A SOLIDFIRE PAPER

Choosing the Right Next Generation Data Center Infrastructure

Planning the best model for your future infrastructure growth

NetApp®

When it comes to deploying IT infrastructure, enterprises small and large now have more options to consider than ever. Making sense of the options, understanding their strengths and weaknesses, and ultimately selecting a model or set of models to utilize are some of the most fundamental decisions an enterprise IT department will make.

This isn't as simple as deciding which model is better. It's about understanding which model is right for you — right now and into the future. To help you make sense of these options is why we designed the Infrastructure Consumption Continuum.

Here's why things are so difficult

The rise of Software as a Service (SaaS) created new and significantly easier ways to consume software applications over the Internet, reducing or even eliminating the need to buy and manage an on-premise infrastructure. However, SaaS is not without tradeoffs and limitations. Therefore, the vast majority of enterprises continue to use a combination of traditional packaged software and custom-developed business applications. Many enterprises also deliver their products and services over the Internet (including SaaS companies of course), requiring traditional IT infrastructure to do it. So while the requirements for IT infrastructure are changing, the need for most enterprises to buy and manage infrastructure is not.

Fortunately, we are seeing new models for IT infrastructure consumption. The range of options available to enterprises today includes:

Best-of-breed appliances

The traditional and still dominant model of buying a "best of breed" mix of individual storage, networking, and compute hardware components, which are then integrated and configured individually.

Infrastructure as a Service (IaaS)

IaaS removes the hardware and data center elements from IT infrastructure completely, by delivering compute, networking, and storage services from a hosted environment over the Internet. IaaS eliminates much of the traditional data center and infrastructure architecture requirements of running IT systems, and in most cases changes the cost model as well, from an up-front capex model to a pay-over-time subscription or usage-based model. The architecture, operational, and financial differences of this model are substantial, which makes it perhaps the most disruptive trend in infrastructure in the last decade.

Converged infrastructure

With converged infrastructure, a vendor or coalition of vendors brings together a set of best-of-breed systems into a validated, tested, and supported package. This may take the form of a reference architecture, a meet-in-the-channel solution, a "single SKU" package, or even a fully integrated rack architecture. This convergence is meant to simplify the selection and testing of individual components by providing a known-good configuration that has been validated for both compatibility and performance against an expected set of workloads.

Hyperconverged infrastructure

Hyperconverged infrastructure takes the idea of convergence further, combining multiple functions (such as compute and storage) in a single functional appliance, and often integrating some software-based networking capabilities into the appliance as well. Hyperconvergence reduces the number of piece-parts that need to be purchased and installed, and often simplifies management through an integrated appliance-level view of the infrastructure.

Infrastructure software on commodity hardware

Since much of modern infrastructure systems is software based, one emerging model is to separate the software and hardware entirely, allowing multiple hardware platforms to be used with a piece of infrastructure software (such as a storage system or network device), and allowing different software platforms to be installed on the same hardware platform. This consumption model brings a new level of flexibility to infrastructure procurement, deployment, and lifecycle management.

Other relevant data center trends. The confusion continues.

New infrastructure consumption models aren't the only trends having an impact on IT departments today. Many other innovations are happening, some of which are intertwined with these new delivery models.

Private cloud

Private cloud is a broad term with many definitions, but generally it refers to delivering public-cloud-type services in a dedicated fashion to a single enterprise or entity. While private cloud infrastructure can be deployed in a service provider data center (e.g., "hosted private cloud") or in an enterprise data center or colocation facility ("on-prem"), the key distinguishing characteristic is that it typically resides on dedicated hardware infrastructure rather than shared infrastructure. While most private cloud deployments use a capex hardware purchase model, some hosted offerings may offer the combination of dedicated hardware and subscription or usage-based pricing.

Any infrastructure consumption model except Infrastructure as a Service (excluded because it typically uses shared infrastructure) can be used to build a private cloud. However, when considering the desired characteristics of a private cloud, including simple scalability, flexibility, and a requirement to run mixed or unspecified workloads, some models may be better suited than others.

Public cloud

Public cloud encompasses a whole range of IT services delivered over the Internet, most notably Software as a Service, Platform as a Service, and Infrastructure as a Service. However, many emerging or more narrowly focused services are being delivered via the public cloud today as well, including communications (UCaaS), business process (BPaaS), and virtual desktops (DaaS). As it relates to infrastructure consumption models, IaaS is typically seen as a subset of the public cloud.

For a service provider looking to offer public cloud services, any infrastructure consumption model can be used to build an "as a service" offering (yes, even IaaS on IaaS). The right consumption model choice will often be critical to both the cost effectiveness and scalability of the service offering.

Software defined data center (SDDC)

Software-defined is perhaps an even broader and more poorly understood term than private cloud. In some cases, it is used to describe the "software on commodity hardware" model of infrastructure deployment. However, this is a narrow definition that misses the true value of the SDDC. The key spirit of the SDDC is around the ability to control, deploy, and reconfigure infrastructure (compute, networking, and storage) via software, and specifically via open API control points. This is often referred to as "separation of control from data plane," however, even that definition is debatable.

While the definition of SDDC may be unclear, it does have a clear relation to infrastructure consumption models. Whichever consumption model is used, SDDC is about being able to discover, deploy, manage, consume, release, and monitor that infrastructure via software-based systems. SDDC is a key enabler of infrastructure agility and flexibility, so considering it in conjunction with any given consumption model (or vendor's implementation of a consumption model) is important. How programmable is the infrastructure? How complete are the APIs? How integrated are the control points into standard software-based data center management stacks (such as VMware, OpenStack, and CloudStack)?

IT as a Service

IT as a Service (ITaaS) refers to the trend of IT departments looking and acting more like an internal "service provider" than the traditional project-based IT organization. With ITaaS, IT organizations strive to offer a set of shared services, including software, development, and data platforms, and even bare infrastructure to their business units in a standardized fashion rather than a set of one-off implementations.

ITaaS encompasses many practices, including service catalogs, self-provisioning, BYOD, and show-back/charge-back. The goal of ITaaS is to make IT more capable and responsive to the business, enable more cost effectiveness with shared underlying infrastructure and practices, and better compete with external infrastructure and service offerings.

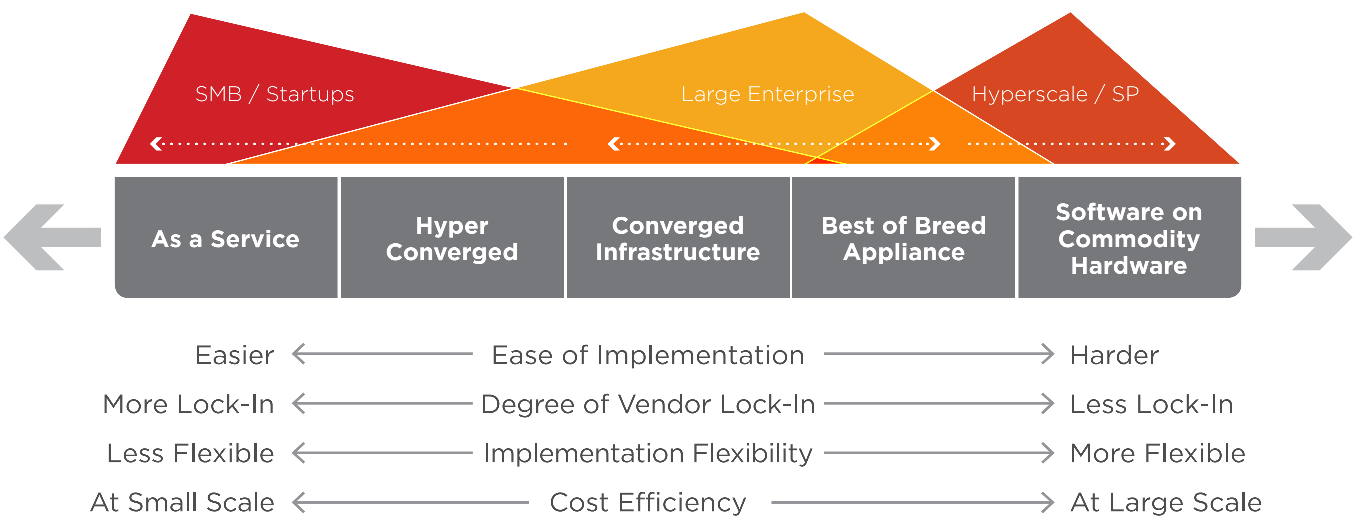
ITaaS can be executed on top of any infrastructure consumption model, and while ITaaS doesn't strictly require a private cloud or SDDC, those approaches are often popular ways to offer standardized and rapidly provisioned service offerings.

Selecting the best consumption model for you

Say hello to the Infrastructure Consumption Continuum

While it would certainly be possible to enumerate a long list of advantages and disadvantages for each consumption model, this approach would make comparing multiple models (or all five) difficult. Furthermore, as the industry evolves new consumption models, the menu of available options and their associated tradeoffs becomes even more complicated. To help enterprises better understand the relationship among the different consumption models, and ultimately decide which model or models are right for them, SolidFire has created a tool called the Infrastructure Consumption Continuum.

The Infrastructure Consumption Continuum



The Infrastructure Consumption Continuum starts with the assumption that each consumption model has a valid reason to exist. That is, each model offers advantages over others, as well as disadvantages or tradeoff. If only one of these were true (e.g., the model offered only advantages over an existing model with no tradeoffs), then there would be no reason for both the exist. The converse is true as well; if a model suffers only disadvantages compared with existing models, without a corresponding advantage, it would have no reason to be considered.

We have chosen a set of viable consumption models and assigned a corresponding set of key considerations (or trade-offs). The Infrastructure Consumption Continuum maps considerations against models to ultimately narrow down and help you select just the most relevant choices.

Prioritize your needs and choose your model

The explosion of options for infrastructure consumption means there simply isn't time to compare every vendor in every consumption model. Beyond that, independent research is hard to come by and hands-on testing is typically required.

What's needed is a method to quickly identify the most viable consumption models, and then look only at the vendors and solutions in each of those models. The Infrastructure Consumption Continuum helps this task by placing the consumption models relative to each other against four key considerations (See the appendix for detailed descriptions of each consideration).

1

Set a relative priority on each of the four key considerations.
Determine how significant your organization values each key consideration. For instance, is ease of implementation or cost efficiency at large scale more important? By understanding the relative importance of factors such as lock-in, cost efficiency, and time to value, you can quickly narrow your list of viable consumption models.

2

Eliminate the two models at the far end of your top priority.
For example, if your top priority is ease of implementation, you would eliminate "best of breed appliances" and "software on commodity hardware." If implementation flexibility is your top concern, you would eliminate "IaaS" and "hyperconverged".

3

Eliminate one of the remaining models at the far end of your second priority.
For example, if your top priority was implementation flexibility, and your second priority was ease of implementation, you would further eliminate "software on commodity hardware" and be left with "converged infrastructure" and "best of breed appliances." If your top priority was "cost efficiency at small scale" and your second priority was "time to value," you would be left with "IaaS" and "hyperconverged".

You now have two remaining consumption models to consider. At this point, it probably makes sense to look more closely at both the remaining priorities and your specific requirements to see if you can select a single consumption model that fits your needs. If not, you may need to further investigate solutions for both of the remaining models to make a final decision.

Multiple options may be valid

For complex IT environments, or environments with widely varying priorities and needs (such as branch-office IT vs. core infrastructure), you may need to repeat this exercise multiple times for different use cases. In the end, you may require multiple consumption models. Of course, using multiple consumption models will always add complexity over adopting a single model, but this may be the only way to meet some business needs, or may simply be more cost effective than a single poorly suited model.

Using multiple consumption models shouldn't be seen as a failure, so long as each model was chosen as the right fit for the business requirements and priorities. Very few complex enterprises will be successfully served by only a single consumption model over time.

Plan to evolve over time

Even if a single consumption model can accommodate all of an enterprise's IT needs today, CIOs should expect to accommodate change over time. We are seeing rapid advancement in technology both within individual consumption models, and even the creation of new models, or models that blend or blur the lines between existing models.

Over time, as models evolve and new consumption models are created, adjusting or adding new models to the continuum can help enterprises understand where the opportunities are to move to a new consumption model — and gain better alignment to existing or changing priorities.

Conclusion

As we've seen over the last decade, the evolution of available infrastructure options has made the task of architecting the Next Generation Data Center more challenging than ever. Indeed, the wide variety of infrastructure consumption models today is a tremendous opportunity for enterprises of all sizes. These options enable both more rapid and more cost-efficient deployment and management of computing infrastructure at scales both small and large. That said, the additional complexity of selecting the right consumption model, vendors, and products often clouds these decisions with uncertainty.

If one thing is clear, it's that no one option fits every need. CIOs need to balance the advantages of each infrastructure option with the sacrifices they should consider while making a decision. Our Infrastructure Consumption Continuum helps you sort through today's options so you can make the decision that makes the most sense for your needs.

Appendix — The key trade-offs

While other trade-offs exist between different consumption models (such as the number of different vendors/providers that each offer, relative maturity, etc), the four we chose represent the most critical defining characteristics of different infrastructure consumption that are unlikely to change over time. We will explore each trade-off below, and further describe where each model is placed, relatively, on the continuum.

Ease of implementation

Ease of implementation rates the relative time to value of each consumption model. How quickly can you go from deciding to add additional infrastructure to having productive infrastructure available?

When it comes to time to value, the simplicity of Infrastructure as a Service is unmatched. Most IaaS providers allow users to quickly spin up (and often spin down) new infrastructure within minutes. Long-range planning, such as data center space plans and scaled network design is eliminated, and short-term turn-up time, such as ordering, delivery, installation, and configuration is eliminated as well. Any enterprise that deals with unpredictable infrastructure requirements, is scaling very quickly, or simply doesn't have the time and expertise to deal with proper data center design would be well served by ITaaS solutions.

By comparison, the various hardware-based infrastructure consumption models all require some level of both long-term planning and short-term turn-up time for additional infrastructure. However, as you move along the continuum to the right, the implementation complexity continues to increase.

Hyperconverged and converged infrastructure models require a reduced number of choices up front, since the vendor has done much of the matching, sizing, and compatibility checking. Installation and configuration is often significantly faster as well, since installation guides or vendor-provided services cover the entire infrastructure stack rather than a single component.

Selecting and installing best-of-breed appliance-based infrastructure can be complex and time consuming, particularly when new systems and vendors are being considered. The number of choices and combinations are limitless, and although most vendors provide compatibility matrices, they are often not comprehensive. Configuration can also be more complex, particularly when different vendors recommend different best practices. Once initial planning, design, and installation are complete, adding additional infrastructure is usually more straightforward. However, the flexibility to constantly bring new best-of-breed technology into the data center means that planning and design is never really complete, and as the environment grows more heterogeneous over time, managing many different systems and vendors can be a complex management burden.

Finally, software on commodity hardware models offer the most complex implementations and take the longest to deliver differentiated value. Choosing hardware and software platforms separately adds an additional layer of decision making to the design process, and testing to validate compatibility is often required. The performance and reliability characteristics of software solutions can vary widely depending on the hardware they are deployed on, so further time is required to characterize those aspects of the system. Once initial selections have been made, the procurement process is often made more complex with additional vendors involved and mixed pricing of software and hardware, and support for both, making cost analysis more difficult. Adding additional infrastructure after the initial deployment can be simpler, so long as the software or hardware components don't change. However since the lifecycles of hardware and software releases will often be mismatched, additional testing, qualification, and version management will add more complexity to managing the infrastructure over time.

Degree of vendor lock-in

Vendor lock-in runs across the infrastructure consumption continuum in the opposite direction of ease of implementation — from most lock-in on the left to least lock-in on the right. While many enterprises may be willing to trade faster time to value for increased levels of lock-in, that lock-in can create significant added cost down the road if they decide to switch vendors, to change consumption models entirely. CIOs should carefully value short-term considerations around ease of implementation with longer-term risk of vendor lock-in.

While on the surface, IaaS may appear to have minimal lock-in, with the ability to quickly spin up infrastructure in other cloud providers only a credit card swipe away, in practice, it has the highest true lock-in if you want to maintain the other benefits it offers, such as ease of implementation. With IaaS, creating a single infrastructure platform (without the complexity and cost of cloud-to-cloud WAN links) requires you to select and stay with a single service provider for the majority of your infrastructure. The capabilities and services offered by different service providers vary widely, on everything from basics such as which Hypervisor is used and what VM configurations are offered to provider-specific platform services such as NoSQL databases and queuing. The more you invest in taking advantage of the unique services of a provider (particularly at the application and API layer), the more difficult it will be to move in the future. Software-based compatibility or cloud-wrapping layers often result in a lowest-common-denominator set of services, or mask critical but subtle differences between services. Data lock-in is a common concern as well. As your data footprint inside a service provider grows from terabytes to petabytes, the practicality of moving that data to a different provider (particularly while keeping applications online) becomes nearly impossible.

Finally, although many IaaS providers offer pay-as-you-go pricing, most also offer longer-term contracts at steep discounts, which will become very important to keep IaaS cost-competitive over time. Signing long-term contracts only creates further lock-in with the service provider.

Hyperconverged infrastructure requires a high level of vendor lock in to provide the key benefits of simple management and critical feature integration. By definition, hyperconverged combines several key infrastructure components from a single vendor, but these systems are only really compatible and integrated with themselves. Although you can choose to deploy a hyperconverged system from a different vendor, or a different consumption model entirely, you will end up with multiple islands of infrastructure, adding complexity to the management and increasing implementation time.

Converged infrastructure shares many of the same lock-in issues as hyperconverged, since to get the full benefits of integration and validation, you need to stick with the prescribed hardware and software combination. However, there is often some more flexibility in these systems to accommodate either changes to the design (in case of a reference architecture), adding in different best-of-breed systems to the architecture over time, or sharing a piece of the infrastructure (like the storage array) with other unrelated systems. Note that in some cases, going outside the proscribed design may eliminate the integrated support offerings that can be a key benefit of converged infrastructure. Finally, because converged infrastructure is often simply made up with a series of pre-selected best of breed appliances, you can always decide to “blow up” the model and simply use the appliances individually.

Best-of-breed appliances offer a high level of flexibility combined with limited long-term lock-in. Standards-based storage and networking protocols can allow multiple vendors appliances to work together as part of a single infrastructure, although you may lose some vendor-specific capabilities as a result. The ability to integrate storage and compute systems from different vendors in particular can significantly reduce infrastructure cost by creating more competition, and also allow flexibility to choose multiple solutions for different needs in the infrastructure.

Software on commodity hardware offers the least vendor-specific lock-in, as it allows you to swap both the hardware and software vendors, often independently and simultaneously. As mentioned, this comes with an added cost of ensuring interoperability of different software and hardware platforms, but gives the greatest control to enterprises who want to fully control their infrastructure destiny independant of any one vendor.

Implementation flexibility

Implementation flexibility determines how much control you have over the final design and capabilities of the infrastructure. Increased flexibility allows infrastructure solutions that are closely tailored to application and business requirements, and can be changed over time as those requirements shift. Less flexibility results in fewer choices to make, but may deliver an infrastructure that is less than ideal from a price, performance, or availability perspective, or starts well aligned but becomes poorly suited over time. In some cases, enterprises may choose to make up for the lack of flexibility in one consumption model by utilizing multiple consumption models, (e.g., IaaS for dev/test environments, and best-of-breed infrastructure for performance-oriented production apps).

While IaaS may offer a wide breadth of functionality “out of the box,” it essentially offers no flexibility on the actual implementation. The services, features, and performance levels offered, and the underlying platforms used to deliver them, are solely at the discretion of the service provider. Offerings and pricing may change on a regular basis, forcing customers to re-evaluate their selections, and while pricing typically goes down over time, it may not fall at a rate as fast as the underlying hardware cost reductions. Enterprises must accept this lack of flexibility and the high switching cost of changing providers in order to obtain the other benefits that IaaS offers.

Of the hardware-based models, hyperconverged infrastructure offers the least flexibility. Each vendor typically has a small number of configurations, and often the compute and storage configurations are linked tightly together, making it difficult to get the correct ratio of either. Changing the capabilities and performance of the infrastructure over time is similarly limited by what the hyperconverged vendor roadmap and compatibility limitations. This has led to hyperconverged initially being used heavily for very specific use-cases (like VDI), where common configurations can work across a wide range of enterprises, and minimal customization is required.

Converged infrastructure offers a higher level of flexibility. Reference-design-based converged infrastructure systems allow customers or partners to customize the system by sizing different components based on their needs, while still maintaining the overall compatibility and simplicity of the system. Because converged infrastructure systems are often made up of individual best-of-breed components, often one or more of those components can be upgraded independently of the others when needed. Longer term, enterprises may still be somewhat limited in their ability to customize the system or add third-party components to it due to the integrated nature of the systems and support.

Best-of-breed appliance-based infrastructure allows tremendous flexibility by allowing customers to not only mix compute, network, and storage systems from different vendors based on their needs, but integrate multiple different systems of a single type when application needs require — such as when utilizing both an all-flash system and an object storage system. The only real limitation on this flexibility is ensuring that the various vendors and systems are compatible with each other (which partners can often help assist with), and the fact that ultimately, you are tied to a specific vendor’s software on an appliance for the life of that appliance. However, the secondary market for hardware can often be used if an enterprise decides to make a wholesale change or upgrade in their infrastructure.

Software on commodity hardware offers the ultimate level of flexibility — in some cases more than an enterprise may want or need. The ability to spec out the hardware for each system down to the component level, and then independently select the software to run on the system, allows systems to be built or rebuilt to suit very specific needs, or simply standardized to the preferences of the enterprise. With this flexibility, much of the burden of ensuring compatibility, performance, and availability now moves from the vendor to the enterprise themselves, adding a significant cost to the increased flexibility.

Cost efficiency

Cost efficiency attempts to answer the question "Which consumption model is least expensive?" … something every CIO is trying to understand. Of course, the answer is not as simple as "This model is the cheapest, and this one is the most expensive." All of the other trade-offs come into play as the ultimate TCO (or ideally, ROI) of the infrastructure is calculated. The most general way to consider cost efficiency on the Infrastructure Consumption Continuum is to align the models based on which are more cost efficient (less expensive) at small scales, and which are more cost effective at very large scales. Since many other trade-offs have a larger or smaller impact depending on scale, this helps simplify the overall question of cost effectiveness. Of course, at exactly which infrastructure scale the balance tips to the next consumption model is very enterprise-specific. For example, an enterprise with a large existing physical data center footprint will have a much lower cost for adding new hardware infrastructure than a startup with no data center space at all. However, here is some general guidance.

At small scale, in the range of 1 to 100 VMs, there is undoubtedly no more cost-efficient consumption model than IaaS, since there is significant "startup" and manpower cost associated with any of the hardware-based consumption models. For many enterprises, IaaS can continue to be cost effective well beyond that — up to 1000 VMs or beyond, due to the reduced human capital cost, the ability to spin up and down resources quickly, and the breadth of capabilities offered out of the gate. However, as the infrastructure size continues to grow, it becomes harder for IaaS to compete against hardware-based infrastructure, particularly for raw resource requirements like compute, bandwidth, and storage capacity. The limiting factor is essentially that service providers need to make a margin on top of their hardware and people cost, and while the best service providers get excellent economies of scale on both, an efficient enterprise at scale can obtain those economies as well. In addition, the lack of flexibility in IaaS means that often an on-premise solution can deliver better results with fewer overall resources.

At a scale of 100 to 1000 VMs, or for smaller environments where regulatory or security requirements eliminate IaaS as an option, hyperconverged can offer a simple way to deploy and manage platform for infrastructure. What it lacks in flexibility due to limited platforms and vendor lock-in, it often makes up for in human capital cost due to significantly simplified setup and management. However, beyond a certain point, the lack of flexibility will start to show up in decreased utilization, over-provisioning, and difficulty running some workloads. As the increased cost of a single vendor infrastructure grows, it may make sense to trade some complexity and higher management cost for lower systems cost.

Converged infrastructure solutions can be cost effective from a scale of 100 VMs to 1000s, depending on the level of integration, support, and flexibility of the system. Often, converged infrastructure offers a path to deliver faster time to value and lower human capital cost than best-of-breed appliances, but with some longer-term flexibility available for more cost efficiency as the infrastructure scales.

Beyond 1000 VMs, or in complex environments with a wide variety of workloads and infrastructure needs, best-of-breed appliances offer the flexibility required to build a cost-effective infrastructure that is much more finely tuned to enterprise needs than a converged of service-based model. While the additional complexity will incur higher startup and management costs, this is amortized over a much larger footprint, and if the proper investments are made in automation, the day-to-day management cost can be very comparable to other models.

Finally, at hyperscale of tens of thousands to hundreds of thousands of VMs, software on commodity hardware starts becoming cost effective. The amount of up-front planning, testing, configuration, and integration for software-based solutions can be significant, as can ongoing tasks such as integration, imaging, upgrading, and multi-vendor support management. These added costs over the other consumption models really only make sense if the level of flexibility offered by software on commodity hardware is fully utilized to drive down the overall cost of the infrastructure significantly. This can be done through volume purchasing, rapid adoption of new lower cost hardware, and continuous optimization of the overall data center design and delivery. This approach has been taken to the extreme by organizations such as Google and Facebook who design both their own hardware and software, but more moderate approaches that combine off-the-shelf hardware and software solutions may also deliver benefits.