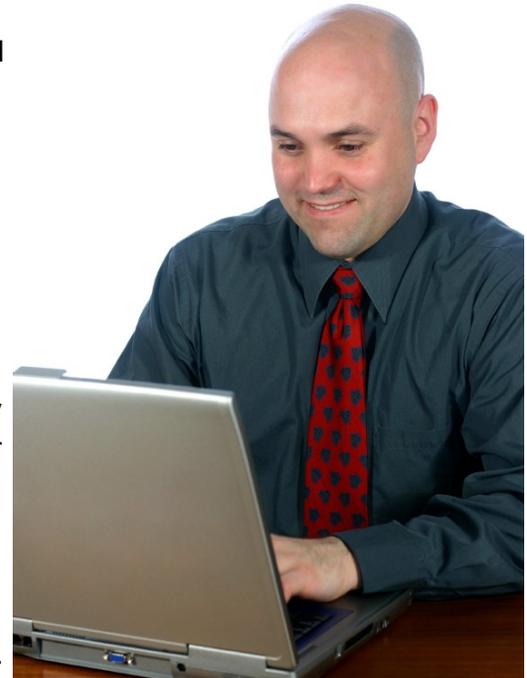
Benchmarking Step 2 – Comparing Bare Metal to the Cloud

After the baseline configuration has been duplicated in the cloud (see "migration strategies to pick from") run the same test and compare the results. Based on how well you've matched your actual hardware to your virtual hardware, you should expect to see much better results, not because virtual hardware is faster than real hardware but because your lab or data center is not likely to have the bandwidth that the cloud provider does. In any case, you now have a way to compare "apples to apples" – now on to the big apple orchard in the sky!

Optimizing and Auto-Scaling

Now that you have your baseline you can move to the really fun part – optimizing and scaling! One of the most basic reasons for moving to the cloud is the "elasticity" of it – capacity on demand provides both an economical <u>and</u> a technical justification of the cloud to begin with! You can now choose which virtual machine sizes, storage options, pricing models and so forth to fit your needs <u>and</u> be able to verify it with empirical data.

Auto-scaling means adding or subtracting capacity based on demand, and the baseline test we used for benchmarking will also be used to set the "trigger" parameters for auto-scaling. This is much easier and far more effective than guessing at what those parameters should be – it's rarely <u>just</u> the CPU or any other measurable factor but a combination and the CapCal benchmark helps you to "fill in the blanks" and then run a test to verify it.