NEW CLOUDS, SAME CHALLENGES

PUBLIC CLOUD GUIDE

VMTURBO

Cloud Déjà vu?

There remain three key things to consider as you transform IT: Performance, Cost, and Agility.

When Amazon Web Services officially launched in 2006 it was fulfilling a growing need for self-service computing. The dot-com era had come and gone. A digital economy was growing out of the ashes. Today, a decade later, applications have become the lifeblood of the enterprise. With a mandate to build more and build faster, on-demand cloud computing has been a godsend to developers. Meanwhile, Infrastructure and IT operations teams have struggled to compete against infrastructure-as-a-service (IaaS) providers, often left dealing with the financial and regulatory consequences of this "Shadow IT."

A growing number of IT leaders are transforming IT's unfortunate reputation as a cost-center into one of a competitive advantage. In doing so they acknowledged that IT cannot be a gate-keeper. Their teams must enable applications teams to deliver on business needs—or they'll be replaced.

With so many options—on-premises private clouds, off-premises public clouds, hybrid cloud bursting, and the mix-and-match approach of "multi-cloud"—deciding how best to move forward can be overwhelming. The crowding landscape of public cloud providers only exacerbates things. New players and new offerings constantly enter the scene, but there remain three key things to consider as you transform IT: Performance, Cost, and Agility.

This white paper is for leaders of Operations, Engineering, or Infrastructure teams who are creating or executing an IT roadmap. It discusses the challenges that must be recognized and questions that must be considered in order to succeed in this new cloud era.

Journey to the Cloud, Any Cloud

Not unlike the decisions that had to be made for on-premises data centers—developing a cloud strategy comes down to Performance, Cost, and Agility. AWS, Google, Azure, IBM Softlayer, and a long list of others promise unlimited cloud resources. But they don’t come free and they still need to be managed.

In any cloud, application performance comes down to a few key questions.

Performance: It's Still Your Problem

IaaS providers deliver the building blocks. Your teams must decide which blocks they need to meet business demands. In this context, application performance comes down to a few key questions:

- Which instances?
- How should I size them?
- Where should I place workloads across instances?
- How should I size the workloads?
- What kind of storage (and how much) do I need to attach to them?

Build

AWS's guidance on selecting the best instance type, sizing, and storage rests largely on your ability to predict resource needs. Answering those questions requires understanding the real-time application demand that will be put on those instances. Underestimate the size of instances your applications need and you have performance issues. Over-provision and

you're wasting budget. Sound familiar? These are the same challenges that exist in on-premises data centers.

AWS's guidance on selecting the best instance type, sizing, and storage rests largely on your ability to predict resource needs.

Run

Once you've selected, sized, and customized—the real work starts: making sure the applications get the resources they need to perform. The AWS Well-Architected Framework outlines best practices for ensuring that instances are performing as expected:

- **Amazon CloudWatch monitoring:** Use CloudWatch to monitor instances.
- **Third-party monitoring:** Use third-party tools to monitor systems.
- **Periodic review:** Periodically review your monitoring dashboards.
- **Alarm-based notifications:** Receive an automatic alert from your monitoring systems if metrics are out of safe bounds.
- **Trigger-based actions:** Alarms cause automated actions to remediate or escalate issues.

Google Cloud Platform and Microsoft Azure offer similar performance best practices: monitoring metrics, setting thresholds and responding to alerts. Public clouds promise many things, but guaranteed application performance is not one of them.

Ask yourself

How is my team managing performance? How much time is spent responding to alerts? Is my team able to scale with business needs in their current mode of operations? How will they guarantee performance in the cloud?

Cost: “Cloud Scale” Still Gets Expensive

Many have pushed for migration to the public cloud only to suffer debilitating bill shock. Operating VMs in the cloud is not cheap or simple.

It’s true, cloud providers can offer a lower cost per transaction or service thanks to economies of scale. Yet many have pushed for migration to the public cloud only to suffer debilitating bill shock. Operating VMs in the cloud is not cheap or simple. AWS offers a plethora of instance types, regions, and pricing options—and these offerings are continuously updated. See Figure 1 for a sampling. Additionally, Cloud Spectator tested 25 of the largest, most recognized public cloud providers with data centers in North America. Their findings are compiled in the Top 10 Cloud Vendor Benchmark and focus on the top ranked vendors based on combined performance and cost (i.e. overall value).

Figure 1 shows a sampling of prices across AWS instances and Figure 2 show prices across different cloud providers. Both provide some insight into how quickly costs can rise. How many VMs are you hosting in the cloud? How many do you plan to host there?

Figure 1: AWS on-demand instances pricing for Linux for the U.S. Northeast Region only. Source: AWS Pricing Page.

	vCPU	ECU	Memory	Instance Storage	Linux/ UNIX Usage	Price Increase from Previous Instance
			(GiB)	(GB)	($ / hour)	(%)
t2.nano	1	Variable	0.5	EBS Only	$0.00650	
t2.micro	1	Variable	1	EBS Only	$0.01300	100%
t2.small	1	Variable	2	EBS Only	$0.02600	100%
t2.medium	2	Variable	4	EBS Only	$0.05200	100%
t2.large	2	Variable	8	EBS Only	$0.10400	100%
m4.large	2	6.5	8	EBS Only	$0.12000	
m4.xlarge	4	13	16	EBS Only	$0.23900	99%
m4.2xlarge	8	26	32	EBS Only	$0.47900	100%
m4.4xlarge	16	53.5	64	EBS Only	$0.95800	100%
m4.10xlarge	40	124.5	160	EBS Only	$2.39400	150%
m3.medium	1	3	3.75	1 x 4 SSD	$0.06700	
m3.large	2	6.5	7.5	1 x 32 SSD	$0.13300	99%
m3.xlarge	4	13	15	2 x 40 SSD	$0.26600	100%
m3.2xlarge	8	26	30	2 x 80 SSD	$0.53200	100%

Figure 2: Monthly Cost of VMs Across CSPs. Source: Top 10 Cloud Vendor Benchmark

	Small	Medium	Price Increase from Small to Medium	Large	Price Increase from Medium to Large	Extra Large	Price Increase from Large to Extra Large
	($)	($)	(%)	($)	(%)	($)	(%)
1&1	$29.99	$49.99	67%	$129.99	160%	$349.99	169%
CentruyLink	$72.81	$138.01	90%	$260.82	89%	$536.84	106%
CloudSigma	$53.99	$107.22	99%	$207.16	93%	$433.06	109%
Google Compute	$68.10	$127.70	88%	$238.40	87%	$493.80	107%
Hostway	$152.80	$273.20	79%	$481.60	76%		
Interoute	$110.24	$212.48	93%	$408.96	92%		
Phoenix NAP	$85.00	$164.00	93%	$316.00	93%		
ProfitBricks	$45.76	$89.51	96%	$175.02	96%	$354.05	102%
Rackspace	$122.27	$219.54	80%	$388.35	77%	$826.70	113%
Ubiquity Hosting	$40.00	$80.00	100%	$160.00	100%		

Over-sizing instances is a costly decision. Consider the example of AWS pricing in Figure 1, where the price increase between sizes reaches 150%. In some cases, an increase of 169%—the price increase between a large instance and an extra-large instance at 1&1, see Figure 2.

Choosing Instances: The Multi-Million Dollar Question

The percentages are startling, but let's talk brass tacks. Let's say one of your Infrastructure guys spins up an AWS m4.10xlarge instance per the request of a demanding Application Owner (that's assuming he's even part of this process at all). He knows that the application probably doesn't need the larger instance; instead the AWS m4.4xlarge instance would be more appropriate. Unfortunately, your guy has no way of easily or definitively proving that, based on the application demand, the Application Owner would be just as satisfied with the smaller instance. How much does this decision cost you in a year?

> **m4.4xlarge cost for 1 year:** $0.95800/hour x 24hrs/day x 365 days/year = $8,392.08
> **m4.10xlarge cost for 1 year:** $2.39400/hour x 24hrs/day x 365 days/year = $20,971.44
>
> **Additional cost for larger instance for 1 year:** $20,971.44 - $8,392.08 = **$12,579.36**
>
> **Total cost of over-provisioning 100 instances for 1 year: $1,257,936**

This particular over-sizing decision multiplied across just 100 instances is over $1.2 million. How many instances are you running in a public cloud? How sure are you that you're using only the resources you need in the public cloud? How do your teams make these sizing decisions? How often do they simply comply with Application Owners to avoid conflict, knowing that the underlying infrastructure is over-provisioned?

The truth is that with any public cloud, it's not as simple as "pay for what you use." Instead, you pay for what you think you'll use. Consider the performance benefits and the cost savings, if your instances were appropriately sized based on real-time application demand? How much would your department save, if you could guarantee the performance of applications, while demonstrating to Application Owners that their applications don't require all the resources they think they require?

Additional costs to be considered:

- **Data transfer costs:** Cloud providers offer free ingress traffic, but there's a price for egress traffic. See AWS Pricing "Data Transfer" as an example.
- **Purchasing options:** For AWS EC2 alone, there are four options—
    - **On-demand**—no upfront costs, pay by the hour; but during periods of high demand your workloads may experience limitations in performance.
    - **Reserved**—heavily discounted hourly rate (compared to on-demand instances), but comes with a one-time fee and purchases cannot be changed later.
    - **Spot instances**—pay only when the hourly rate suits you, but make sure your apps don't have to be running all the time.
    - **Dedicated hosts**—specify physical EC2 servers to save money on licensing costs and/or meet regulatory requirements.
- **Support, monitoring, and other services** further increase costs.

The truth is that with any public cloud, it's not as simple as "pay for what you use." Instead, you pay for what you think you'll use.

For every cloud provider there are numerous decisions that must be made initially and continuously. Ultimately, sizing instances, data transfer—even determining the best AWS purchasing option—depends on application resource demands.

Ask yourself

How do your teams currently make these decisions? How much time do they spend looking at metrics in spreadsheets? Are your budgets and forecasts susceptible to human error? Will your current processes scale as the business grows? Will those processes need to change? How?

Agility: Easily Talked About, Not Easily Done

Often, what enables Developer agility challenges Infrastructure and Operations teams' agility.

IT continuously strives for "agility." But, it refers to different things in different contexts. When it comes to cloud services, there are two sides to agility:

- The agility it offers to Developers who can create applications for the business more quickly with self-service resources.
- The agility of Infrastructure and Operations teams to empower Developers, managing the use of those resources, while continuously maintaining performance.

There are tradeoffs on both ends. Often, what enables Developer agility challenges Infrastructure and Operations teams' agility. Elastic Beanstalk (EBS), for example, is the platform as a service (PaaS) offering from AWS. The promise to developers is that they can focus on application code, while Elastic Beanstalk handles every stage of deployment: from capacity provisioning, load balancing, auto-scaling to monitoring the health of the application. The caveat is that there are architectural specifications that must be implemented in order for EBS to work its magic. Considerations that—to varying degrees—lock developers and cloud architects into a specific cloud ecosystem are not unusual.

Ready for the Cloud—Which Cloud?

Commonly cited "best practices" for cloud-readiness include:

- **Think small**—applications should be designed as a set of loosely coupled services, a.k.a. (micro)service-oriented architecture.
- **Be stateless**—applications should get environment-specific information from the environment in order to seamlessly move between different environments or clouds.
- **"Live" in the moment**—applications should be designed to seamless start up and shut down, as scaling out is often the approach used to handle user load in the cloud.

These and other tenets are key to a "cloud first" development strategy, but every cloud is different. A best practice is to find out what design specifications a PaaS (or IaaS) provider recommends.

Moreover, EBS only supports a one-metric auto-scaling decision. Developers must decide which metric is most important to them. How do they make this decision? No application requires *just* CPU or *just* Memory or *just* Storage or *just* Network and so on—contention in any of those resources will cause performance degradation. The "most important metric" decision requires a fair amount of thought, but the bigger question is, why is that a decision that a Developer must make at all? Guaranteeing performance of applications requires an understanding of real-time application demand across all metrics. Decisions of whether to auto-scale out, up, or in should consider all the resources that applications need to perform as well as the impact of those decisions on the surrounding environment.

The cloud makes agility possible, but does not guarantee it. More importantly, without guaranteed performance agility is moot.

For Infrastructure and Operations teams using a public cloud offering to serve their developers, they must keep up with the dynamic demand that inevitably occurs as a result of easy self-service. If they don't, they risk performance issues—again, it's still your problem, it doesn't magically get taken care of in the cloud—and/or high costs from instance sprawl or over-provisioning.

In fact, Developers so value the instant gratification of public cloud providers that ease-of-use has become a differentiator for some cloud providers in their effort to compete against established leaders. But, easy self-service can quickly become untenable at scale, if not managed correctly. The cloud makes agility possible, but does not guarantee it. More importantly, without guaranteed performance agility is moot.

Ask yourself

How are your teams currently managing their cloud deployments? Will this approach scale as you offer developers on-demand resources? What happens when you have to support micro-service architectures with containers?

Choosing the Right Cloud

So you have a mandate to “go all-in on the cloud.” What’s driving that directive? For some it’s a sure way to cut hardware costs. For others, “the cloud” will streamline operational efficiency and speed up development. Meanwhile, a few here and there have simply resigned themselves to the inevitability of Shadow IT: if you can’t fight it, work with it. In most cases, there’s a mix of reasons. In all cases, as previously discussed, it’s not as easy or straightforward to “go to the cloud” as you think.

As you consider Performance, Cost, and Agility, you must also make decisions about where you host your workloads. Do you host them all in the public cloud or just some? Production, QA, or Dev? Mission-critical applications or non-mission-critical? Do you move those workloads between clouds? Why?

All-In on the Public Cloud: For younger organizations that have not yet invested in on-premises infrastructure, it may make sense to leap-frog private cloud computing investments and simply host all workloads in a public cloud. Your teams will still have the responsibility of guaranteeing the performance of those workloads while making sure cloud resources are being used efficiently, but you will not have to worry about the massive capital investment in hardware and the operational cost of building a private cloud.

Hybrid Cloud Elasticity: Established organizations that have been around for a while, have likely made investments in on-premises cloud infrastructure. For these companies, a public cloud offers a means to drive greater efficiency on-premises without risking performance. When demand spikes, workloads can be burst to the public cloud to ensure they get the resources they need. The question remains, however: Which workloads? Where? And when?

Multi-Cloud: For organizations not yet ready to embark on hybrid cloud bursting, hosting different workloads (or environments) across different clouds is still an option. In many cases, it's the only solution to servicing multiple business needs and requirements. Perhaps it makes sense to host Dev and QA environments in the public cloud, while Production stays on-premises—or, visa versa. Perhaps, due to compliance rules, applications and data for certain lines of business cannot go to the cloud.

No matter which cloud you choose, the responsibility of guaranteeing application performance, navigating costs, and driving agility still falls on Infrastructure and Operations Teams—not the cloud provider.

Ultimately, public- vs hybrid- vs multi-cloud decisions must be made within the context of business goals and realities. These choices reflect business policies or constraints, data security, and/or regulatory compliance. They cannot be a siloed IT activity.

Yet no matter which cloud you choose, the responsibility of guaranteeing application performance, navigating costs, and driving agility still falls on Infrastructure and Operations Teams—not the cloud provider. Are you prepared?

Conclusion

Public clouds are not a panacea. No matter what cloud or combination of clouds you select, your teams will have to make numerous, complicated, and on-going decisions in order to guarantee Performance, minimize Costs, and drive Agility. Application performance is still your responsibility in the cloud and it largely rests on your teams' ability to predict demand. Budgets quickly get lost in a quagmire of sizing, placement, and pricing decisions that have very expensive consequences. Agility does not come without a thoughtful approach to the cloud ecosystem you choose and that ecosystem's requirements—more importantly, it cannot exist if you do not guarantee performance. These challenges do not dissipate in the cloud, they only become more complex and the consequences more definitive.

As you design and implement your cloud strategy, it is important to understand how your teams currently operate, as well as how they will operate to support the highly dynamic demands that cloud computing inevitably enables. Will the processes and platforms they use today carry them into the cloud-first era?

Are You Ready for the Cloud?

Will the processes and platforms your teams use today carry them into the cloud-first era?

Today's cloud environments have thousands, tens of thousands, and perhaps even hundreds of thousands of workloads. Should the burden of navigating the complex multi-dimensional compute, storage, network resource—and cost—decisions be put on your teams? Consider this:

Monitoring does not scale; alerts and thresholds are reactive. These approaches also look at individual metrics in isolation when in reality applications require multiple resources in order to perform. Resource constraints must be evaluated simultaneously, in real-time, and with the context of the entire cloud environment—from the application layer to its underlying physical infrastructure.

Orchestration tools will automate, but they do not tell you what workloads to place where and when. Infrastructure teams must largely guess at placement and sizing, implementing those guesstimates via scripts and thresholds. Here again these systems consider metrics in isolation and often need consistencies in the underlying infrastructure in order to deliver scalability.

Cloud environments are heterogeneous—whether by business constraints, legacy, or in preparation for next-gen technologies. The platforms that make your cloud strategy a tried and tested implementation, must be infrastructure agnostic. Today you may have a mandate to go to the cloud. Tomorrow that may evolve into a multi-cloud or hybrid cloud strategy—meanwhile, your favorite Applications Team will need your teams to support their micro-services cloud-first architectures.

Visit **vmturbo.com** to learn more about Application Performance Control for your private, public, hybrid and multi-cloud environments.

Any journey to the cloud requires a real-time understanding of application demand, as well as every layer of the underlying infrastructure that supports it. It requires software that can make the real-time workload placement and sizing decisions that guarantee performance of any workload on any cloud. The cloud solution you choose today must assure performance in today's environments and as you scale with tomorrow's technologies.

About VMTurbo

Put application performance on autopilot.

VMTurbo's Application Performance Control platform is trusted by enterprises around the world to guarantee the performance of any application on any infrastructure, cloud or virtualized. VMTurbo's patented decision-engine dynamically analyzes application demand and automatically allocates shared resources to all applications, maintaining a perpetual state of health.

Launched in 2010, VMTurbo is one of the fastest growing technology companies on the market. Leveraging VMTurbo's control platform, customers can confidently accelerate their adoption of virtualized, cloud, and container deployments for all mission-critical applications today and in the future.

Leveraging VMTurbo, customers can guarantee application performance, maximizing infrastructure and human efficiency and positively impacting their companies' business goals.

To learn more, visit vmturbo.com.

Try VMTurbo Today

Private & Public Cloud Control

Deploy VMTurbo in your on-premises cloud environment to control workloads in private and public cloud.

[Download Free Trial]

Public Cloud Control

Deploy VMTurbo as an instance in AWS or Microsoft Azure to control workloads in the public cloud.

[Get Started Now in AWS]

[Get Started Now in Azure]

Supported Platforms

amazon web services™

Windows Azure

SOFTLAYER an IBM Company

openstack™ CLOUD SOFTWARE

vmware vCloud Director

Microsoft System Center Virtual Machine Manager