Public Cloud Performance – Solving the Problem of Inconsistency-as-a-Service

I don't often dig into the specific work I do at Turbonomic, but there are some times when I really want to share the powerful stuff that I've been able to be a part of. Some recent stuff came up which really hit home. As I'm at the start of my first AWS re:Invent in Las Vegas, the topic of cloud performance has bubbled up to the surface quite a bit.

The question around your consumption of cloud resources has many aspects to how you make the decision on what best belongs in the cloud, but how is it that we are tackling performance? This is my quick story on that.

In public cloud, the same performance test will have different results when you run it more than once. It's not consistent by design. That alone should be a driver for many of the traditional data center architects to look carefully as you spread your wings into the public cloud realm.

The fundamental economics of the public cloud are rooted in the idea of sharing infrastructure and delivering capacity, not performance. Performance is secondary. The reason that capacity is the first target is that the public cloud promise of elastic capacity of pooled resources provides a way for us to consume what we need, when we need it.

All of the more economic ways to leverage elasticity within the cloud lean towards finding resources to add in interesting ways such as the spot instances with AWS, and scalable options through various means. This leads to the real thing that is being sold to you by public cloud providers. It also doesn't do what you really think that it does.

Performance at a Price

There are two fundamental ways that the public cloud wants you to deal with performance, and both are costly. These are through over-provisioning resources (increased flavor sizes, or more nodes in a horizontally scalable application) or through buying provisioned capacity which provides a sort of guarantee that you have a minimum level of access to resources.

Either of these methods comes at a price, quite literally. Choose wisely as you are going to be selecting these experimentally. If you elect to tackle the performance challenge using more capacity, you really aren't tackling performance. You're simply putting more money into the pile which hasn't solved the issue. This has just hidden the issue, in an expensive way. That also requires that you're architected to take advantage of horizontally scalable infrastructure, or else you're really about to overspend on capacity in vertical sizing, again, using over-provisioning as a failed way to attempt to provide performance.

The other key factor that you're dealing with here is that performance must be measured. None of the public cloud providers provide the in-guest data that you will need to be able to assess the real performance. This requires the use of other systems in order to know that the actual performance is at any given time, and over time to show the real results of the capacity-based way that the cloud providers are handing you as your supposed performance solution.

The (Performance) Struggle is Real

GitLab found out about the struggles of performance inconsistency, and shared their story. This is a must read IMHO. I'm not going to tell you that this signals everyone to run back to the data center as the only solution. As a huge proponent of private cloud, and of embracing the best of breed solution for every part of your IT portfolio, the public cloud has an extremely powerful way of solving specific challenges that we have within the business of IT.

If you listen to many of the pundits, they are abstracting the infrastructure further with containers, PaaS, and other product alternatives which in and of themselves solve tooling issues, and not performance issues. If anything, they decrease performance even further. This is where we have to change our approach as an industry.

How to Solve Performance in the Public Cloud

The GitLab story still misses out on the happy ending. The method of solving the performance challenge is actually just furthering the long-standing methods of monitoring performance to find out in hindsight when issues have occurred. This is the core of what my team is doing at Turbonomic. There really is a better way. The approach is all about an autonomic way of assuring application performance, in real-time, while utilizing your data center and cloud infrastructure as efficiently as possible.

By understanding the real-time performance of the applications, as well as every layer of the virtual architecture from traditional data center products to the public cloud, we have the ability to control the supply of infrastructure to meet the demand of your applications.

Scale up, or out? Burst to cloud? Migrate to cloud? These are questions that cannot be answered using tribal knowledge and peeking through reams of performance data which only tells you how things performed in the past. Haven't you heard the phrase "past performance is not indicative of future results" when you read a financial prospectus? The same holds true for application performance.

Imagine being able to assure performance within a budget on the public cloud. Well, imagine no longer, because it's available for you today. Welcome to the autonomic future, Powered by Turbonomic. Coming up next week is a more full view of what I'm talking about.

Hopefully you'll be as excited as I am about what we are doing