Enterprise Strategy Group | **Getting to the bigger truth.™**

White Paper

Performance Evaluation Criteria for Hyperconverged Infrastructures

Speed, Scale, and Stability

By Mike Leone, ESG Senior Analyst

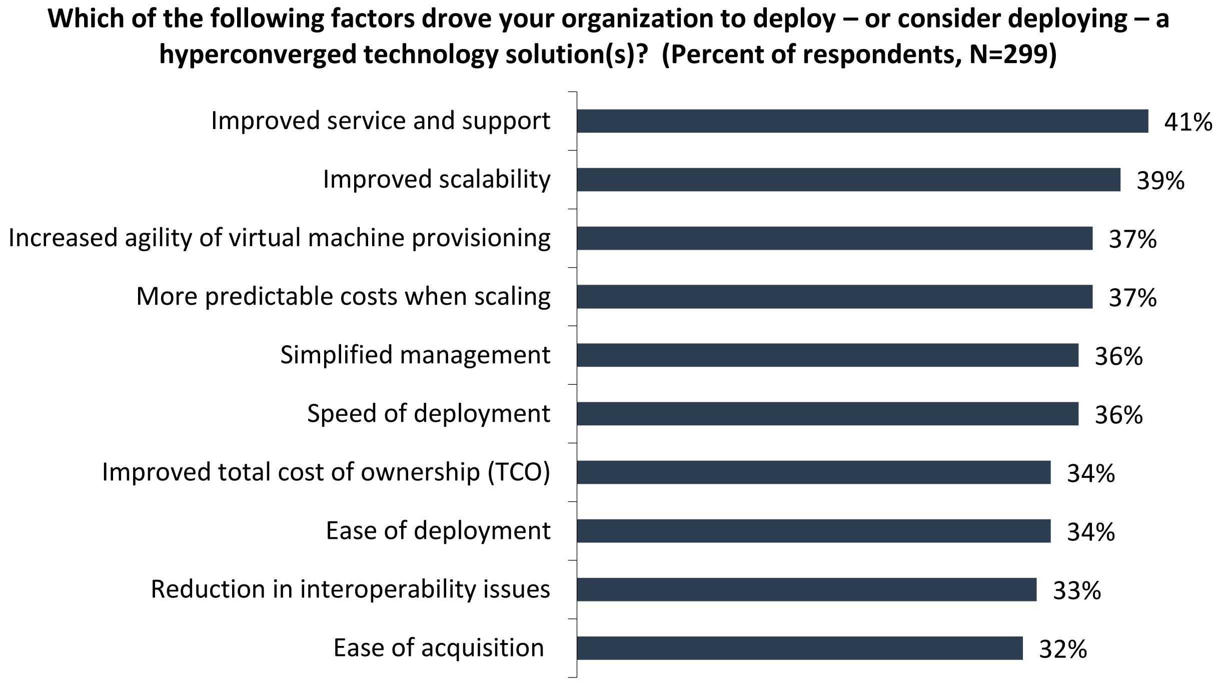
November 2016

This ESG White Paper was commissioned by Nutanix and is distributed under license from ESG.

Adoption of Hyperconverged Infrastructures

When hyperconverged infrastructures (HCIs) first hit the scene, they were all about consolidation, simplicity, and cost. Most organizations were looking for a way to improve operational efficiency by virtualizing their infrastructures and minimizing infrastructure complexity without breaking the bank. In fact, recent ESG research shows that most of the factors driving organizations to adopt hyperconverged technologies are related to savings, simplicity, and the speed at which those two goals can be achieved. [1]

Figure 1. Top Ten Factors Driving the Deployment of Hyperconverged Infrastructures



Source: Enterprise Strategy Group, 2016

Organizations use many of these potential benefits—as well as traditional technology features like high availability, data protection, and performance—as initial buying criteria. At first, as with any new technology, many organizations were hesitant to deploy their tier-1, mission-critical applications on HCIs; they were still immature and unproven, a far cry from traditional 3-tier architectures built with enterprise-class compute, network, and storage technologies from leading IT vendors. Consequently, many organizations kept their traditional infrastructures for important applications, and deployed HCIs either as an experiment, or as a place for lower priority applications like virtual desktop infrastructure (VDI).

As HCIs continue to mature and become more widely adopted, the key themes move beyond simplicity and cost savings. All HCI vendors can tout their ability to simplify, save on, and improve IT. HCIs are not only disrupting the thought processes around deploying traditional IT infrastructures, but also competing with and outright beating industry-leading vendors that have owned the traditional infrastructure space for years. Organizations are now confident enough in the technology to not only adopt it, but also leverage it as their primary infrastructure housing their most important business applications. As such, buying criteria have shifted from answering for "*Can* this offering support my requirements?" to "*How well* can it support my requirements?" Of course, many of these important requirements are related to performance.

ESG research shows that…

85% of survey respondents currently use HCI technology or plan to over the next 12 months.

[1] Source: ESG Research Report, *The Cloud Computing Spectrum, from Private to Hybrid*, March 2016. All ESG research references and charts in this white paper have been taken from this research report.

Performance: A Key Buying Criteria for Hyperconverged Infrastructures

Traditionally, performance buying criteria were all about speeds and feeds. "Hero" numbers showing millions of I/Os per second (IOPS) and massive throughput would nearly guarantee a vendor's place on a buyer's short list. However, the availability and cost-effectiveness of flash storage today has made it significantly easier for vendors to produce eye-popping IOPS and throughput results. Using lightweight workload generators that exercise only the storage, workloads can be configured to "superficially" simulate most business applications (e.g., file server, database, email, etc.) and generate big numbers. The keywords here are "exercise only the storage." In today's modern IT world, where many organizations leverage the cloud to satisfy some of their application workloads, a different performance testing methodology that evaluates more than the storage is required.

With HCIs, compute and storage are combined into a single building block that, when clustered with other building blocks, creates a large pool of compute and storage resources. Therefore, as with cloud infrastructures, HCIs also require a different performance testing methodology that exercises more than the storage. Emulated real-world workloads that test compute and storage together are essential for a true evaluation of HCI performance. Simply testing a single real-world workload on a four-node cluster is only the beginning. HCI buyers should ask three important questions when evaluating HCI performance:

1. How fast can my applications perform on a shared infrastructure?
2. What happens as my applications and supporting infrastructure grow?
3. How is my application impacted if a failure occurs?

Speed

The first important performance aspect of any system is the speed at which it can handle different workloads. This can be broken down into two test areas: theoretical and real-world.

- ***Theoretical performance*** testing focuses on understanding how the system handles different workloads and where potential bottlenecks can arise. This type of performance measurement traditionally exercises a single resource to understand its maximum capabilities, and therefore testing can be done much faster using a plethora of publicly available and well-documented tools that have been around for many years. Specifically, with storage, three metrics are traditionally and universally used as key performance indicators with theoretical performance testing: *IOPS*, *throughput*, and *response time*. IOPS represents the number of I/O operations (e.g., read or write requests) that can be serviced by the underlying storage at a given time. Throughput represents the amount of data processed (e.g., MB/second, GB/second, etc.), and response time represents how quickly a single operation occurs (e.g., milliseconds, microseconds, etc.). These three metrics merely serve as a subset of the performance metrics that are required to fully evaluate HCI performance. Additional metrics are required and important to understanding the true performance of an HCI infrastructure, such as resting CPU and memory usage, impact of simultaneously running workloads ("noisy neighbor"), application response times, etc.
- ***Real-world performance*** testing goes a step further by exercising multiple resources in a simulated environment using real applications. Results from theoretical testing are often used to set expectations for real-world testing. This type of testing is particularly important in hyperconverged environments since underlying resources are shared, so testing only a single resource tells an incomplete (and often misleading) story. The trouble with real-world performance testing comes from the length of time required to properly configure, validate, and test real applications with real datasets. IOPS, throughput, and response time remain important metrics, but application-specific metrics provide greater understanding. For example, in testing an OLTP database workload, adding a transactions/sec metric expands the evaluation because that is a database-specific metric that takes the completion of interdependent processes into account, making it harder to artificially inflate single resource performance results (i.e., gaming benchmarks). Further, truly evaluating the real-world performance of *your* specific application should involve migrating *your* workload to the platform under evaluation. This is the ultimate test that should always be done before making a final decision.

Understanding the speed of an individual application is obviously an important first step, but testing additional aspects of the underlying infrastructure also factors into properly evaluating the speed of a hyperconverged solution. For hyperconverged infrastructures to function properly, multiple nodes should be clustered together to create pools of shared compute and storage resources. Background services are constantly running to control everything from basic internode communication and cluster-wide data movement for data protection and resiliency, to deduplication and compression for efficient capacity utilization. These services consume cluster resources, and add to the list of non-application-specific circumstances that can impact application performance. Well-engineered systems will have built-in mechanisms to minimize resource consumption by underlying services to prevent application impact. An example would be the software knowing to limit the use of deduplication or compression services during peak hours in order to not impact production workloads. In all, key questions to focus on during this phase of performance analysis include:

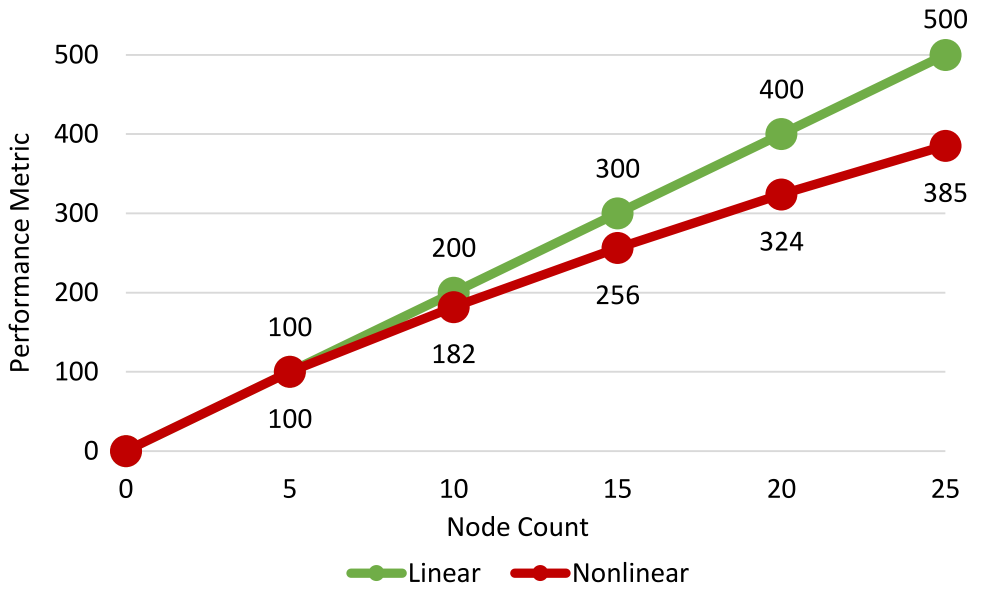
1. What component is being tested and is the right tool being used to test it?
2. Has enough information been gathered from theoretical testing to understand results from real-world testing?
3. How quickly, easily, and reliably can a POC environment be built using my real application(s) and dataset(s)?
4. Regardless of the default level of resources consumed by background services, is my application workload truly impacted by them?

Scale

There are several ways to view scale in HCIs. First, there is individual ***application scale***. Much of individual application scale can be accounted for in the "speed" phase, as you evaluate the impact of scaling on IOPS, throughput, and response time. It is also important to understand current and potential resource consumption if application performance requirements grow due to an increased user base, a growing underlying dataset, a newly added application feature, etc. This is where the configuration flexibility of hyperconverged infrastructures comes into play. Need more CPU? Add a compute-heavy node to the cluster. Need faster storage? Add a node with flash. This directly ties into the next aspect of scale: cluster scalability.

The theory behind ***cluster scale*** is simple: If more resources are needed, add another node specific to that need. The expectation is an immediate increase in performance. Simply put, if a cluster doubles in size, it should have twice the resources and therefore the potential for twice the performance. If this is achieved, the solution can provide linear performance scalability, which is ideal. Figure 2 shows the difference between linear and nonlinear performance scalability. The performance metric is irrelevant in this example, but starting out, a five-node cluster delivered 100 measured units in both cases. After adding five nodes, the linear scenario perfectly doubled to 200, while the nonlinear one did not. This continues as more nodes are added. Understanding whether or not a solution scales linearly is very important, as this understanding plays a crucial role in future planning.

Figure 2. Linear vs Nonlinear Performance Scalability



Source: Enterprise Strategy Group, 2016

The final aspect of scalability focuses on the ***tenant scale***. Early in the hyperconverged adoption curve, organizations were leveraging small-sized clusters to handle a single application, such as VDI. Today, organizations are more likely to leverage large HCI clusters and deploy multiple applications with multi-tenancy. Since HCIs are fully virtualized and share underlying resources, even when you use the fastest hypervisor technology with minimal impact to the underlying infrastructure, you must account for additional resource consumption overhead. Further, more complex virtualization processes can have an even greater impact on the overall cluster performance. A common concern is the "noisy neighbor," in which one virtual application steals resources from other applications in the cluster. This can have a significant impact on HCIs, especially in those that engage critical background services to ensure a healthy, fully functional cluster. This is where understanding the speed of a single application comes in handy. An easy way to validate multi-tenant scalability is to independently test the performance of each tenant or application based on existing requirements to establish a baseline, and then rerun the tests together. The goal is to run all applications simultaneously with minimal impact to performance when compared with the baseline. If available, additional features like quality of service (QoS) serve as a bonus to help prevent issues like the "noisy neighbor" from ever occurring.

Organizations should ask several questions when evaluating the scale of an HCI platform:

5. What is the maximum supported cluster size?
6. Does performance scale linearly as the cluster grows?
7. Can resources be scaled independently (specific nodes with more storage or more compute)?
8. Can the cluster scale without disrupting existing application workloads?
9. As new platform generations are made available, can they be added to preexisting platforms?

Stability

This last performance evaluation phase for properly vetting an HCI is determining how performance is affected when failures occur—i.e., the underlying infrastructure stability. Testing for this phase can be both fun and terrifying. Start a workload with an understanding of the infrastructure architecture and performance expectations. The fun part comes from intentionally injecting failures, such as pulling a network cable, unplugging a drive, or powering off a node, and observing the impact to the workload. The terrifying aspect comes when something unexpected happens, such as a VM not properly migrating or the cluster becoming unresponsive.

In short, the most important aspect is verifying that the HCI under evaluation is architecturally designed to work under failure with little to no front-end impact. In well-engineered hyperconverged infrastructures, most components are redundant (highly available) and/or shared. Each node will be made up of the same core components and therefore, in theory, any other node could be used to satisfy the workload. That means that if a node goes down, another node with available compute and storage can take over the workload, albeit at a degraded level of performance until everything in the background is sorted out. Once the VM has established a new primary node and storage has been rebuilt and redistributed, performance should return to the same level as before the failure occurred. Again, this is all in theory and may drastically differ between hyperconverged offerings. One offering may show a mere blip in performance that is completely undetectable to an end-user, while another without the level of automation needed to redistribute and recover must endure longer performance dips.

Numerous variables combine to define the overall stability of an HCI offering, demonstrating how a solution handles an unplanned (or planned) failure. To start, more nodes in the cluster means more resources are available and a higher level of data replication can be used, enabling a faster recovery. At the same time, the size of the application and its underlying dataset will directly impact the speed at which the recovery may occur.

Infrastructure stability is crucial to HCI evaluations and as such, key questions to ask include:

1. What happens when <insert component here> fails?
2. Are there any single points of failure?
3. How fast can a system get out of "degraded performance" mode?
4. Can all aspects of the system be upgraded without experiencing downtime?

The Bigger Truth

As hyperconverged offerings continue to mature and beat out traditional enterprise offerings, the evaluation criteria and priority list is shifting. Like the public cloud, with HCIs, it is no longer just about cost-effectiveness and simplicity. While these remain important benefits to evaluate when selecting a cloud provider or an HCI, performance is rising on the priority list and the traditional storage-centric definition of performance is no longer valid. With public cloud offerings, storage represents one aspect of the infrastructure (never mind one of the many costs). When running a mission-critical application in the cloud, the performance of both the storage and compute are part of the equation. This same idea holds true for HCIs.

Performance evaluation is so much more than generating a huge IOPS or throughput performance number from a 20-year-old workload generation tool. It's about the ability to predictably deliver that speed at scale while recovering from failures. If an HCI can do that, it represents the holy grail of IT infrastructure performance evaluation.

Enterprise Strategy Group is an IT analyst, research, validation, and strategy firm that provides actionable insight and intelligence to the global IT community.