VMware vCloud® Architecture Toolkit™
for Service Providers

Architecting a VMware vSphere® Compute Platform for VMware Cloud Providers™

Version 2.9

January 2018

Martin Hosken

VMware, Inc.
3401 Hillview Ave
Palo Alto, CA 94304
www.vmware.com

Contents

List of Tables

List of Figures

Overview

The VMware Cloud Provider™ Program software-defined solution provides on-demand access and control of network bandwidth, servers, storage, and security while maximizing asset utilization. Specifically, the VMware software-defined solution allows VMware Cloud Providers to integrate all of the key functionality that business customers demand, including:

- Self-service portal for end-user and administrative provisioning
- Catalog of available compute services
- Rapid, precise, and automated service provisioning
- Multi-tenant monitoring, reporting, and billing integration capabilities

These features help the service provider's enterprise business consumers maintain control over infrastructure and management, while realizing the benefits of service provider economies of scale through flexible and commercial delivery models.

The VMware Cloud Provider Program software-defined architecture also helps service providers offer highly customized services that meet the demanding needs of their business consumers. The focus of this document is to assist the service provider community in creating solutions that will drive a scalable foundation in the infrastructure fabric, and to deliver value-added services and create new revenue streams. Through the VMware Cloud Provider Program, end consumers of services benefit from their providers' ability to meet agreed upon service level agreements (SLAs) and from the rapid provisioning of new services in anticipation of ongoing and changing business requirements.

Scope

This document includes design guidelines, considerations, and patterns for building the software-defined compute (SDC) components of a VMware Cloud Provider Program instance. The purpose of this document is to provide VMware Cloud Providers with guidance to:

- Establish an enterprise-class cloud compute offering as an alternative or a complement of existing dedicated, single-tenant compute offerings.
- Design efficient infrastructures typically associated with service providers' economies of scale.
- Design a best-in-class compute service capable of responding quickly to consumers' unpredictable business conditions and usage habits.
- Leverage VMware proven high availability to quickly redeploy servers in the event of a problem, improving service levels for consumers.
- Accelerate time to market for new SDC-based offerings by leveraging validated VMware proven solutions.
- Reduce infrastructure and operational complexity when deploying proven cloud-based services based on VMware technology.

This document addresses key design and architectural decisions relating specifically to the compute aspects of VMware Cloud Provider Program based services as implemented by the VMware service provider community. It provides a design framework but does *not* provide a specific solution for either service providers or enterprise customers that want to offer IT as a Service to their internal business units.

Use Case Scenario

The sample conceptual and logical design configurations outlined in the following sections are based on use cases from a UK-based VMware Cloud Provider that has data centers across Europe. This service provider wants to provide virtual data centers across a shared multitenant platform and also provide dedicated private cloud service offerings to their consumers based on the next-generation VMware platform.

The service provider has an extensive hosting portfolio, which has assisted the growth of the company over a number of years. Currently there are many different platforms and this separation of products is leading to manual workflows and limitations in the automation and self-service capabilities they currently offer to their consumers. For this reason, there is a requirement to consolidate these offerings under a single physical infrastructure and consumer portal, and provide a new level of automation and self-service to the new generation of platforms. The primary aim of this modernization of existing platforms is to reduce the support overhead and to free operational resource time and work effort to focus on higher value customers. The increase of agility within the self-service and API driven portals enables the business to engage customers higher up the stack, while offering greater automation in terms of scaling, and adding, removing, or changing services.

The following sections address two different use cases within the hosting portfolio:

- A virtual data center service with a multitenant shared hardware platform.
- A hosted private cloud service that offers a dedicated hardware platform with private cloud agility and flexibility across multiple highly available data center locations.

3.1 Service Definition – Virtual Data Center Service

Currently this service is sold on a “per server (blade)” basis. A customer contracts for a certain number of servers and receives role-based and locked-down access to the shared VMware vCenter Server® system to manage their servers. Storage can be provided through dedicated arrays, but it is more common for service consumers to receive a percentage of shared storage (dedicated LUN) from a shared array. Firewalls and load balancers are provided from the providers’ shared platforms, but all changes must be undertaken by operational staff on receipt of a change request from a consumer.

Next-generation models will need to increasingly support hybrid capabilities. Link the virtual data center environment to a customer’s “on-premises” VMware environment, enabling easy migration of virtual machines and the addition of new services in future releases.

In addition, a consumer must be able to manage all their resources in one place, while the provider continues to assist with capacity management, infrastructure, and hypervisor support, and creates new value-added services, such as disaster recovery, backup and archive, network, firewalls, and so on, on a per-component basis.

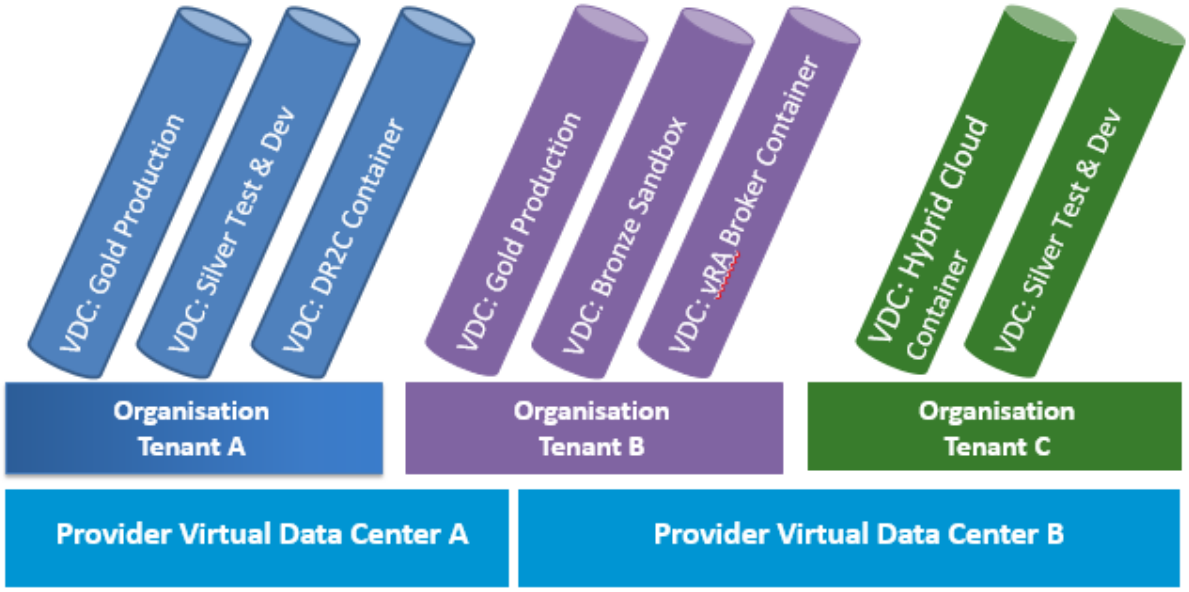
The virtual data center based solutions are a low margin product for the provider, so make every effort to verify that the operational team can work in as effective a manner as possible to keep management work effort low. This can be achieved through the increasing use of automation, orchestration, and self-service, while maintaining the consumer’s high level of flexibility.

The provider’s next-generation virtual data center service has the following key requirements:

- Multitenant hardware (compute and storage).
- Containers of guaranteed compute resources.
- Compute (CPU and RAM) must be provided as a virtual resource pool, with no direct link to physical servers, and scalable in smaller increments allowing consumers to order RAM in 10 GB increments, and vCPU in multiples of 2.
- Storage (tiers) must also be provided by pools.
- Retain the option for a “per blade” solution per customer. This should be the exception, not the norm.

- Retain the option for the provider to decide whether to share the resource pools across blades, or to assign certain customers to specific hardware.
- Ability to move the resource pools around the platform to perform tasks such as maintenance or migration.
- Consumers can create all of their VMs.
- Access to a template library (provider catalog) of available images.
- Consumers can create their own template and images, for example, to support an application running on a new operating system.
- Automation is required to create virtual pools of resources per customer, driven by the provider, as well as set up VLANs, networks (firewalls, switching, load balancing, IP addressing, and so on), and any images needed. There should be minimal manual effort required on the part of the provider to implement these services.
- Provide customers the ability to self-service firewall and load balancer changes, as well as create and remove networks. (Ideally provide in phase 1.)
- The provider must be able to manage capacity across the platform (both shared and dedicated) to manage the infrastructure and address any performance concerns.
- Ability to bill customers for VMware resource usage based on what they consume, as the cost will vary based on the number and size of virtual machines each customer installs.
- Ability to offer operating system licensing as an option (Windows and Red Hat Linux) in case the customers do not want to use their own.
- Burst capability must be an option on a virtual data center resource pool.
- Self-service can be API or GUI-based (with feature parity between the two options), and role-based (so customers can have some read-only users, for example, and other users limited to changing only certain environments).
- Portal access must be through a link (ideally with SSO) from the providers' existing online portal.
- All changes on the portal must be logged and be visible to the provider.
- Within an environment, consumers can use their own host names for their virtual machines.
- Support for snapshot-based backup.

Figure 1. Virtual Data Center Service Conceptual Design



3.2 Service Definition – Hosted Private Cloud Service

While the hosted private cloud service is a higher value than the virtual data center service, developing automation that is more effective is required. In addition, self-service and the ability to perform adds, changes, and deletions must be provided, both for initial deployment of an environment by the provider, and afterwards.

Today, if a customer orders a hosted private cloud service (either as part of an initial environment deployment or as an in-life addition), the service provider supplies all implementation services. While this is required to verify that all virtual machines are licensed, installed, and operated correctly, further automation in the building and management of these servers is needed.

Many of the virtual data center requirements described earlier are applicable here, with the exception of some of the self-service capabilities which, in a managed environment, must be controlled to a certain degree. In addition, resources (and commercials) will be on a per-virtual machine basis, rather than a pool of resources.

The provider's next-generation hosted private cloud service has the following key requirements:

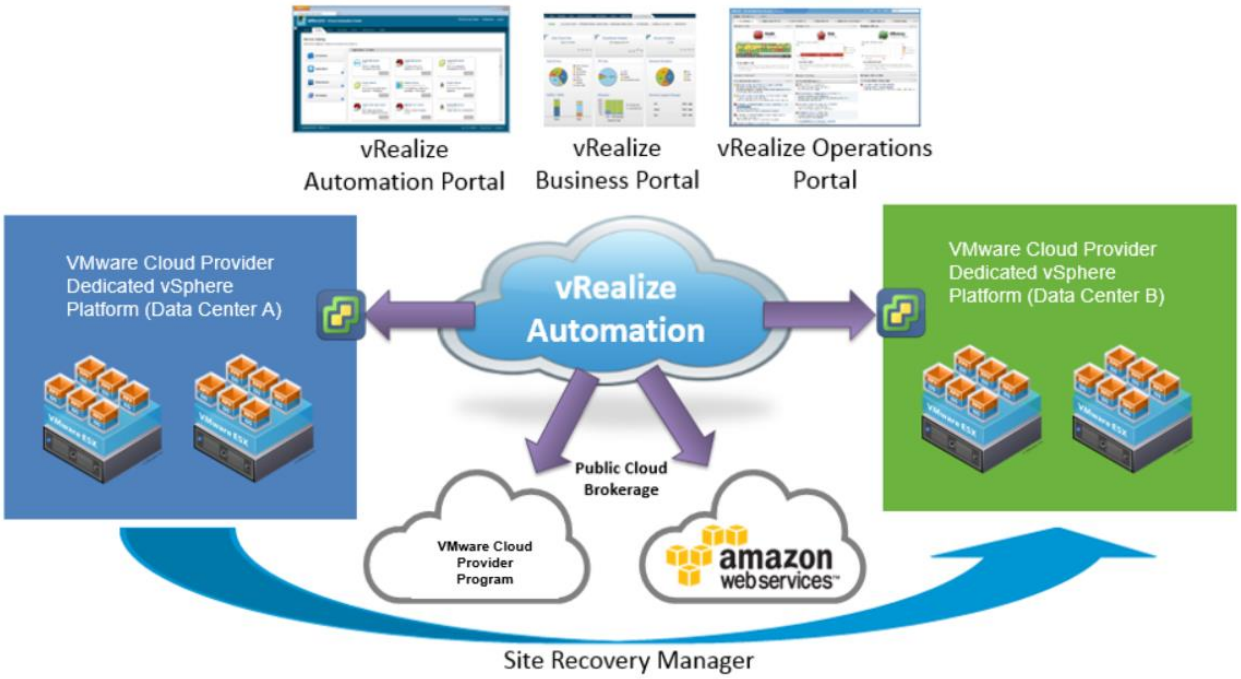
- Compute (CPU and RAM) is variable by virtual machine, and can be changed, dynamically through a self-service portal.
- Storage (tiers) must also be provided by pools, but different tiers can be attached by drive to VMs.
- Solution is per customer or a "private cloud," so that the provider can assign resources delivering a customer's virtual machines to specific hardware.
- VMware and OS licensing will be provided by the service provider as standard, per virtual machine, and built into the price.
- Within certain bounds, a customer must have some self-service capability, such as requesting additional resources (RAM, storage, and so on).
- A customer must be able to create virtual machines from a controlled catalog of images.

- A customer must be able to request that a new template be created based on their template. However, tasks (ideally automated, but possibly manual) will be required to create this template in such a way that a managed virtual machine can be deployed from it.
- Storage provided by drive to be able to support QoS for specific applications.
- Automation is required for the provider to manage implementation of these services, including creating virtual pools of resources per customer, attaching these resources to specified virtual machines, setting up VLANs and networks (firewalls, switching, load balancing, IP addressing, and so on), installing the operating system, and preparing backups (file level) and storage.
- The provider must be able to manage capacity across the platform (both shared and dedicated) to address any performance concerns and manage the infrastructure.
- All operating system patching must be able to be automated, with rollback and self-service capabilities, allowing a customer to select which patches they require and when.
- (Ideally phase 1) Provide customers the ability to self-service firewall and load balancer changes, as well as create and remove networks. This must be limited to basic requests from a service catalog. More complex changes will be designed and implemented by the provider.
- As a phase 2 feature, provide the capability to auto-scale virtual machines based on monitoring thresholds and criteria being met (possibly with a customer action to reboot the virtual machine at an acceptable time). In addition, support options, such as bringing up powered-down virtual machines within a load balanced group.
- Develop options for “mothballing” a virtual machine at a lower price (as a phase 2 feature) so customers can turn off unused virtual machines (for example, a staging/test environment) and drop to a lower level of billing.
- Self-service must be API or GUI-based (feature parity between the services) and role-based (so customers can have read-only users, for example, and users limited to changing certain environments only).
- Portal access should be through a link (ideally with SSO) from the providers existing online portal.
- All changes on the portal should be logged and visible to the provider.

Key service requirements include:

- Consumer access to vCenter Server and VMware vRealize® Automation™ dedicated private cloud stack
- Provider-managed hypervisor
- Shared storage (Dedicated LUNs) with optional add-on service of dedicated storage hardware
- Snapshot, application-aware, and file-level backups
- Dedicated VMware vCenter Server Appliance™ / management stack / physical resource / fabric

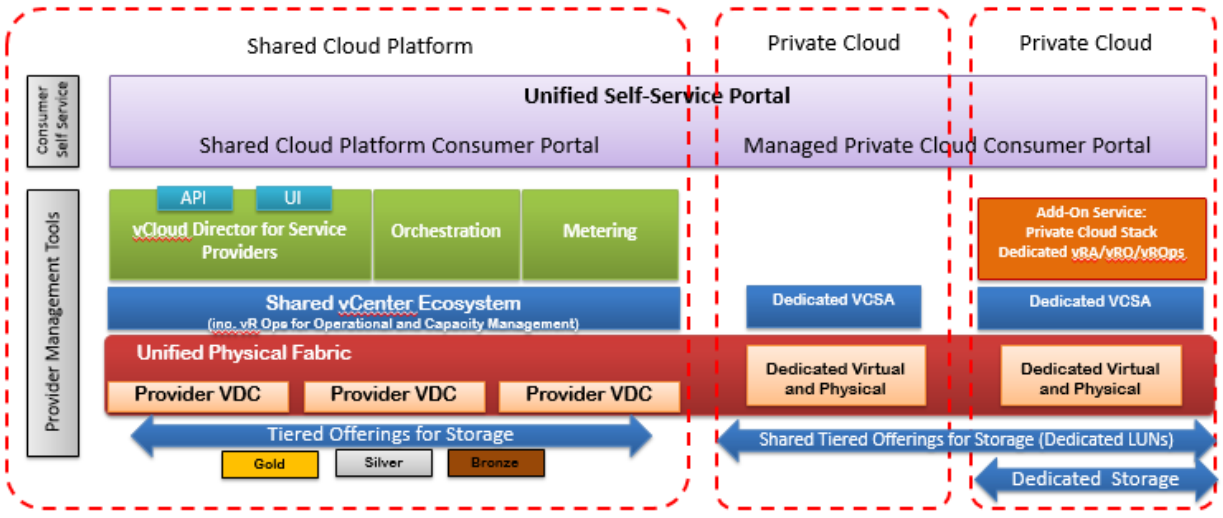
Figure 2. Hosted Private Cloud Service Conceptual Design



3.3 Integrated Service Overview – Conceptual Design

The main differentiator is not at a platform level (which must be consistent across both next-generation hosting offerings), but at the management and service level.

Figure 3. Services Overview Conceptual Design



The provider must be able to position the various components within an applicable service level suitable to a consumer's application or requirements. Whether a consumer uses virtual data center, hosted private

cloud, or a blend of both hosting service offerings, these must form part of an integrated solution, regardless of the underlying infrastructure.

To achieve this, the provider must standardize across the hosting infrastructure platforms and networks, and layer on clear, repeatable, and supportable orchestration and automation to manage these on a per-resource, per-consumer, and per-platform level.

Future hybrid capability is a key strategy and includes disaster recovery, management of third-party clouds, and management of replication from a customer's infrastructure off-premises to the service provider's data center. To achieve these goals, the provider cannot afford to write a full orchestrator, automation, and customer portal from scratch. In working with VMware, the provider wants to consider best practices and off-the-shelf tools, such as VMware vCloud Director®, to avoid large development efforts and delays that would be involved in creating something from scratch.

Software-Defined Compute and Hypervisor Concepts

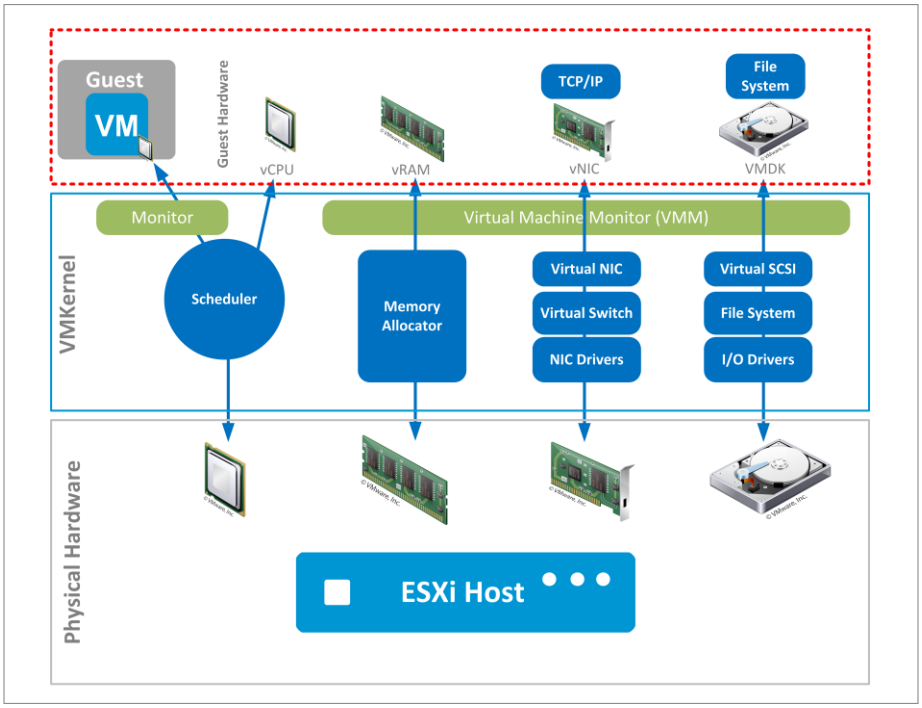
The software-defined compute platform for the VMware Cloud Provider Program is provided by the VMware ESXi™ hypervisor, which enables service providers to build an enterprise-grade, scalable, multitenant platform for complete compute service lifecycle management.

The hypervisor replaces the traditional operating system, such as Microsoft Windows or Red Hat Linux, and provides the ability to create a number of virtual servers on the same hardware. This technique provides multiple benefits, one of which is isolation of the virtual server from the underlying hardware, allowing hardware resources to be utilized more fully, and providing the mechanism for dense server consolidation that is the basis for enabling cloud-based computing.

Employing this VMware based software-defined computing layer gives the service provider the ability to seamlessly deliver highly-scalable, on-demand infrastructure services to consumers, while reducing power, saving space, maintaining reliability, and reducing the overall cost to serve.

While a virtual machine is a logical entity, to its operating system and end user, it appears to be a physical host with its own CPU, memory, network controller, and disks. All virtual machines running on a host share the same underlying physical hardware, but each consumes its share in an isolated manner. From the hypervisor's perspective, each virtual machine is a discrete set of files that include a virtual machine configuration file, data files, and so on.

Figure 4. Hypervisor Architecture



Scalability and Designing Physical Resources

With any virtual infrastructure design that is required to scale extensively across hundreds or even thousands of hosts, provide petabytes of storage, and support large complex networks, extensibility is a key factor. For a successful service provider deployment, scaling a large physical platform while maintaining control, compliance, and security is critical. Taking a predefined building block approach to this type of architecture, from day one, is paramount in planning for scalability.

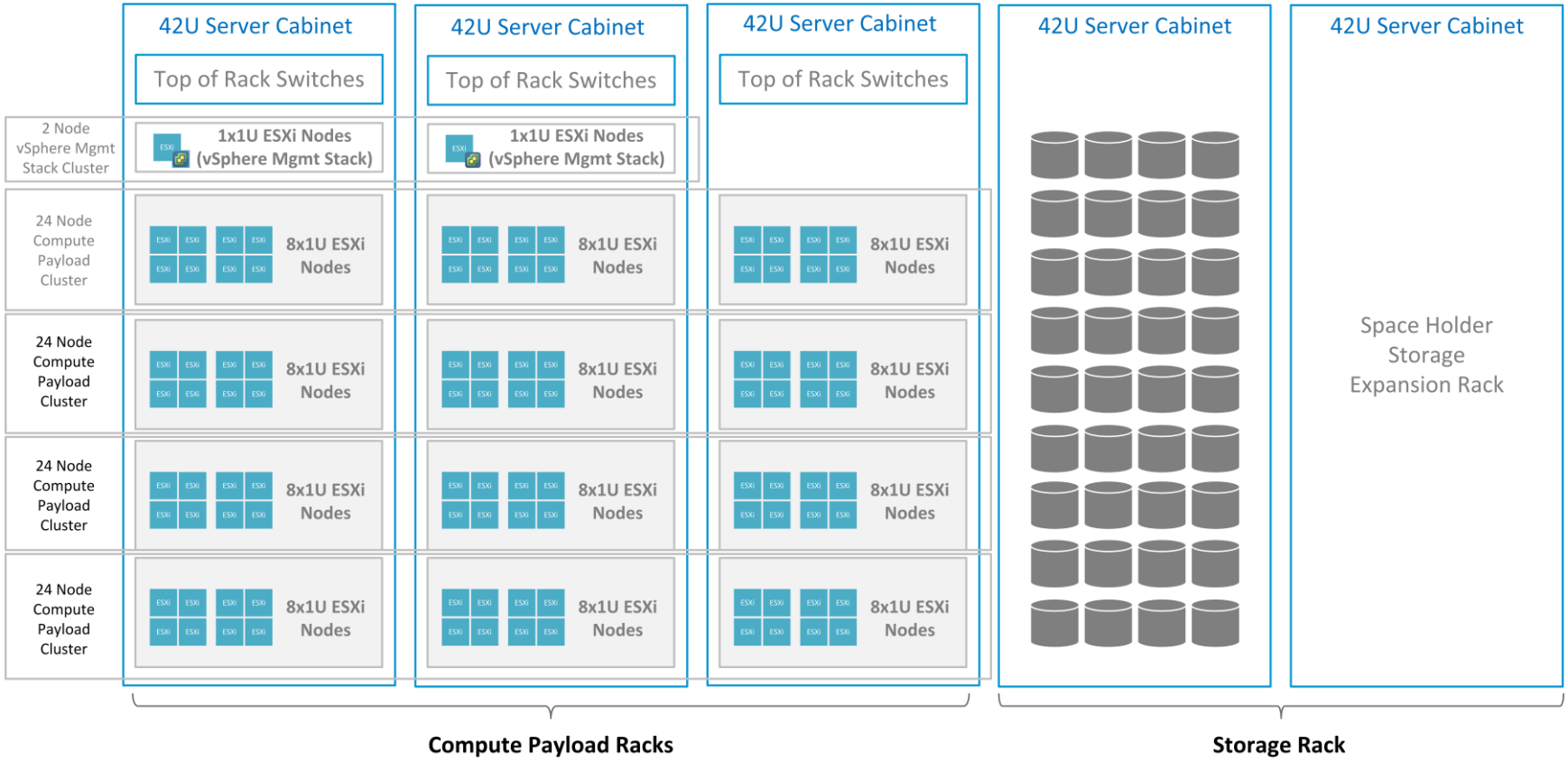
In addition, the configuration and assembly process for each system must be standardized, with all components installed identically. Standardizing the configuration of the physical components is critical in providing a manageable and supportable infrastructure by eliminating variability, which in turn, reduces the amount of operational effort involved with patch management and helps provide a flexible building block solution that meets a service provider's requirement for elasticity.

While the configuration and scaling is likely to be hardware vendor dependent, this model must form part of the platform design, which will be linked to commercial growth estimates of the services offered.

For instance, the sample design in the following figure represents a possible building block scenario where the service provider for the compute payload clusters is employing a traditional approach to storage, whereas a hyper-converged architecture is utilized for the hardware associated with the management and edge clusters, with VMware vSAN™ ready compute nodes deployed.

From a physical platform perspective, each "Payload vPod" is made up of 96 rackmount ESXi compute hosts configured as four 24-node VMware vSphere® clusters split equally across three server cabinets and a two-node vSphere local management component cluster. Each Payload vPod also houses two 48-port 10-GbE "leaf" switches, two 48-port 8-GB multilayer fabric switches, and two 1-GbE IPMI management switches for out-of-band connectivity. Each of these Payload vPods is designed to provide a fault domain for compute, network, and storage.

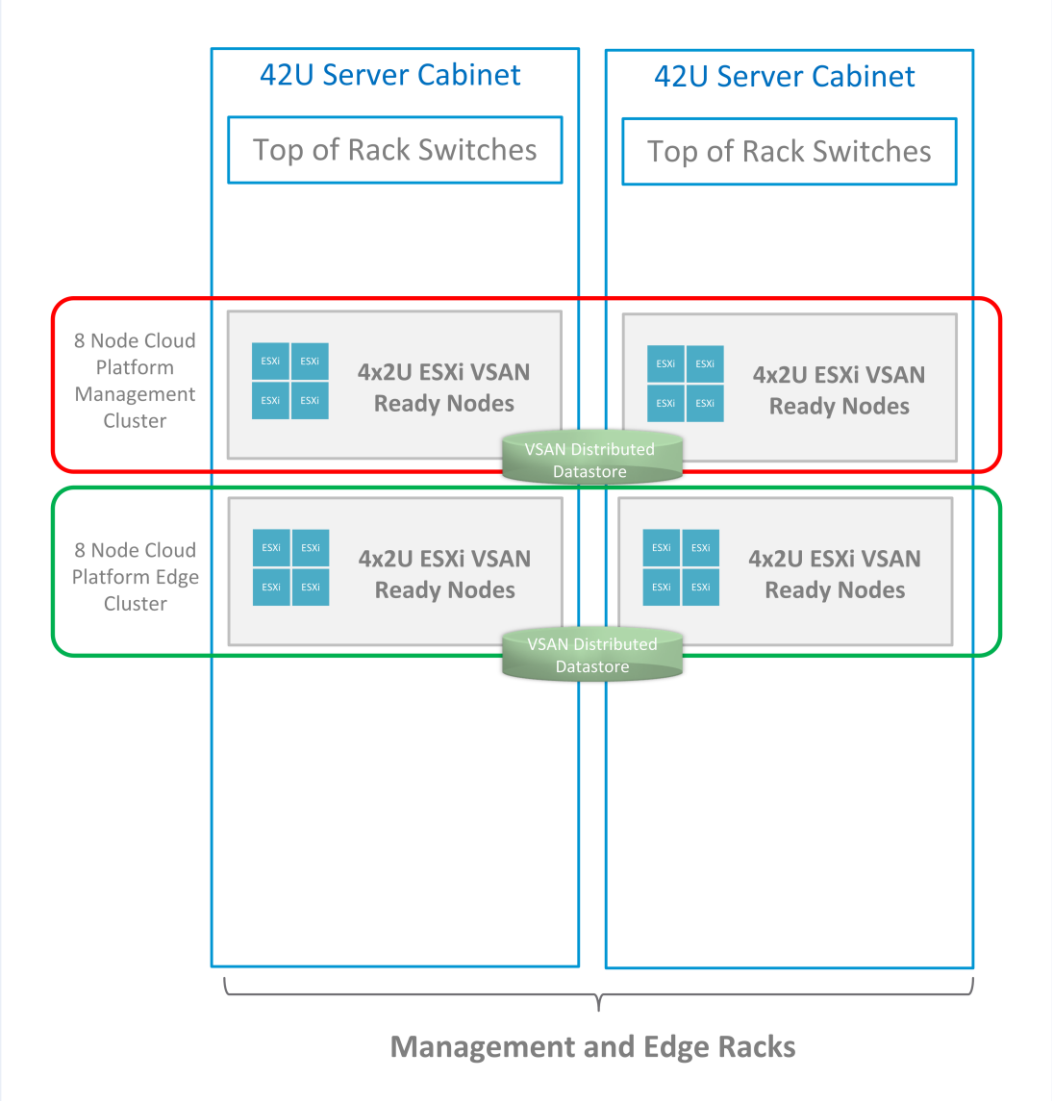
Figure 5. Sample Payload vPod Logical Architecture



The number of Payload vPods in this sample design can be scaled out accordingly, depending on hardware, software, and power limitations. The vSphere components in each Payload vPod are managed by a single vCenter Server instance.

In addition to the Payload vPod, this sample cloud-based architecture requires a Management vPod to house the Top Level Management (TLM), Cloud Management Platform (CMP), and VMware NSX® Edge™ components required to support all the Payload vPods within the "VMware Cloud Provider Program Data Center Block." In this sample architecture, the Cloud Platform Management Cluster and Edge Cluster are hosted on distributed virtual datastores provided by the vSAN ready nodes.

Figure 6. Sample Management vPod Logical Architecture



In this sample design, there is a design constraint of 16 Payload vPods per availability zone, due to power limitations in the data center halls. In this architecture, this entity, made up of one Management vPod and 16 Payload vPods, is referred to as a VMware Cloud Provider Program Data Center Block.

This building block architecture is further represented in the data center layout figures that follow, demonstrating one VMware Cloud Provider Program Data Center Block located at a single physical data center across two availability zones.

Figure 7. Logical Data Center Layout – Single VMware Cloud Provider Program Data Center Block

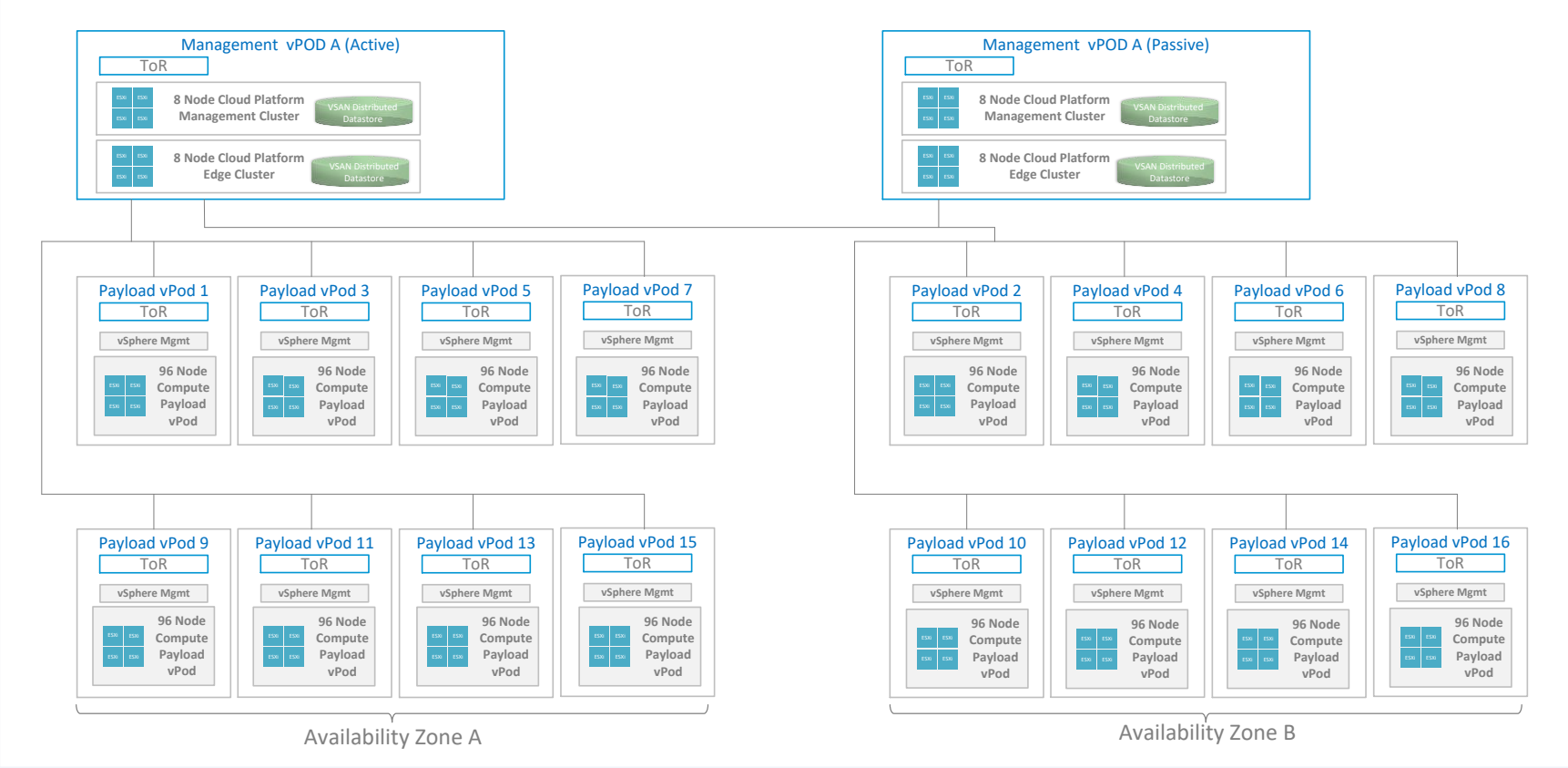
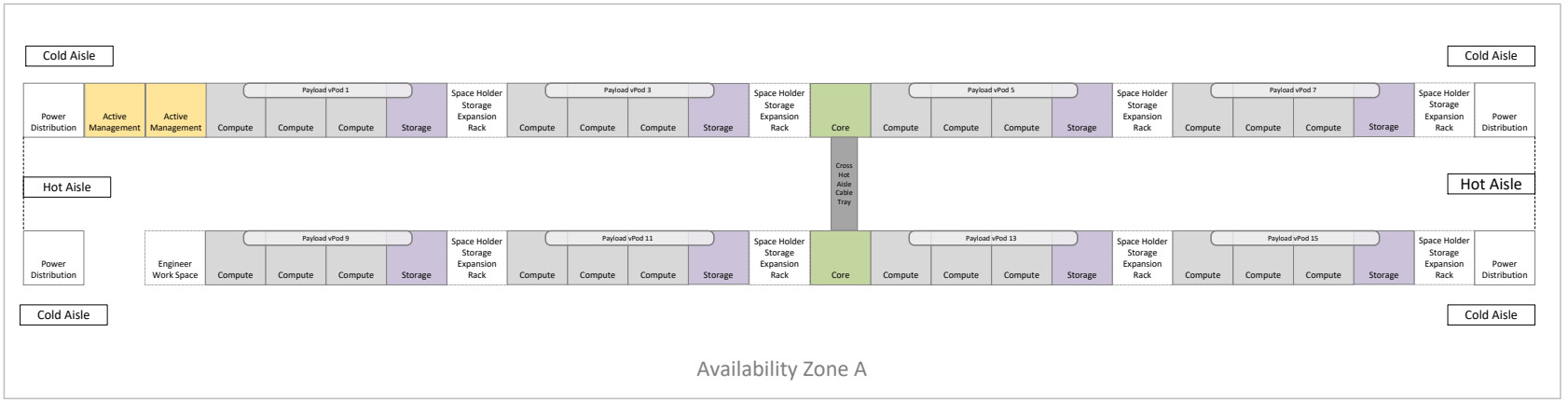
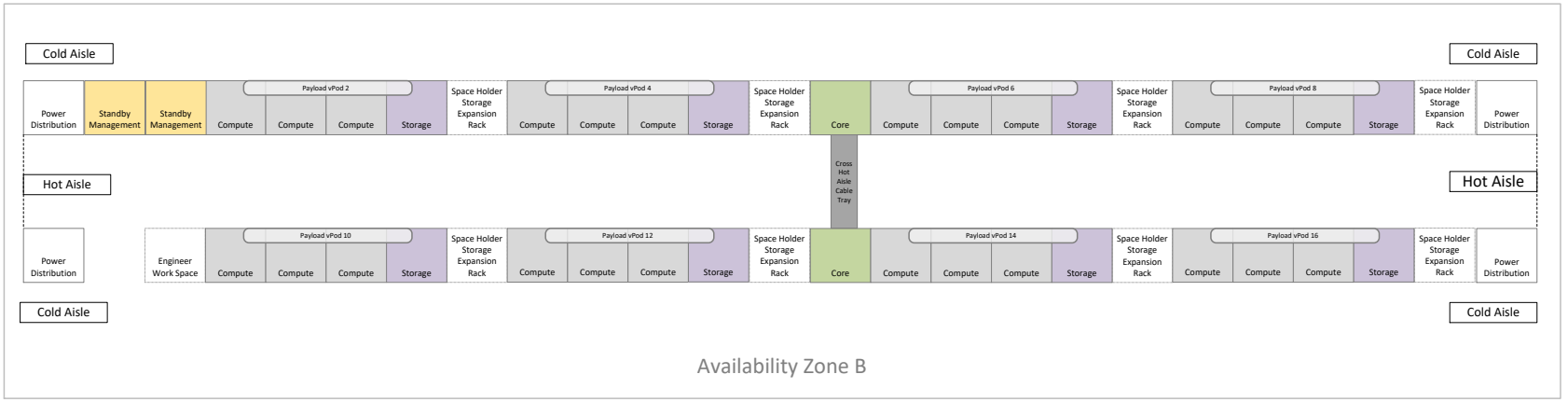


Figure 8. Physical Data Center Layout – VMware Cloud Provider Program Data Center Block





The compute, storage, and network resources available from each component layer of this building block architecture, as specified in this sample design, are provided in the following table.

Table 1. Capacity Scalability of Building Block Architecture

Resource	Host	Cluster	Payload vPod	VMware Cloud Provider Program Data Center Block
Memory	512 GB DDR3	10.5 TB (24 nodes with 3 reserved for HA)	42 TB	672 TB
CPU	2 x Intel E5 8-Core 3.1 GHz = 49.6 GHz	1,041.6 GHz (24 nodes with 3 reserved for HA)	4,166.4 GHz	66,662.4 GHz
Fast Storage	N/A	Flexible Configuration	180 TB	2.9 PB
Standard Storage	N/A	Flexible Configuration	300 TB	4.8 PB
Network Bandwidth	20 Gbps	420 Gbps	1,680 Gbps (80 Gbps MLAG to Spine)	10 Gbps to Internet

There is no single solution to scaling the VMware Cloud Provider Program physical infrastructure platform. During the design phase, a number of factors play an important role in designing the building blocks including, but are not limited to:

- Expectations regarding services and provider growth
- Hardware availability and lead times
- Physical hardware scalability limitations (such as with blade system management tools)
- Capital expenditure and hardware depreciation considerations
- Data hall power, space, and cooling limitations

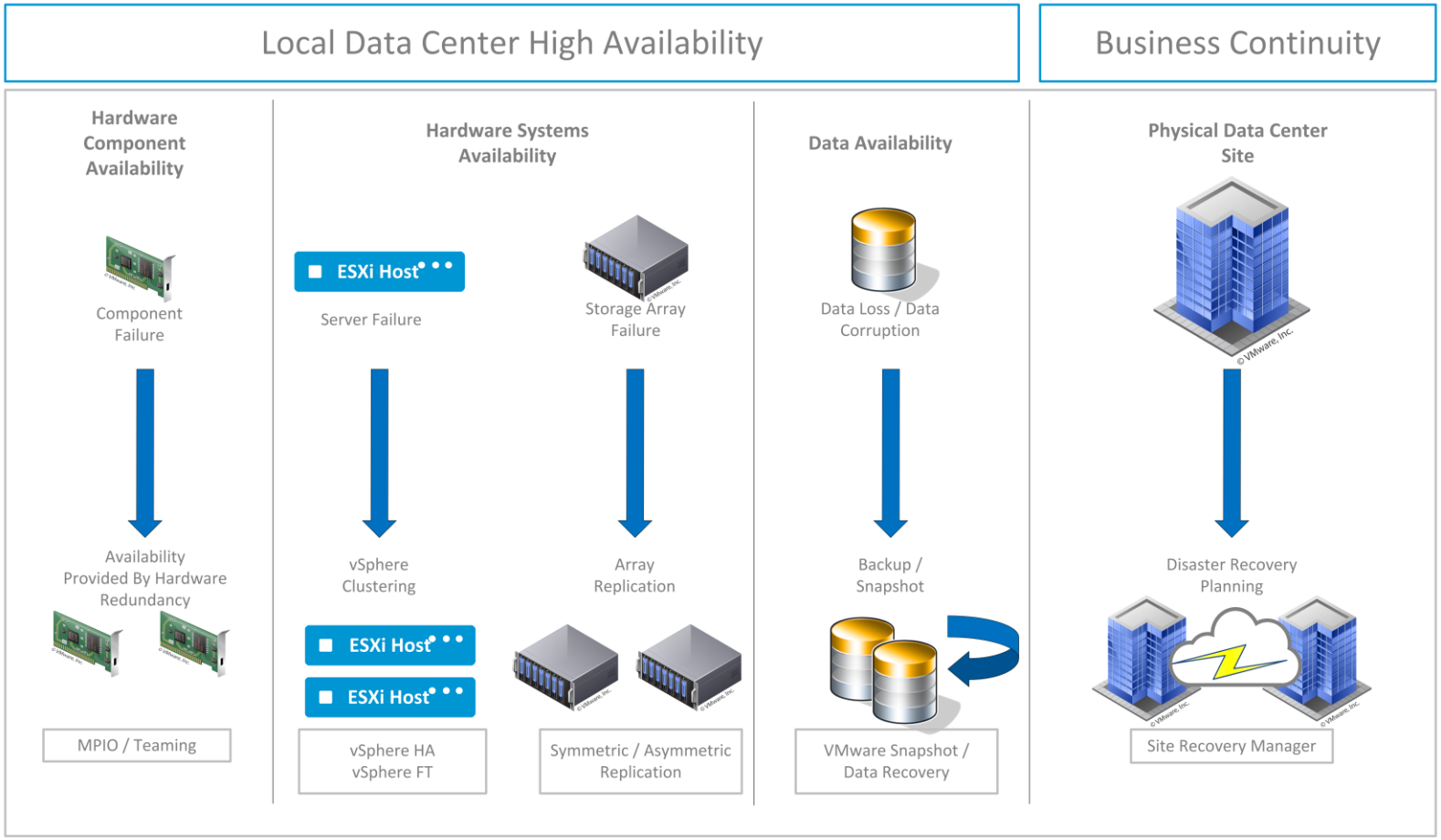
5.1 Infrastructure Protection

There are two considerations for protecting the building blocks that make up the physical infrastructure—local availability, referring to the single physical data center, and cross-site availability, referring to the business continuity processes that are undertaken when either a partial or complete data center incident takes place on the first site.

Local availability is achievable through employing a number of technologies and strategies to prevent service disruption when either a single component or multiple components fail. The following tactics can together form a strategy to mitigate these types of component failures:

- Redundant hardware to eliminate single points of failure.
- Employing vSphere clustering and HA technologies to return services to an online state in case of host outage.
- Use of snapshots to quickly return to a previous healthy state in response to application failure or corruption resulting from operating system of application updates.
- Securing data both onsite and off-site to prevent data loss through traditional backup and replication mechanisms.

Figure 9. Infrastructure Component Availability



5.1.1 Power Distribution

VMware recommends the use of uninterruptable power supplies (UPS) with these designs to provide high availability in the event of a power outage from utility providers. The UPS systems is typically placed outside of the data halls to avoid additional space usage, heat dissipation, and power consumption. Two redundant panel boards inside the data hall must be fed from the UPS systems and provide the power required for all active equipment within this design.

5.1.2 Cable Management

By employing cable management best practices, the design of the “vPod” racks will provide easy access to device components during maintenance windows and provide proper cooling efficiency. It is imperative that cabling does not block the easy insertion or removal of any field replaceable units (FRUs) on any piece of equipment, nor block hot-air exhaust outlets within the cabinets. The copper and fibre patch cables themselves must be easy to trace though standardized naming labels and, as such, simplify troubleshooting scenarios. It must also be possible for cables to be easily moved, added to, or changed during proactive or corrective maintenance.

5.1.3 Environmental Considerations

To achieve maximum efficiency and optimization in this solution, an appropriate cooling design, power distribution, and grounding configuration must be employed. Best practices typically followed in large data center implementations are applicable for these designs. In the cooling design, the objective is to provide adequate cold air supply to the equipment and properly remove hot exhaust air. A sustainable cooling system that follows industry best practices is essential to the stability of the hardware components. With this approach, not only is the equipment safe from unplanned downtime due to overheating, but efficient energy usage provides significant OpEx savings. With a properly implemented power distribution system, the equipment can remain available even during power outages or service disruptions. Finally, the grounding and bonding of systems must be able to maximize equipment uptime, maintain system performance, and protect engineers from injury during maintenance.

5.2 Eliminating Single Points of Failure

In addition to protecting the physical infrastructure, any VMware Cloud Provider Program platform must have high availability at the core of every design decision. Therefore, eliminating single points of failures is a key design requirement.

A single point of failure could render an entire cloud platform unavailable if a central component were to fail, with dire consequences for the service provider. Typically, single points of failure are mitigated by the use of redundant hardware limiting the impact of and helping to avoid service interruptions. For this reason, it is paramount that VMware Cloud Providers design their core platform with the following considerations:

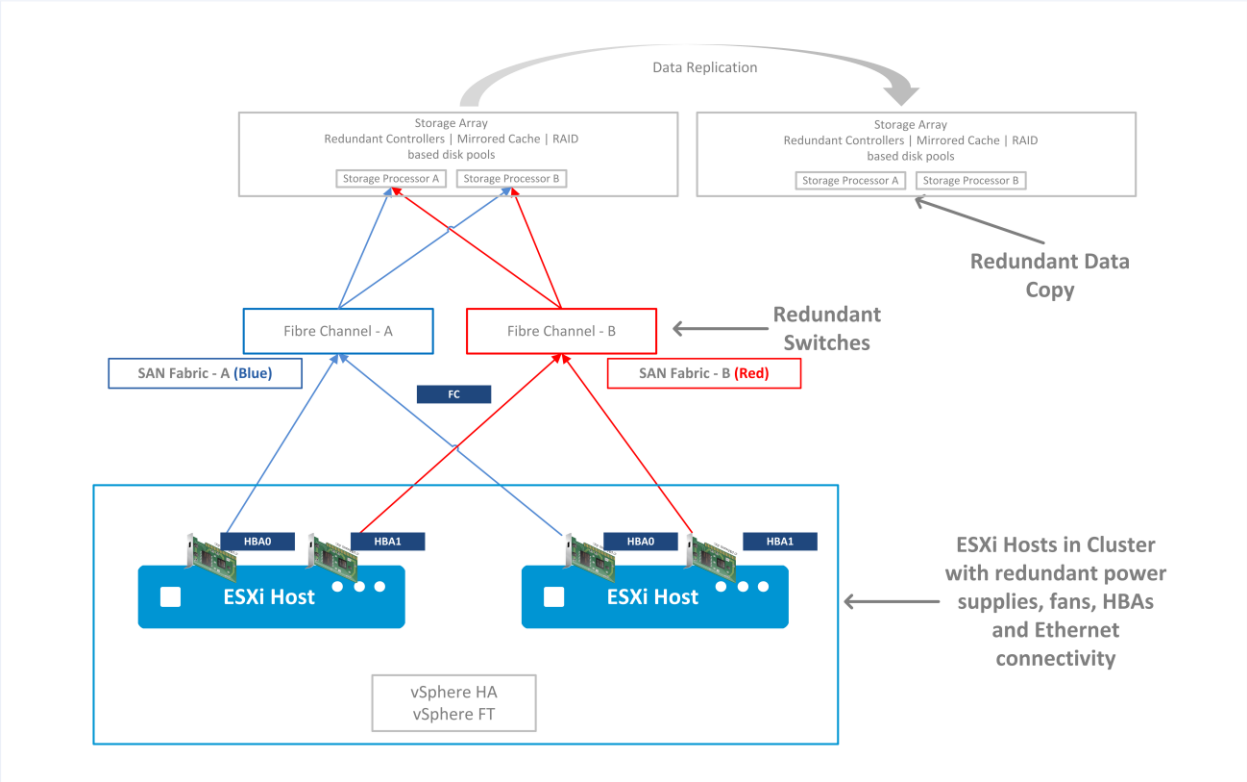
- Physical hard disk drives with spinning spindles have a relatively low mean time between failures (MTBF) and must be protected because they contain customer and provider data that is frequently accessed. To mitigate against single or multiple hard disk failures, employ RAID or similar technologies.
- Servers, switches, and other critical cloud platform hardware must support multiple power supplies, fed from separate circuits. Fans and other moving parts that are susceptible to failure must be fully redundant within the hardware.
- Ethernet or converged network adaptor cards can be protected by employing the IEEE 802.3ad or similar protocol to aggregate links.
- A storage area network (SAN) must be designed with redundant components throughout. This includes, but might not be limited to, Host Bus Adaptor (HBA) cards, Fibre Channel (FC) switches, and storage controllers. Many storage vendors also support additional technologies within the array

such as ALUA, mirrored cache memory, and multiple paths to access disks. Redundant access paths will also maintain access to data from the ESXi host across the redundant fabric through a process referred to as *multipathing* or *multipath I/O (MPIO)*. This technology is native in vSphere and forms part of the hypervisor Pluggable Storage Architecture (PSA). However, third-party products, such as the EMC PowerPath might be recommended by some storage vendors to further enhance this native functionality.

- Other less common technologies for maintaining hardware availability might also provide good value depending on the hardware vendors chosen. For instance, technologies such as memory mirroring or the NEC Fault Tolerant or Stratus Server System, where the motherboard architecture makes these components redundant, might also be considered as part of the design process.

The following figure illustrates a redundant storage architecture through the removal of single points of failure, and outlines the logical storage design employed by the sample use cases.

Figure 10. Redundant Storage Architecture Logical Design



5.3 Blade Servers Compared with Rackmount Servers

The long-standing discussion regarding blade servers compared with rackmount server systems for use with cloud platforms has no definitive correct answer because every use case is different, and no one solution has superiority over the other.

The likelihood is that each service provider has their own preference when it comes to their cloud platform compute hardware. However, it is important to be able to work through the advantages and drawbacks of each option when it comes to server form factor.

Blades have the density advantage, and with rack space often provided at a significant cost, higher density is a good thing. Centralized management also provides a major advantage. A single interface where you can see and work with all of the blades, creating server profiles, adding networks, adding SAN connectivity, configuring VLANs, and managing power provides a serious advantage over rackmount

servers. The ability to perform all of these tasks, and many more, on hundreds of blades from a single pane of glass has many operational advantages.

On the downside, blades are expensive and require special power connections. You are also locked into a single vendor when you buy a chassis or enclosure. The initial setup of blade systems can be time consuming and require additional skills and experience. Their setup is often a manual process although once done, adding additional blades to an existing chassis is simple and is often extremely simple, cloning the configuration for an existing system.

Though blade system density is higher, there is often a maximum of two enclosures per cabinet because of the weight of a full enclosure. Two full enclosures can weigh in at over 900 pounds so it is necessary to spread the load across additional floor space as well as spread the power consumption across multiple 32 amp connectors.

References to rackmount systems typically describe 19-inch rack-mounted 1U, 2U, or 4U servers with their own power connections, network connections, local disks (possibly), and HBA cards.

Rackmount servers are independent, which means that you can insert them into any 19-inch server cabinet without the need for special wiring, power, or enclosure. Rackmount systems are often more affordable than their blade counterparts and typically come equipped with KVM access. Older rack systems can also generate much more heat and pull a great deal of power compared to modern blade systems. Therefore, rackmount systems can require much cooling and high volumes of airflow to keep their components at a working temperature, although these factors are improving all of the time.

So, when balancing the advantages and drawbacks of blades compared with rackmount servers, typically employ blade systems when:

- You need higher density and can support the weight.
- You have skilled, trained blade administrators.
- Centralized management provides operational benefits.
- Rapid and simple deployment of new compute nodes on the existing cloud platform is required.
- You want to continue with an established vendor relationship and are amenable to lock-ins.
- The numbers of servers required justifies the cost of the chassis.

Employ rackmount systems when:

- CapEx budget is tight.
- You do not want to be locked into a specific hardware vendor.
- You do not mind standard (“old school”) management procedures.
- You have sufficient cooling, power, and airflow to accommodate standard systems.

No service provider data center has just one compute form factor in it. Most large data centers have every kind of system imaginable, with systems from every vendor, often dating back decades, and as discussed earlier, there are advantages and disadvantages to both. There is no single correct answer or a single ultimate architecture for a VMware Cloud Provider Program cloud platform.

Whichever compute architecture type you choose, the aim is to create a building block approach for the platform that will enable both horizontal and vertical scaling of resources. Standardizing the configuration and using a consistent hardware platform helps to provide a manageable and supportable infrastructure by eliminating variability. Maintaining this consistent and standardized building block approach to compute resources:

- Simplifies capacity planning and scaling-out
- Simplifies automated installation, patching, and configuration of hosts
- Simplifies troubleshooting, fault finding, and configuration management

- Preserves VMware vSphere vMotion® compatibility across the entire platform
- Minimizes unused resources in VMware vSphere High Availability (vSphere HA) configurations

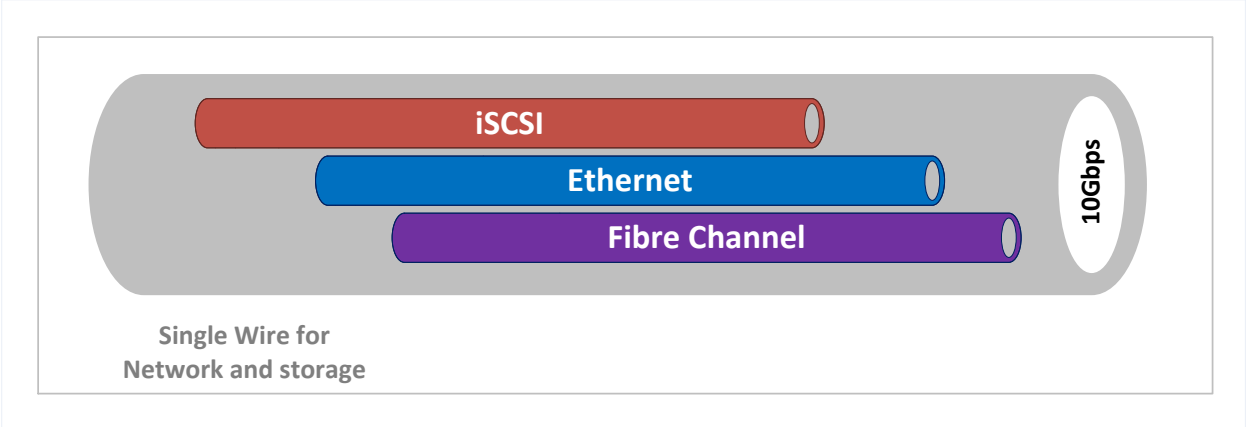
With the rapid adoption of next generation hyper-converged VMware vSAN based storage, a new dimension has been added to the blade versus rackmount debate. Due to their small form factor, blade systems do not typically provide adequate local SFF disk spaces to deliver sufficient capacity, and as such, make vSAN a viable option. For this reason, purchasing a new blade system for cloud platform projects typically removes the option of either deploying vSAN as part of the initial implementation, or later repurposing hardware to support vSAN distributed datastores as part of, or as a complete, storage solution.

In addition, if you consider the commodity-based hardware concept of the software-defined data center, the vendor tie-ins associated with blade systems are not typically conducive to this model.

5.4 Converged and Hyper-Converged Infrastructure

As 10-Gbps Ethernet has grown and has become a widely accepted standard for deploying data center networks, so has the concept of converged infrastructure. The converged infrastructure approach packages or integrates networking and storage technologies into an easily consumable, deployable, and manageable solution. Data Center Bridging (DCB), also sometimes referred to as Converged Enhanced Ethernet (ECC), merges traditional Fibre Channel and Ethernet networks. While traditional 10-Gbps Ethernet supports IP SAN technologies, such as iSCSI and NFS, Data Center Bridging also supports the Fibre Channel over Ethernet (FCoE) protocol over the same physical cabling infrastructure. With FCoE you get a unique combination of the functionality and reliability offered by Fibre Channel with the flexibility of integrated Ethernet.

Figure 11. Converged Infrastructure



Converged infrastructure is typically deployed and aligned with a specific hardware vendor's technology, such as Cisco UCS or HP BladeSystem. This converged approach aims to combine multiple types of resources into a single logical management entity by collapsing the traditional silos of IT infrastructure. This brings with it several distinct advantages, such as less cabling, reduced hardware stack, and improved flexibility and visibility from a single pane of glass.

Hyper-convergence takes this concept to the next level by grouping compute, storage, and networking into a single virtual computing appliance, with no need for a remote storage array. In the hyper-converged infrastructure, the storage controller function is running on each compute node in the vSphere cluster. The internal server disks on the ESXi hosts provide scalable-shared storage with cloud agility, efficiency, scalability and resilience. The VMware implementations of hyper-converged infrastructure include VMware EVO:RAIL™ and its larger relative system, VMware EVO™ SDDC™.

5.5 Compute Host Sizing – Scale-Up Compared with Scale-Out

When it comes to sizing hosts to meet the service provider's requirements for CPU, memory, and network throughput, there are two basic strategies:

- The "scale-up" approach (fewer hosts, but larger in capacity)
- The "scale-out" approach (greater in number, but lower in capacity)

The scalability of the hypervisor platform will determine the level of consolidation that can be achieved, as well as the underlying design of the compute layer. When examining host hardware choices, there are several key characteristics that influence the design:

- The number of physical CPUs (cores) and the virtual CPU-to-physical CPU ratio, which will impact the number of tenant virtual machine workloads.
- The amount of supported memory.
- The expandability, including the number and type of I/O cards for network and storage connectivity—the key factor in the amount of bandwidth available to a host.
- The number and size of storage devices and storage controllers, particularly relevant in a hyper-converged or vSAN architecture.
- The virtual machine consolidation ratio will be the definitive limit for host scaling.

A design strategy on the scale-out or scale-up approach will be made based on weighing the advantages and drawbacks of each option against the service provider's design requirements. Each strategy has its pros and cons as described in the following table.

Table 2. Scale-Up Compared with Scale-Out

Scale-Up Strategy	Scale-Out Strategy
• Uses a fewer number of servers, but each server has greater capacity (more cores more RAM, and so on) • Because of the greater capacity, each server will typically run more virtual machines • Typically means a higher virtual machine-to-core ratio • Also might mean more virtual machines will be affected by an issue with a single host • Typically uses rackmount servers • Might provide benefits in power/cooling density • Can also provide additional expandability	• Uses a greater number of servers, but each server is smaller in capacity • Due to fewer resources per server, the number of virtual machines on each server is usually fewer • The result is often lower virtual machine-to-core ratios • Fewer virtual machines are affected by an issue involving a single host • Can involve blade or rackmount servers. Blade servers can help in situations with limited physical space which might also potentially offer manageability benefits

Typically, this hardware choice is transparent to the consumers. From the perspective of the tenant's virtual machines, the host must provide enough virtual CPUs and virtual memory to facilitate the virtualization of the guest operating system and running applications.

If the design includes the use of the Sphere HA mechanism (which will almost certainly be the case, particularly for upper tier offerings), be sure to weigh the hosts' capabilities against the clusters' capabilities. For instance, because a limitation of 64 hosts per cluster exists in vSphere 6, the amount of resources available will be capped, although as discussed further in the cluster design section of this document, this is not a factor that will typically limit any design. The reason for this is that it is considered a good practice that standard compute configurations for the host design include the same blade or rackmount server type within each cluster. If possible, set up each cluster to span across chassis' or server cabinets to maximize availability. Therefore, very large clusters might not offer the levels of availability that a more determined building block approach to cluster design will achieve.

5.6 Determining Host CPU and Memory Requirements

There are two considerations for compute sizing—processing requirements and memory requirements. On a dynamic cloud platform, designing for empirical data with regard to CPU and memory requirements is unlikely to be possible. Instead, sizing will typically be based on the anticipated workloads that will run on the infrastructure.

Typically, CPU and memory requirements are defined during the project requirements analysis and specific metrics are based on anticipated workloads and expected growth for the lifecycle of the platform. From this information, the architect can determine the aggregate CPU and memory requirements. When designing an environment, look at current requirements and also design a solution that will allow the environment to grow without re-designing the platform every time you need to add capacity.

The processing capabilities of each compute node can be determined by multiplying the number of cores times the speed of the processors. For sizing purposes, we typically want to plan for no more than 80 percent utilization of the processing capabilities.

To fully utilize the processing capabilities that the compute nodes offer, systems must be configured with sufficient memory. In recent years, memory costs have been reduced and the memory density of hardware has increased. With this new extended memory capability of the servers, as much as 768 GB is available on a single half-size blade, and rackmount servers are able to offer in excess of 1 TB of memory (at time of this writing). The balancing point between CPU and memory capacity might not favor multi-core CPUs so strongly.

After the total memory requirements, established during the planning process, have been calculated, divide by the memory per host to determine the number of compute nodes that are required. For this calculation, you can usually ignore the overhead of the hypervisor, which is minimal when you consider the available memory and the efficiency of consolidation.

The final predicted values not only take into account the aggregate CPU and memory requirements, but also the service provider's desired maximum utilization thresholds, anticipated growth factors (as expressed by business stakeholders), and any expected benefit from virtualization technologies, such Transparent Page File Sharing (TPS), if enabled.

The use of TPS in sizing calculations highlights a relatively new design consideration now facing an architect. An important shift in the VMware policy on Transparent Page File Sharing that must be considered is that as of ESXi 5.5, 5.1, and 5.0 patches in Q4, 2014, TPS is disabled on these hosts, and on all future versions of the core hypervisor. For further details, see the VMware knowledge base article, *Additional Transparent Page Sharing management capabilities in SDXi 5.5, 5.1, and 5.0 patches in Q4, 2014 (2091682)* at http://kb.vmware.com/kb/2091682.

Also, consider that if TPS is enabled as part of the platform design, an important design note is that estimated savings from memory sharing must intentionally be kept low where the guest operating systems will be 64-bit and as such, large memory pages will be used. For more details, read VMware knowledge base articles, *Transparent Page Sharing (TPS) in hardware MMU systems (1021095*) at http://kb.vmware.com/kb/1021095 and *Use of large pages can cause memory to be fully allocated (1021896)* at http://kb.vmware.com/kb/1021896.

Finally, when sizing the platform, it is important to document all assumptions about how you calculated the service provider's requirements. Because these are estimations, in most cases, based on theoretical workloads, it is important that the business stakeholders accept the methodology that has been employed and understand the confidence level in the information that was used in assessing the compute platform requirements.

5.7 VMware Cloud Provider Program Compute Sizing Example

In this sizing example, the VMware Cloud Provider Program product owner has provided the following design requirements to the business stakeholders and architect based on commissioned market research.

The following table provides a summary of anticipated workload requirements for a new VMware Cloud Provider Program platform over three years. The numbers illustrate a sample design based on a mean calculation derived from small (35%), medium (30%), large (25%) and x-large (10%) virtual machines.

Table 3. Sample Design Scaling Requirements

Growth Metric	Values
Anticipated Total Number of VMs in Year 1	5000
Anticipated Total Number of VMs in Year 2 (140% Growth)	12,000
Anticipated Total Number of VMs in Year 3 (66.5% Growth)	20,000

Figure 12. Sample Design Scaling

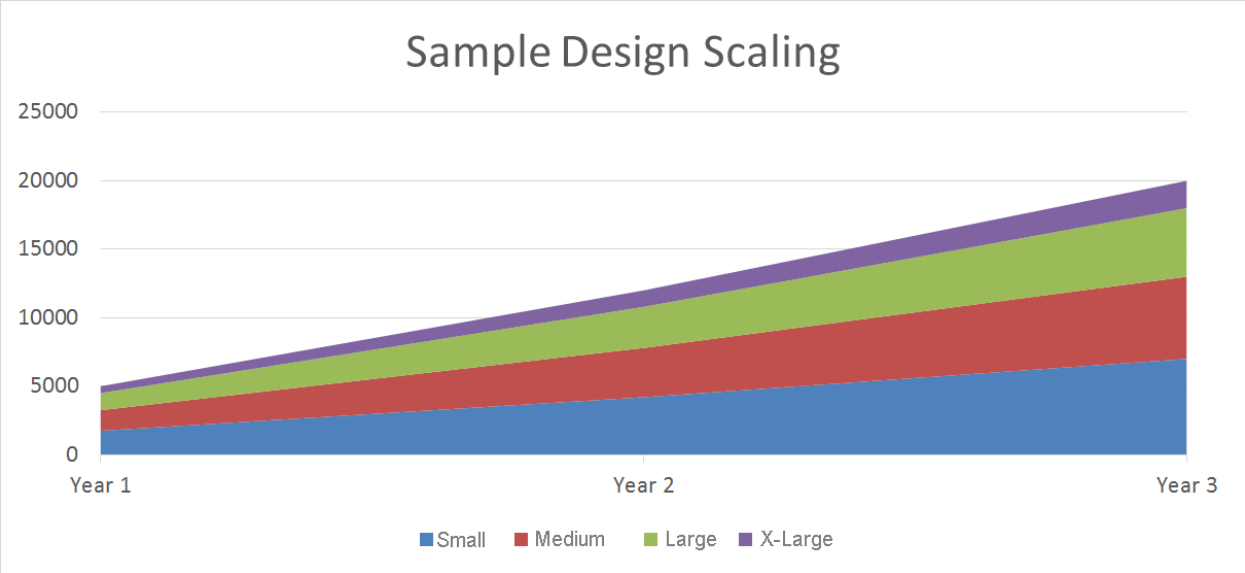


Table 4. Mean Virtual Machine Requirement Metrics

Performance Metric For T-Shirt Size Templates	Small (35% of 5000 VM Load)	Medium (30% of 5000 VM Load)	Large (25% of 5000 VM Load)	X-Large (10%) of 5000 VM Load)	Mean VM Resource
Projected average number of vCPUs per VM	1	2	4	8	2.75 vCPUs
Projected average utilization per vCPU	350 MHz	350 MHz	350 MHz	350 MHz	350 MHz
Projected peak vCPU utilization	600 MHz	600 MHz	600 MHz	600 MHz	600 MHz
Projected average vRAM per VM	2 GB	4 GB	8 GB	16 GB	7.5 GB
Projected average memory utilization per VM	60% (1.3 GB)	60% (2.5 GB)	60% (4.92 GB)	60% (9.83 GB)	60% (4.64 GB)
Projected peak memory utilization per VM	72% (1.5 GB)	72% (2.95 GB)	72% (5.9 GB)	72% (11.8 GB)	72% (5.5 GB)
Assumed memory-sharing benefit when enabled (TPS) (*)	10%	10%	10%	10%	10%
Projected average network utilization per VM	2.2 Mbps (3.8 Gbps)	4.2 Mbps (6.2 Gbps)	8.4 Mbps (10.25 Gbps)	16.2 Mbps (7.91 Gbps)	7.7 5Mbps (7.05 Gbps)
Projected peak network utilization per VM	6.0 Mbps (10.5 Gbps)	7.0 Mbps (10.5 Gbps)	12.0 Mbps (14.6 Gbps)	32 Mbps (15.6 Gbps)	9.20 Mbps (12.5Gbps)
Projected average VM I/O requirement	24 IOPS (42k)	48 IOPS (72k)	60 IOPS (75k)	120 IOPS (60k)	63 IOPS (62k)
Projected peak VM I/O requirement	48 IOPS (84k)	60 IOPS (90k)	100 IOPS (125k)	200 IOPS (100k)	102 IOPS (100k)

*TPS now disabled by default.

Table 5. Year 1, 2, and 3 Scaling Requirements

Performance Metric	Year 1 Required Resources	Year 2 Required Resources	Year 3 Required Resources
Total CPU resources for all virtual machines at peak	3,000 GHz	7,200 GHz	12,000 GHz
Projected Total RAM for all virtual machines at peak	27,500 GB (26.9 TB)	66,000 GB (64.5 TB)	110,000 GB (107.4 TB)
Total RAM for all virtual machines at peak (including TPS benefit memory sharing)	24,750 GB (24.17 TB)	59,400 GB (58 TB)	99,000 GB (96.7 TB)

Using this performance information provided by the cloud platform product manager, it is possible to derive the high-level CPU, memory, network bandwidth, and disk requirements that the platform must deliver to fulfill the design. The following table details the high-level specifications of the server hardware that is pertinent to this analysis as it has been selected by the service provider to deliver the compute resource to the tenant workload.

Table 6. Server Hardware Specification

Hardware Attribute	Specification
Hardware vendor	Vendor *X*
Form factor	1U Rackmount
Number of CPUs (sockets) per host	2
Number of cores per CPU (Intel)	Intel Xeon Processor E5-2687W (20M cache, 3.10 GHz, 8.00 GT/s Intel QPI) 8 Core 16 Threads
Hyperthreading	Enabled (16 logical cores per CPU)
MHz per CPU core	3.10 GHz
Total CPU GHz per CPU	24.8 GHz
Total CPU GHz per host	49.6 GHz
Proposed maximum host CPU utilization	80%
Available CPU MHz per host	39.68 GHz
Total RAM per host	512,000 GB
Proposed maximum host RAM utilization	80%

Hardware Attribute	Specification
Available RAM per host	409.6 GB
Number of Ethernet adaptor ports for network	2 x 10 GB
Installation destination	Boot from SAN (20 GB Boot LUN)
ESXi server version	ESXi 6.0 server. Build 2494585

In determining the compute node required, the service provider has compared CPU requirements, memory requirements and the hardware cost to establish the “sweet spot” for the chosen server type. For instance, as with the example shown in the following table, while you might be able to meet the memory requirements with 61 hosts, the number of hosts required to meet the CPU requirements is higher at 76. Therefore, you would be required to implement 76 hosts to meet the workload requirement.

Alternatively, depending on other design factors, you might be able to look at modifying your CPU choice or look at hardware that could facilitate higher memory capacity to achieve the required balance. For instance, if the current calculations for CPU were based on 8 cores per socket, would modifying the choice in favor of 12-core alternatives balance the CPU/memory requirements and maintain availability calculations while reducing costs?

Also, remember to allow for growth and the vSphere HA admission control policy. While 76 hosts cover the memory and CPU requirement for year one, to meet the service provider’s availability SLA with its consumers, the final server count for the design is likely to be 85 nodes.

Table 7. Compute Hardware Requirements

Type	Available per Host	Number of compute nodes Year 1	Number of compute nodes Year 2	Number of compute nodes Year 3
CPU Memory	39.68 GHz 409.6 GB	76 + 9 for HA = 85 /24 Nodes (4x24 Node Clusters)	182 + 23 for HA = 205 / 24 Nodes (9x24 Node Clusters)	303 + 39 for HA = 342 / 24 Nodes (15x24 Node Clusters)

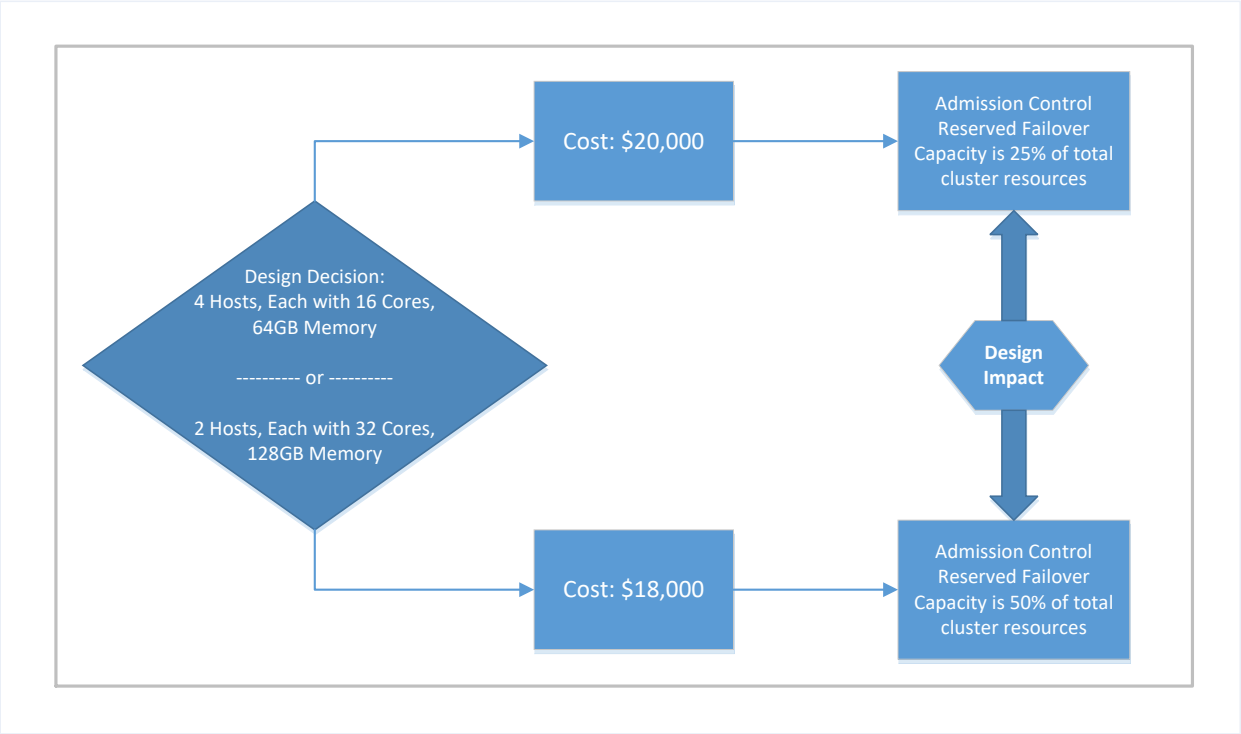
Other factors that can affect host sizing include:

- When workloads must to be split into two different data centers or availability zones, the vSphere HA N+*x* requirements and also the growth factor are increased.
- Lower utilization targets on CPU and memory resources.
- Licensing costs, which might be affected with more CPU sockets.
- Virtual CPU-to-host CPU ratios. Having a higher vCPU-to-pCPU ratio might mean you cannot meet the consumers SLAs.

Another consideration that is critical when evaluating vendor hardware for the implementation of host resources is ongoing hardware maintenance. Upgrading and patching can be time consuming for operational teams and is simpler with fewer vendors, as is maintaining firmware and the BIOS on the servers and other associated components.

Also, keep in mind the hardware cost. Two smaller servers can often be procured for a lower cost than a comparable larger server. An example of this type of design decision is highlighted in the following flowchart.

Figure 13. Design Decision Example



In this design decision example of a vSphere cluster that is required to support HA, the admission control policy must be configured with its default setting of allowing a single host failure (N+1). Each of the two options in the flowchart meets the customer's CPU and memory requirements for the workload that will utilize the clusters resources. However, as you can see, the two-node cluster might appear less expensive, although 50 percent of the total available resources are reserved by the admission control policy for failover capacity. While the four-node cluster option is more expensive, only 25 percent of the total available resources are reserved by admission control to provide appropriate failover capacity. Therefore, the likely design decision would be to scale out the solution with the four-node option to reduce the total reserved failover capacity and as such, lower the amount of resource that sits unused under normal operating conditions.

This example demonstrates a design decision that is based on sound, rational best practices. It is important that the architect involve the project stakeholders who have an understanding of the business goals in these types of design decisions, because these are the best people to help create a design that meets requirements and business goals. It is also important that all design decisions are documented and the rationale behind each decision is made clear to the project team.

5.8 Determining an Appropriate vCPU-to-pCPU Ratio

In a virtual machine, processors are referred to as virtual CPUs or vCPUs. When the vSphere administrator adds vCPUs to a virtual machine, each of those vCPUs is assigned to a physical CPU (pCPU), although the actual pCPU might not always be the same. There must always be enough pCPUs available to support the number of vCPUs assigned to a single virtual machine or the virtual machine will not boot.

However, one of the major advantages of vSphere virtualization is the ability to oversubscribe, so there is of course no 1:1 ratio between the number of vCPUs that can be assigned to virtual machines and the number of physical CPUs in the host. For vSphere 6.0, there is a maximum of 32 vCPUs per physical core, and vSphere administrators can allocate up to 4,096 vCPUs to virtual machines on a single host, although the actual achievable number of vCPUs per core depends on the workload and specifics of the hardware. For more information, see the latest version of the *VMware Performance Best Practices for VMware vSphere 5.5* at http://www.vmware.com/pdf/Perf_Best_Practices_vSphere5.5.pdf.

For every workload beyond a 1:1 vCPU to pCPU ratio to get processor time, the vSphere hypervisor must invoke processor scheduling to distribute processor time to virtual machines that need it. Therefore, if the vSphere administrator has created a 5:1 vCPU to pCPU ratio, each processor is supporting five vCPUs. The higher the ratio becomes, the higher the performance impact will be, because you have to account for the length of time that a virtual machine has to wait for physical processors to become available. The metric that is by far the most useful when looking at CPU oversubscription, and when determining how long virtual machines have to wait for processor time, is CPU Ready.

The vCPU-to-pCPU ratio to aim to achieve in your design depends upon the application you are virtualizing. In the absence of any empirical data, which is generally the case on a heterogeneous cloud platform, it is a good practice, through the use of templates and blueprints, to encourage your service consumers to start with a single vCPU and scale out when it is necessary. While multiple vCPUs are great for workloads that support parallelization, this is counterproductive in the case for applications that do not have built in multi-threaded structures. Therefore, while a virtual machine with 4 vCPUs will require the hypervisor to wait for 4 pCPUs to become available, on a particularly busy ESXi host with other virtual machines, this could take significantly longer than if the VM in question only had a single vCPU. This performance impact is further extended as the vSphere ESXi scheduling mechanism prefers to use the same vCPU-to-pCPU mapping to boost performance through CPU caching on the socket.

Service providers must, where possible, try to educate their consumers on provisioning virtual machines with the correct resources, rather than indiscriminately following physical server specifications that software vendors often refer to in their documentation. In the event that they have no explicate requirements, advise the consumer to start with a single vCPU if possible, and then scale up once they have the metric information on which to base an informed decision.

So that the vCPU-to-pCPU ratio is optimized and you are able to take full advantage of the benefits of over provisioning, in an ideal world you would first engage in dialog with the consumers and application owners to understand the application's workload prior to allocating virtual machine resources. However, in the world of shared platform and multitenant cloud computing, where this is unlikely to be the case, and the application workload will be unknown, it is critical to not overprovision virtual CPUs, and scale out only when it becomes necessary. Employing VMware vRealize Operations™ as a monitoring platform that can trend historical performance data and identify virtual machines with complex or mixed workloads is highly beneficial and its capacity planning functionality assists in determining when to add pCPUs. CPU Ready Time is the key metric to consider as well as CPU utilization. Correlating this with memory and network statistics, as well as SAN I/O and disk I/O metrics, enables the service provider to proactively avoid any

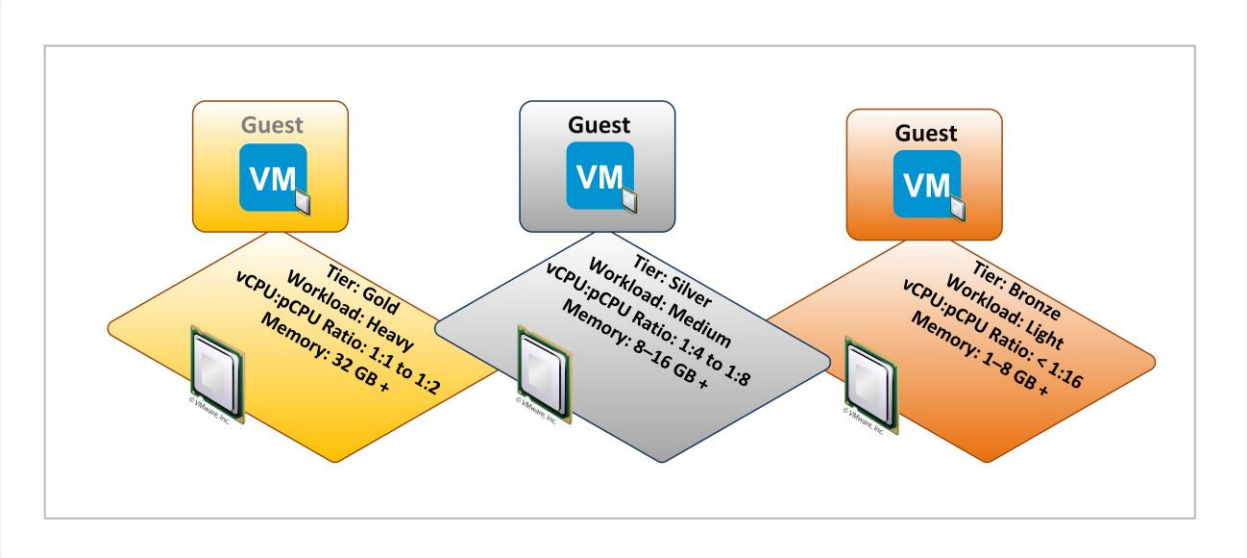
bottlenecks and correctly size the VMware Cloud Provider Program platform to avoid performance penalizing or overprovisioning.

The actual achievable ratio in a specific environment depends on a number of factors:

- The vSphere version – The vSphere CPU scheduler is always being improved. The newer the version of vSphere, the more consolidation is possible.
- Processor age – Newer processors are much more robust than older ones. With newer processors, service providers should be able to achieve higher vCPU:pCPU ratios.
- Workload type – Different kinds of workloads on the host platform will result in different ratios.
- As a guideline, the following vCPU:pCPU ratios can be considered a good starting point for a design:
  - 1:1 to 3:1 is not typically an issue
  - With 3:1 to 5:1, you might begin to see performance degradation
  - 6:1 or greater is often going to cause a significant problem for VM performance

The use of vCPU-to-pCPU ratios can also form part of a tiered service offering. For instance, offering lower CPU consolidation ratios on higher tiers of service or the reverse. Use the following figure as a starting point to calculate consolidation ratios in a VMware Cloud Provider Program design, but remember for every single rule there are exceptions and calculating specific requirements for your tenants is key to a successful architecture. This figure is for initial guidance only.

Figure 14. Virtual CPU-to-Physical CPU Ratio



As a general guideline, attempt to keep the CPU Ready metric at 5 percent or below. For most types of platforms, this is considered a good practice.

5.9 Performance Tuning with NUMA

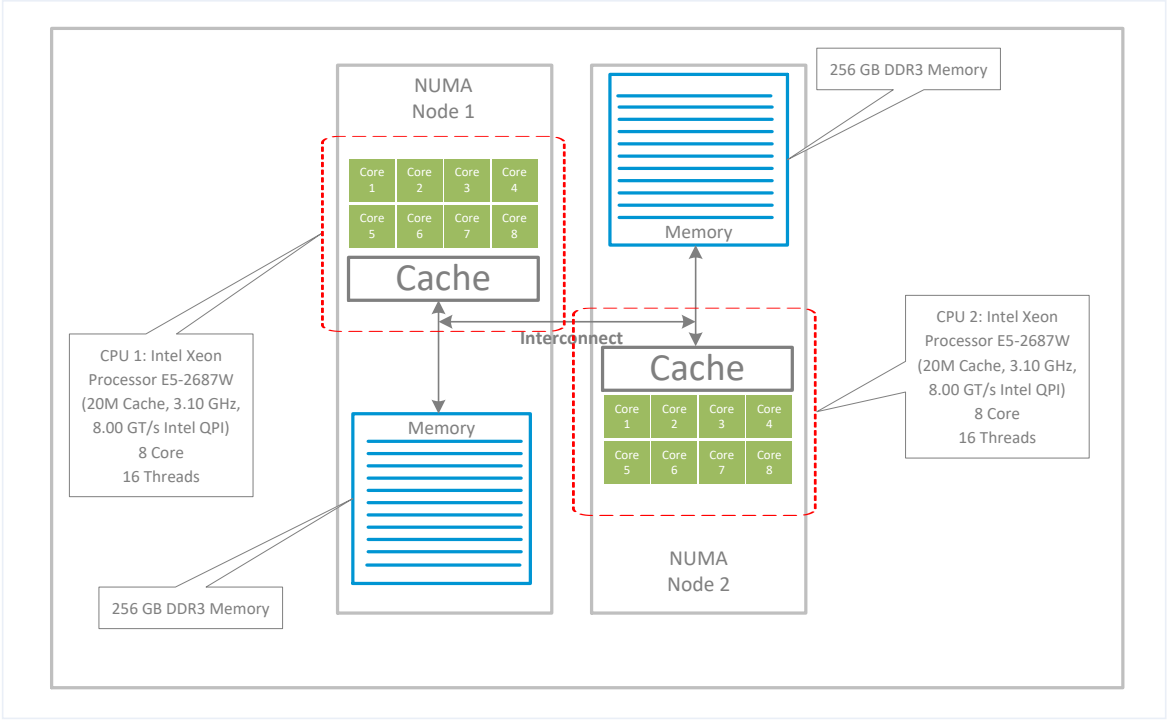
When sizing virtual machines for a service catalog, keep non-uniform memory access (NUMA) recommendations in mind to optimize platform performance.

Most modern servers have CPUs with directly attached memory. The CPU scheduler is aware of this physical architecture when it is available in the hardware, and targets processes to run on a CPU with fast access to the local memory as shown in the following figure. Because the process of accessing memory on a different CPU is not as efficient, the scheduler tries to keep processes on the same physical CPU to take advantage of the CPU cache and local memory (unless you manually override this functionality). Some NUMA systems provide the ability in the BIOS to disable NUMA by enabling node interleaving. Typically, you will get optimum performance by disabling node interleaving (that is, leaving NUMA enabled).

VMware recommends not assigning more vCPUs to a virtual machine than a physical CPU has cores. Consider this recommendation when designing your virtual machine service catalogs and virtual machine "t-shirt" sizing. Employing this recommendation as part of your service design means that the scheduler will not split vCPUs across multiple CPUs logical cores.

Likewise, in the service design, do not assign more memory to a virtual machine than is available to a single NUMA node. If necessary, check the server configuration to see how much memory each CPU can directly access. When under CPU contention, the scheduler might move vCPUs to other NUMA nodes, which will have a temporary performance impact.

Figure 15. NUMA Architecture



As the architect, examine the service design for high performance, oversized virtual machines and make the recommendation to stakeholders to stay within the size of a single physical NUMA node for vRAM and vCPU. This way local memory with the highest speed access is employed, with CPU and memory within the virtual machine being maintained from a single NUMA node, where possible.

5.10 vNUMA

If a virtual machine has more than eight vCPUs, virtual non-uniform memory access (vNUMA) is automatically enabled (although this behavior can be modified if required). Virtual NUMA is especially useful with large, high-performance virtual machines or multiple virtual machines within a VMware vSphere vApp or multi-machine blueprint. With vNUMA awareness, the virtual machine's memory and processing power is allocated based on the underlying NUMA topology, as outlined previously, even when it spans more than one physical NUMA node. vNUMA-aware virtual machines must use at least VMware virtual machine hardware version 8 and operate on vSphere 5.0 or later, with NUMA-enabled hardware.

As the ESXi hypervisor creates a NUMA topology that closely mirrors the underlying physical server topology, it allows the guest operating systems to intelligently access memory and processors in the single NUMA node. A virtual machine's vNUMA topology will mimic the topology of the host on which it starts. This topology does not adjust if the virtual machine migrates to a different host, which is one of the reasons why using consistent hardware building blocks is the recommended approach for cluster design.

In previous releases of vSphere, enabling the Hot Add feature on a virtual machine disabled vNUMA functionality. However, with the release of vSphere 6.0, this is no longer the case.

5.11 ESXi Host Server Advanced BIOS Settings

When configuring the compute node system BIOS, maintaining consistent settings that conform to the manufacturer’s recommendations is the starting point for any design.

The default hardware BIOS settings provided out-of-the-box on servers might not always be the optimal choice to maximize performance when the ESXi hypervisor is configured.

When designing the configuration of a new server farm, BIOS settings to consider include the following points (again, the manufacturer's recommended settings for ESXi is always be your starting point):

- Verify you are running the latest version of the BIOS available from the manufacturer for that system.
- Verify that the BIOS is configured to enable all populated processor sockets as active and to enable all cores in each socket.
- Enable “Turbo Boost” in the BIOS, if supported by the processors.
- Verify that hyperthreading is enabled in the BIOS for processors that support this technology.
- Verify that that any hardware-assisted virtualization features such as VT-x, AMD-V, EPT, and RVI, are enabled.
- Disable any devices you will not be using such as serial, USB, or network ports.
- If the BIOS allows for the memory-scrubbing rate to be configured, VMware recommends leaving it at the manufacturer’s default setting.

If VMware vSphere Fault Tolerance is employed in the design, disable CPU power-saving modes, such as Intel SpeedStep and AMD PowerNow!, which can reduce the CPU speed.

ESXi has the ability to manage the power capabilities of the servers in the software, and as such, reduce power consumption when the host is not being fully utilized. VMware recommends configuring the BIOS settings to allow ESXi the most flexibility in using the power management features offered by the hardware, and configure your power-management settings from within ESXi. While these features can provide a reduction in power consumption with little or no impact on performance, for workloads that are sensitive to I/O latency, overall performance could be impacted.

Other "C-states" deeper than C1/C1E (that is, C3 and C6) allow for further power savings, though with an increased chance of performance impact. Again, it depends on the design requirements as to whether or not these features are employed, but an ESXi host can take advantage of advanced processor

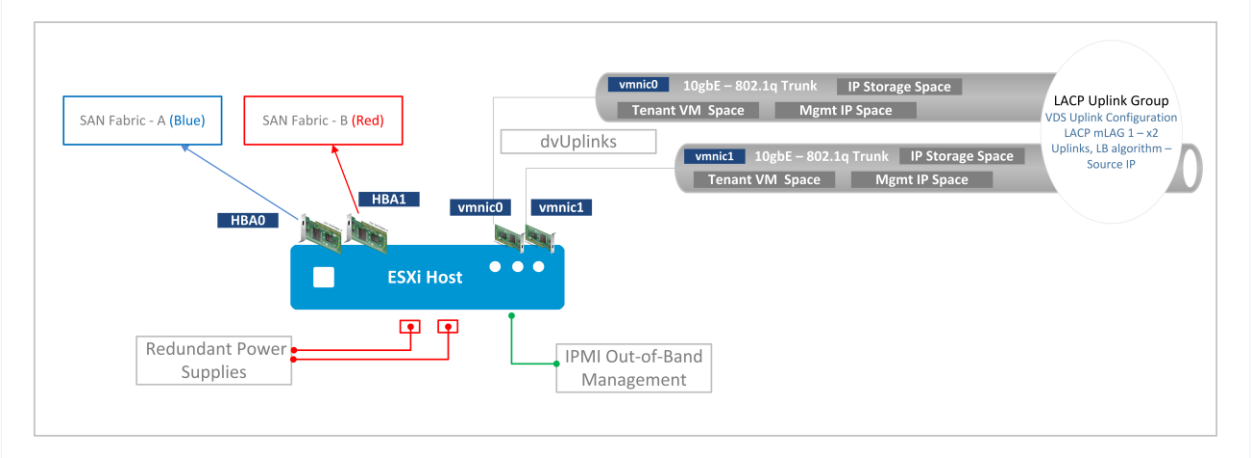
technologies such as Intel Enhanced SpeedStep or AMD PowerNow! to adapt the frequency of the physical CPUs based on the service provider's actual needs.

5.12 Host Connectivity

The way in which hosts are connected upstream for network and storage access can vary significantly, depending on hardware choices, legacy infrastructure, vendor constraints, cost, and protocols employed.

However, because a number of virtual machines are typically consolidated onto a single host, it is important that the bandwidth capabilities within the infrastructure not be overlooked. With smaller form factor commodity hardware, the number of I/O slots is typically limited, and those slots might have to provide storage connectivity in addition to IP network connectivity. These types of restrictions promote the use of 10-gigabit Ethernet interfaces or Converged Network Adaptors (CNAs), both of which will allow for the consolidation of multiple interfaces and data types into a single, or ideally dual, sizable bandwidth interface.

Figure 16. Host Connectivity



Regardless of the type of NIC, CNA or HBA employed, two ports of each type are typically required for redundancy. For Fibre Channel or FCoE connectivity, each interface is treated as a separate entity, with the load balancing being carried out using multipathing (MPIO) software. From an IP network interface controller (NIC) perspective, load balancing can be achieved through VMware vSphere Network I/O Control. In addition, the NICs can be configured in two different ways. In Active/Active mode, the interfaces are aggregated into a single port channel or used as separate entities. In Active/Passive mode, the passive interface waits for the active interface to fail before becoming active and transmitting and receiving data packets.

When designing and scaling host connectivity, in addition to virtual machine traffic, it is important to identify other system traffic types and their impact on the platform. IP storage, vSAN, and system-level traffic, such as heartbeats and vMotion will all affect the design.

Finally, if data is being replicated directly from the host, as is the model with VMware vSphere Replication™, verify that the design allows for sufficient bandwidth from the host's interface, as well as from the LAN and WAN perspective, to accommodate this data.

5.13 Single Hypervisor Compared with Mixed Hypervisor

The implementation of more than one hypervisor platform has become more common in the service provider industry in recent years. However, to host a VMware Cloud Provider Program service for business customers, the hypervisor platform must consist of a single native VMware offering. A single hypervisor environment can be managed through a single toolset, providing a global view of the entire

environment. A multi-hypervisor environment might require multiple tools to manage and administer systems.

A single hypervisor environment also allows the relocation of resources anywhere within the environment. For instance, unused hypervisors can be moved to different clusters depending on where the capacity is required, without having to redeploy the hypervisor binaries. This provides flexibility with resources and also allows for simpler short-term and long-term capacity planning by not having to analyze multiple environments independently.

The use of a single hypervisor approach simplifies business continuity and disaster recovery (BCDR) planning because not all hypervisors have the same options available. Having a multi-hypervisor approach in a BCDR scenario will complicate the storage and network configuration and limit recovery options. Also, consider that a single hypervisor approach will, in most cases, cost significantly less in terms of operational management.

Table 8. Advantages and Drawbacks of Multiple Hypervisor Platforms

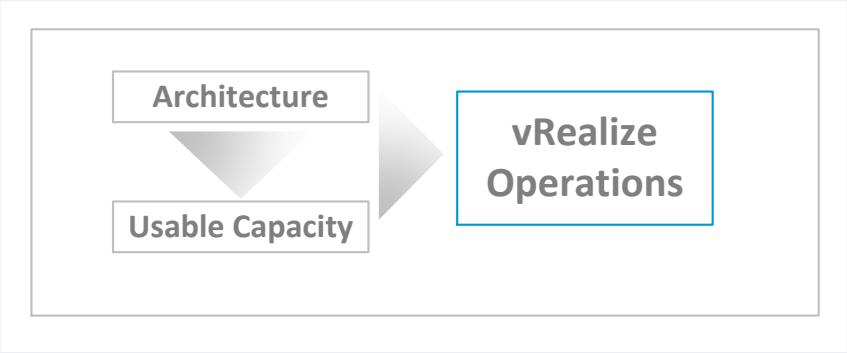
Advantages	Drawbacks
• Might realize optimal performance for some applications • Support for more operating systems • Interoperability with specific cloud providers	• Might require multiple different management tools and different operational processes • More complex capacity planning • Limitations in terms of allocating resources to different environments • Inhibits cross-environment migration • Multiple BCDR plans will be required • More complex capacity planning across multiple platforms

5.14 Capacity Management for vCloud Service Providers

Achieving true elasticity is a challenge that every service provider faces. Allowing tenants to scale up and scale down their infrastructure based on demand requirements takes carefully considered capacity planning, hardware contingency and customer trend metrics. The ability to forecast demand and either allocate or reserve resources of future needs is a key requirement for VMware Cloud Providers.

The first crucial factor in capacity management is to get the architecture correct—the wrong design makes management and operations unnecessarily complex.

Figure 17. Capacity Management



It is also important that the service provider is constantly aware of their usable capacity and lead-time for expansion of compute, storage, and network resources. The final step for capacity management is configuring VMware vRealize Operations™ accordingly, to address and alert, based on the following core questions:

1. What is our typical virtual machine profile?
2. How many more virtual machines can we support based on actual workload as opposed to theoretical workload?
3. When do we need to add capacity for compute, storage or network?

Planning Host Deployment

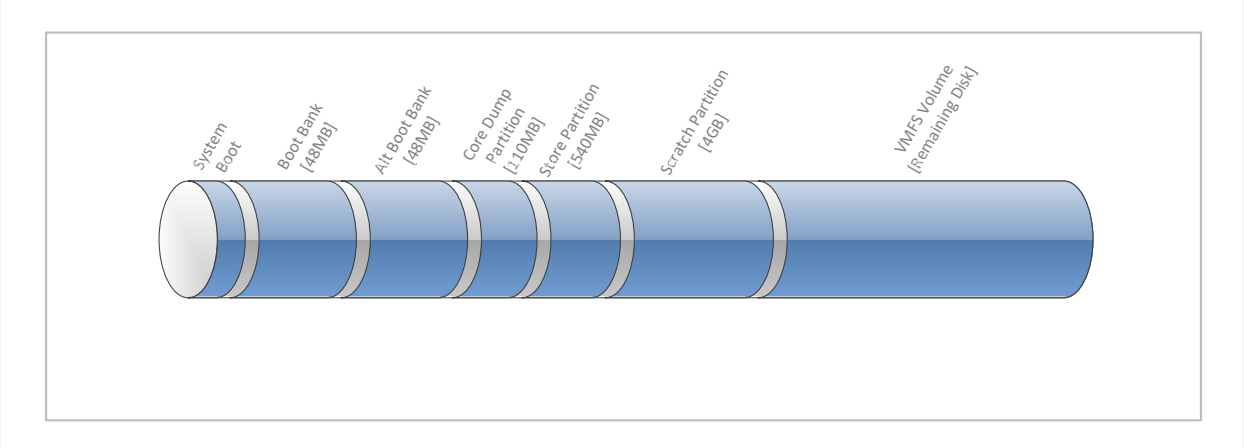
Deploying ESXi hypervisor hosts into the data center can be achieved in a number of different ways, which gives us a number of deployment scenarios to consider, depending on the hardware choice made. The first key decision is whether to opt for a traditional stateful installation of the ESXi hypervisor or to go stateless. The second pivotal decision to make is the destination media, where the hypervisor binaries will be installed, assuming that they are not being stored only in memory as part of a stateless implementation.

These options are discussed in later sections of this document. This section describes the structure and layout of the ESXi hypervisor so that an informed choice for the planning of host deployment can be made, and to determine how that decision will affect other virtual infrastructure components.

The way the hypervisor is deployed and the hardware it detects during deployment determines which partitions are created, and where items, such as logs and dump files, are stored. There is no way to manually define the partition layout during the ESXi installation procedure.

When ESXi is installed on to local hard disks (typically configured as a RAID 1 mirror), all possible partitions are created as shown in the following figure.

Figure 18. ESXi Boot Device Architecture



However, during an installation to either removal media (such as SD or a USB device) or a remote installation such as a SAN LUN (in a boot from SAN scenario), several of these partitions are absent post-installation.

For instance, the physical location where logs are written depends on the installation device used during the ESXi deployment. When the ESXi installation device is an SD card, USB key, or a remote boot LUN from an SAN device, a local scratch partition is not created on the installation media automatically during the deployment. Despite its size, ESXi 6.x always sees this type of installation as remote, and as such, logs are stored in RAM disk (a disk drive that is made up of a block of volatile memory) and therefore lost when the host is rebooted.

The reason for this is that USB/SD devices are sensitive to high I/O volumes, so the installer will not place the scratch partition on this type of device. During installation, the ESXi installer first scans for a local 4-GB VFAT partition. If it is unable to find one, it will then scan for a local VMware vSphere VMFS volume to use to create a scratch directory. If no local VFAT partition or VMFS volume is found, the last resort is to put the scratch partition in `/tmp/scratch` (a scratch location on the local RAM disk).

After this type of installation, you will see a warning on your ESXi hosts in vCenter Server indicating that their log files are stored on non-persistent storage (see the VMware Knowledge Base article *Syslog not configured messages on ESXi host console or in logs (1032460)* at http://kb.vmware.com/kb/1032460). Where this is the case, scratch space can be manually configured on the ESXi host using the VMware vSphere Client™, VMware vSphere Web Client, VMware vSphere PowerCLI™, or as part of a scripted installation procedure.

Because log messages that are stored on RAM disk are not retained after a reboot, troubleshooting information contained within the logs and core files is lost. In addition, if a persistent scratch location on the host is not configured properly, you might experience intermittent issues due to lack of space for temporary files, and the log files will not be updated. This can be problematic in low-memory hosts, but is not a critical issue for ESXi operation.

If the installation device is considered local during deployment, as illustrated previously, the ESXi host is not typically required to be manually configured with a scratch partition. The ESXi installer automatically creates a 4-GB Fat16 partition on the target device during the installation process, if there is sufficient space to do so. If persistent scratch space is configured, most of the logs are located on the scratch volume, and the `/var/log/` directory contains symlinks (symbolic links—a type of file that contains a reference to another file in the form of an absolute or relative path) to the persistent storage location.

6.1 Preparing for Host Deployment

The ESXi installer, unlike most other operating systems, copies the system image to the installation destination, and therefore does not actually install in the traditional sense. This makes the deployment process very fast, requiring little interaction from the deployment engineer. The source of the system image is typically provided to the server hardware through a CD/USB for local interactive deployment, through an ISO image for remote KVM deployment, or through a PXE boot from one of the server's network cards.

As of vSphere 5.1, in addition to stateless deployments offered by VMware vSphere Auto Deploy™, a stateful installation of the system image can also be initiated from a vSphere Auto Deploy server through PXE, which is covered in more in depth in the upcoming section on vSphere Auto Deploy.

An interactive installation, the default and most straightforward method for smaller deployments, is a simple routine where the deployment engineer responds to different options and provides the answers, either locally, or remotely through a remote console system, such as HP Integrated Lights-Out (iLO), Dell Remote Access Card (DRAC) or Cisco Integrated Management Controller (CIMC).

However, where the number of servers being deployed best suits a "light touch" approach, a scripted installation might offer several advantages. Employing the use of a kickstart script for the mass deployment of hosts or ongoing deployment of hosts, not only provides a means to fully automate the operating system deployment, but also:

- Provides a repeatable and consistent process for ESXi deployment
- Provides a standardized build, reducing operational overhead
- Speeds up the deployment process
- Provides additional post-deployment configuration options

The ESXi answer file (`ks.cfg`) mimics the syntax found in Red Had Linux kickstart script, and can provide a scalable and consistent approach to host deployment by being practically "hands-free" after the initial effort to set up, test, and customize is complete. Therefore, this type of build environment is ideally suited to large service providers who are potentially deploying hundreds of servers each week.

The next consideration required for host deployment is the destination of the system image itself. The ESXi hypervisor binaries can be deployed to several different locations. Traditionally, VMware ESX® or ESXi was deployed to local hard drives often configured as a RAID 1 mirrored pair for improved availability. However, in recent years and for various reasons, such as cost, power, and simplification, there has been a wider adoption of the use of internally mounted USB flash and SD cards. This trend is set to continue with the constant growth in the use of vSAN and VMware vSphere Virtual Volumes™. However, these locations are considered removable by the ESXi installer and are therefore subject to the caveats and considerations for log files discussed previously.

Another destination that is considered remote by the ESXi installer is a boot from SAN (BfSAN) LUN. A bootable SAN LUN can be configured over FC, FCoE, or iSCSI, and the hypervisor can boot from it directly through an appropriate initiator.

The following section addresses the advantages and drawbacks of each system image destination option to provide the service provider the information to make the most educated and up-to-date design decision.

6.2 Boot from Local Disk

Booting from local server disks is the most traditional mechanism for the ESXi system image. However, with more and more enterprise-level servers, such as the HP ProLiant BL490c (Intel) or BL495c (AMD), geared towards virtualization and so having to maximize physical memory space, they do not come with space for internal disks. For these server hardware types (typically blades), the hypervisor can either be booted from SAN, from a USB memory stick, or from an SD memory card.

Table 9. Advantages and Drawbacks of Boot from Local Disk

Advantages	Drawbacks
Simplicity – A known entity for data centers to adopt, with both HDD and SSD supported. **VM Swapping** – You can configure an ESXi host to use local storage for virtual machine swap files. This reduces load on the SAN and can improve performance under some workloads. Because these swap files are small in size, and temporary in nature, you can use small (70 GB to 150 GB) 15k RPM SAS drives to get good performance for a low price. **Flash Cache** – New in ESXi 5.5 was VMware vSphere Flash Read Cache™, which allows you to use SSDs local to the ESXi host to intelligently cache VM data. This reduces load on the SAN and improves performance for VMs. This is not cheap, but it can speed up some workloads significantly. While the same disks cannot be used for flash cache and as a boot device, investment in one could lead to investment in the other.	**Higher cost compute nodes** – Due to the need for locally installed disks. **Higher power consumption per compute node** – Due to increased power requirement for local disks. **More heat generated by server** – As a result of heat generated by spinning disks and additional power consumption. **Additional disk types to manage** – Additional standby hardware to be kept in stock. **Moving parts on compute nodes** – The meantime to failure of spinning disks is relatively low compared to other hardware types. Replacing disks might increase operational costs (HDD spindles only).

6.3 Boot from SAN

When you configure a host to boot from a SAN, the hypervisor's boot image is stored on a single LUN in the SAN attached storage system. When the host is powered on, it boots from the LUN through the SAN rather than from any local media. A boot from SAN environment can provide numerous benefits to the infrastructure, including providing a completely stateless compute environment. However, it can be complex to support. In addition, in certain use cases, do not use boot from SAN for ESXi hosts. For instance, where vSAN is being employed on the same hardware. Before you decide whether boot from SAN is appropriate for your environment, the advantages and drawbacks for an environment are outlined in the following table.

Table 10. Advantages and Drawbacks of Boot from SAN

Advantages	Drawbacks
Less power, less heat, less state – Removing internal hard drives from servers means they consume less power and generate less heat. That means they can be packed more densely, and the need for localized cooling is reduced. Without local storage, the servers effectively become "stateless" compute resources that can be pulled and replaced without having to worry about locally-stored data.	**Compatibility problems** – Some operating systems, system BIOS, and especially HBAs, might not support boot from SAN. Upgrading these components might change the economics in favor of local boot or vSphere Auto Deploy.
Less server CapEx – Boot from SAN enables organizations to purchase less expensive diskless servers. Further savings can be made through reduced storage controller costs, although servers still need bootable HBAs.	**Single point of failure** – If a server hard drive fails, the system will be unable to boot, but if a SAN or its fabric experience major problems, it is possible that *no* servers will be able to boot. Although the likelihood of this happening is relatively small because of the built-in redundancy in most SAN systems, it is nevertheless worth considering.
More efficient use of storage – Whatever the footprint of a server's operating system, it will always be over-provisioned in terms of internal storage to accommodate it. Using boot from SAN, the boot device can be configured to match the capacity it requires. That means a large number of servers running a range of operating systems can boot from a far smaller number of physical disks.	**Boot overload potential** – If a large number of servers try to boot at the same time—after a power failure, for example—this might overwhelm the fabric connection. In these circumstances, booting might be delayed or, if timeouts occur, some servers might fail to boot completely. This can be prevented by ensuring that boot LUNs are distributed across as many storage controllers as possible and that individual fabric connections are never loaded beyond vendor limits.
High availability – Spinning hard drives with moving internal components are disasters waiting to happen in terms of reliability, so removing reliance on internal hard drives guarantees higher server availability. The servers still rely on hard drives, but SAN storage arrays are much more robust and reliable, with far more redundancy built in so that servers can boot.	**Boot dependencies** – The SAN and array infrastructure must be operational to boot ESXi hosts. After a complete data center outage, these components must be started and be operational prior to restarting hosts.
Rapid disaster recovery – Data, including boot information, can easily be replicated from one SAN at a primary site to another SAN at a remote disaster recovery site. That means that in the event of a failure, servers are up and running at the remote site very rapidly.	**Configuration issues** – Diskless servers can easily be pulled and replaced, but their HBAs have to be configured to point to their SAN-based boot devices before they boot. Unexpected problems can occur if a hot-swappable HBA is replaced in a running server. Unless the HBA is configured for boot from SAN, the server will continue to run but fail to boot the next time it is restarted.

Advantages	Drawbacks
Lower OpEx though more centralized server management – Boot from SAN provides the opportunity for greatly simplified management of operating system patching and upgrades. For example, upgraded operating system images can be prepared and cloned on the SAN, and then individual servers can be stopped, directed to their new boot images, and rebooted, with very little down time. New hardware can also be brought up from SAN-based images without the need for any Ethernet networking requirements. LUNs can be cloned and used to test upgrades, service packs, and other patches or to troubleshoot applications.	**LUN presentation problems** – Depending on your hardware, you might find that some servers can only boot from SAN from a specific LUN number (LUN0). If that is the case, you must have a mechanism in place to present the unique LUN that you use to boot a given server as the LUN it (and other similar servers) expects to see. This is now considered a legacy issue that does not affect a new implementation.
Better performance – In some circumstances the rapidly spinning, high-performance disks in a SAN may provide better operating performance than is available on a lower performance local disk.	**Additional complexity** – There is no doubt that boot from SAN is far more complex than common local booting, and that adds an element of operational risk. As IT staff become accustomed to the procedure, however, this risk diminishes. However, do not discount the potential for problems in the early stage of boot from SAN adoption. For example, boot-from-SAN configurations require individual fabric zoning for each server and potentially a much more complex HBA/CNA configuration.
	Cost – SAN storage is typically more expensive than local storage, so any savings on server storage is lost on the extra SAN disks.
	Storage team overhead – A SAN LUN must be provisioned and managed for every server, which can create significant additional work for a storage team.
	Performance – Periods of heavy VMkernel disk swapping I/O can affect the virtual machine's disk performance, because they share the same disk I/O channels.
	Microsoft clustering – In vSphere 4, virtual machines configured with Microsoft Clustering (MSCS or failover clustering) are not supported on boot from SAN configurations.
	Scratch partitions –ESXi does not automatically create scratch partitions in a boot from SAN environment because it sees the disks as remote. The creation of scratch partitions can be easily configured manually or scripted.

6.4 Boot from Removable Media

The boot from removable media option is gaining momentum, particularly with the VMware introduction of vSAN with vSphere 5.5 U1, allowing maximum disk slots for virtual machine datastore capacity. Before you decide whether the removable media option is suitable for your environment, consider the advantages and drawbacks listed in the following table.

Table 11. Advantages and Drawbacks of Removable Media Deployment

Advantages	Drawbacks
Simplicity – The simplest option available whether a RAID1 mirror or USB/SD solution is used. **Lower cost compute nodes** – Boot from removable media allows organizations to purchase less expensive diskless servers. Servers are potentially lower cost because no local disks or RAID card is required. **Lower power consumption per compute node** – Due to decreased power requirement for local disks. **Less heat generated by server** – As a result of less moving parts and lower power consumption. **No moving parts on host** – Having no moving parts on the server results in lower mean times to failure for the hardware, reducing operational costs. **vSAN** – In a typical vSAN environment, hosts are configured to boot from USB, SD, or other non-persistent storage to maximize the number of persistent disk slots available on the hardware, thereby increasing the potential for virtual machine storage.	**Scratch partitions** – ESXi never creates scratch partitions on USB flash drives or SD cards, even if they have the capacity, because the potentially heavy disk I/O from the userworld swap could damage them. **Media quality** – The design must specify industrial grade SD cards or USB devices for the ESXi hypervisor installation (redundantly configured in RAID1 if supported by the hardware). For more VMware recommendations when booting hosts from non-persistent storage, see the *VMware vSphere Installation and Setup* document at https://pubs.vmware.com/vsphere-60/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-60-installation-setup-guide.pdf.

6.5 vSphere Auto Deploy

The vSphere Auto Deploy network boot mechanism enables quick provisioning and configuration of ESXi hosts. Essentially, instead of managing the state of each host separately, with vSphere Auto Deploy you put in place centralized configurations and rules that map configurations to sets of ESXi hosts, ensuring that every time a host boots, its gets the same required state. When you boot ESXi hosts using vSphere Auto Deploy, the host server loads the ESXi image directly into memory and therefore does not have to store the ESXi state in any type of persistent storage. Managing hosts in this type of centralized manner allows you to easily scale out the management of significantly more physical servers and have a more deterministic environment, simplifying day-to-day operational tasks.

When you start a physical host that is configured to use it, vSphere Auto Deploy employs a PXE boot infrastructure in conjunction with vSphere host profiles to provision and customize that host. No state information is typically stored on the hypervisor itself. Instead, the vSphere Auto Deploy server manages the state information for each host individually.

Introduced in vSphere 5.1, the stateless caching feature allows the host to cache its state information by forcing the hypervisor to store the ESXi image and configuration on a local disk, a remote disk, or a USB drive. Subsequent boots, when the stateless caching feature is enabled, continue to provision the host

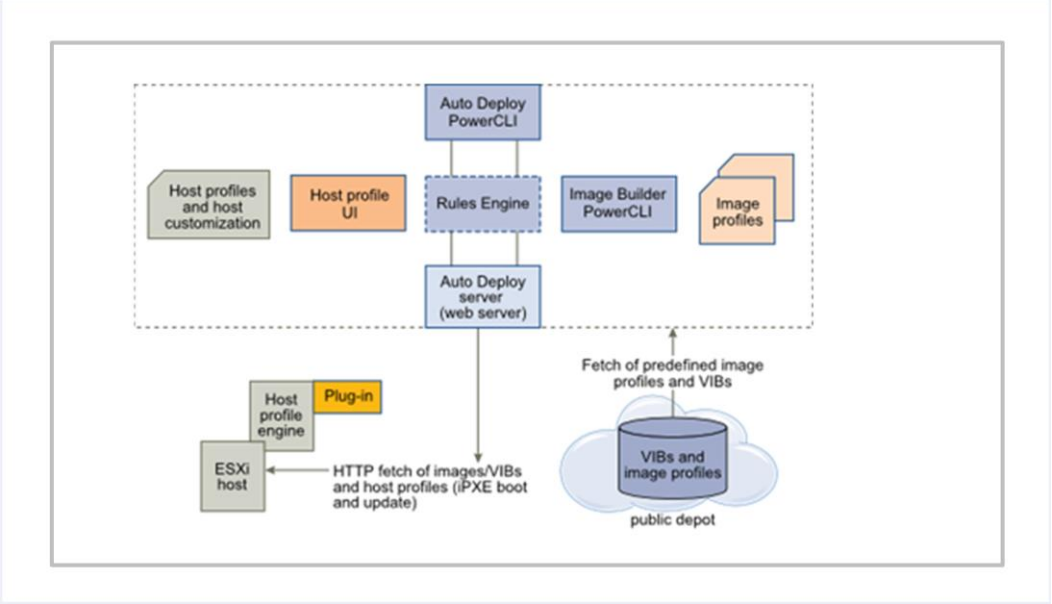
through the vSphere Auto Deploy server. However, if the vSphere Auto Deploy server is not available, the host boots from cached image.

In addition to stateless caching, vSphere 5.1 introduced the ability to use vSphere Auto Deploy to perform stateful installations. In some designs, a vSphere Auto Deploy infrastructure can be employed to provision hosts that will perform all subsequent boots from its local persistent storage. This mechanism allows vSphere Auto Deploy to provision an ESXi host and configure a host profile that forces the host to store the ESXi image and configuration on the local disk, a remote disk, or a USB drive in the way described previously. In this type of use case, all subsequent boots of the ESXi host take place from the local image with no further interaction with the vSphere Auto Deploy infrastructure. This process is similar to performing a scripted installation and not dissimilar to a kickstart scripted installation, where the script provisions the host and the host then boots from disk. However, in this case, vSphere Auto Deploy provisions the host and the host then boots from its own media. This feature has meant, to some extent, that other community-driven deployment appliances, such as the ESX Deployment Appliance (EDA) and the Ultimate Deployment Appliance (UDA), have lost momentum.

6.5.1 vSphere Auto Deploy Architecture

The vSphere Auto Deploy architecture can be broken down into the components illustrated in the following figure.

Figure 19. Auto Deploy Components



vSphere Auto Deploy components include:

- vSphere Auto Deploy server – This component serves images and host profiles to ESXi hosts. The vSphere Auto Deploy server is at the heart of the vSphere Auto Deploy infrastructure.
- Auto Deploy rules engine – This component tells the vSphere Auto Deploy server which image profile and which host profile to assign to a particular host. Administrators use the vSphere PowerCLI to define the rules that assign image profiles and host profiles to hosts. These rules are organized into the following categories:
  - Active rule set – Ready to be deployed to physical hosts.
  - Working rules set – Rule testing that takes place prior to deployment.

- Image profiles – Define the set of VIBs to boot ESXi hosts with:
  - Default images – VMware and VMware partners make image profiles and VIBs available in public depots. Use the VMware vSphere ESXi Image Builder CLI to examine the depot, and the vSphere Auto Deploy rules engine to specify which image profile to assign to which host.
  - Custom images – VMware customers can create a custom image profile based on the public image profiles and VIBs in the depot, and apply that image profile to the host.
- Host profiles – Maintained by the vCenter Server and defined by machine-specific configurations, such as networking or storage setup. Administrators create host profiles by using the host profile user interface with the vSphere Web Client. You can create a host profile for a reference host and apply that host profile to other hosts in your environment for a consistent configuration.
- Host customization – Referred to as an answer file in previous releases of vSphere Auto Deploy. Stores information that the user provides when a host profile is applied to the host. Host customization might contain an IP address or other unique information.

The use of vSphere Auto Deploy in the service provider data center must be considered carefully because a number of other design decisions regarding infrastructure and hardware are dependent on it. These design decisions might differ significantly if vSphere Auto Deploy is adopted. As part of this design review, consider the advantages and drawbacks described in the following table.

Table 12. Advantages and Drawbacks of vSphere Auto Deploy

Advantages	Drawbacks
Patch management – Provides ease of deployment and patch management of large scale environments: • Ability to re-deploy new versions of ESXi to a large number of ESXi hosts. Increases refresh agility when managing many ESXi hosts. • Uses stateless ESXi hosts, memory is resident, and there is no need to have dedicated hard disks/volumes to run ESXi hosts. (Although some sort of storage is required for stateless caching.)	**Complexity** – Additional complexity when deploying to small number of servers.

Advantages	Drawbacks
Eliminate configuration drift – By sharing a standard ESXi image profile across multiple hosts, all ESXi hosts are running the same ESXi version. In addition, each time a host is rebooted, it is like performing a fresh install of ESXi.	**Management** – Managed using vSphere PowerCLI, which might be less familiar to the customer (although a fling is available). • Heavily dependent on knowledge of vCenter Server host profiles and PowerCLI. • Heavily dependent on vCenter Server, which poses recoverability issues in the event of a site failure. (Use of vCenter Server heartbeat is supported.) • If vCenter Server is virtualized, the vCenter Server cluster of ESXi hosts must be deployed using installable ESXi. This cluster must be installed, patched, and updated differently from the vSphere Auto Deploy hosts. VMware recommends that vSphere Auto Deploy is not used when provisioning ESXi hosts in the case where a management cluster will be used to host the vCenter Server. The management cluster running vCenter Server must be configured with stateful booting. • VMware recommends installing vSphere Auto Deploy on a dedicated server. For each vCenter Server, a dedicated vSphere Auto Deploy server is required.

Advantages	Drawbacks
Reduced storage costs – Because vSphere Auto Deploy installs directly into the host's memory, there is no need to dedicate a boot disk for each server. This not only saves money when purchasing new hardware and storage, but when booting from SAN, it helps to simplify the storage architecture by eliminating the need to map dedicated LUNs to each ESXi host.	**Additional Infrastructure** – vSphere Auto Deploy introduces several additional "mission critical" applications to the infrastructure management stack, such as TFTP, DHCP, PXE and vSphere PowerCLI. Issues with blade or access switching infrastructure might cause problems in a disaster recovery scenario after powering off and powering on the host. vSphere Auto Deploy might pose a single point of failure if the image profile is incorrectly configured and the PXE and DHCP architecture is unavailable. In this case, the servers would not be able to boot. (The vSphere Auto Deploy server can be protected with vCenter Server heartbeat.) Two options available in vSphere Auto Deploy can be configured to mitigate against this risk: • **Stateless caching** – Caches the image locally on the ESXi host, ready for an event where the PXE, DHCP, or vCenter Server architecture is not available. • **Stateful Install** – Installs the ESXi image in a stateful manner on the local disks, removable media or SAN LUN that will act in the same way as if it were installed manually. Both options negate some of the benefits of vSphere Auto Deploy by having to maintain local storage for cache/installation, but they reduce risk to the architecture. vSphere Auto Deploy uses the PXE network boot feature of an ESXi host and is fully dependent on an operational DHCP and a TFTP server. Additional DHCP scope options are required when configuring vSphere Auto Deploy: • 066 Boot Server Host Name (address of the TFTP server) • 067 Boot File Name (iPXE binary will be used to boot the ESXi hosts) vSphere Auto Deploy uses a PXE boot infrastructure that supports only IPv4. You can use vSphere Auto Deploy in a mixed IPv4/IPv6 environment or an IPv4-only environment, but not in an IPv6-only environment.

Advantages	Drawbacks
Fast server provisioning – Deploying a new ESXi host is as simple as enabling PXE boot and powering on a new server. The vSphere Auto Deploy server identifies the new server, assigns an appropriate ESXi image profile and host profile, and places the server in the correct vCenter Server folder or cluster. Consider the following design points: • The vCenter Server that provisions vSphere Auto Deploy hosts must not be dependent on vSphere Auto Deploy. • Keeping the management cluster separate from the vSphere Auto Deploy infrastructure facilitates role-based access control. • It is easy to grow the environment if additional instances of vCenter Server and the vSphere Auto Deploy server are needed.	vSAN is not supported on a stateless host.
vSphere Auto Deploy stateful deployment – vSphere Auto Deploy can also be used to provision an ESXi host and set up a host profile that causes the host to store the ESXi image and configuration on the local disk, a remote disk (boot from SAN), or a USB drive. Subsequently, the ESXi host boots from this local image. vSphere Auto Deploy no longer provisions the host. This process is similar to performing a scripted installation. With a scripted installation, the script provisions a host and the host then boots from disk. In this case, vSphere Auto Deploy provisions a host and the host then boots from disk.	

6.5.2 vSphere Auto Deploy Server Security Considerations

For vSphere Auto Deploy, secure your network as you would for any other PXE-based deployment method. vSphere Auto Deploy transfers data over SSL to prevent casual interference and snooping. However, the authenticity of the client or of the vSphere Auto Deploy server is not checked during a PXE boot.

The boot image that the vSphere Auto Deploy server downloads to a machine can have the following components:

- The VIB packages that the image profile consists of are always included in the boot image.
- The host profile and host customization are included in the boot image if vSphere Auto Deploy rules are set up to provision the host with a host profile or a host customization setting.
    - The administrator (root) password and user passwords that are included with host profile and host customization are MD5-encrypted.

  - Any other passwords associated with profiles are in the clear. If hosts have been configured on an Active Directory domain, the passwords are not protected, and therefore the vSphere Authentication Proxy must be employed to avoid exposing Active Directory passwords being stored in plain text in the host profiles.

- The host's public and private SSL key and certificate are included in the boot image.
- As mentioned previously, reduce the security risk of vSphere Auto Deploy by completely isolating the network where it is used.

6.5.3 Stateful Compared with Stateless Compute

The concept of a stateless ESXi host was introduced with vSphere 5 along with vSphere Auto Deploy. A stateless host would not under normal operating conditions boot from a disk or other media. Stateless hosts take advantage of the ESXi small memory-resident footprint to boot the system image over the network on every startup or restart. In this context, the stateless host must leverage vSphere Auto Deploy as its deployment mechanism. However, this is the not only option to achieve stateless compute within the ESXi platform. For instance, the Cisco UCS Blade system, when employed with boot from SAN, takes a very different approach to stateless hosts.

Cisco UCS abstracts the logical server from the physical hardware. With this concept of “stateless computing,” each compute node’s hardware does not have a set configuration. MAC addresses, UUIDs, firmware, and BIOS settings for instance, are all configured within UCS manager and stored as part of a service profile, and then applied to the server hardware. This allows for consistent configuration and ease of repurposing of physical equipment. A new service profile can be applied to a physical blade server within a matter of minutes, and includes the following key elements:

- LAN addressing (MAC addresses)
- SAN addressing (WWPN and WWNN addresses)
- Blade firmware versions
- Boot order
- Network VLANs
- Physical port configuration (Ethernet vNICs and vHBAs)
- Quality of service (QoS) policy
- BIOS versions and parameters

The Cisco service profile concept allows for a significant amount of flexibility and control. You can disassociate the service profile from one server and then associate it with a different blade. The burned-in settings on the new blade, such as UUID and MAC address, are overwritten with the configuration in the service profile. As a result, the change to the server is transparent to your network and the boot image. You do not need to reconfigure any component or application on your network to begin using the new blade.

The concept of abstracting the hardware away from the system image remains the same in both vSphere Auto Deploy and Cisco UCS, but the mechanism of achieving “stateless compute” differs significantly in each architecture.

A stateful installation of the hypervisor boots from a copied installation of the system image on local disk or other stateful media. In a vSphere Auto Deploy architecture, a stateful installation is carried out the first time the server is booted and receives the image. That system image is then copied to the server’s local disk, removable media, or SAN LUN, and the host boots from that installation on all subsequent reboots, not the vSphere Auto Deploy infrastructure. This is typically achieved by modifying the server’s BIOS so that the first boot attempt was from local media and the second a PXE boot. The patching of stateful hosts must be carried out using conventional methods such as VMware vSphere Update Manager™ with the continued risk of configuration drift within the environment.

6.6 Customizing ESXi Images with Image Builder

Because VMware provided images can quickly become out-of-date and because not all hardware drivers are included, VMware allows you to create your own custom ESXi images for all types of deployment methods. VMware customers can create custom images by slipstreaming the latest hardware drivers, VMware patches, hardware CIM providers or third-party plug-ins, solutions or agents. To create customized images, use the vSphere ESXi Image Builder CLI to provide a set of commandlets that package together vSphere installation bundles (VIBs) into new system images. These custom system images can be configured as *depot* packages for a vSphere Auto Deploy mechanism, or saved as ISO images for deployment either through a remote KVM or burned onto a recordable CD for manual installation.

Traditionally, hardware vendors such as HP, Cisco, and Dell have provided pre-packed vendor-specific images that include all component-specific drivers for newly released hardware, hardware CIM providers (Common Information Model providers allow management functions, such as reporting health monitoring information), and vendor-specific tools. In some cases, the use of these vendor-specific images might negate the requirement to build your own service provider custom images.

6.7 Impact of vSAN

There are two specific areas where the use of vSAN can affect the ESXi system image deployment mechanism chosen by a service provider:

- vSAN is not supported on stateless hosts.
- Persistent disks in the host server that are employed by the hardware as a boot device cannot be used by vSAN and the other way around.

In a typical vSAN environment, you configure hosts to boot from USB, SD, or other non-persistent storage in order to maximize the number of persistent disk slots available for virtual machine storage. This approach allows you to take full advantage of your vSAN distributed datastore capacity, or at least provide you with the option of increasing vSAN capacity in the future.

6.7.1 Post-Installation Design

Post-deployment configuration of hosts is an important aspect of any vSphere design. Most, if not all, aspects of post deployment configuration, can be scripted as part of a kickstart deployment, configured through vSphere Auto Deploy host customization or vSphere host profiles.

6.7.2 Host Name and IP Address

Each host requires a unique host name and IP address. During an interactive installation, this is typically manually configured by an engineer at the customer's data center. If vSphere Auto Deploy or a scripted installation is being used, these details are provided during the installation process or by a mechanism of DHCP with reserved IP addresses. The use of DHCP without a specific MAC-to IP-address reservation is not recommended.

6.7.3 DNS

Domain Name Service (DNS) must be configured on all hosts. The configured DNS servers must be able to resolve both short names and fully qualified domain names (FQDNs) using forward and reverse lookups.

6.7.4 NTP

A Network Time Protocol (NTP) server provides a precise time source (radio clock or atomic clock) to synchronize the system clocks of all devices within the infrastructure. NTP is transported over User Datagram Protocol (UDP/IP) and all NTP communications use Universal Time Coordinated (UTC) time.

An NTP server receives its time from an upstream reference time source, such as a radio clock or atomic clock. The NTP server then distributes this time across the network.

A consistent NTP configuration is highly recommended because it synchronizes all the clocks in each component in the infrastructure and eases the tasks of reconciling logs, debugging, and information tracing. NTP must be configured on each host and share the same time source as the vCenter Server, VMware Platform Services Controller™, and all other management and production components.

6.7.5 Additional Network Configuration

Additional network configuration is likely to be necessary either before or after the connection to vCenter Server is made. For instance, configuration of a management VLAN, a VMkernel port for a secondary management address, additional VMkernel ports for vSAN, NFS storage, or third-party distributed virtual switches might also be required. As previously highlighted, in an interactive installation, these configurations are likely to be configured either manually or a through a post-deployment script. However, during an automated deployment, it is more likely that they are configured either through a post-installation script or through vSphere host profiles.

6.7.6 Host Certificate

Each host creates its own self-signed certificate post-installation based on the FQDN of the host. In most environments, it is considered a good practice to replace these X.509 v3 base 64 encoded SSL certificates with a certificate from a trusted certificate authority before joining the host to the vCenter Server. New options and improvements in host certificate management are described in the section on host security.

6.7.7 Core Dump Collector

A core dump is a copy of the state of working memory at point of failure. By default, a core dump is saved to the vmkcore (type 0xFC) partition on disk in the event of host failure. If available, the core dump from the VMkernel includes everything seen on the physical console screen when a purple screen of death (PSOD) occurs. A core dump can expedite the resolution of hardware issues, but the analysis can be performed only by a member of VMware technical support staff.

The VMware vSphere ESXi Dump Collector allows you to keep core dumps on a centralized network server for use during the debugging process and, if necessary, for long-term retention. The vSphere ESXi Dump Collector is especially useful in a vSphere Auto Deploy environment (where local disks might not exist), but is supported for all ESXi host deployment methods. The ESXi hosts must be configured to send core dumps to the appropriate network location, which is typically located in the management ecosystem.

6.7.8 Licensing

Each host is required to have a license key assigned despite the licensing mechanism typically being the VMware Cloud Provider Program (formally called VMware Service Provider Program or VSPP). Licensing keys are typically managed through vCenter Server, although they can also be locally assigned. If no appropriate product key is available immediately at time of deployment, a host will run in evaluation mode for 60 days prior to management capability being dropped from the vCenter Server.

6.7.9 Scratch Partition

Remember to redirect the scratch partition, if applicable. In a design that employs SD/USB or boot from SAN as the installation destination, the host installer does not allow for the creation of a scratch partition during the initial setup process. As part of a manual or scripted post-deployment configuration, configure a local `.locker` directory on a shared datastore for each host to act as a scratch partition. In addition, configure each host to log into the remote syslog system to set up host log file availability after a reboot. For more information on VMware recommendations when booting hosts from non-persistent storage, see

the *VMware vSphere Installation and Setup* document at http://pubs.vmware.com/vsphere-60/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-60-installation-setup-guide.pdf.

6.7.10 Remote Logging Configuration

VMware recommends that you configure logging to an external syslog service from all hardware, including ESXi hosts, physical servers, and network components, because the centralization of logs increases administration and security investigation capabilities. By configuring hosts to use a central logging server, aggregate analysis and searches become possible, providing visibility into events that affect multiple hosts.

VMware vRealize Log Insight™ provides a much more comprehensive solution for syslog than the VMware vSphere Syslog Collector or VMware vSphere Management Assistant. vRealize Log Insight gives administrators the ability to consolidate logs, monitor, and troubleshoot vSphere, and perform security auditing and compliance testing. This scalable virtual appliance includes a syslog server, log consolidation tool, and a log analysis tool that works for any type of device that can send syslog data. vRealize Log Insight administrators can also create custom dashboards based on saved queries, which can then be exported, shared, and integrated into VMware vRealize Operations Manager™.

In a multisite architecture, every device for which you would like to collect events is typically configured to send events to a syslog aggregator, and the syslog aggregator is configured to forward events to one or more vRealize Log Insight instances. For more information on designing a single or multisite syslog architecture, refer to the *VMware vCloud Architecture Toolkit™ for Service Providers VMware vRealize Log Insight Architecture for Service Providers* document.

6.7.11 SNMP Configuration

In addition to centralized logging and the CIM agentless monitoring visible from with the vSphere Web Client, ESXi can be configured to use SNMP to send monitoring traps, potentially highlighting problems with hardware, to an SNMP manager. Since the release of vSphere 5.1, ESXi has also supported the use of SNMPv3. To enable and configure SNMP on the host requires the use of the `esxcli system snmp` command.

6.7.12 Local User Permissions

As part of your security and access procedures, you might be required to create additional local accounts on each host. The mechanism for doing so has improved dramatically in vSphere 6.0 and is addressed in detail in the section on host security.

6.7.13 Active Directory Integration

Integrating ESXi hosts with a Microsoft Active Directory provides a secure and simple way to manage local host access. This is detailed in the section on host security.

6.7.14 Lockdown Mode

Enabling the ESXi Lockdown Mode prevents root access to the hosts over the network. With Lockdown Mode enabled, all configuration changes to the vSphere environment must be made by accessing the vCenter Server. Lockdown Mode restricts access to host services on the ESXi host, but does not affect the availability of these services. There have been several significant enhancements in Lockdown Mode with the release of vSphere 6.0, which is addressed more fully in the section on host security.

vSphere Cluster Design

ESXi hosts and their resources are typically pooled together into clusters. These clusters contain the CPU, memory, network, and storage resources available for allocation to virtual machines. Clusters can scale up to a maximum of 64 nodes in vSphere 6.0 and can support thousands of virtual machines.

When designing clusters, there are two basic strategies:

- "Scale-out clusters," which have fewer ESXi hosts, but the architecture will end up with a larger overall number of clusters configured.
- "Scale-up clusters," which each have more ESXi hosts, but a fewer number of overall clusters.

Therefore, the design decision is whether to have many smaller clusters or fewer larger clusters.

As with "scale out" or "scale up" at the host level, each approach to cluster design has a number of advantages and drawbacks, but the decision will be driven by a number of potential factors as described in the following table.

Table 13. Advantages and Drawbacks of vSphere Cluster Design Options

Advantages	Drawbacks
Scale-out clusters (fewer numbers of hosts per cluster)	
• You can be less concerned about staying within the vSphere HA cluster virtual machines per host maximum.	• Depending on the admission control policy, more resources might be reserved for failover, reducing the amount of resources available to host virtual machines. • Reduced VMware vSphere Distributed Resource Scheduler™ (DRS) migration options.
Scale-up clusters (larger numbers of hosts per cluster)	
• Fewer resources might be reserved for failover, increasing the amount of resources available for virtual machines. • If DRS is enabled, more hosts offer greater migration choices and more opportunities to achieve a better workload balance across the cluster.	• There is DRS algorithm overhead on vCenter Server due to the larger number of calculations required. • Depending on the storage employed, presenting LUNs to large numbers of hosts could affect performance when locks are placed on datastores.

Other general factors, including the following, might also play a part in the design decision:

- Licensing (consider Oracle)
- Shared storage presentation or SAN zoning
- Redundancy (N+1, N+2, N+3 and N+4 models)
- CPU architectures (Intel/AMD)
- Network connectivity
- Total numbers of virtual machines

7.1 Designing vSphere Host Clusters

The vSphere cluster is typically the boundary of shareable resources. Therefore, when planning the design of clusters, consider the following key design considerations:

- **Capacity planning** – It may be simpler to plan for growth with a small number of large clusters. However, limitations on the number of hosts per cluster, and therefore the number if virtual machines per cluster, might warrant a scale-out approach to cluster design.
- **Hardware cost** – Because each cluster requires a defined amount of spare resources to accommodate failures, depending on the scale of the environment, having a large number of smaller clusters will result in a higher hardware cost to virtual machine ratio.
- **Security** – Isolating tenants or tenant applications into dedicated clusters is one way to segment workloads and control access through role-based access control (RBAC).
- **Performance** – Separating tenant's workloads or specific tenant applications into dedicated clusters provides a mechanism to verify that resources are constantly available for those consumers.

The number of hosts in a cluster will affect consolidation ratios. For example, you have eight compute nodes, and you have to decide whether to deploy two 4-node clusters or one 8-node cluster. You would have to reserve one ESXi host in each of the two 4-node clusters for HA failover to achieve N+1. To achieve the same level of availability in the 8-node cluster, you would still only need to reserve a single node, which will provide the design with one extra ESXi host for the running of virtual machine workloads.

In a production environment, always consider at least one host as a failover minimum per 8 to 10 ESXi servers to achieve N+1. Therefore, in a 16-node cluster, do not stay with only one host for failover. Aim to increase this number to two. The reason for this is that you must cover the risk of dual failure as much as possible by providing an additional node for this failover scenario, but also provide the ability to carry out maintenance on a host without a single host failure affecting the customer's environment. The vSphere HA calculations must not be overlooked, and are detailed further in Section 8, Planning for Server Failure.

The minimum size of a cluster is two nodes for vSphere HA to protect workloads in case one host stops functioning. However, in most use cases, a 3-node cluster is far more appropriate because you have the option of running maintenance tasks on an ESXi server without having to disable HA.

Configuring large clusters has its benefits too. You will typically have a higher consolidation ratio, but they might have a downside as well if you do not have enterprise-class or correctly-sized storage in the infrastructure. Keep in mind that if a datastore is presented to a 32-node or a 64-node cluster, and if the virtual machines on that datastore are spread across the cluster, there is a chance you will run into SCSI locking contention issues. Using a VMware vSphere Storage APIs – Array Integration aware array helps reduce this problem with ATS. However, if possible, consider starting small and gradually growing the cluster size to verify that your storage behavior is not impacted.

Another situation you might encounter is having separate ESXi servers for DMZ workloads or other isolated environments. While this approach might be considered "old school," some tenants might have security requirements or compliance requirements that require this type of architecture, which creates a physical boundary between servers, zones, or virtual machines. You might be able to use separate network cards and physical network fabric to achieve the customer's isolation goals, but still run the workload on the same ESXi server, giving you better consolidation ratios and still ensuring the level of security required for the customer.

When hosting tenant mission-critical applications, which are of the utmost importance and must have consistent performance at all times, you might need to place them in their own dedicated cluster. Whether you place multiple different applications in the same cluster or only place, one application per cluster (the concept of an "island cluster") will depend on the critical nature and resource requirements of the application, and possibly other factors such as isolation.

It is considered a VMware best practice to not run mixed-host clusters operating on different versions of ESXi code. However, typically during upgrade or patching activity there is likely to be a period of mixed-mode operation.

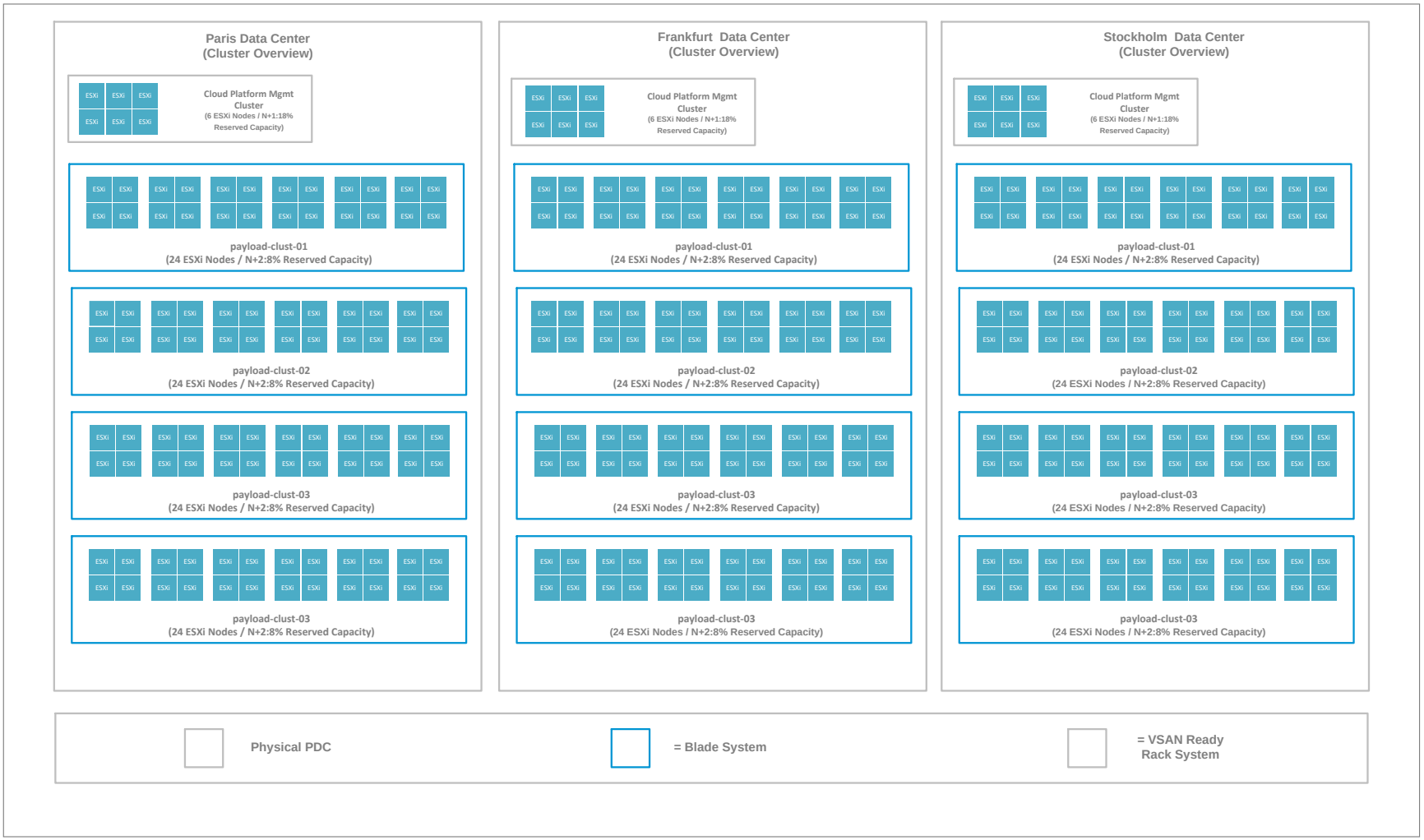
7.2 Building Block Clusters and Scale-Out Architecture

One simple and scalable approach to cluster design is the building block approach that is employed by many service providers and large private cloud platforms hosted by enterprise customers. The idea is that each cluster is a standard container of resources that is provisioned consistently to provide a simple, scalable, and building block approach to compute resource provisioning. Not only does this methodology scale consistently across data centers, but it also eliminates variability, configuration drift, and the amount of operational effort involved with patch management and day-to-day operations. This approach is the simplest and most effective way to provide a flexible building block solution that meets the service provider's requirement for elasticity. This building-block approach standardizes the configuration of the ESXi hosts, clusters, and even server cabinets to help provide a manageable and supportable infrastructure.

Standardizing not only the model, but also the physical and logical configuration of the ESXi hosts and clusters, is critical to providing a manageable and supportable infrastructure in large-scale deployments by eliminating variability. The aim is to utilize vSphere host profiles to configure additional values consistently across hosts and clusters wherever applicable, post-installation.

In the off-premises, hosted private cloud use case scenario, cluster sizing is dictated and configured according to the specific customer's requirement, because a dedicated management stack, including vCenter Server, is mandated by the design. However, for the shared multitenanted virtual data center use case, this building block approach aligns with the service provider's scaling and elasticity requirements.

Figure 20. Building Block Cluster Architecture



7.3 Cloud Platform Management Cluster

With the growth seen in new VMware management components and services, a dedicated, out-of-band cloud platform management cluster now has significant advantages for all VMware customers, but for service providers hosting a large and often complex cloud management platform (CMP), it is essential. The cloud platform management cluster contains all of the core components and the services required to run the virtual infrastructure and CMP, and is maintained separately from tenant workloads. Separating the management infrastructure components from tenant workloads allows for better segmentation of resources and improves the manageability and security of the vSphere and CMP infrastructure.

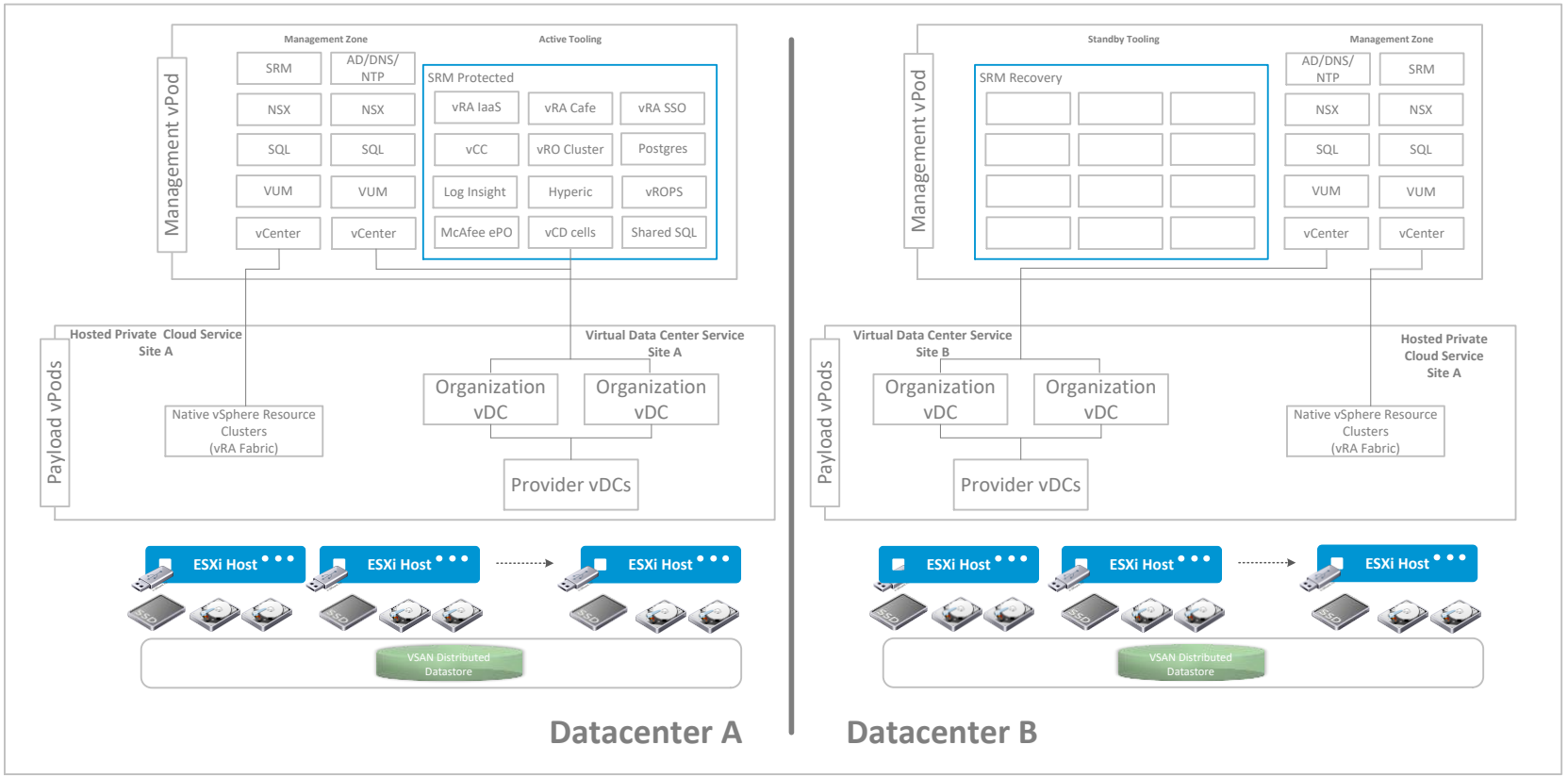
Providing the service provider infrastructure with a dedicated management cluster and management component separation has the following benefits:

- Separates the management components from the resources they are managing.
- Facilitates quicker troubleshooting and problem resolution, because management components are strictly contained in a relatively small and manageable cluster.
- Isolates resources between workloads running in the tenanted environment and the actual systems used to manage the cloud platform to avoid resource contention.
- Improves the ability to upgrade the vSphere environment and related components without affecting the tenant workload clusters.

The following host, storage, and networking considerations specifically apply to the design of a dedicated management infrastructure:

- Do not employ vSphere Auto Deploy or boot from SAN in the management environment. Avoid having the tools and services that are used to manage these components running on this same management platform, creating a circle of dependency.
- Avoid booting from local disks. This makes them unavailable for use in a vSAN disk group.
- Provide a highly available vSphere cluster configuration with redundancy at each component layer.
- Provide high availability of virtual and physical network management switching.

Figure 21. Highly Available Cloud Platform Management Cluster (Logical Architecture)



7.4 Cloud Platform Edge Cluster

A cloud platform edge cluster is typically included as part of a design that also incorporates VMware NSX for vSphere. The primary purpose of the cloud platform edge cluster is to place the controller virtual machines for NSX for vSphere and all customer edge devices that are provisioned as part of the onboarding and ongoing provisioning process onto a dedicated and segmented cluster.

A dedicated edge cluster hosting all VMware NSX Edge™ devices is able to act as a demarcation line to the Internet, or to corporate VLANs, so that the network administrator can provide management in a more secure and centralized way. For instance, this might mean that external connectivity needs to be configured only on the edge cluster hosts.

Three controllers must be deployed for NSX to provide sufficient redundancy and majority decisions from the controllers, and therefore this is considered the minimum number of hosts that can be configured to act as a cloud platform edge cluster.

7.5 Dedicated Island Clusters

The concept of island clusters is straightforward. An island cluster hosts workloads with special license requirements and is sometimes also referred to as a “dedicated application cluster.” Some software vendors apply special licensing policies on their applications, middleware, or databases that are not conducive to virtualization or hosted cloud environments, especially where vSphere DRS is employed and the application could potentially touch a high number of physical CPUs. Island clusters is one approach of dealing with this challenge.

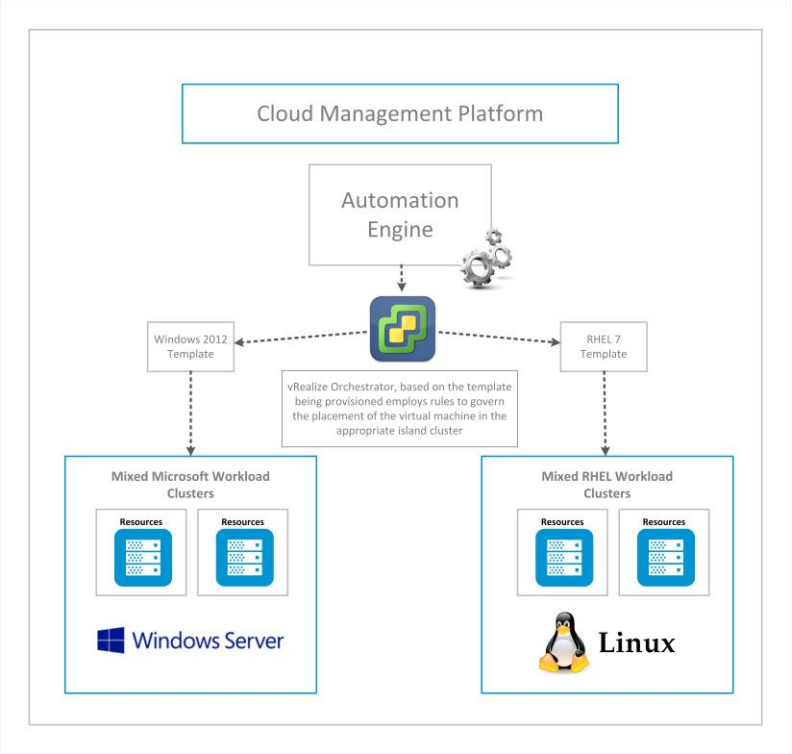
Use cases for island clusters might include:

- Running Oracle databases, middleware, or applications on dedicated clusters will not only provide that you are able to consolidate more and more Oracle virtual machines on a small cluster of ESXi hosts, but also reduce licensing costs by limiting the number of physical CPUs that require licensing (if this is your Oracle licensing model).
- Customers, particularly service providers, also use island clusters of operating systems such as Windows or RHEL. This helps you save money on data center Windows OS socket licenses, which is typically the most cost efficient way of licensing large numbers of Windows virtual machines running on a host. Another side benefit of this approach is that it helps ESXi to take advantage of the memory management technique of Transparent Page Sharing (TPS), if enabled. This is more efficient because there are higher chances that when you are running many of the same operating system virtual machines, duplicate pages will be spawned by these VMs in physical memory, making your ESXi servers more efficient.

Figure 22. Island Clusters

In addition, service providers can back this strategy with automation, ensuring that specific operating system templates are only deployed to the appropriate island cluster.

Figure 23. Island Cluster Provisioning Workflow

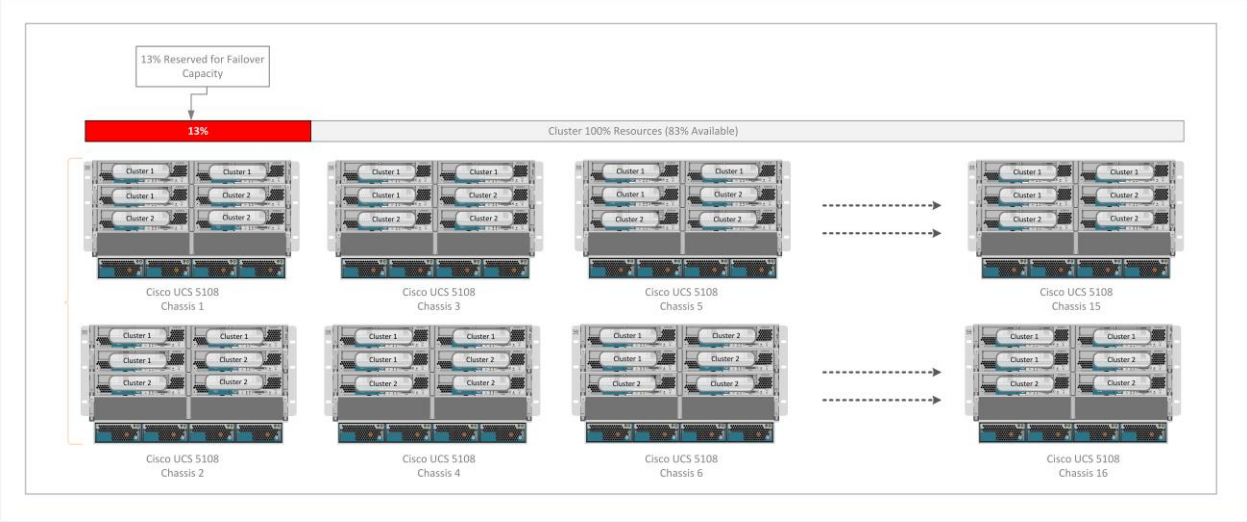


7.6 Host Placement for Optimized Availability

VMware recommends, where possible, using multiple server cabinets for the ESXi hosts, and distributing the vSphere clusters across the cabinets to minimize the impact of a single component failure.

This is also something to take into consideration when designing vSAN fault domains / rack awareness and vSphere HA admission control policies. If your tenant SLA dictates that you must maintain consumer services despite a blade chassis or compute cabinet failure, the number of hosts in the same failure domain that can be placed inside one of those entities is dictated by your admission control policy reservation. For instance, if the service provider guarantees to tenants that a blade chassis is not a single point of failure and the design has employed a Cisco UCS Blade System with 5108 blade chassis that are capable of supporting a maximum of 8 half-width blades, in a 24-node vSphere cluster, which can tolerate up to 3 hosts failing (13% of CPU and memory resource), each 5108 blade chassis must not accommodate more than 3 nodes that form part of the same vSphere cluster.

Figure 24. Physical Host Placement to vSphere Cluster Mapping



While a blade chassis or physical server rack is not typically considered a single point of failure, additional protection can provide higher levels of mitigation against major component outage.

The next consideration beyond chassis failure is mitigating against rack failure or even data center zone failure. The level of attention given to these requirements for availability and reducing the possibility of single points of failure must be stipulated in the design requirements.

7.7 Virtual Machine Mobility

In previous releases of vSphere, the cluster has been the boundary for sharable resources and live migration of virtual machines. The ability to live migrate a virtual machine across hosts was revolutionary at the time. However, with the release of vSphere 6.0, the vSphere vMotion capabilities have been enhanced significantly, enabling users to perform live migration of virtual machines across virtual switches, vCenter Server systems, and long distances with up to 150 millisecond (ms) RTT between hosts.

These new enhancements enable much greater flexibility when designing vSphere architectures, which had previously been limited to a single vCenter Server. This flexibility of scalability and live migration also provides enhanced options for multisite or metro designs, as vCenter Server 6 scale limits are no longer a physical boundary for compute resources. This means significantly larger and more flexible vSphere environments are now possible.

When a live migration occurs across vCenter Server instances, the metadata and virtual machine settings are preserved. This includes the virtual machine UUID, events, alarms, and task history, in addition to resource settings, such as shares, reservations, and limits.

vSphere HA and vSphere DRS settings are also maintained after a long-distance vSphere vMotion instance, along with affinity and anti-affinity rules, automation level, startup priority, and host isolation response. This allows for a seamless operational experience because the virtual machine live migrates throughout the multisite infrastructure. Virtual machine MAC addresses are also preserved as they are moved across different vCenter Server instances. When a virtual machine is moved from one vCenter Server instance to another, the MAC address is added to an internal blacklist to provide that a duplicate MAC address is never generated.

Increasing the latency thresholds for vSphere vMotion to 150 ms host-to-host RTT allows live migration to occur across larger geographic spans and potentially intracontinental distances. This feature will play a key role for many service providers in data center migrations, disaster avoidance scenarios, and multisite load balancing.

With the new features offered by vSphere 6, service providers can change the compute resource, storage resource, virtual machine network, and vCenter Server instance without disrupting consumer application services that reside on the virtual machines, thus enabling a wide array of new data center design opportunities.

Table 14. Online Migration Design Options

Mobility Technology	Use Cases	Business Benefits	Design Requirements
Cross virtual switch vSphere vMotion	Perform a seamless migration of a virtual machine across different virtual switches. Migrate a virtual machine to a new cluster with a separate VMware vSphere Distributed Switch™ (VDS) without interruption.	You are no longer restricted by the networks you created on the virtual switches to use vSphere vMotion to move a virtual machine. vSphere vMotion works across a mix of switches (standard and distributed). Previously, you could only use vMotion from a vSphere Standard Switch to another vSphere Standard Switch, or within a single VDS. This limitation has been removed. Cross virtual switch vSphere vMotion transfers the VDS metadata (network statistics) to the destination VDS. Increased agility by reducing the time it takes to replace/refresh hardware. Increased reliability with increased availability of business applications and increased availability during planned maintenance activities.	Requires the source and destination port groups to share the same L2 address space. The IP address within the virtual machine will not change. The following cross virtual switch vSphere vMotion migrations are possible: • vSphere Standard Switch to vSphere Standard Switch • vSphere Standard Switch to VDS • VDS to VDS Migrating back from a VDS to vSphere Standard Switch is not supported.

Mobility Technology	Use Cases	Business Benefits	Design Requirements
Cross vCenter vSphere vMotion	Built on enhanced vSphere vMotion, shared storage is not required. Simplify migration tasks in public/private cloud environments with large numbers of vCenter Server instances. Migrate from a vCenter Server Appliance to a Windows version of vCenter Server and the reverse. Replace or retire a vCenter Server without disruption. Resource pool across vCenter Server instances where additional vCenter Server instances are used due to vCenter Server scalability limits. Migrate virtual machines across local, metro, and continental distances. Increase reliability of migration to a Windows vCenter Server with an SQL cluster. This can increase availability of vCenter Server services, and increase availability during planned maintenance activities, such as vCenter Server upgrades. Upgrades can now be made without affecting managed virtual machines.	Perform the following virtual machine relocation activities simultaneously, and seamless to the guest operating system: • Change compute (vSphere vMotion) – Performs the migration of virtual machines across compute hosts. • Change storage (VMware vSphere Storage vMotion) – Performs the migration of the virtual machine disks across datastores. • Change network (cross virtual switch vSphere vMotion) – Performs the migration of a VM across different virtual switches. • Change vCenter (cross vCenter vMotion) – Performs the migration of the vCenter Server, which manages the VM. Reduced cost with migration to a vCenter Server Appliance, eliminating the need for Windows and SQL licenses.	As with virtual switch vSphere vMotion, cross vCenter vSphere vMotion requires L2 network space connectivity, because the IP of the virtual machine will not be changed.

Mobility Technology	Use Cases	Business Benefits	Design Requirements
Long-distance vSphere vMotion	Long-distance vSphere vMotion is an extension of cross vCenter vSphere vMotion. However, it is targeted for environments where vCenter Server instances are spread across large geographic distances, and where the latency across sites is 150 ms or less between source and destination hosts. Migrate VMs across physical servers that are spread over a large geographic distance without interruption to applications. Perform a permanent migration for VMs in another data center. Migrate VMs to another site to avoid imminent disaster. Distribute VMs across sites to balance system load.	Follow the sun global support teams can be employed. Increased reliability with greater availability of business applications during a disaster avoidance situation.	Although spread across a long distance, all the standard vMotion guarantees are honored. VMware vSphere Virtual Volumes™ is not required, but this technology is supported along with VMFS/NFS datastores. The requirements for long-distance vSphere vMotion are the same as cross vCenter vSphere vMotion, with the exception that the maximum latency between the source and destination sites must be 150 ms or less, and there must be 250 Mbps of available bandwidth.

Mobility Technology	Use Cases	Business Benefits	Design Requirements
Long-distance vSphere vMotion (cont.)			The VM network must be a stretched L2 because the IP address of the guest operating system will not change. If the destination port group is not in the same L2 address space, network connectivity to the guest OS will be lost. This means that in some topologies, such as metro or cross-continental, you will need a stretched L2 technology in place. The stretched L2 technologies are not specified. Any technology that can present the L2 network to the vSphere hosts will work, because the ESXi does not know how the physical network is configured. Some examples of technologies that would work are VXLAN, VMware NSX L2 gateway services, Cisco OTV, or GIF/GRE tunnels. There is no defined maximum distance that will be supported as long as the network meets these requirements. Long-distance vSphere vMotion performance will vary, because you are constrained by the laws of physics. For a complete list of requirements, refer to the VMware Knowledge Base article, *Long Distance vMotion requirements in VMware vSphere 6.0 (2106949)* at http://kb.vmware.com/kb/2106949.

Planning for Server Failure

An important consideration for cluster design is planning for server failure, or planned maintenance. vSphere clusters are typically used to provide increased availability and resource load balancing, and are a key component of a VMware Cloud Provider Program platform design. One part of the cluster design, discussed previously, is the proportion of the cluster's total capacity that is reserved for a failure event or reserved to undertake operational maintenance tasks. Determining the number of compute nodes in each cluster and the total capacity you want to reserve (which makes it unavailable for tenant workloads) will provide the percentage of capacity that must be allocated for failure scenarios.

8.1 vSphere High Availability

vSphere High Availability (vSphere HA) operates at the cluster level, and when enabled, provides the capability to monitor vSphere hosts for failures and automatically restart one or more virtual machines if deemed necessary. At a cluster level, vSphere HA monitors all hosts in the cluster through both network and datastore heartbeats. In the event that the network heartbeat between a slave node and master node in a cluster is lost, the master node attempts to use the datastore heartbeat to verify that the slave node is still operational. If the datastore heartbeat has stopped as well, the slave node is determined to have failed and the master node begins restarting the appropriate virtual machines on other nodes in the cluster. This is the reason for the requirement for reserved unused capacity within the vSphere cluster. The VMware Cloud Provider Program platform uses this vSphere core functionality as the foundation for service availability to their enterprise business tenants.

In a vSAN based cluster, this mechanism works slightly differently. On a vSAN platform, datastore heartbeats become irrelevant, and the vSphere HA agent uses the vSAN network to communicate instead of the management network. However, the management gateway is still used by the host to detect if it has become isolated. For more information about vSphere HA behavior in a vSAN enabled cluster, see the *vSAN Support Center* at http://www.vmware.com/uk/support/virtual-san.

Designing a scalable and reliable vSphere HA environment for the service provider does not differ significantly from designing the same mechanism for the enterprise business customer. However, a service provider's platform is heterogeneous by its very nature due to its lack of control over the applications provisioned by the tenants.

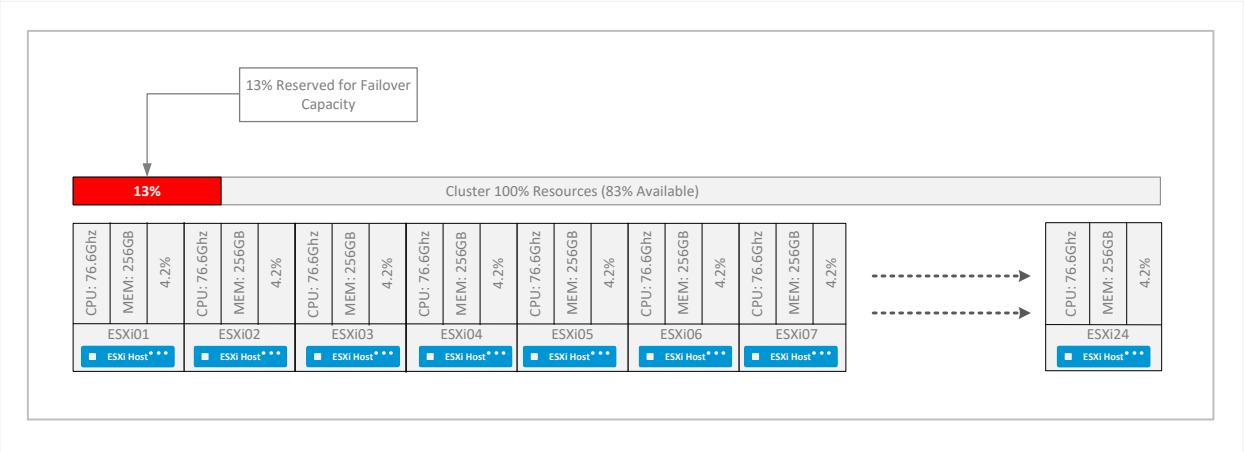
Each vSphere cluster within the VMware Cloud Provider Program is typically configured for vSphere HA to automatically recover virtual machines should any ESXi host fail, or even if there is a specific virtual machine failure. An exception to this rule might exist where a lower cost service offering is being provided to the tenants that does not include a strict availability clause in its SLAs.

8.1.1 Determining the Number of Host Failures to Tolerate

As previously discussed, enabling vSphere HA on a cluster reserves some host resources for a vSphere HA failure event, and therefore reduces the available capacity for running virtual machines. The vCenter Server reserves sufficient unused resources in the vSphere HA cluster to support the failover capacity specified by the chosen admission control policy.

For example, the following figure shows an eight-node cluster in which you would typically reserve the equivalent of a single host's resources as failover capacity, thus allowing for a single server failure within the cluster without impacting the performance of the virtual machines once restarted on remaining hosts.

Figure 25. Calculating the Number of Failures to Tolerate



It is the vSphere HA admission control policy that enforces this availability constraint, and preserves host failover capacity so that the vSphere HA cluster can support as many host failures as specified.

The admission control policy allows you to configure reserved capacity in any one of three ways:

- Host Failures Cluster Tolerates (default)
- Percentage of Cluster Resources Reserved
- Specify a Failover Host

With the default Host Failures Cluster Tolerates policy, vSphere HA performs admission control by calculating the available slot sizes. In brief, a slot is a logical representation of memory and CPU resources. By default, it is sized to satisfy the minimum requirement for any powered-on virtual machine in the cluster but can, and often should be, modified using advanced vSphere HA settings.

The Fault Domain Manager (FDM) determines how many slots each host in the cluster can hold, and calculates the Current Failover Capacity of the cluster. This determines the number of hosts that can fail in the cluster and still leave enough slots available to satisfy the virtual machines that will need to be powered on in the event of a server failure.

With the Percentage of Cluster Resources Reserved admission control policy of vSphere HA, instead of using slot sizes, calculations are used to provide that a percentage of the cluster's resources are reserved for failover.

The FDM carries out its calculations by determining the total resource requirements for all powered-on virtual machines in the cluster. It then calculates the total number of host resources available for VMs, and finally, it calculates the current CPU failover capacity and current memory failover capacity for the cluster. If the FDM determines there is less than the percentage that is specified for the configured failover capacity, the admission control policy will be enforced. For further information about the calculations relating to these first two options, refer to VMware documentation.

Finally, with the Specify a Failover Host admission control policy, you can configure vSphere HA to designate a specific host as the failover host. With this policy, when a host fails, vSphere HA attempts to restart its virtual machines on the specified failover host, which under normal operating conditions remains unused. If vSphere HA is not able to reboot all the virtual machines of the failed server, for example if it has insufficient resources, vSphere HA attempts to restart those virtual machines on other hosts in the cluster.

Specify a Failover Host is not the most commonly used policy because it means not all hosts are being utilized, but it is sometimes seen in scenarios where customers are required to demonstrate to auditors

that sufficient failover capacity exists. Also note that if this policy is used, the standby host must have the required resources to replace any host within the cluster.

Table 15. Admission Control Policy Use Cases

Policy	Recommended Use Cases
Host Failures Cluster Tolerates admission control policy	When virtual machines have similar CPU/memory reservations and similar memory overheads.
Percentage of Cluster Resources Reserved admission control policy	When virtual machines have highly variable CPU and memory reservations.
Specify a Failover Host admission control policy	To accommodate organizational policies that dictate the use of a passive failover hosts, most typically seen with the use of virtualized business critical applications.

When it comes to a VMware Cloud Provider making the right design decision regarding which policy to adopt, consider the following points:

- It is crucial to avoid resource fragmentation, which can occur when there are enough resources in aggregate for a virtual machine to be failed over, but the resources are spread across multiple hosts, and are therefore unusable. The Host Failures Cluster Tolerates policy manages resource fragmentation by defining the slot as the maximum virtual machine reservation. The Percentage of Cluster Resources policy does not address this problem, and so it might not be appropriate in a cluster where one or two large virtual machines reside with a number of smaller virtual machines. When the policy configured is Specify a Failover Host, resources are not fragmented because the host is reserved for failover, assuming the failover host has sufficient resources to restart all of the failed host's virtual machines.
- Consider the heterogeneity of the cluster. Service provider cloud platform clusters are typically heterogeneous in terms of the virtual machines resource required. In a heterogeneous cluster, the Host Failures Cluster Tolerates policy can be too conservative because it only considers the largest virtual machine reservations when defining slot size, and assumes the largest host will fail when computing the current failover capacity. As mentioned previously, this can and often should be modified to define a specific slot size for both CPU and memory manually. However, in a dynamic cloud environment this can be operationally difficult to calculate.
- Percentage based and dedicated failover host admission control policies are not affected by cluster heterogeneity. Because a Specify a Failover Host admission control policy requires one to four dedicated standby nodes that are not utilized during normal operations, the percentage based policy is typically recommended for service provider use. The Percentage of Cluster Resources Reserved policy also allows you to designate up to 50 percent of cluster resources for failover, while the Specify a Failover Host policy allows you to specify failover hosts. For these reasons, the recommended solution for VMware Cloud Providers is to employ the Percentage of Cluster Resources Reserved policy for admission control.
- When calculating the percentage reserved for the admission control policy, the service provider must consider the cluster size and SLAs provided to the consumers on service uptime. This typically takes into account both unplanned outages, caused by hardware failure or human error, as well as planned maintenance. The larger the cluster capacity, the larger the requirement for spare capacity, because the typical mean time to failure of a hardware component within a single cluster increases. The following table provides guidance for the percentage based admission control policy for the most commonly configured building block cluster sizes such as 8, 16, 24 nodes, and so on. For business critical production systems, VMware recommends that service providers provide a minimum of 1:8 to

1:10 reserved resource capacity for admission control. This means for a 24-node compute cluster, three nodes would provide the reserved capacity to tolerate multiple host failures, or a host failure during a maintenance period where one or more hosts were already unavailable for use (for instance, during an orchestrated patching cycle).

Table 16. Percentage Failed Resource to Tolerate (Percentage Based Admission Control Policy)

Avail-ability Level	Number of Nodes in vSphere Cluster															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
N+1	N/A	N/A	33%	25%	20%	18%	15%	13%	11%	10%	9%	8%	8%	7%	7%	6%
N+2	N/A	N/A	N/A	50%	40%	33%	29%	26%	23%	20%	18%	17%	15%	14%	13%	13%
N+3	N/A	N/A	N/A	75%	60%	50%	43%	38%	33%	30%	27%	25%	23%	21%	20%	19%
N+4	N/A	N/A	N/A	N/A	80%	66%	56%	50%	46%	40%	36%	34%	30%	28%	26%	25%
	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
N+1	6%	6%	5%	5%	5%	5%	4%	4%	4%	4%	4%	4%	3%	3%	3%	3%
N+2	12%	11%	11%	10%	10%	9%	9%	8%	8%	8%	7%	7%	7%	7%	6%	6%
N+3	18%	17%	16%	15%	14%	14%	13%	13%	12%	12%	11%	11%	10%	10%	10%	9%
N+4	24%	22%	22%	20%	20%	18%	18%	16%	16%	16%	14%	14%	14%	14%	12%	12%

vSphere 6.0 supports cluster sizes of up to 64 nodes. While this doubles the supported nodes in a cluster, this enhancement brings with it new considerations for designing vSphere HA for such a large failure domain:

- While designing for very large clusters comes with benefits, such as higher consolidation, and so on, this approach might also have a negative impact if you do not have enterprise-class or correctly sized storage in your infrastructure. Remember, if a datastore is presented to a 32-node or 64-node cluster, and if the virtual machines on that datastore are spread across the cluster, there is a possibility that there might be contention for SCSI locking. If the storage being employed supports vSphere Storage APIs – Array Integration, this is partly mitigated by ATS. However, verify that the design takes into account this possibility and that the storage performance is not impacted.
- With 64 nodes in a cluster, having such a large failure domain should mean you can increase your consolidation ratio by reducing the number of reserved hosts for admission control. However, you must provide the design allows for sufficient host failures to guarantee SLAs can be met in a multiple host failure scenario, and during host maintenance and patching.
- Although we are currently addressing vSphere HA, DRS performance is based on the DRS algorithm thread on the vCenter Server, and the amount of calculations required for resource scheduling. For large clusters, this calculation overhead will increase accordingly. Consider this as part of the design decision process.

Understanding the vSphere HA mechanism is key to a good, resilient and functioning design. The following sections provide an overview of the different vSphere HA components and the multiple complex mechanisms that maintain virtual machine availability in event of a host failure.

8.1.2 vSphere HA Components

The vSphere HA mechanism is essentially made up of three different but equally important components:

- The Fault Domain Manager (FDM)
- The hostd agent installed on each ESXi host server
- The vCenter management server

In vSphere 5.0, the FDM replaced the Legato Automated Availability Manager (AAM). The FDM agent runs on ESXi hosts and communicates with the other hosts in the cluster with regard to available resources and the state of running virtual machines. It is also responsible for the heartbeat mechanism, the placement of virtual machines, and the restarting of VMs in a vSphere HA event in relation to the hostd agent. The FDM communicates directly with both the hostd agent and the vCenter Server. The hostd agent, which runs on the host server, is required for vSphere HA to function correctly. If the hostd agent is not functioning, the FDM puts all vSphere HA functions on hold until the agent is operational again. The vCenter Server deploys and manages the FDM agents after a host is added to an HA cluster. The vCenter server also provides all the configuration changes within a cluster to the host elected as the master.

8.1.3 The Master Slave Model

During the creation of a vSphere HA cluster, the FDM agent is deployed on each ESXi host. In addition, one host is elected as the master and all other hosts are assigned as slaves. The role of the master host is to continuously monitor the state of the slave hosts and detect issues within the cluster. The master also holds the list of virtual machines that are currently being protected by vSphere HA and is responsible for the replacement and restarting of those virtual machines should a vSphere HA event occur. If the master host fails, becomes isolated, partitioned, or disconnected from vCenter Server, it is placed into maintenance mode, has its vSphere HA agent reconfigured, or is rebooted, and another host is elected as master.

8.1.4 HA Failure States

If the master host can no longer communicate with the FDM agent of a slave server but the heartbeat datastore responds, that server is still functioning from the perspective of vSphere HA. In this type of failure scenario, the slave host is considered isolated or partitioned from the network.

A server is considered isolated when it is no longer able to communicate with the master host's heartbeat and the master can no longer ping the failed hosts management IP address. There is also the scenario where several hosts within the same cluster become isolated but can communicate among themselves through the management network. When this occurs, it is referred to as a network partition. In this type of failure scenario, a second host is elected as a master within the same partitioned network, so the result can mean that several different partitions each with its own master exist. When communication within the cluster is re-established, the cluster will return to having a single master with the other hosts returning to a slave role.

What happens to virtual machines when a host is isolated depends on the defined policy. This is referred to as the Host Isolation Response.

The policy options are:

- Leave Powered On (default) and allow the virtual machine to continue functioning.
- Power Off or Shut Down (if VMware Tools™ is installed) and initiate a reboot of the VM on a different surviving host in the cluster.

The design decision to select an appropriate Host Isolation Response policy is based on a number of factors and configurations. Use Leave Powered On if you want your virtual machine to be accessed continuously by users and believe that an outage of the management network might not necessarily impact virtual machine traffic networks. You typically select Shut Down to so that your virtual machines

only run in a stable environment. Note that Power Off is not a graceful shutdown for applications or operating systems.

In addition to these policy options, there is a general recommendation that where IP-based storage, such as iSCSI or NFS is employed as the protocol to access the heartbeat datastores, and the same physical connectivity is used for storage, virtual machines, and management networking (converged), that you modify the default option, and configure the host isolation response policy to Power Off or Shut Down. The reason for this is that any outage in this type of architecture is likely to affect all network segments across the converged infrastructure and not only the management IP space.

8.1.5 Heartbeat Communication

The term heartbeat refers to the vSphere HA mechanism that determines whether or not a host server is functioning normally. Each slave server in the cluster sends a heartbeat signal to the master server every second and likewise, the master server sends a heartbeat to each of the slave servers in the cluster every second. When the master node does not receive a heartbeat communication, it might mean that network communication has failed, but does not necessarily mean that the slave server has failed. To verify that the slave server is still functioning and able to support the running of virtual machines, the master checks its health status by using two different mechanisms:

- Sends a ping to the slave host's management IP address
- Performs a datastore heartbeat

A host is declared failed when the vSphere HA master node cannot communicate with it by means of a network heartbeat, or a storage heartbeat, and the host cannot be pinged.

8.1.6 Heartbeat Network Path Redundancy

Redundancy between cluster nodes provides the best reliability for the mechanism that protects the virtual machines. A single host management network is potentially a single point of failure and might result in failover scenarios, even though only a single network component has failed. Without heartbeat path redundancy, any failure between the host and the cluster could potentially cause an unnecessary failover event.

Hardware failures, including network interface card failures, network cable failures, network cable removal, and switch resets, can be possible sources of failure between hosts, and therefore any redundant design mitigates against this risk, and try to minimize the impact of failure. Typically, this can be achieved by providing network redundancy at every component of the physical network.

It is possible to implement network redundancy either at the NIC level with NIC teaming, or at the management network level with a secondary management network. In most service provider implementations, NIC teaming will provide sufficient redundancy, but you can use or add a secondary management network for redundancy if required. Implementing a redundant management networking allows the reliable detection of host failures and prevents isolation event conditions from occurring because heartbeat traffic can be sent over multiple networks.

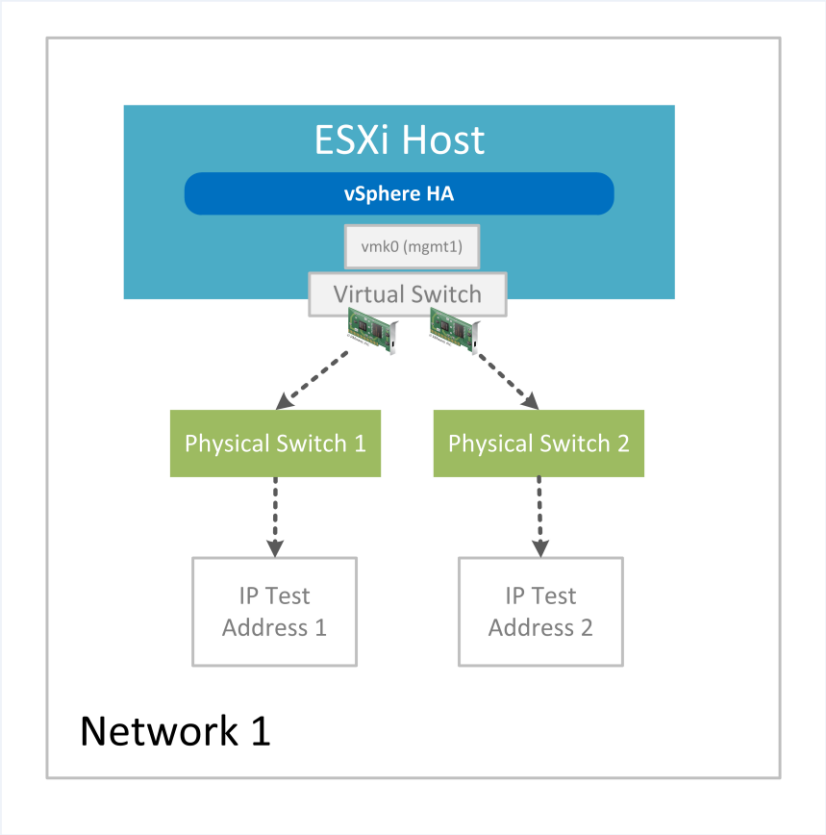
Aim to configure as few as possible hardware segments between the servers in a cluster. The goal is to limit single points of failure, which is best achieved through simplicity. In addition, too many network hops can cause networking packet delays for heartbeat traffic and increase the possible points of failure.

Option 1: Heartbeat Network Path Redundancy (NIC Teaming)

You can create a network interface card (NIC) team for vSphere HA management network redundancy, which is typically the recommended configuration for service providers. Each NIC in the team must be connected to a separate physical switch.

In this design, the NIC team helps prevent a switch failure from initiating a vSphere HA isolation response. See the following figure for an example of NIC teaming.

Figure 26. Heartbeat Network Path Redundancy (NIC Teaming)

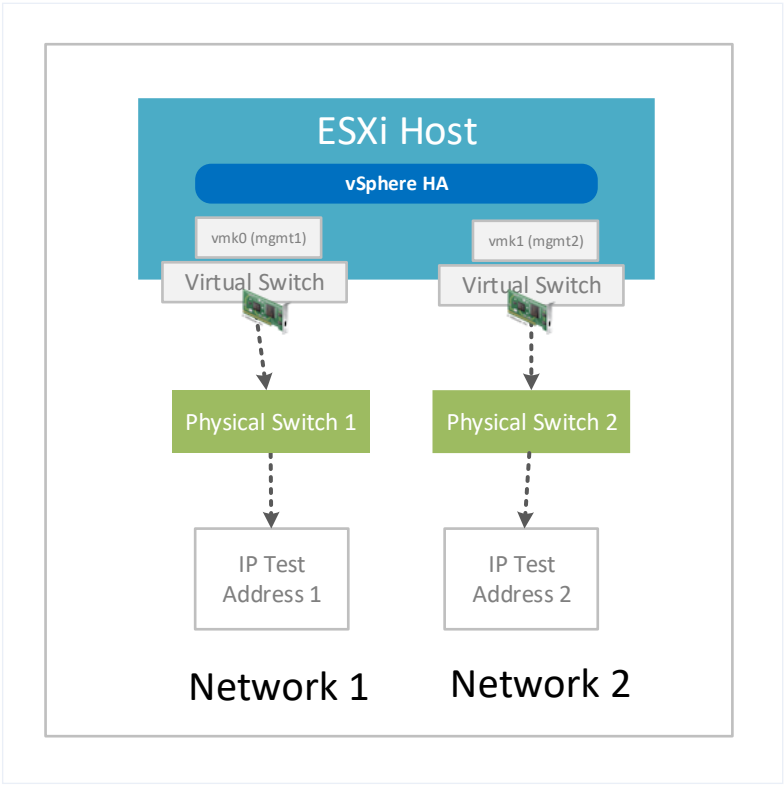


Option 2: Heartbeat Network Path Redundancy (Secondary Management Network)

This second option creates a second VMkernel port for ESXi, which is attached to a separate virtual switch or port group. In this design, all management network interfaces are employed to send heartbeats.

As shown in the following figure, virtual switches are configured on separate physical switches, eliminating all single points of failure. For this design, you will also need to use the `das.isolationaddress` parameter to add an isolation address to each additional management network segment, which removes the isolation address as a single point of failure.

Figure 27. Heartbeat Network Path Redundancy (Secondary Management Network)



8.1.7 Virtual Machine and Application Monitoring with vSphere HA

Virtual machine monitoring can determine whether a VM is not functioning through a heartbeat exchange between the host server and the virtual machine's VMware Tools instance. This is performed every 30 seconds. If this exchange does not take place, the virtual machine is rebooted. From the service provider perspective, this might not be something you want to occur. When enabled, a virtual machine is declared failed if the heartbeat inside the guest operating system tools is no longer received, and there is no network or storage I/O observed.

To detect an application crash, VMware provides an application recognition API that allows software vendors and in-house application developers to develop solutions that are monitored for health and that tell vSphere HA to reboot a virtual machine if a running application fails. For more information about the VMware application recognition API, see the *VMware vSphere Guest SDK Documentation* at https://www.vmware.com/support/developer/guest-sdk/.

Enable virtual machine monitoring only if the application or service running on the virtual machine is able to recover after a virtual machine restart. Disable virtual machine monitoring on a virtual machine when it is already protected by either VMware vSphere Fault Tolerance or Microsoft Cluster Services.

8.1.8 Heartbeat Datastores

When the master host in a vSphere HA cluster is unable to communicate with a slave host over the ESXi management network, the master host resorts to using datastore heartbeats to establish whether the slave host has become unavailable, is in a partition state, or is network isolated. Where the slave host has stopped the datastore heartbeats, it is considered to have failed, and its virtual machines are restarted on other surviving hosts.

By default, vCenter Server selects a preferred set of datastores for heartbeats to maximize the number of hosts that have access to a specific heartbeat datastore and therefore minimize the likelihood that the datastores are backed by the same storage array. However, it is a simple task to replace the default-selected datastores by using the Cluster Settings dialog box in the vSphere Web Client to specify specific heartbeat datastores.

It is also possible to use the advanced attribute `das.heartbeatdsperhost` to change the number of heartbeat datastores selected by vCenter Server for each host in the cluster. The default is two and the maximum valid value is five.

When enabled, vSphere HA creates a directory at the root of each of the selected datastores. The name of the directory is `.vSphere-HA`. Make sure that this directory is never deleted or modified by your operational teams, because this action will almost certainly have a serious impact on the vSphere HA mechanism to maintain operations.

In a vSAN environment, vSphere HA behavior is slightly different from the traditional mechanism. Datastore heartbeats are no longer relevant, and the vSphere HA agent uses the vSAN network to communicate instead of the host management network. However, the management gateway remains used by the host to detect whether it has become isolated.

Datastore heartbeat key design implications include:

- Allowing vCenter Server to select a preferred set of heartbeat datastores. vCenter Server uses certain guidelines for choosing the preferred set of heartbeat datastores:
  - Choose a datastore that is accessible by the maximum number of hosts.
  - Prefer VMFS datastores to NFS datastores.
  - Prefer datastores that are backed by different storage arrays.
- vSphere HA uses approximately 3 MB of disk space on each heartbeat datastore, which is negligible for most environments.
- The vSphere HA datastore heartbeat mechanism adds a negligible overhead on the storage system that has no performance effect on other storage operations.

The following points are considered best practices for a service provider when designing a solution that employs vSphere HA:

- Always configure strict admission control to protect tenant's workload. While this reserves resources that cannot be used under normal operating conditions, and therefore increases hardware costs, it protects critical business services to tenants. Always explain the risks of not enabling a strict admission control policy to the key stakeholders and SMEs.
- The size of the cluster and percentage of reserved capacity are closely interrelated.
- Verify that the amount of resources reserved for failover is not proportionally too high, and that it does not negatively affect the resources available to tenants.
- Always reserve sufficient failover capacity to accommodate host failures during scheduled maintenance and unplanned downtime.
- Make strict admission control a matter of a change control policy, ensuring that powering on unprotected virtual machines becomes a choice made by managers and not operational staff or administrators.

8.1.9 Sample HA Cluster Configuration

A service provider's VMware Cloud Provider Program platform typically configures all vSphere HA parameters consistently across all clusters in all data centers. This helps to limit variability and simplify operational management. The following table provides design guidance for service providers by addressing the standard settings and options configured for vSphere HA attributes.

Table 17. Sample vSphere HA Settings

Attribute	Configuration
Cluster Name	boston-dc-01-payload-003
Number of ESXi Hosts	24
Host Monitoring	Enabled
Admission Control Response	Prevent virtual machines from being powered on if they violate availability.
Admission Control Policy	Enabled: Percentage of resources reserved: 13% CPU 13% Memory N+3 for 24 node cluster (based on 1 to 8 Ratio)
Default Virtual Machine Restart Priority	Medium (majority of VMs) Modify if necessary at VM level for High (critical VMs) / Disabled (non-critical VMs)
Host Isolation Response*	Shut down or power off
Virtual Machine Monitoring	Disabled
Virtual Machine Monitoring Sensitivity	N/A
Heartbeat Datastores	Automatically select datastores accessible from the host (default)

* Host isolation response refers to the action that vSphere HA takes when the host becomes isolated. The best practice in a vSAN environment might differ from a traditional storage vSphere HA design, depending on the deployed configuration. For information about the specific vSAN host behavior during a host isolation event, see the *vSAN Support Center* at https://www.vmware.com/support/virtual-san.

8.2 vSphere Fault Tolerance

vSphere Fault Tolerance (vSphere FT) provides continuous availability for applications in the event of a server failure by creating a live shadow instance of a virtual machine that is in lockstep with the primary instance. By allowing instantaneous failover between the two instances in the event of hardware failure, vSphere FT eliminates even the smallest chance of data loss or service disruption. vSphere FT automatically triggers:

- Seamless stateful failover when the protected virtual machines fail to respond, providing zero downtime, zero data loss, and continuous service availability.
- The creation of a new secondary virtual machine after failover, to provide continuous protection of the application.

In releases previous to vSphere 6.0, all virtual machines that were to be protected by vSphere FT were restricted to only one vCPU and a range of other limitations that inhibited adoption, including a requirement to have all of the virtual machine VMDKs configured as eager-zeroed.

These previous limitations made it impractical for use with the majority of virtual machines. However, through the development of a completely new fast checkpointing technology, vSphere FT now supports protection of virtual machines with up to 4 vCPUs and 64 GB of memory. This means that the vast majority of mission-critical tenant workloads can now be protected regardless of application or guest operating system.

In addition, VMware vSphere Storage APIs - Data Protection can now be used with virtual machines protected by vSphere FT. With 6.0, vSphere FT also enables vSphere administrators to use VMware snapshot-based tools to back up virtual machines protected by vSphere FT, enabling easier backup administration, enhanced data protection, and reduced risk.

There has also been a significant change in how vSphere FT handles storage. vSphere FT now creates a complete copy of the entire virtual machine, resulting in total protection for virtual machine storage, in addition to compute and memory. It also enables the files of the primary and secondary virtual machines to be stored on shared as well as local storage. The result is increased protection, reduced risk, and improved flexibility.

Other improvements have been made to vSphere FT virtual disk support and host compatibility requirements. Previous vSphere releases required a very specific virtual disk type—eager-zeroed thick. They also had very limiting host compatibility requirements. vSphere FT now supports all virtual disk formats—eager-zeroed thick, thick, and thin, and host compatibility for vSphere FT is now the same as for vSphere vMotion.

Symmetric Multiprocessing Fault Tolerance (SMP-FT) provides zero data loss and downtime with the ability to recover a virtual machine instantly and continue working in the case of a hardware failure.

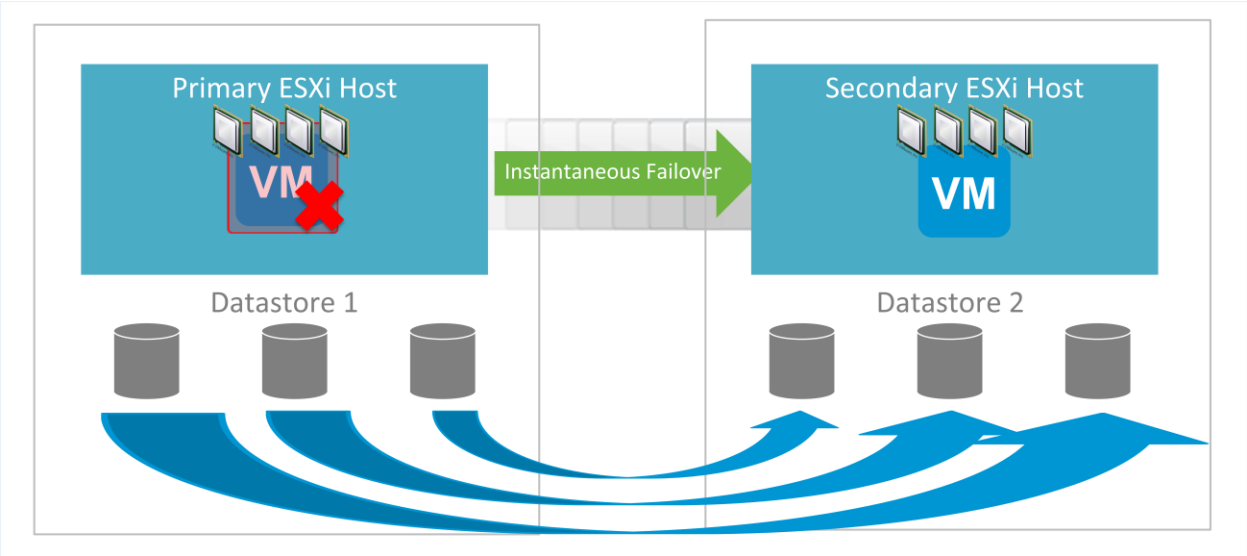
In addition, SMP-FT:

- Supports up to 4 vCPU
- Supports up to 64 GB memory
- Supports vMotion for both the primary and secondary virtual machines
- Creates a secondary copy of VMs files and disks
- Does not support user-created snapshots
- Supports snapshot-based backup solutions
- Creates a full copy of a VM for redundancy
- Uses an XvMotion operation to create an initial copy of the VM
- With SMP-FT, the primary and secondary are continuously updated to stay in sync

- If the primary host crashes, the VM can be resumed on the secondary host
- A 10-Gbps NIC is recommended for the SMP-FT network
- Support for multiple 10-Gbps NICs on the vSphere FT network is not yet available
- As before, Storage vMotion is not possible with vSphere FT running multiple vCPUs
- Virtual machines in vCloud Director, vSAN, vSphere Virtual Volumes, and vSphere Replication are not supported on SMP-FT machines

The new technology used by SMP-FT is called fast checkpointing and is basically a heavily modified version of XvMotion that runs continually, and executes many more checkpoints (multiple times per second).

Figure 28. SMP Fault Tolerance – Two Complete Virtual Machines



Although SMP-FT seems similar to the previously available Uniprocessor FT (UP-FT), it is, in fact, a new technology only available with vSphere 6.0. However, uniprocessor virtual machines can continue to use the legacy Record-Replay FT or the new SMP-FT technology, and UP FT virtual machines can run alongside SMP-FT virtual machines without issue. The following tables examine use cases, business benefits, design requirements, and capabilities for incorporating UP-FT and SMP-FT based services as part of a VMware Cloud Provider Program platform.

Table 18. Symmetric Multiprocessing Fault Tolerance Design Options

Use Cases	Business Benefits	Design Requirements
Use cases include any workload that has up to 4 vCPUs and 64 GB memory that is not latency-sensitive (for instance, VOIP or high-frequency trading). There is VM/application overhead to using vSphere FT which depends on a number of factors, such as the application, number of vCPUs, number of vSphere FT protected virtual machines on a host, host processor type, and so on.	Protect mission-critical, high-performance applications regardless of operating system. Continuous availability. Zero downtime and zero data loss for infrastructure failures. Fully automated response. SMP-FT greatly expands the use cases for vSphere FT to approximately 90 percent of workloads.	vSphere FT logging (traffic between hosts where primary and secondary are running) is very bandwidth intensive and will use a dedicated 10-GB network interface on each host. This is not required, but highly recommended, because at a minimum, a vSphere FT protected virtual machine will use more bandwidth. If FT does not get the bandwidth it needs, the impact is that the protected VM will run slower. There is a limit of either 8 vCPUs or 4 vSphere FT protected VMs per host—whichever limit is reached first: • 2 VMs with 4 vCPUs each (total 8 vCPUs) • 4 VMs with 2 vCPUs each (total 8 vCPUs) • 4 VMs with 1 vCPUs each (total 4 vCPUs) In addition, vSphere FT now creates a second copy of the VMDKs associated with a protected virtual machine. This means that storage is now redundant (where with the previous version it used shared storage so it was not). However, this also means that storage requirements are doubled for every protected virtual machine.

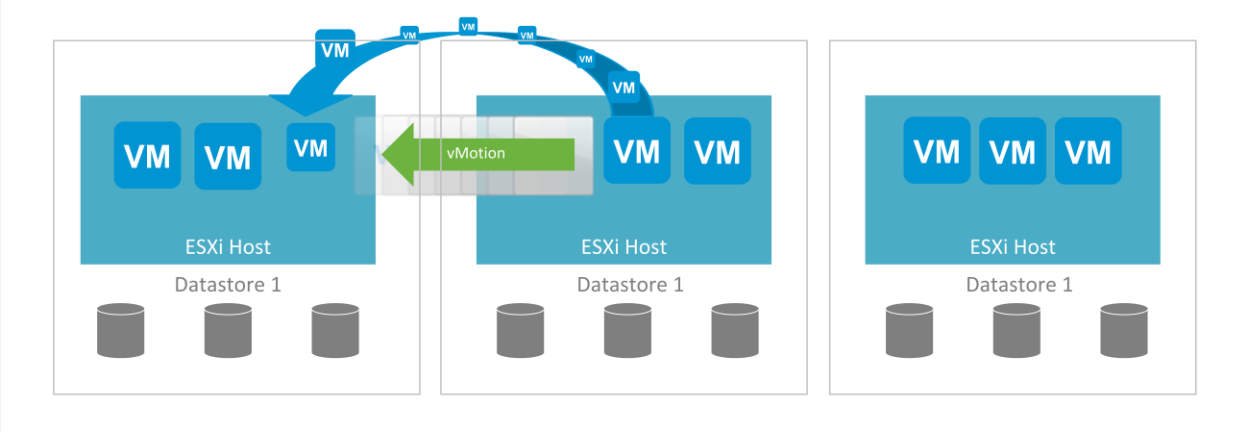
Table 19. Fault Tolerance Capabilities by vSphere Version

Feature	vSphere FT (vSphere 5.5)	vSphere FT (vSphere 6.0)
vCPUs	1	4
Virtual Disks	Eagerzeroed	Any
Hot Configure FT	No	Yes
H/W Virtualization	No	Yes
Backup (Snapshot)	No	Yes
Paravirtual Devices	No	Yes
Storage Redundancy	No	Yes
vSAN / vSphere Virtual Volumes	No	No
High Availability	Yes	Yes
vSphere DRS	Partial (Initial Placement)	Partial (Initial Placement)
VMware vSphere Distributed Power Management™	Yes	Yes
VMware Site Recovery Manager™	Yes	Yes
vSphere Distributed Switch	Yes	Yes
Storage DRS	No	No
vCloud Director for Service Providers	No	No
vSphere Replication	No	No

Resource Balancing and Transparent Maintenance

vSphere Distributed Resource Scheduler (DRS) dynamically balances computing capacity across a collection of hardware resources aggregated into logical resource pools based on CPU and memory load status. vSphere DRS continuously monitors utilization across resource pools, or the root cluster resource pool, to intelligently allocate available resources among virtual machines based on a predefined set of rules that reflect business or application requirements. These vSphere vMotion migrations of virtual machines can be performed automatically or manually based on the defined parameters.

Figure 29. vSphere Distributed Resource Scheduler (DRS)

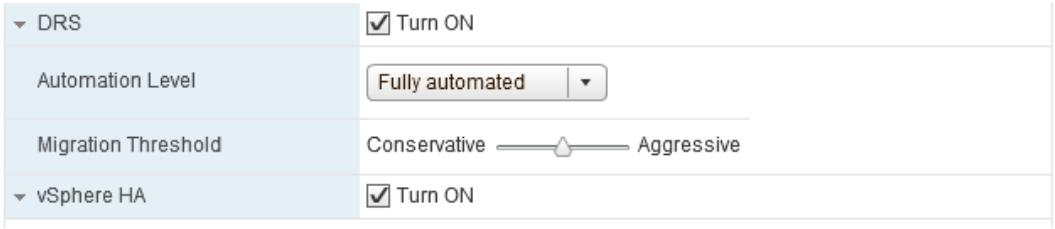


An additional benefit of DRS is realized when there is an operational requirement to place a host in maintenance mode. For instance, when servicing is required or to install additional memory on a host, all virtual machines that are running on the targeted host will be automatically migrated to other hosts within the cluster (assuming the DRS policies are configured appropriately to allow automatic migration). This provides a significant additional benefit gained from the vSphere HA admission control policy reserving spare capacity for both planned and unplanned outages. This can prove invaluable when performing rolling hardware maintenance or orchestrated patching through vSphere Update Manager. These types of actions can be carried out without any need to disrupt tenant services and are completely transparent to the consumers, providing non-stop IT services, maintenance without downtime, and greatly improved availability.

9.1 DRS Automation

vSphere DRS employs vSphere vMotion to migrate virtual machines in either a fully automated, partially automated, or manual manner, depending on the parameters defined.

Figure 30. vSphere DRS Configuration Options



As shown in the preceding figure, there are three different automation levels. When **Fully automated** is selected, a migration threshold must be configured.

Table 20. DRS Automation Levels

Setting	Initial VM Placement	Load Balancing
Manual	Recommendation is displayed to administrator	DRS makes a recommendation but will not migrate VMs without validation from the administrator.
Partially Automatic	Automatic placement	While the initial placement is completed automatically by DRS, the migration of powered on virtual machines will only be performed after the administrator validates the recommendation in vCenter Server.
Automatic	Automatic placement	Fully automated migrates powered on virtual machines automatically. The level of aggressiveness employed by DRS for this automatic migration is based on a threshold corresponding to five different recommended levels from conservative (5 stars) to aggressive (1 star).

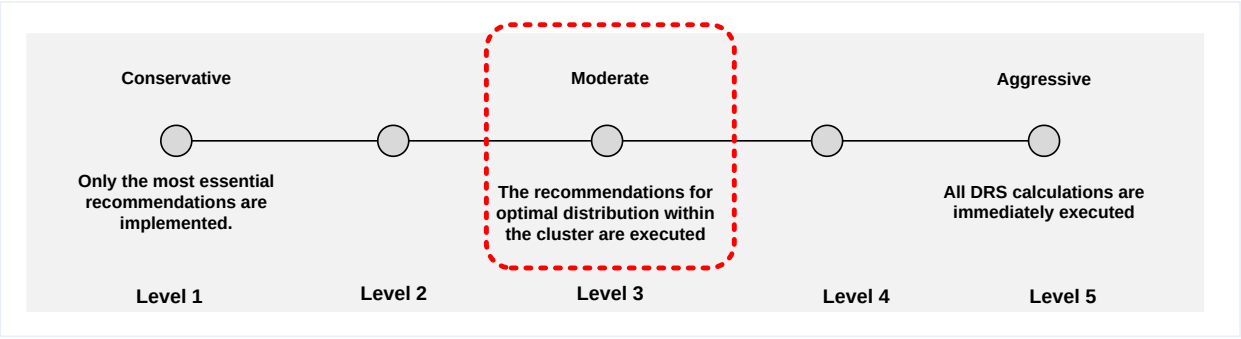
Table 21. Migration Threshold Options

Level	Stars	Description
Level 1	5	Conservative. Migrations will only take place if rules are not being respected or if a host is placed into maintenance mode.
Level 2	4	A migration will only take place if level 1 is met or if a migration will bring about significant improvements in performance.
Level 3	3	A migration will only take place if the first two levels are met or if a migration brings about a good amount of improvements to virtual machine performance.
Level 4	2	A migration will only take place if the first three levels are met or a migration brings about moderate improvements to virtual machine performance.
Level 5	1	Aggressive. Migration will occur only if all recommendations from Level 1 to 4 are met or if the migration will bring about minor improvements to virtual machine performance.

As you would expect, the conservative setting leads to fewer migrations whereas the aggressive configuration will lead to more frequent virtual machine migrations.

For a service provider environment, in which clusters are usually heterogeneous in nature, it is typically desirable to set the vSphere DRS automation level to Fully Automatic and the migration threshold to the default moderate (Level 3) configuration to avoid automatic vSphere vMotion actions that might have either only a minimal or short-term performance benefit.

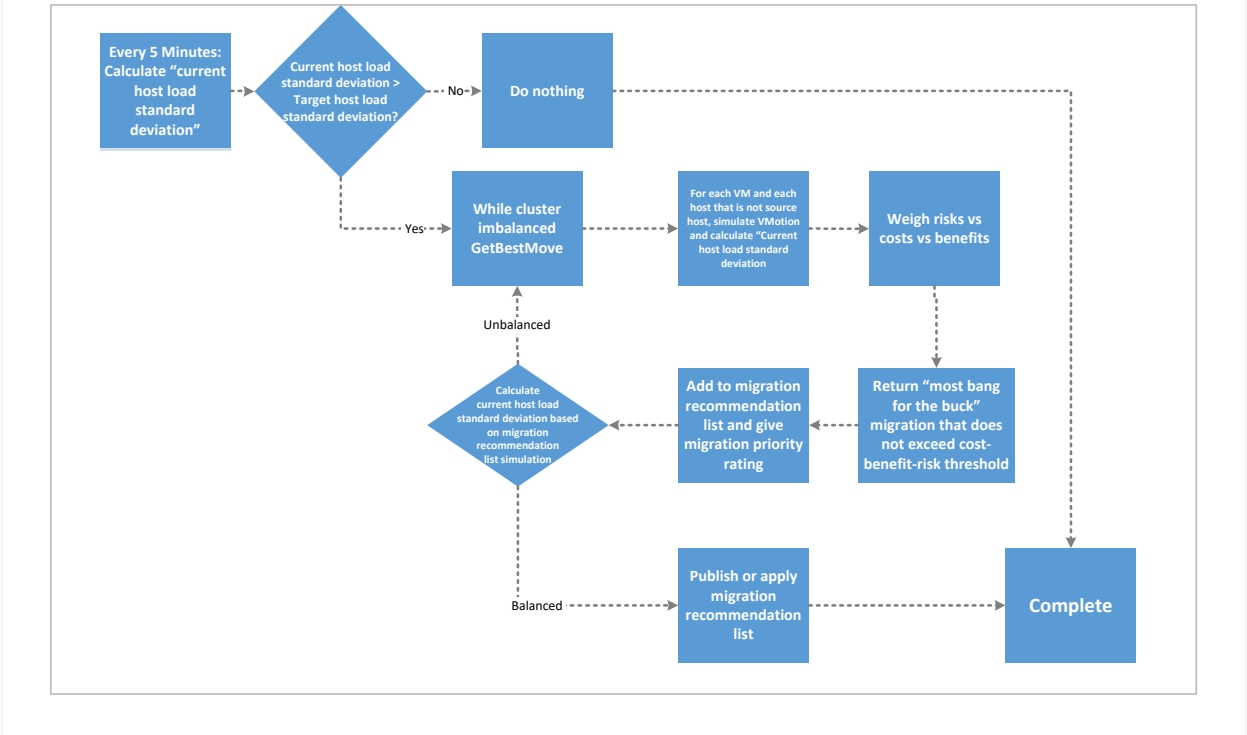
Figure 31. Migration Threshold Slider – Recommendation for VMware Cloud Providers



By default, an automation level is specified for the whole cluster. However, you can also specify a custom automation level for individual virtual machines, if appropriate. For instance, you might do this if you require a specific vSphere DRS setting, such as manual, defined on explicit virtual machines within the cluster. Where this is configured, vCenter Server does not take automatic actions to balance those explicit resources but instead, the vSphere DRS Summary page indicates that migration recommendations are available, and the Migration page displays the recommendation.

Although it is not essential that you configure vSphere DRS on a service provider's payload cluster, VMware recommends using this mechanism as a way of balancing workloads across hosts in the cluster for optimal performance.

Figure 32. vSphere DRS Automation Workflow



A service provider's VMware Cloud Provider Program platform typically configure all DRS cluster parameters consistently across all clusters in all data centers. This helps to limit variability and simplify

operational management. The following table provides design guidance for service providers by addressing the standard settings and options configured for vSphere DRS attributes.

Table 22. Sample vSphere DRS Settings

Attribute	Configuration
Cluster Name	boston-dc-01-payload-003
Number of ESXi Hosts	24
DRS	Enabled
Automation Level	Fully Automated
Migration Threshold	Moderate, Level 3 (Default)
vSphere Distributed Power Management (DPM)	N/A
Enhanced vSphere vMotion Compatibility	Disabled

For a service provider's implementation, typically DRS is enabled to automate workload balancing. DRS will always benefit the overall infrastructure improving performance, scalability, and manageability while providing transparent maintenance to the consumers. The only exception to this is if the specific tenant, on a dedicated cluster, is running applications that scale and balance at the application level.

The following table examines use cases, business benefits, and design requirements for incorporating vSphere DRS as part of a VMware Cloud Provider Program platform.

Table 23. DRS Use Cases, Business Benefits, and Design Requirements

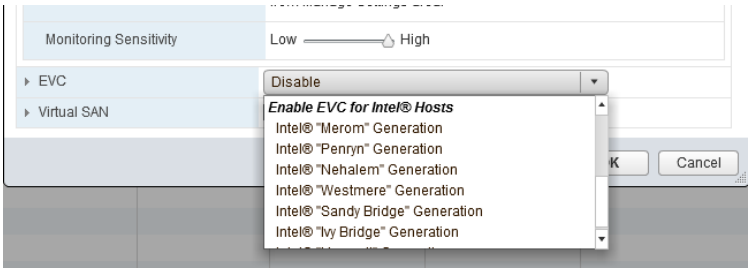
Use Cases	Business Benefits	Design Requirements
Redistribute CPU and/or memory load between ESXi hosts in the cluster. Migrate virtual machines off an ESXi host when it is placed into maintenance mode. Rules to keep virtual machines together on the same host (affinity rule) optimizing communication by ensuring host adjacency of VMs or separating virtual machines on to different ESXi hosts (anti-affinity) in order to maximize availability of services. Use anti-affinity rules to increase availability for service workloads as appropriate, such as in rare cases where applications with high-transactional I/O workloads might require an anti-affinity rule to avoid an I/O bottleneck on the local host.	vSphere DRS collects resource usage information for all hosts and virtual machines in the cluster and will migrate virtual machines in one of two situations: • Initial placement – When you first power on a virtual machine in the cluster, DRS places that virtual machine on the most appropriate host. • Load balancing – DRS aims to improve resource utilization across the cluster by performing automatic migrations of running virtual machines (through vSphere vMotion). • Configuring DRS for full automation, using the default migration threshold: • Reduces daily monitoring and management requirements. • Provides sufficient balance without excessive migration activity.	vMotion migration requirements must be met by all hosts in the DRS cluster. Whether or not to enable Enhanced vMotion Compatibility (EVC) at the appropriate EVC level on hosts. DRS load balancing benefits from having a larger number of hosts in the cluster (scale-out cluster) rather than a smaller number of hosts. DRS affinity and anti-affinity rules should be the exception rather than the norm. Configuring many affinity and anti-affinity rules limits migration choices and could collectively have a negative effect on workload balance. An affinity rule is typically beneficial in the following situations: • Virtual machines on the same network share significant network traffic where the affinity rule localizes traffic within the host's virtual switch, which reduces traffic on the physical network components. • Applications can share a large memory working set size where Transparent Page Sharing (TPS) can reduce the actual amount of memory used.

9.2 Enhanced vMotion Compatibility

Enhanced vMotion Compatibility (EVC) addresses the challenge that each generation of processor hardware presents through improvement with new functionalities. EVC makes sure that all host servers, where slight variations in processor generation exist, offer the same CPU instruction sets to virtual machines, safeguarding vSphere vMotion compatibility among the mixed processors.

EVC employs a baseline across the CPU instruction sets and this baseline enables a set of functionalities supported by all processors found in the cluster, defining a common level for all processors.

Figure 33. Enhanced vMotion Compatibility



The ESXi hypervisor works directly with Intel VT Flex Migration and AMD-V Extended Migration processors and technologies to show only the common instruction sets and hide those that would create vSphere vMotion compatibility problems. For further information on which baseline and instruction sets are masked, see the VMware Knowledge base article *Enhanced vMotion Compatibility (EVC) processor support (1003212)* at http://kb.vmware.com/kb/1003212.

Note, however, that EVC only supports the use of a single vendor's processors within the cluster (Intel or AMD), and the NX/XD functionality must be activated in the server hardware's BIOS.

The CPU is the most restrictive component when it comes to vSphere vMotion. Enabling EVC permits the masking of some differences, offering more compatibility between servers with different generations of processors. When employed, EVC does not hinder performance and does not affect the number of processor cores or the size of the CPU cache. The only possibility of degraded performance is where certain processor instructions, such as SSE 4.2 for instance, have been masked and therefore not used.

The design decision on whether or not to enable EVC will depend on a number of questions that you will need to address:

- Do you intend to mix hardware in the cluster?
- Do you think there is a high possibility of adding newer hardware to your cluster?
- If cross vCenter vSphere vMotion or long-distance vSphere vMotion forms part of a design or service offering, do you know what hardware types and processer generations exist at the other target data centers?

If the answer to any of these questions is yes, the general advice is to enable EVC on all clusters. Doing so makes adding hosts easier and saves having to split the cluster in the future if you have the foresight to enable it from the outset.

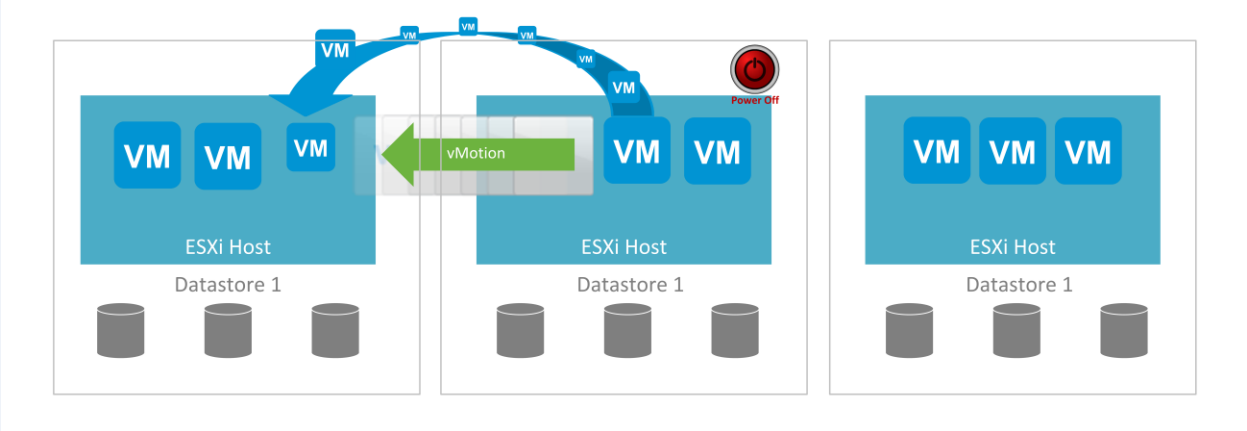
In a world where the cluster is no longer the boundary for live migration, and vSphere vMotion can occur across not only different vCenter Server instances, but also geographically dispersed physical data centers, EVC takes on a new meaning.

In the past, enabling EVC has often been overlooked by service providers and VMware enterprise customers because all nodes within the cluster housed an identical processor type, and was as such deemed unnecessary. However, with the introduction of cross vCenter vSphere vMotion and long-distance vSphere vMotion, where the local data center cluster is no longer the limiting factor for performing a vMotion operation, enabling EVC at the outset of a design is a logical step.

9.3 Distributed Power Management

The vSphere DPM feature significantly reduces the power consumption used in the data center by consolidating virtual machines so that the minimum number of ESXi hosts run during periods of low utilization, such as overnight. DPM works in conjunction with DRS to migrate virtual machines from hosts with low utilization to reduce the number of required hosts, powering down unused hosts automatically, which in turn reduces power consumption through both lower hardware power draw and lowered costs associated with air-conditioning units.

Figure 34. vSphere Distributed Power Management



The DPM mechanism takes advantage of remote power-on and power-off technologies such as Intelligent Power Management Interface (IPMI) or out-of-band remote access cards such as the HP Integrated Lights-Out (iLO) or, alternatively, the Wake-on-LAN (WoL) functionality embedded into network interface cards.

Service providers benefit from DPM where workloads vary significantly over time, reducing operational costs with only a minimal initial setup overhead. The design decision to support DPM in the infrastructure requires the service provider to consider:

- Purchasing server hardware that includes an iLO or IPMI network interface.
- Only running DPM during non-business hours because consumers of service might not want to shut down servers during business hours.
- If the host platform does not support iLO or IPMI, purchase NICs for the vSphere vMotion network that supports WoL functionality, although there are configuration prerequisites to use WoL.
- Configuring the vSphere DPM automation level for automatic operation, and using the default vSphere DPM power threshold, which decreases power and cooling costs, and administrative overhead.

Designing Host Security for Multitenanted Clouds

The hypervisor is the construct underlying the integrity of the tenant's virtual machines. Protecting it effectively is of the utmost importance. Any compromise of the hypervisor could seriously affect the virtual machines it hosts, leading to performance issues, data corruption, data loss, or even data exposure. In addition, because these attacks can occur below the guest operating system, they can be challenging to detect and have a much larger impact.

For instance, the effect of a Denial of Service (DoS) attack against an ESXi host is magnified because it affects all the virtual machines running on it. On a classic physical server, a DoS attack that monopolizes the system's CPUs affects only that host. However, in a virtual infrastructure, that same attack against an ESXi host can starve the physical CPUs of resources, affecting all virtual machines hosted on that hypervisor. In addition, if that host is part of a vSphere cluster and virtual machines are relocated as a result of the attack, this might impact all hosts, and in turn, all virtual machines in the cluster, leading to serious performance degradation of the entire infrastructure.

The next consideration is protection of the virtual machines. Protecting just the hypervisor is not enough. The virtual machines themselves must be secured in the same way as a classic physical server. While in a vSphere infrastructure, virtual machines are isolated in terms of having a separate guest instance on top of a dedicated VMM, they still communicate with each other over the network, the same way other hosts do. Unfortunately, traditional mechanisms, such as physical firewalls, would have had limited effectiveness, because much of the intercommunication takes place on the hypervisor itself. Fortunately, VMware NSX mitigates many of these risks.

In the following sections, we address some of the security measures that can provide protection on complex service-provider, multitenanted platforms.

10.1 Hypervisor Secure Communication

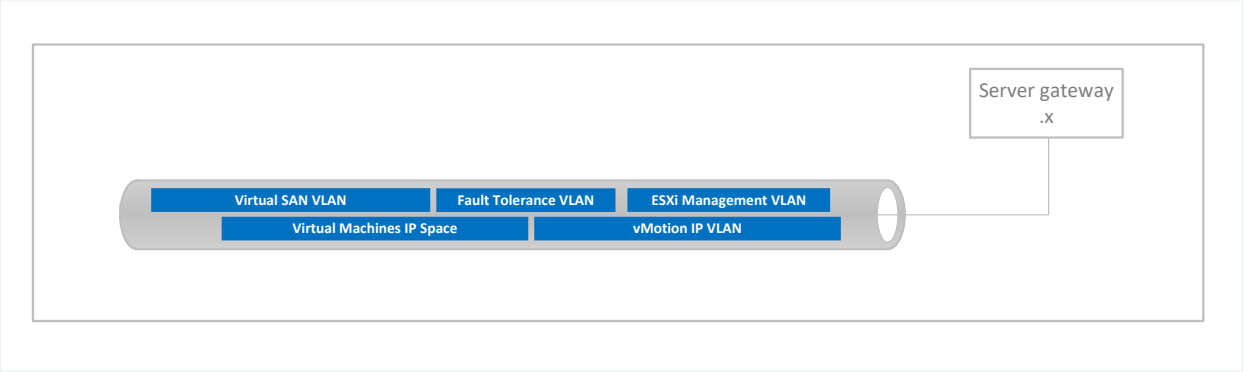
Several of vSphere capabilities require components to communicate over management, or other dedicated networks. Because networks that provide management communication provide direct access to core functionality, VMware recommends that ESXi host management communication only occur over a dedicated and isolated network segment.

Communication between vCenter Server and each host is encrypted by default using standard X.509 version 3 certificates. However, other traffic types described in this section are not encrypted by default, making it necessary to evaluate any requirement on data traversing the network for these components to be made secure and unreadable.

Consider other traffic flows from a security perspective, including vSphere vMotion traffic (which is now no longer tied to a L2 data center LAN), vSphere FT, and storage traffic, such as iSCSI, NFS, replication, or vSAN.

These traffic types must be isolated and strongly secured from all other traffic flows going to and from virtual machines. The most typical way of achieving this is by isolating networks through the creation of separate VLANs for each traffic type. This way, virtual and physical switches can be shared as long as traffic remains logically isolated for tenant virtual machines.

Figure 35. Network Segmentation



The principle goals of the service provider in securing hypervisor communication include:

- Verifying that tenants or external attackers cannot gain privileged access to the hypervisor through services running on the management, backup, or other secure networks.
- Ensuring tenants or external threats cannot sniff vSphere vMotion, vSphere FT, vSAN, or other privileged traffic to obtain memory or file system contents of a virtual machine or other data that could assist in the staging of a man-in-the-middle attack.
- Making sure that all replicated storage or storage access traffic, which is typically not encrypted, cannot be viewed by anyone.

Despite the need to protect the VMware vCloud platform from attacks, appropriate operational teams still need access to the vCenter Server instances and ESXi hosts. However, rather than configuring direct access to these protected networks, consider limiting access only through a virtual private network (VPN) or through the configuration of “jump boxes” that reside on the management network.

As highlighted previously, a management cluster also provides resource isolation and can satisfy the requirement to have physical isolation between management and tenant workloads. This further protects access to the management virtual machines running monitoring, management, and cloud platform services.

10.2 Certificate Configuration and Usage

On any production platform, the self-signed certificates generated when you install the components must always be replaced with those from a verifiable trusted Certificate Authority (CA) to enable certificate verification on all internal vSphere and external client connections.

Self-signed certificates are vulnerable to man-in-the-middle attacks and are unlikely to comply with your organization’s security policy. In vSphere 6.0, the Platform Services Controller hosts the VMware Certificate Authority (VMCA) and the VMware Endpoint Certificate Store (VECS). The VMCA and VECS are part of the Platform Services Controller, and take full advantage of the multi-master replication model that is offered by the Directory Service (VMDir) to provide high availability.

The VMCA is a Certificate Authority that allows you to:

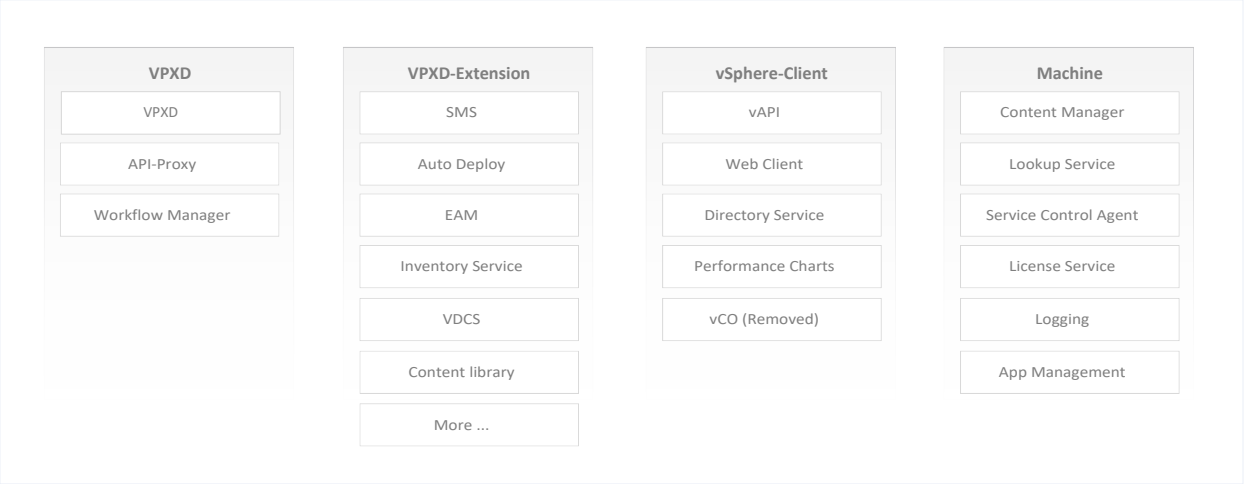
- Generate certificates
- Generate CRLs
- Use the UI
- Use the command-line interface to replace certificates

The VECS is where the certificates within the Platform Services Controller are stored, with exception of the ESXi certificates, which are stored locally on the individual vSphere hosts. The VECS can:

- Store certificates and keys
- Sync trusted certificates
- Sync CRLs
- Use the UI
- Use the command-line interface to perform various actions

In previous vSphere releases, every service had its own user and as such required its own certificate. However, this is no longer the case because vSphere now uses solution users. In vSphere 6.0, there are four main solution users that hold the certificate used for a number of services.

Figure 36. Certificate Solution Users

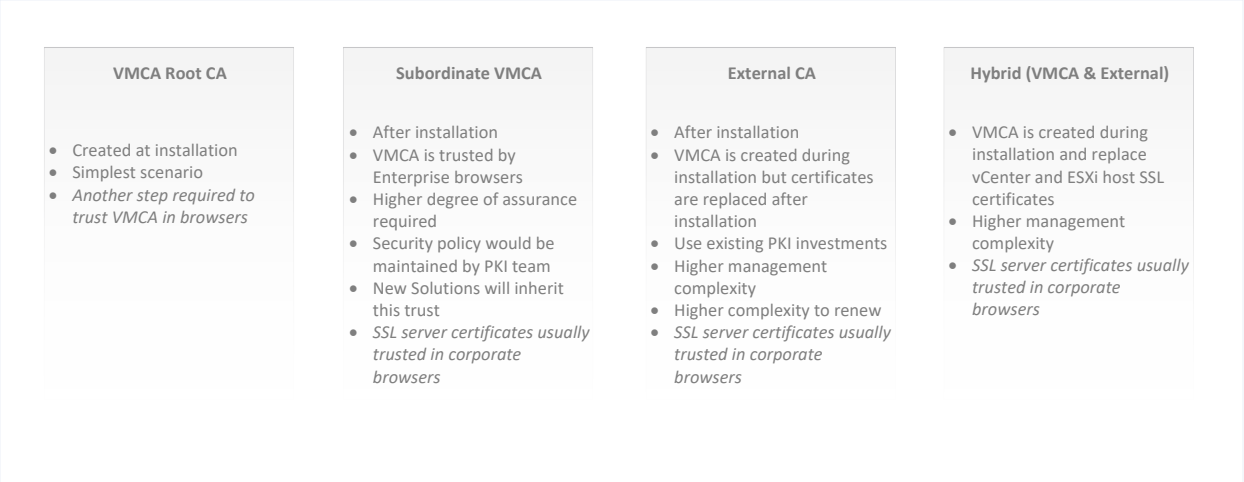


The VMCA can pose as either a root certificate authority or a subordinate certificate authority that issues signed certificates to all vSphere 6.0 components through the solution users. This secures the environment by using a centralized CA to generate certificates, as opposed to using self-signed certificates as in previous releases.

Because the VMCA can also be configured as a subordinate CA, it provides the ability to issue certificates based on an existing enterprise PKI. Because most service providers and enterprises already have an existing key infrastructure, non-disruptively adding the VMCA as a subordinate protects the investment already made. Allowing service providers that already have an investment in a PKI to incorporate the VMCA into their existing infrastructure adds a level of integration over certificate management that is applicable to a large number of use cases, including simplifying certificate lifecycle management in the vSphere environment.

Additional VMCA use cases are illustrated in the following figure.

Figure 37. VMware Certificate Authority Use Cases



10.3 Local Account Management

It is likely that occasionally, some advanced configuration and troubleshooting of an ESXi host might still require local privileged access through the classic C# client, or console access through the DCUI or SSH. While better managed through Active Directory integration, local host user accounts are often required, if for no other reason than "in case of emergency" accounts.

Previous releases of vSphere required you to create local accounts on each host corresponding to business and security requirements. However, in vSphere 6.0, you can manage local accounts on the hypervisor using new ESXCLI commands, providing the ability to script and inject such configuration during deployment. The new ESXCLI commands allows us to add, list, remove, and modify accounts across all hosts in a cluster from the vCenter Server, whereas previously the account and permission management functionality for ESXi hosts was only available with direct host connections. Setting, removing and listing local permissions on ESXi servers can also be centrally managed.

However, despite this simplification of administration of local host accounts which strengthens the security of the vSphere platform, the number of local user accounts created on the hypervisors must be limited to those that are absolutely essential.

In addition, in previous versions of ESXi, local host account password complexity changes had to be made manually by editing the `/etc/pam.d/passwd` file on each ESXi host. In vSphere 6.0, this has been moved to an entry in the hosts "Advanced System Settings," enabling centrally managed changes for all hosts in a cluster.

10.4 Host Active Directory Configuration Status

ESXi hosts can be joined to Active Directory, or more precisely, can use Active Directory for authenticating users, which allows for assigning permissions to domain users at host level. The advantage of this is that you can manage user accounts using Active Directory for authentication, authorization, and compliance, which is significantly easier and more secure than trying to manage local accounts.

VMware recommends taking advantage of this functionality by employing the use of a tenant administrative or federated domain as a security repository for permitting easy authentication and authorization with unique administrative credentials.

Using the pre-created Active Directory group “ESXi Admins” provides root access to authorized administrators as well as a way to audit direct access to the ESXi hosts.

For more information, refer to the *VMware vSphere Security* document at http://pubs.vmware.com/vsphere-60/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-60-security-guide.pdf.

10.5 Authentication Proxy

Authentication proxy is particularly useful for stateless ESXi hosts deployed from vSphere Auto Deploy, where the hypervisor configuration is provided by a host profile. The authentication proxy enables ESXi hosts to join a domain without requiring AD credentials, and as such, removes the need for Active Directory credentials to be provided in the host profile in plain text.

10.6 Transparent Page Sharing Security

Transparent Page Sharing (TPS) is a mechanism used by the VMkernel for more effective use of physical memory resources by only storing once memory pages that are identical on two or more virtual machines. Each virtual machine has only read access to shared memory pages, and as soon as a virtual machine tries to modify a shared page, a new private copy is created by the VMkernel.

Shared pages are most commonly seen when a host is running multiple virtual machines with the same guest operating system. However, the advent of more and more x64-based guest operating systems, where the guest leverages large page tables, has seen the effective benefits of TPS being significantly reduced.

More recently however, prompted by academic research in 2014 that leveraged TPS to gain unauthorized access to data under certain highly-controlled conditions, changes to the default TPS settings in the ESXi update releases of Q4 2014 and Q1 2015 were made.

Although VMware believes the risk of TPS being used to gather sensitive information is low, so that ESXi ships with default settings that are as secure as possible, TPS management options have been introduced and inter-virtual machine TPS is no longer enabled by default. Administrators are able to easily revert this new default behavior to enable TPS if they want to do so.

For more information with regard to TPS security and any concern you might have, refer to the following knowledge base articles:

- *Security considerations and disallowing inter-Virtual Machine Transparent Page Sharing (2080735)* at http://kb.vmware.com/kb/2080735
- *Additional Transparent Page Sharing management capabilities and new default settings (2097593)* at http://kb.vmware.com/kb/2097593

10.7 SNMP Hardware Monitoring

In addition to all the monitoring options available to hosts, ESXi includes an SNMP v3 daemon that can both send SNMP and receive polling requests to send SNMP traps for monitoring the server’s hardware and virtual machines. In addition, vCenter Server can be configured to send SNMP traps, which are typically sent to another management system.

Nothing in either ESXi or vCenter Server SNMP provides you with anything that you cannot already discover through the VMware vSphere API, PowerShell, vCenter Server CIM monitoring, or syslog. However, the primary situation in which SNMP might still be implemented is when a pre-existing SNMP management application within the service provider’s infrastructure is already in place, and the organization wants to continue with its use.

10.8 Host Lockdown Mode

Enable lockdown mode to increase security of ESXi hosts and to further mitigate the risk of unauthorized access to the ESXi console by limiting it to only the appropriate operational team through vCenter Server.

In previous releases of vSphere, there was one version of lockdown mode. However, with the release of vSphere 6.0, two different lockdown mode options exist.

With "normal lockdown mode," no users other than the `vpxuser` account have authentication permission. So other accounts cannot perform any actions against a host directly. Normal lockdown mode forces all operations to be carried out through the vCenter Server, although DCUI access is *not* stopped, and users on the `DCUI.Access` list can access the DCUI. However, in vSphere 6, "strict lockdown mode," the DCUI access is stopped.

vSphere 6 also introduces a new functionality called "Exception Users." These can be local accounts or Active Directory accounts with permissions defined locally on the host, where these users have host access. Exception Users are not recommended for general user accounts but are recommended for use only by third-party applications. For example, "Service Accounts," where the host needs access when either normal or strict lockdown mode is enabled. Permissions on these accounts must be configured as the absolute minimum required for the application in question to carry out its task and with an account that needs only read-only permissions to the host.

10.9 ESXi Firewall

Since the release of vSphere 5, the ESXi hypervisor has included a firewall that is enabled by default to only allow the incoming and outgoing connections that are necessary for managing virtual machines. The ESXi firewall allows for low-level control over network access, provides the ability to restrict access to specific network segments, and is centrally configurable through host profiles to reduce the operational overhead of managing changes globally within the virtual infrastructure.

10.10 Compute Component Patching

Maintaining an up-to-date IT infrastructure is critical for the health, performance, and security of the entire environment, and host and vSphere component patching constitutes one part of this task. This maintenance, while a daunting undertaking for operational teams, if not performed dependably and routinely, puts the entire platform at risk.

Verify that the scheduled and emergency patches protect systems from security vulnerabilities, as well as provide stability of performance. As with other systems, vSphere components must be maintained and devices patched and updated in line with the service provider's internal policies and customer SLAs.

The patch and update process must include the following practices:

- Documentation on the version of each hardware and software component within the environment.
- Documentation on risk acceptances for patches delayed or not installed in a timely manner.
- Research done to mitigate or to reduce the risk when patches cannot be installed.
- Following change management procedures to provide appropriate documentation and internal approvals.
- Establishing regular patch cycles for high and for low priority patches (for example, weekly and monthly).
- Establishing and testing of processes for emergency out-of-cycle patching.
- Ensuring that virtual machines are re-patched if restored from a snapshot prior to a scheduled patch date.

VMware vSphere Update Manager™ does not support the patching of vCenter Server or the Platform Service Controller. Therefore, administrators need to routinely check for, and evaluate new vCenter

Server and vSphere management component updates. These must be installed in a timely fashion following their release and proper internal testing.

Further guidance with reference to the implementation of vSphere Update Manager is provided in Section 11, Host Management.

10.11 ESXi Logging Service

By default, in ESXi, logs are stored on a local scratch volume or in RAM disk, depending on the host's installation device and configuration. To preserve the logs in a centralized location, the ESXi hosts and other devices must be configured to send their logs across the network directly to a central syslog server or alternatively, to a syslog aggregation server, which in turn forwards the syslog messages to the centralized location.

With each ESXi host generating a large number of component logs, on an average day with default logging settings, ~250 MB of data per host, even with a relatively small number of hosts, querying this log data when troubleshooting a problem quickly becomes very difficult, and correlating the information even more so. When, as a service provider, you are maintaining hundreds or even thousands of hosts across multiple geographically dispersed data centers in different regions, effective log management becomes paramount.

VMware recommends logging messages from the VMkernel, and other system components, to a centralized syslog target such as vRealize Log Insight. For more information on designing a multisite vRealize Log Insight infrastructure, refer to the *Designing an Enterprise Syslog Infrastructure with VMware vRealize Log Insight* white paper at http://www.vmware.com/files/pdf/Designing-an-Enterprise-Syslog-Infrastructure-with-VMware-vRealize-Log-Insight.pdf. This white paper outlines design options for every aspect of a local or distributed syslog design and provides several sample design scenarios.

Previously when an administrator executed an action from vCenter Server against an ESXi server, the administrator's user name would not be logged in the ESXi logs. The action would be logged as `vpxuser`. However, in vSphere 6.0, the user name that the administrator is logged into vCenter Server as is now included in the logs of the action that executes against ESXi.

Figure 38. vSphere Audit Trail Logging

This new functionality provides better forensics and auditing. In vSphere 6.0, all actions, including parent actions taken on vCenter Server and child actions run on ESXi hosts for user CORP\smithj (for instance), can be tracked in vRealize Log Insight and other logging solutions. Matching user names to actions provides accountability, auditing, and forensics and is a key requirement of compliance objectives.

10.12 ESXi Host Hardening

To provide an ESXi security baseline, consider the requirements for hardening the hypervisor. VMware guidance on security hardening and the recommendation level depends on the rating that corresponds to the operational environment in which it is to be applied. Each service provider will need make their own determination as to the applicability of each level.

VMware provides the following three baseline levels for hardening:

- **Enterprise Security Level 1 (Enterprise L1)** – This includes most enterprise production environments. The recommendations are meant to protect against most security attacks and provide protection of confidential information to the level required by all major security and compliance standards.

- **Specialized Security Level 2 (SSL2)** – This includes environments that are particularly susceptible to targeted attacks. Examples include: Internet-facing hosts and internal systems with highly confidential or regulated data.
- **Specialized Security Level 3 (SSL3)** – This represents unique and specialized environments that have some aspects that makes them especially vulnerable to attacks. Recommendations at this level might result in loss of (ease-of-use) functionality or purposefully cause the inability to use certain features. Careful consideration must be given to determining the applicability of these recommendations, including the possibility of using alternate compensating controls.

For instance, based on the provider's security policy, the following configuration steps to harden each host could be taken during the implementation phase of a new multitenant platform.

Table 24. Sample Host Hardening Configuration

Configuration	Description
Enabling the ESXi normal lockdown mode to prevent root access to the hosts over the network	Lockdown mode is enabled on each ESXi host. All configuration changes to the vSphere environment must be made by accessing the vCenter Server. Lockdown mode restricts access to host services on the ESXi server, but does not affect the availability of these services.

Disabled managed object browser (MOB)	The managed object browser provides a way to explore the VMkernel object model. Attackers can use this interface to perform malicious configuration changes or actions.
	Use the following ESXi shell command to determine if MOB is enabled:
	`vim-cmd proxysvc/service_list`
	`(vim.ProxyService.NamedPipeServiceSpec) {`
	`dynamicType = <unset>,`
	`serverNamespace = "/mob",`
	`accessMode = "httpsWithRedirect",`
	`pipeName = "/var/run/vmware/proxy-mob",`
	If MOB is enabled, use the following ESXi shell command to disable the MOB:
	`vim-cmd proxysvc/remove_service "/mob" "httpsWithRedirect"`

The VMware security hardening guides provide detailed guidance for customers on how to deploy and operate VMware products in a secure manner. For more information on ESXi host hardening, refer to the VMware Security Hardening Guides at https://www.vmware.com/security/hardening-guides.

Host Management

VMware vCenter Server and its supporting services are at the heart of the vSphere and vCloud platform. Within the vSphere infrastructure, vCenter Server is employed to provide the following functionality:

- Cloning of virtual machines
- Creating templates
- vSphere vMotion and Storage vMotion
- DRS and initial configuration of vSphere high availability clusters

vCenter Server also provides monitoring and alerting capabilities for hosts and virtual machines. System administrators can create and apply alarms to all managed objects in vCenter Server. These alarms include:

- Data center, cluster and host health, inventory and performance
- Datastore health and capacity
- Virtual machine usage, performance and health
- Virtual network usage and health

vCenter Server 6.0 simplifies the planning and deployment from previous releases by offering only two deployment components:

- Platform Services Controller
- vCenter Server

All vCenter Server services, such as Inventory Service, Web Client, vSphere Auto Deploy, and so on are installed along with vCenter Server Manager. There are no longer separate installers for these components, simplifying the architecture by combining multiple functions onto a single server. One exception remains, vSphere Update Manager currently continues as a standalone Microsoft Windows installation.

The Platform Services Controller combines common services across the vSphere infrastructure such as single sign-on, licensing, and certificate management. The Platform Services Controller replicates information such as licenses, roles and permissions, and tags with other Platform Services Controllers. For further information on the configuration of these vCenter Server components, refer to the *VMware vSphere 6 Documentation* at https://www.vmware.com/support/pubs/vsphere-esxi-vcenter-server-6-pubs.html.

While more and more host and virtual machine administration is being moved away from vCenter Server up the management stack to applications, such as vCloud Director, VMware vRealize Orchestrator™, VMware vRealize Automation, or VMware Horizon™ View™, vCenter Server remains central to ESXi host and virtual machine administration and this is unlikely to change anytime soon.

11.1 vCenter Server Appliance

The vCenter Server Appliance was new in vSphere 5 and is built on SUSE Linux Enterprise Server 11. Significant advancements were made in the 5.1 and 5.5 releases, but the 6.0 release has few limitations, with the vCenter Server Appliance now having the same scalability as the Windows-installable vCenter Server.

This scalability is supported with the embedded PostgreSQL database or with an external Oracle database. This enables service providers to choose the platform that best suits their specific use cases, without sacrificing vCenter Server performance or scalability.

Table 25. vCenter Server Installable Windows and Appliance Scalability

Metric	Windows vCenter Server	vCenter Server Appliance (Embedded VMware vFabric Postgres)
ESXi hosts per vCenter Server	1000	1000
Powered on VMs per vCenter Server	10,000	10,000
Hosts per cluster	64	64
Virtual machines per cluster	6,000	6,000
Linked mode	10 vCenter Server instances	10 vCenter Server Appliance instances
IP v6 support	Yes	Yes
Site Recovery Manager support	Yes	Yes
vSphere PowerCLI support	Yes	Yes

So that host resources are not wasted, the vCenter Server Appliance can be deployed in one of four scalable options. The following table outlines the different size options available during the initial deployment, which can easily be modified post deployment, if required.

Table 26. vCenter Server Appliance Deployment Scalability Options

Size	vCPU	vRAM (GB)	Hosts (Max)	VMs (Max)
Tiny	2	8	20	400
Small	4	16	150	3,000
Medium	8	24	300	6,000
Large	16	32	1,000	10,000

Now that the vCenter Server Appliance is a viable option for most deployment use cases and is, in most respects, equivalent to the Windows-installable option in terms of functionality and scalability, service providers must understand its advantages and drawbacks when choosing to employ it.

Table 27. vCenter Server Appliance Compared with Installable Windows vCenter Server

Advantages to the vCenter Server Appliance (vSphere 6.0)	Drawbacks to the vCenter Server Appliance (vSphere 6.0)
Do not need a windows license. Do not need an SQL license. Easy to deploy and easy to upgrade (built-in update function).	It is not possible to install VMware Update Manager (you must install it on a Windows virtual machine and then a Windows license is required). Plug-in support for third-party software might not be available. The only option for external database support is Oracle.

11.2 Physical or Virtual vCenter Server

The design decision on whether or not a Windows-installable vCenter Server must reside on a physical or virtual machine has been around as long as vCenter Server and virtualization itself.

Some vSphere administrators are concerned that if a virtual vCenter Server goes down, they would lose all centralized management of the virtual platform. This is true, however, if vCenter Server is down, that does not mean that the hosts and virtual machines are affected. However, if your vCenter is a virtual machine in a HA / DRS management cluster, it must be started back up very quickly on another host. In addition, with a virtualized vCenter Server, you can take advantage of advanced vSphere features, such as snapshots, vSphere vMotion, and Storage vMotion.

Another consideration is how vCenter Server has changed and will change in future. As we have seen already, there is now the option of taking a scale-out approach to the vCenter Server infrastructure by distributing the Platform Services Controller onto a different Windows installation, and in most instances, this would not be scalable in the physical world. Also, consider that the vCenter Server Appliance is gaining strength and is likely to be the primary deployment mechanism for vCenter Server very soon.

For these reasons, vCenter Server is typically deployed as a virtual machine and as our goal is to achieve 100 percent server virtualization, vCenter Server is no exception. If a physical vCenter Server is deployed, list it as a risk and a single point of failure for the service owner.

Table 28. Virtual vCenter Server Compared with Physical Server

Virtual Machine	Physical Machine
Easily backed up and restored by image level backup solution.	Traditional backup mechanisms must be employed.
Can be moved across hosts with vSphere vMotion allowing hardware maintenance without downtime.	Fixed location.
Performance / SLAs can be maintained by enabling DRS and vSphere Storage DRS.	Fixed to server hardware resource.
Can be protected by vSphere HA.	Expensive to provide HA. Requires vCenter Server Heartbeat (now end-of-life) or a third-party clustering solution.
Can be protected by vSphere SMP-FT.	No continually available mechanism.
Virtual machine CPU and memory resources can be resized easily.	Physical server must be correctly sized.
Can utilize virtual machine snapshots.	No snapshot mechanism available.

11.3 vCenter Server High Availability Options

High availability of your vCenter Server and supporting services will depend on whether you have opted for a physical machine, Windows virtual machine, or the SUSE Linux-based appliance. Whichever platform has been chosen, protecting the vCenter Server and its supporting infrastructure is crucial for managing and monitoring hosts. Multiple options exist, so as with all design decisions, the service provider and consumer requirements dictate the design options available. Key answers that must be obtained regarding the availability of vCenter Server are:

- How much downtime can be tolerated?
- Should failover be a manual or automated process?
- What is the cost or impact of vCenter Server management service downtime?

For some environments, such as those using View or a service provider's vCloud Director platform, where not even a minute or two of vCenter Server downtime can be tolerated, the architect is required to examine the options closely.

Table 29. vCenter Server Virtual Machine Availability Options

Availability Method	Windows vCenter Server	Windows Platform Services Controller Server	Windows vSphere Update Manager Server	Windows SQL Server	vCenter Server Appliance
vSphere HA	Yes	Yes	Yes	Yes (risks DB corruption or non-consistent state)	Yes
vSphere SMP-FT	vSphere SMP-FT	Supported only in specific user cases	Supported only in specific user cases	Yes	Yes
Cold Standby VM and manual failover	Yes	No	Yes	No	Yes
Microsoft failover clustering	No	No	No	Yes	No
Third-party solutions	Yes	Yes	Yes	Yes	No
Application integrated log shipping	N/A	N/A	N/A	Yes	No
Active / passive load balanced configuration	No	Yes	No	No	No

While cloning and manual failover is a viable option, it will typically take significantly longer to recover from than SMP-FT or vSphere HA. vCenter Server is essentially stateless. However, with this manual approach, it takes time to not only create the initial cold clone, but also to keep the redundant server up-to-date. It is also possible to keep a cold standby virtual machine of a physical vCenter Server by using physical-to-virtual conversion software to create the clone.

11.4 Role-Based Access Control

Role-based access control (RBAC) can be employed in a number of different scenarios for VMware Cloud Providers. Internally, from an operational perspective, to verify that only the appropriate operational or administrative teams have access to internal and tenant environments. Also, from the perspective of giving tenants access to either shared or dedicated vCenter Server instances or to hosts directly, allowing for self-managed hosting use cases. No matter what use cases are defined by the service provider, if proper role-based access controls are not in place within the environment, virtual machines will be vulnerable.

The following is a list of common and best practices for designing RBAC for the VMware Cloud Provider Program service provider platform:

- Any vSphere administrative permission granted to a user account must use a privileged account (PA).
- All permissions must be assigned to AD groups and not individual user accounts. Do not use local accounts or groups for day-to-day administration.
- Grant permissions only where needed for job role, using the minimum number of permissions makes it easier to understand and manage the permissions structure.
- Create new groups for vCenter Server users. Avoid using Windows built-in groups or other pre-existing groups.
- If you assign a restrictive role to a group, check that the group does not contain the Administrator user or other users with administrative privileges. Otherwise, you could unintentionally restrict administrators' privileges in parts of the inventory hierarchy where you have assigned that group the restrictive role.
- Use folders to group objects to correspond to the differing permissions you want to grant for them.
- Use caution when granting permission at the root vCenter Server level. Users with permissions at the root level have access to global data on vCenter Server, such as roles, custom attributes, vCenter Server settings, and licenses. Changes to licenses and roles propagate to all vCenter Server systems in a linked mode group, even if the user does not have permissions on all of the vCenter Server systems in the group.
- In most cases, enable propagation on permissions. This makes sure that when new objects are inserted in to the inventory hierarchy, they inherit permissions and are accessible to users.
- Use the No Access role to mask specific areas from the hierarchy that you do not want particular users to have access to.
- Certain privileges can be harmful to hosts and should be assigned to users only when required. This includes any privilege that allows a user to delete, rename, remove, or create items that can cause data loss or datastores to be filled up. This can cause a denial of service attack on your VMs (for instance, prevent snapshot creation).

- Create roles that are customized to a user's/tenant's requirements. For example, to create a role for an operations team that is responsible for monitoring VMs, create one that allows only VM interactions (for instance, power on, power off, reset, and console interaction). This allows team members to look at the console of a VM to see what is happening and power-cycle a VM.
- A privilege that you assign sparingly is the datastore low-level file operations privilege, which allows users to upload and download files to a host datastore. This privilege can create a security risk.
- Other potentially dangerous privileges are in the network and virtual switch categories, which can allow a user to move a VM to any available virtual LAN that is configured on your virtual switches. This can be particularly risky if you have public and private network virtual switches on a host where you definitely do not want a VM moved between them or connected to both at the same time (for instance, edge cluster hosts). Assigning the network privileges to your network admins and denying them to everyone else is a good practice.

Designing a vCenter Server Ecosystem

With vSphere 6.0, a number of changes relating to vCenter Server architecture for both the vCenter Server and vCenter Server Appliance were made. For this reason, some design aspects and the deployment use cases have changed from previous releases. However, some considerations remain the same, such as database placement, choosing between a physical or virtual deployment, or designing an effective highly available architecture.

12.1 Platform Services Design

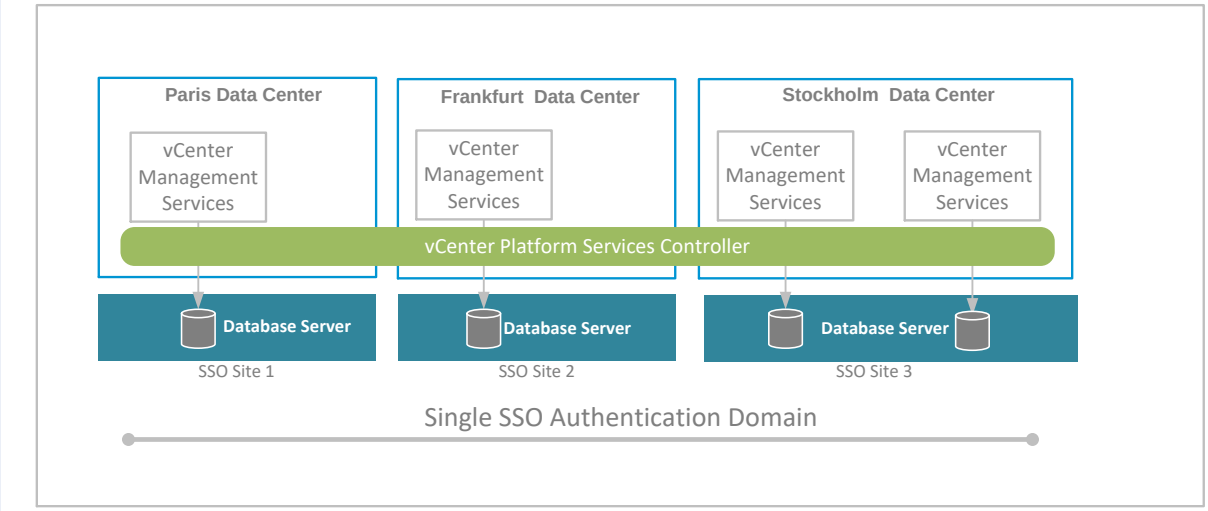
The vSphere 6.0 Platform Services Controller is made up of more than only the Single Sign-On component. The Platform Services Controller includes a set of common infrastructure services that are used by vSphere, but also other products, such as vRealize Operations, vCloud Director, and vRealize Automation.

Figure 39. Platform Service Controller Components

When installing vCenter Server you have the option to embed the Platform Services Controller on the same server as vCenter Server (single system) or to deploy it externally (multiple systems). The use case options provide maximum deployment flexibility:

- vCenter Server with embedded Platform Services Controller (suitable for single system, but also for multi-system and multisite)
- vCenter Server with external Platform Services Controller (suitable for multiple systems in a single site or multisite design)

Figure 40. Platform Services Controller Design Example



12.1.1 Embedded Platform Services Controller

A single system that includes the vCenter Server Manager with embedded Platform Services Controller is sufficient for many use cases, and not only limited to small environments. This design is the easiest to maintain and deploy and is fully supported for a Windows or appliance vCenter Server deployment.

Installing vCenter Server with an embedded Platform Services Controller has the advantages and drawbacks described in the following table.

Table 30. Advantages and Drawbacks to an Embedded Platform Services Controller

Advantages to an Embedded Platform Services Controller	Drawbacks to an Embedded Platform Services Controller
The connection between vCenter Server and the Platform Services Controller does not go over the network, and vCenter Server is not prone to outages because of connectivity and name resolution issues between vCenter Server and the Platform Services Controller. If you install vCenter Server on a Windows host machine, you will require fewer Microsoft Windows licenses. Less virtual machines to manage, reducing operational overhead. No requirement for a load balancer to distribute the load across Platform Services Controller systems.	There is a Platform Services Controller for each product, which might be more than is required. This will consume more host resources. The maximum number of Platform Services Controllers per site is eight. If you deploy multiple products with embedded Platform Services Controllers, after you install eight products at the same site, you have reached the maximum recommended number.

Typical use cases for the embedded model are for standalone sites, where this vCenter Server will be the only VMware vCenter Single Sign-On™ integrated solution, and replication to other Platform Services Controllers is not a design requirement. In general, there is a recommendation to deploy external Platform Services Controllers in any environment where there is more than one vCenter Single Sign-On enabled solution (vCenter Server, vRealize Automation, and so on), or where replication to other Platform Services Controllers, such as another site, is required in the design.

12.1.2 External Platform Services Controller

An external Platform Services Controller is typically suitable for customers with numerous vCenter Server instances who can reduce their footprint by sharing Platform Services Controllers across several vCenter Server instances and other Platform Services Controller enabled VMware products. Installing vCenter Server with an external Platform Services Controller has the advantages and drawbacks described in the following table.

Table 31. Advantages and Drawbacks to an External Platform Services Controller

Advantages to an External Platform Services Controller	Drawbacks to an External Platform Services Controller
Fewer resources consumed by the services in the Platform Services Controllers. Because the Platform Services Controllers are not embedded with every vCenter Server instance, you are less likely to reach the maximum of 8 Platform Services Controllers per site.	The connection between the vCenter Server and the Platform Services Controller goes over the network, and is, therefore, dependent on connectivity and name resolution. If you install vCenter Server and the Platform Services Controller on Windows virtual machines, you will need more Microsoft Windows licenses. More virtual machines to manage.

As we will see in the sample design that follows, it is not obligatory that a design consist of the same deployment types. It is perfectly possible to design a mixed environment, which consists of vCenter Server instances with both embedded and external Platform Services Controllers as well as both Windows and appliance-based vCenter Server and Platform Services Controller instances. As long as the Platform Services Controllers are in one vCenter Single Sign-On domain and replicate information with each other, when you log into the vSphere Web Client, you will see the inventories of all vCenter Server instances.

12.2 vCenter Server Management Services Design

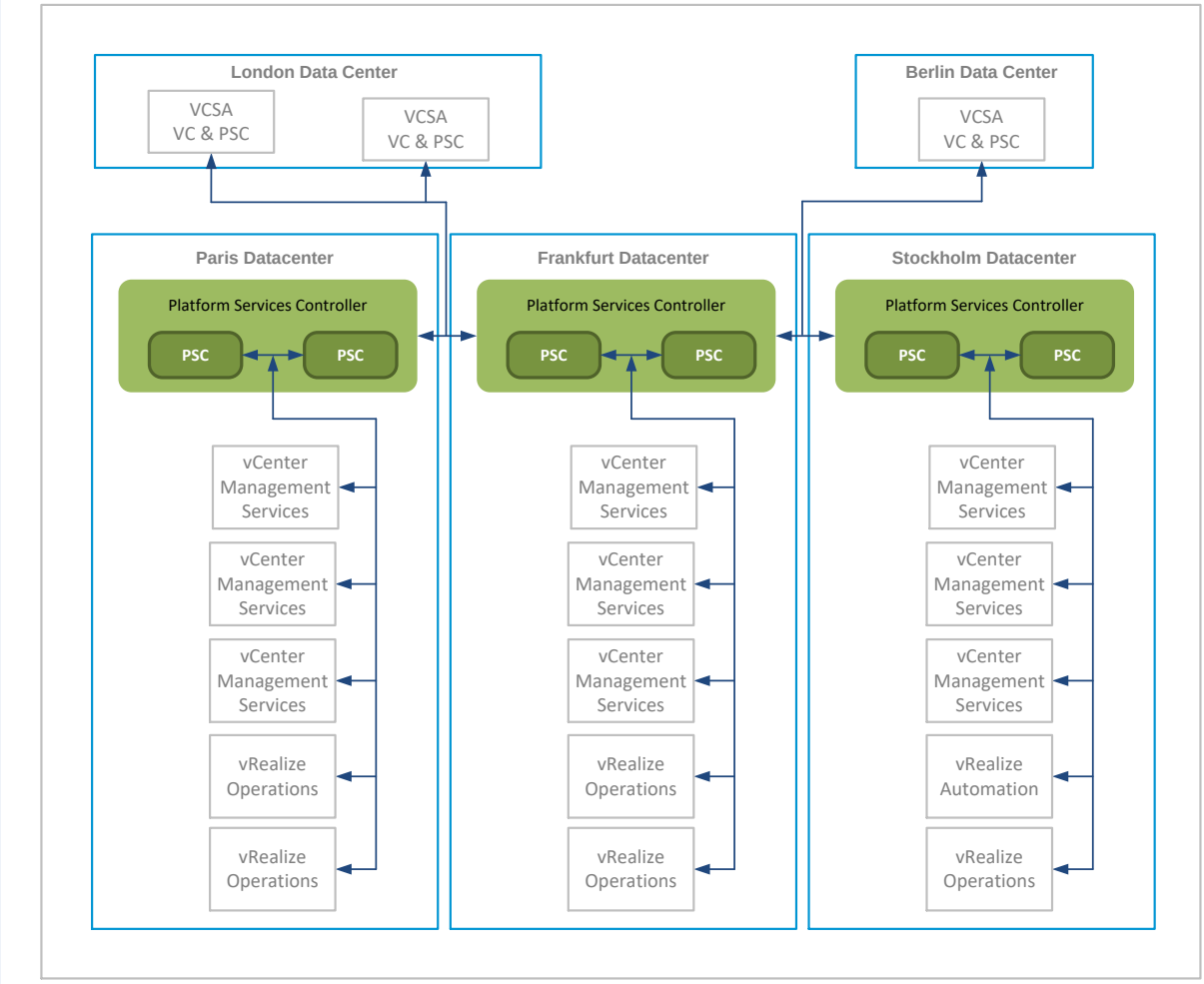
With vSphere 6.0, all additional vCenter Server services, such as the vSphere Web Client, Inventory Service, vSphere Auto Deploy, Syslog Collector, vSphere ESXi Dump Collector, and so on, are installed on the same server as the vCenter Server service. During an upgrade, should any of these services be installed on a different host server, the upgrade will pull the configuration from these services and apply it to the service installed on the vCenter Server 6.0 instance.

There is no longer a way to deploy these components on different servers from the vCenter Server. The exception to this is vSphere Update Manager, which, at the time of writing, remains a separate installation that can, and generally should, be installed on a different server. Where the vCenter Server Appliance has been deployed, the vSphere Update Manager server must still be installed on a separate Windows server and registered with the vCenter Server Appliance. Currently, vSphere Update Manager is a Windows-only service.

12.3 Sample Service Provider Deployment Scenario

The following deployment use case depicts a service provider architecture that spans five geographically separated data centers with a requirement to maintain a vCenter Single Sign-On domain. The sites with larger vSphere deployments host multiple load-balanced Platform Services Controllers supporting multiple vCenter Server instances with additional VMware products, such as vRealize Operations 6.0. Smaller sites have been deployed with a single vCenter Server Appliance with an embedded Platform Services Controller for an efficient, small footprint design.

Figure 41. Mixed vCenter Server Appliance and Windows Sample Platform Services Design



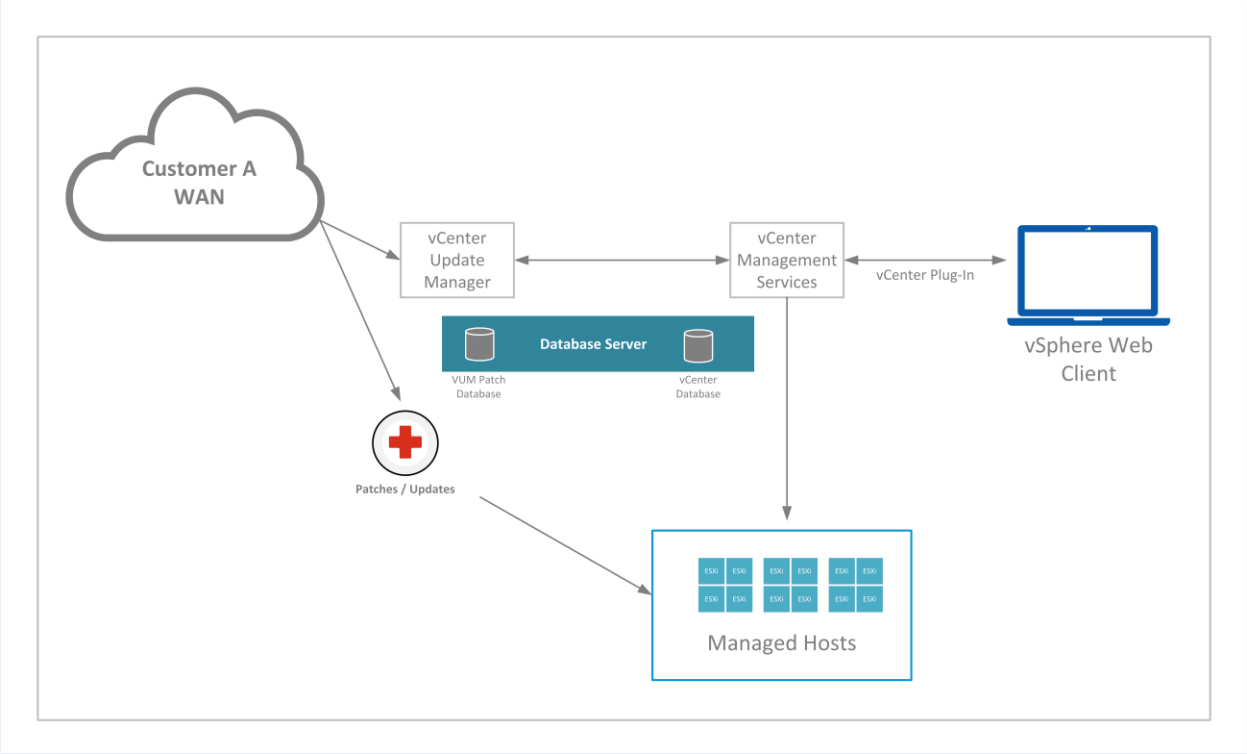
12.4 vSphere Update Manager

vSphere Update Manager is an enterprise patch management automation tool that is often implemented as part of a vSphere platform to keep the ESXi hosts, virtual machine hardware, and VMware Tools up-to-date. VMware recommends installing vSphere Update Manager on a separate virtual machine to enable future expansion and for improved security. Like vCenter Server, a dedicated database for vSphere Update Manager is required, which typically resides alongside the vCenter Server database in a Windows and SQL based design. There is no reason to host another database server simply for the vSphere Update Manager database.

At time of writing, vSphere Update Manager requires a separate installation on a Windows host even where the vCenter Server Appliance has been deployed. vSphere Update Manager has a one-to-one relationship with the vCenter Server it is registered to, so a dedicated vSphere Update Manager instance is required for each vCenter Server deployed.

vSphere Update Manager can also perform orchestrated vSphere upgrades. An orchestrated upgrade allows you to upgrade the objects in your vSphere inventory in a two-step process: host upgrades, followed by virtual machine upgrades. You can configure the process at the cluster level for higher automation, or at the individual host or virtual machine level for more granular control. You can upgrade clusters without powering virtual machines off, as long as vSphere DRS is available to the cluster. To perform an orchestrated upgrade, first remediate a cluster against a host upgrade baseline, and then remediate the same cluster against a virtual machine upgrade baseline group that contains the virtual machine hardware upgrade and VMware Tools upgrade to match host baselines.

Figure 42. vSphere Upgrade Manager Architecture



For further details of deployment of vSphere Update Manager, see *VMware Update Manager Documentation* at https://www.vmware.com/support/pubs/vum_pubs.html.

12.4.1 vSphere Update Manager Configuration

To determine database patch repository sizing, use the *VMware vSphere Update Manager Sizing Estimator* at http://pubs.vmware.com/vsphere-60/topic/com.vmware.ICbase/PDF/vsphere-update-manager-60-sizing-estimator.xls. For instance, you can base a vSphere Update Manager design on the following assumptions:

- No remediation of VMs
- Remediate ESXi 6.0+ hosts
- 1 concurrent ESXi host upgrade
- Patch scan frequency for hosts: 4 per month
- Upgrade scan frequency for hosts: 1 per month

The following table provides a sample design configuration for vSphere Update Manager. As with other platform design considerations, the goal for a service provider, who deploys multiple instances of vSphere Update Manager across multiple platforms, is to maintain consistency in order to simplify operational management of the environment.

Table 32. Sample vSphere Update Manager Configuration

Attribute	Specification
Patch download sources	Select Download ESXi 6 patches De-select Download ESXi 3, 4 and 5 patches or as appropriate for the environment
Shared repository	`D:\vCenterUpdateManager\vSpherePatches`
Proxy settings	None
Patch download schedule	Every Sunday at 12:00AM EST
Email notification	TBD by Customer
vSphere Update Manager baselines to leverage	Critical and non-critical ESXi host patches VMware Tools upgrade to match host
Virtual machine settings	N/A
ESXi host settings	Host maintenance mode failure: Retry Retry interval: 30 Minutes Number of retries: 3
vApp settings	N/A

Note vSphere Update Manager does not address the issue of vCenter Server or Platform Services Controller component patching. Administrators are required to evaluate patches and updates to vCenter Server and the Platform Services Controller and apply those manually as required.

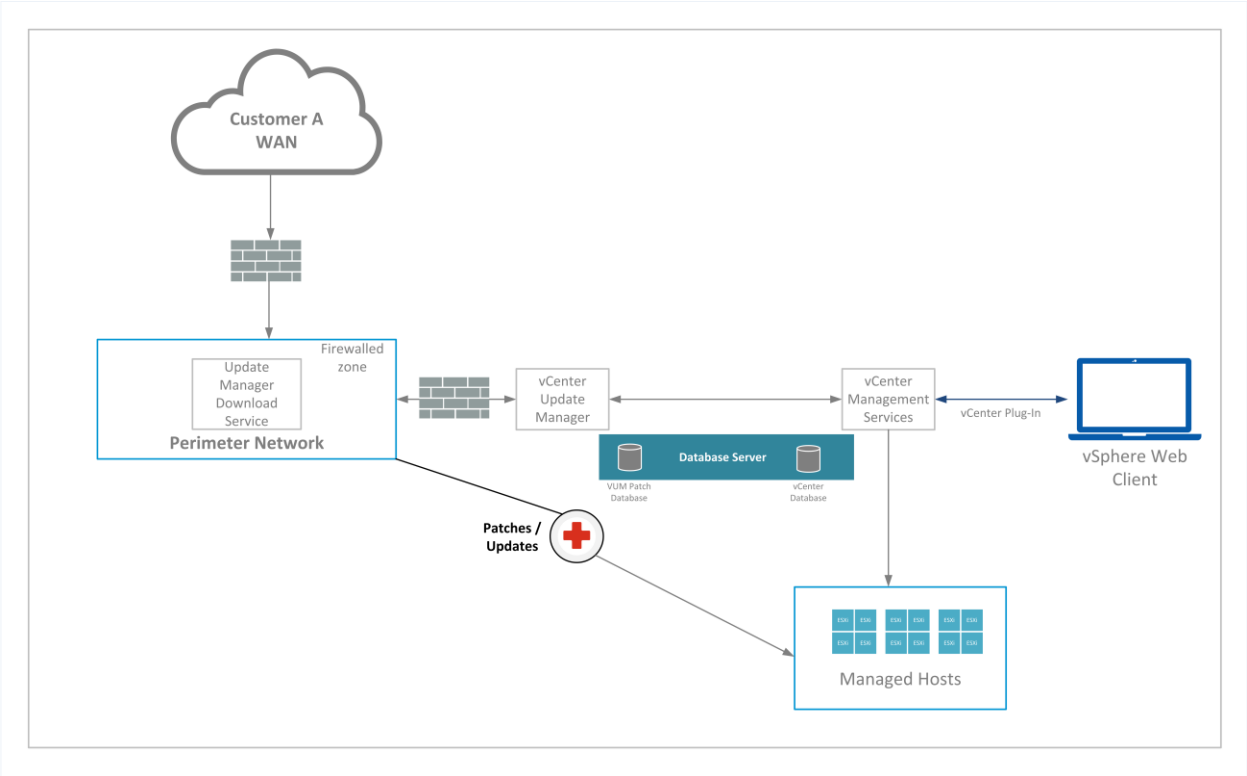
12.4.2 vSphere Update Manager Download Service

In some environments, the vCenter Server and vSphere Update Manager server might not have direct access to the Internet to download patches directly from vmware.com. For these use cases, the architecture could include the use of the vSphere Update Manager Download Service (UMDS), in order to download patches and updates first to a staging location on a perimeter network.

To implement the UMDS, first create the shared repository using UMDS and host it on a web server or a local disk. The UMDS must be running a version compatible with vSphere Update Manager, but cannot co-exist on the same server as vSphere Update Manager. For further details, see *VMware vSphere Update Manager Documentation* at https://www.vmware.com/support/pubs/vum_pubs.html.

The following figure provides a logical overview of vSphere Update Manager Download Service and how this component meets specific security design requirements. The UMDS is a shared repository that can be accessed by multiple vSphere Update Manager instances.

Figure 43. vSphere Update Manager Download Server Architecture



12.5 vSphere Management Assistant Appliance

VMware vSphere Management Assistant is a Linux-based virtual appliance that comes pre-installed with a command-line interface (CLI), VMware Tools, vSphere SDK for Perl and CLI, a logging component, and vi-fastpass authentication. You can also use Active Directory for authentication.

The vSphere Management Assistant appliance stores ESXi host and vCenter Server management scripts and programs and allows you to access them from any SSH client. This appliance is often deployed in a management cluster to facilitate centralized and auditable command-line management of ESXi hosts and vCenter Servers.

12.6 VMware vCenter Support Assistant

The VMware vCenter Support Assistant™ is a vSphere Web Client plug-in that collects support bundles on a regular basis, analyzes the environment, and sends alerts and recommended fixes for potential problems.

The vCenter Support Assistant also provides an easy-to-use, timesaving application for filing and managing support requests and for generating and uploading vSphere and vCenter Server support bundles and other files to VMware technical support. VMware recommends that you deploy vCenter Support Assistant alongside the vCenter Server in the management cluster.

Operational Verification

The purpose of operational verification is to perform post-implementation configuration and redundancy testing of the accepted design. Furthermore, the aim is to demonstrate physical and virtual connectivity between all of the infrastructure components that form part of the vCloud platform and to validate that they are operating as expected.

The following figure illustrates the minimum operational testing to be carried out to validate the implementation of the compute components of the VMware Cloud Provider Program design.

Figure 44. Operational Verification of Compute Components