



# VMware® NSX Network Virtualization Design Guide

## Table of Contents

Intended Audience .....	3
Overview .....	3
Components of the VMware Network Virtualization Solution .....	4
Data Plane.....	4
Control Plane.....	5
Management Plane .....	5
Consumption Platform.....	5
Functional Services .....	5
Network Virtualization Design Considerations .....	6
Physical Network.....	6
Data Center Fabric Attributes.....	6
Simple .....	7
Scalable.....	8
High Bandwidth .....	9
Fault Tolerant .....	10
Differentiated Services – Quality of Service .....	11
Data Center Access Layer Deployment Scenarios .....	12
Layer 3 in the Data Center Access Layer.....	12
Compute Racks.....	12
Connecting Hypervisors .....	13
VXLAN Traffic.....	13
Management Traffic .....	14
vSphere vMotion Traffic .....	14
Storage Traffic.....	14
Edge Racks .....	14
Infrastructure Racks .....	17
VLAN Provisioning .....	17
Multitier Edges and Multitier Application Design Considerations .....	18
Logical Switching.....	21
Components.....	20
Logical Routing.....	24
Components.....	24
Logical Firewall.....	29
Logical Load Balancer.....	31
Conclusion.....	32

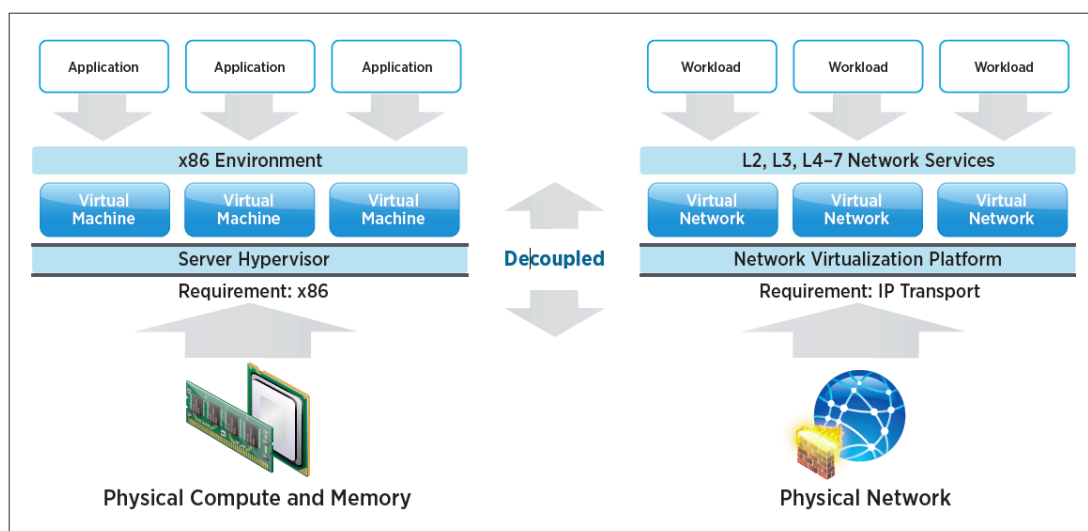
## Intended Audience

This document is targeted toward virtualization and network architects interested in deploying VMware® network virtualization solutions.

## Overview

IT organizations have gained significant benefits as a direct result of server virtualization. Server consolidation reduced physical complexity, increased operational efficiency, and the ability to dynamically re-purpose underlying resources to quickly and optimally meet the needs of increasingly dynamic business applications are just a handful of the gains that have already been realized.

Now, VMware's Software Defined Data Center (SDDC) architecture is extending virtualization technologies across the entire physical data center infrastructure. VMware NSX, the network virtualization platform is a key product in the SDDC architecture. With NSX, virtualization now delivers for networking what it has already delivered for compute and storage. In much the same way that server virtualization programmatically creates, snapshots, deletes and restores software-based virtual machines (VMs), NSX network virtualization programmatically creates, snapshots, deletes, and restores software-based virtual networks. The result is a completely transformative approach to networking that not only enables data center managers to achieve orders of magnitude better agility and economics, but also allows for a vastly simplified operational model for the underlying physical network. With the ability to be deployed on any IP network, including both existing traditional networking models and next generation fabric architectures from any vendor, NSX is a completely non-disruptive solution. In fact, with NSX, the physical network infrastructure you already have is all you need to deploy a software defined data center.



**Figure 1.** Server and Network Virtualization Analogy

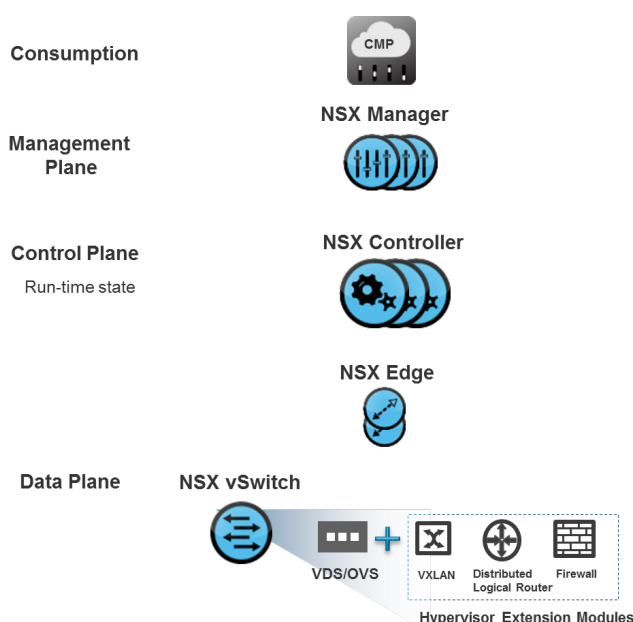
Figure 1 draws an analogy between compute and network virtualization. With server virtualization, a software abstraction layer (server hypervisor) reproduces the familiar attributes of an x86 physical server (e.g., CPU, RAM, Disk, NIC) in software, allowing them to be programmatically assembled in any arbitrary combination to produce a unique virtual machine (VM) in a matter of seconds.

With network virtualization, the functional equivalent of a “network hypervisor” reproduces the complete set of Layer 2 to Layer 7 networking services (e.g., switching, routing, access control, firewalling, QoS, and load balancing) in software. As a result, these services can be programmatically assembled in any arbitrary combination, to produce unique, isolated virtual networks in a matter of seconds.

Not surprisingly, similar benefits are also derived. For example, just as VMs are independent of the underlying x86 platform and allow IT to treat physical hosts as a pool of compute capacity, virtual networks are independent of the underlying IP network hardware and allow IT to treat the physical network as a pool of transport capacity that can be consumed and repurposed on demand. Unlike legacy architectures, virtual networks can be provisioned, changed, stored, deleted and restored programmatically without reconfiguring the underlying physical hardware or topology. By matching the capabilities and benefits derived from familiar server and storage virtualization solutions, this transformative approach to networking unleashes the full potential of the software defined data center.

With VMware NSX, you already have the network you need to deploy a next-generation software defined data center. This paper will highlight the design factors you should consider to fully leverage your existing network investment and optimize that investment with VMware NSX.

## Components of the VMware NSX Solution



**Figure 2.** VMware Network Virtualization Solution Components

### Data Plane

The NSX Data plane consists of the NSX vSwitch. The vSwitch in NSX for vSphere is based on the vSphere Distributed Switch (VDS) (or Open vSwitch for non-ESXi hypervisors) with additional components to enable rich services. The add-on NSX components include kernel modules (VIBs) which run within the hypervisor kernel providing services such as distributed routing, distributed firewall and enable VXLAN bridging capabilities.

The NSX vSwitch (VDS or OVS-based) abstracts the physical network and provides access-level switching in the hypervisor. It is central to network virtualization because it enables logical networks that are independent of physical constructs such as VLAN. Some of the benefits of the vSwitch are:

- Support for overlay networking with protocols such as VXLAN, STT, GRE and centralized network configuration. Overlay networking enables the following capabilities:
  - Creation of a flexible logical layer 2 (L2) overlay over existing IP networks on existing physical infrastructure without the need to re-architect any of the data center networks
  - Provision of communication (east-west and north-south) while maintaining isolation between tenants

- Application workloads and virtual machines that are agnostic of the overlay network and operate as if they were connected to a physical L2 network
- NSX vSwitch facilitates massive scale of hypervisors.
- Multiple features—such as Port Mirroring, NetFlow/IPFIX, Configuration Backup and Restore, Network Health Check, QoS, and LACP—provide a comprehensive toolkit for traffic management, monitoring and troubleshooting within a virtual network.

Additionally, the data plane also consists of gateway devices which can either provide L2 bridging from the logical networking space (VXLAN) to the physical network (VLAN). The gateway device is typically an

NSX Edge virtual appliance. NSX Edge offers L2, L3, perimeter firewall, load-balancing and other services such as SSL VPN, DHCP, etc.

The functionality of L2 bridging from virtual to physical networks can also be done by a physical network switch which supports the functionality to de-encapsulate VXLAN traffic.

### Control Plane

The NSX control plane runs in the NSX controller. In a vSphere optimized environment with VDS the controller enables multicast free VXLAN, control plane programming of elements such as VDR. In a multi-hypervisor environment the controller nodes program the vSwitch forwarding plane.

In all cases the controller is purely a part of the control plane and does not have any data plane traffic passing through it. The controller nodes are also deployed in a cluster of odd members in order to enable high-availability and scale. Any failure of the controller nodes does not impact any data plane traffic.

### Management Plane

The NSX management plane is built by the NSX manager. The NSX manager provides the single point of configuration and the REST API entry-points in a vSphere environment for NSX.

### Consumption Platform

The consumption of NSX can be driven directly via the NSX manager UI. In a vSphere environment this is available via the vSphere Web UI itself. Typically end-users tie in network virtualization to their cloud management platform for deploying applications. NSX provides a rich set of integration into virtually any CMP via the REST API. Out of the box integration is also available through VMware vCloud Automation Center, vCloud Director and OpenStack with the Neutron plug-in for NSX.

### Functional Services of NSX for vSphere

In this design guide we will discuss how all of the components described above give us the following functional services:

- **Logical Layer 2** – Enabling extension of a L2 segment / IP Subnet anywhere in the fabric irrespective of the physical network design
- **Distributed L3 Routing** – Routing between IP subnets can be done in a logical space without traffic going out to the physical router. This routing is performed in the hypervisor kernel with a minimal CPU / memory overhead. This functionality provides an optimal data-path for routing traffic within the virtual infrastructure. Similarly the NSX Edge provides a mechanism to do full dynamic route peering using OSPF, BGP, IS-IS with the physical network to enable seamless integration.
- **Distributed Firewall** – Security enforcement is done at the kernel and VNIC level itself. This enables firewall rule enforcement in a highly scalable manner without creating bottlenecks onto physical appliances. The firewall is distributed in kernel and hence has minimal CPU overhead and can perform at line-rate.
- **Logical Load-balancing** – Support for L4-L7 load balancing with ability to do SSL termination.
- SSL VPN services to enable L2 VPN services.

## Network Virtualization Design Considerations

VMware network virtualization can be deployed over existing data center networks. In this section, we discuss how the logical networks using VXLAN can be deployed over common data center network topologies. We first address requirements for the physical network and then look at the network designs that are optimal for network virtualization. Finally, the logical networks and related services and scale considerations are explained.

### Physical Network

The physical data center network varies across different user environments in terms of which network topology they use in their data center. Hierarchical network design provides the required high availability and scalability to the data center network. This section assumes that the reader has some background in various network topologies utilizing traditional L3 and L2 network configurations. Readers are encouraged to look at the design guides from the physical network vendor of choice. In the following sections, we will examine the most common physical network topologies and analyze how network virtualization can be enabled in each of these scenarios.

#### Data Center Fabric Attributes

One of the key goals of network virtualization is to provide virtual-to-physical network abstraction. The physical fabric must provide a robust IP transport with the following parameters:

- Simple
- Scalable
- High-bandwidth
- Fault-tolerant
- QoS-providing

The following sections offer some detail for each of these parameters. In the following discussion, the terms *access layer switch*, *top-of-rack (ToR) switch* and *leaf switch* are used interchangeably. A leaf switch is typically located inside a rack and provides network access to the servers inside that rack. The terms *aggregation* and *spine layer*—which effectively provide connectivity between racks—refer to the location in the network that aggregates all the access switches.

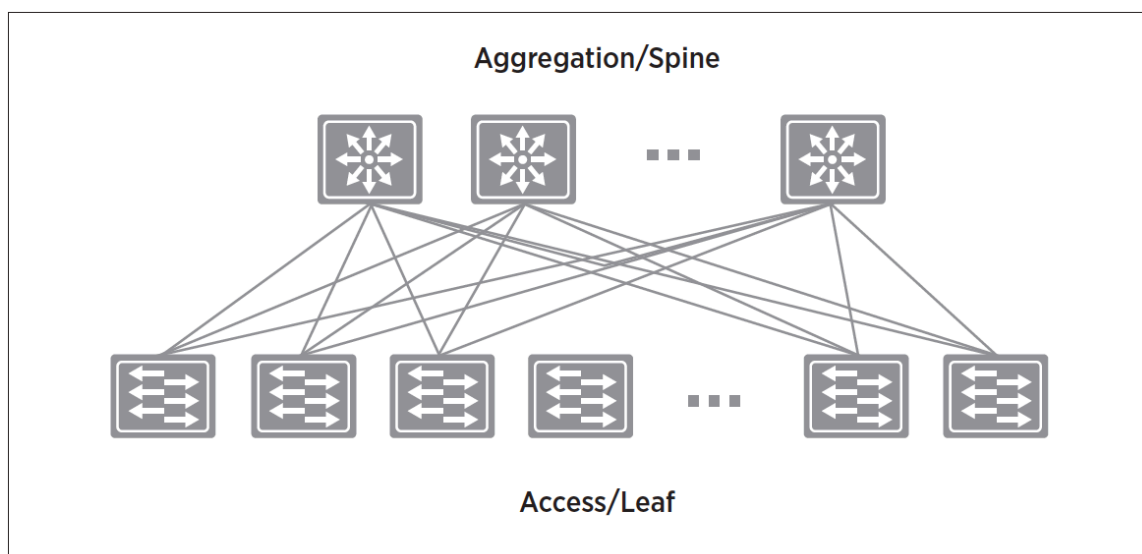


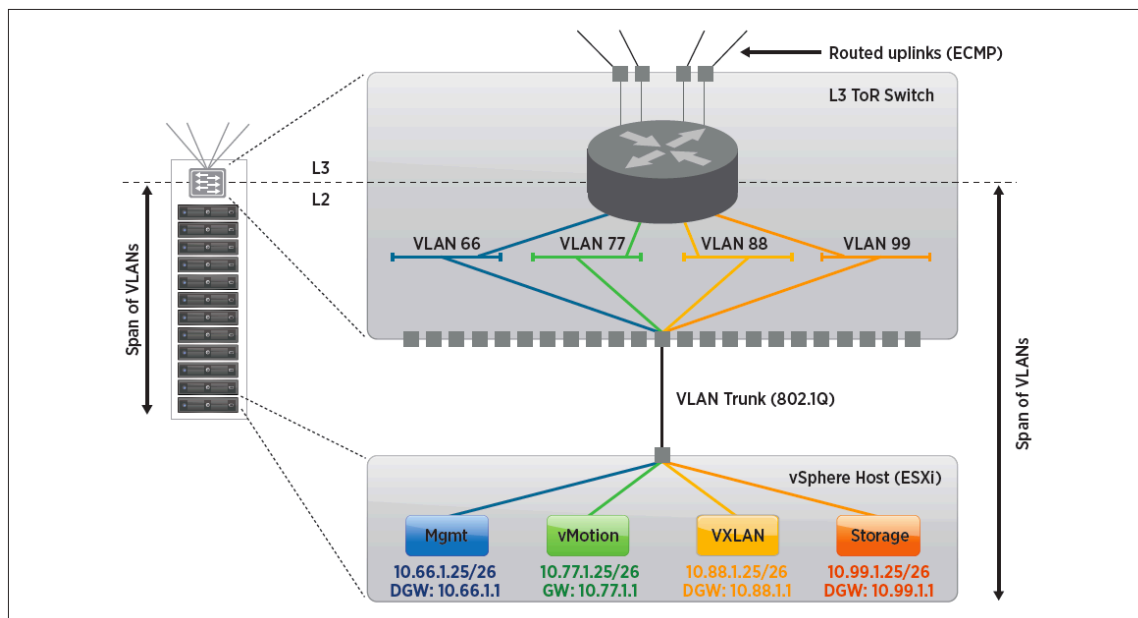
Figure 3. Leaf-Spine Topology

## Simple

Configuration of the switches that compose the overall fabric inside a data center must be simple. General or global configuration such as AAA, SNMP, SYSLOG, NTP, and so on, should be replicated almost line by line, independent of the position of the switches. The following are the primary examples of data center fabric design connectivity:

### Leaf Switches

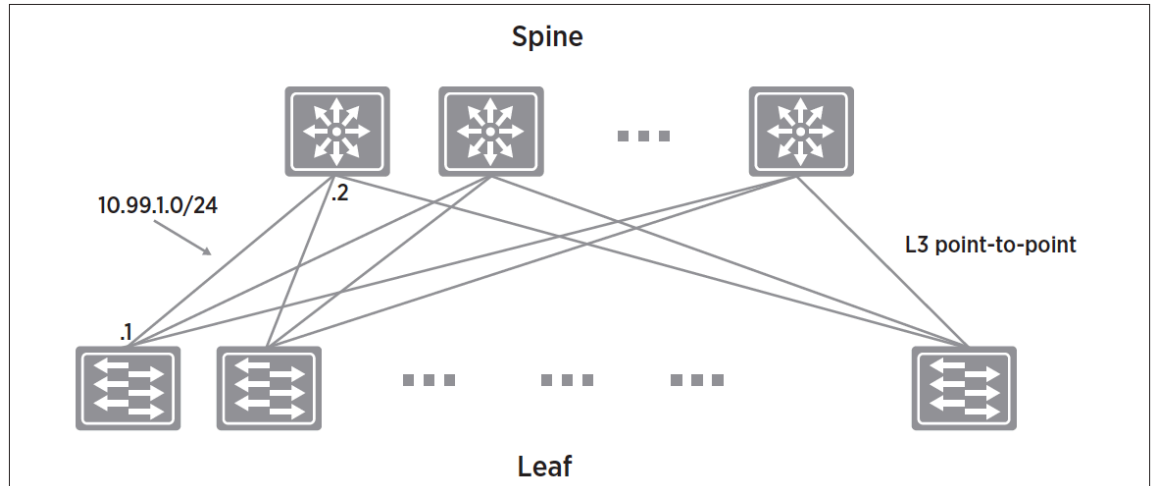
Ports facing the servers inside a rack should have a minimal configuration. Figure 4 is a high-level physical and logical representation of the leaf node.



**Figure 4.** High-Level Physical and Logical Representation of the Leaf Node

Assuming the server has multiple interfaces of the same speed, link aggregation can be used. There are various link aggregation options available on the vSphere Distributed Switch. The two significant options are load-based teaming, with route based on virtual network adapter load, and IEEE 802.3ad standard-based Link Aggregation Control Protocol (LACP). The teaming options enable optimal use of available bandwidth while providing reliability in case of a link failure.

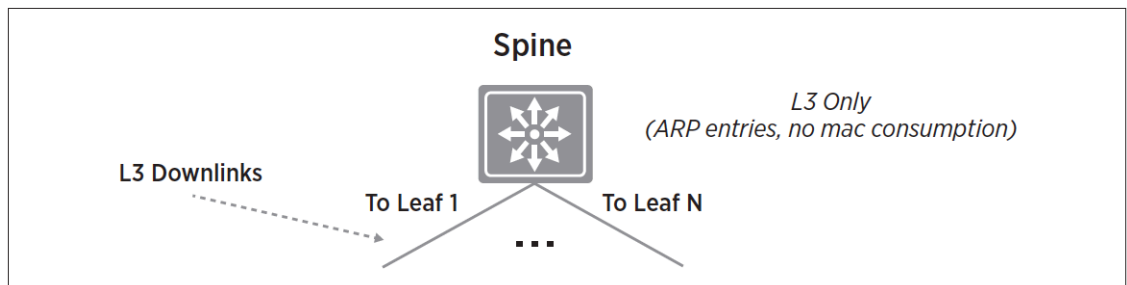
Typically, 801.Q trunks are used for carrying a small number of VLANs; for example, VXLAN tunnel, management storage and VMware vSphere vMotion® traffic. The switch terminates and provides default gateway functionality, respectively, for each VLAN; that is, it has a switch virtual interface (SVI) for each VLAN. Uplinks from the ToR switch, or leaf switch, to the aggregation or spine layer are routed point-to-point links. VLAN trunking on the uplinks is not allowed—not even for a single VLAN. A dynamic routing protocol—OSPF, ISIS, BGP, for example—is configured between the leaf and spine switches. Each ToR switch in the rack will advertise a small set of prefixes, typically one per VLAN or subnet it has present. In turn, it will calculate equal cost paths to the prefixes received from other ToR switches. In the vSphere environment, there are some design restrictions regarding vSphere vMotion and storage networks. These will be discussed in detail in the “Layer 3 in the Data Center Access Layer” section.



**Figure 5.** L3 Connectivity Between Leaf and Spine Switches

#### Spine Switches

The spine switch has only interfaces that connect to leaf switches; all interfaces are configured as routed point-to-point links, effectively acting as the “other end” of the leaf switch’s point-to-point uplinks.



**Figure 6.** Spine Switch Interfaces

Links between spine switches typically are not required. If there is a link failure between a spine switch and a leaf switch, the routing protocol will ensure that no traffic for the affected rack is attracted to the spine switch that has lost connectivity to that rack.

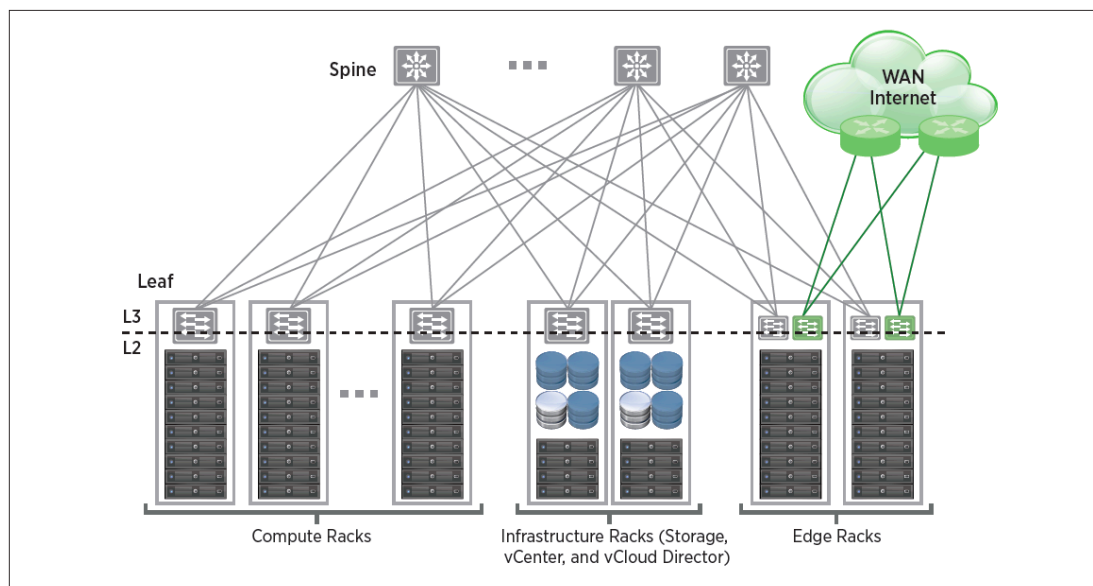
#### Scalable

Factors concerning scale include the number of racks supported in a fabric, the amount of bandwidth existing between any two racks in a data center, the number of paths a leaf switch can select from when communicating with another rack, and so on.

The number of racks supported in a fabric is dictated by the total number of ports available across all spine switches and the oversubscription that is acceptable. More details are provided in the “High Bandwidth” section.

Different racks might be hosting different types of infrastructure. For example, there might be racks with filers or other storage systems. Such racks by nature might attract or source more traffic than others in a data center. In addition, traffic levels of compute racks—that is, racks hosting hypervisors with workloads or virtual machines—as opposed to edge racks, which provide connectivity to the outside world, might have different bandwidth requirements. Link speed as well as number of links vary to satisfy different bandwidth demands. This can be done for each rack without sacrificing any of the architecture of the spine switch or leaf switch.



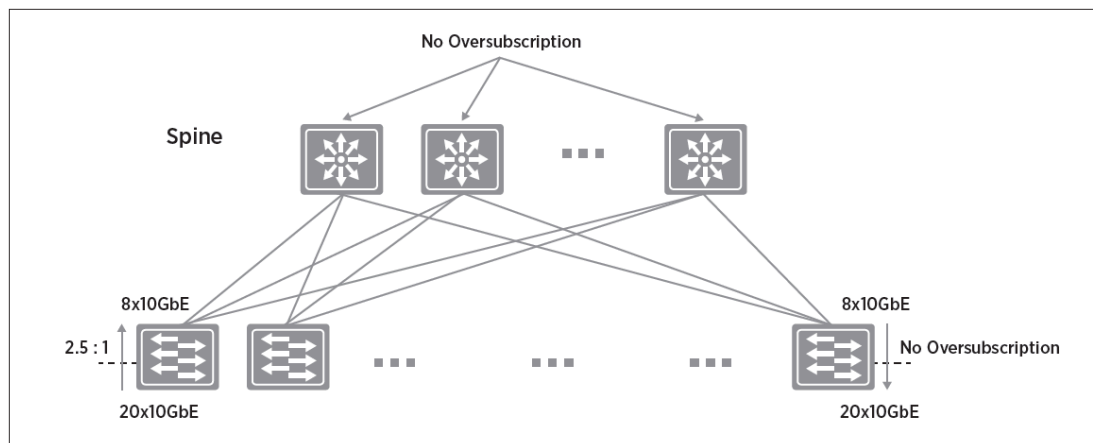


**Figure 7.** Compute, Infrastructure, and Edge Rack Design in the Leaf-Spine Topology

The number of links to the spine switches dictates how many paths that traffic from this rack to another rack can be chosen from. Because the number of hops between any two racks is consistent, equal-cost multipathing (ECMP) can be utilized. Assuming traffic sourced by the servers carry a TCP or UDP header, traffic spray can occur on a per-flow basis.

### High Bandwidth

In spine-leaf switch topologies, oversubscription typically occurs—if at all—at one point: the leaf switch. The calculation is simple: total amount of bandwidth available to all servers connected to a given leaf switch divided by the aggregate amount of uplink bandwidth provides the oversubscription. For example, 20 servers with one 10 Gigabit Ethernet (10GbE) port each create up to 200Gbps of bandwidth. Assuming eight 10GbE uplinks to the spine—a total of 80 Gbps—results in a 2.5:1 oversubscription.



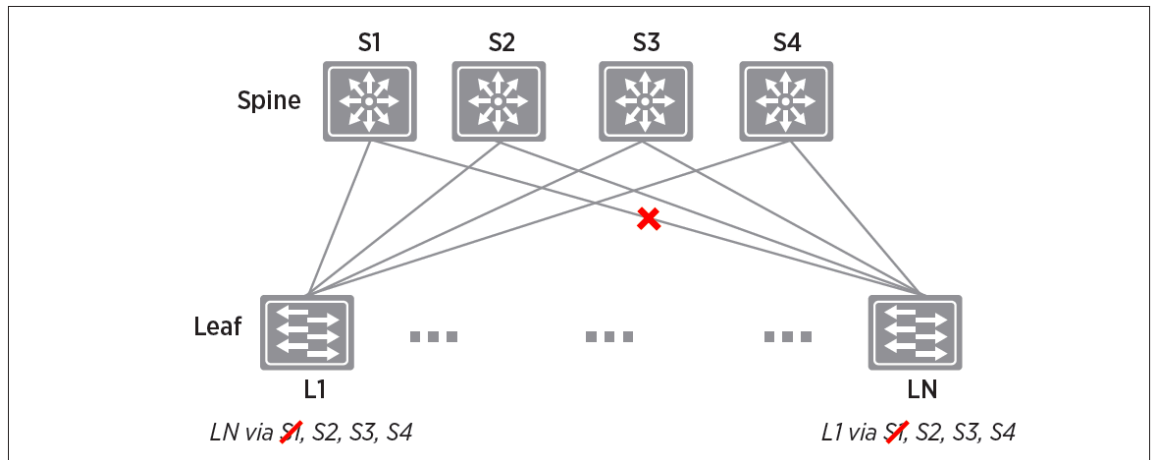
**Figure 8.** Oversubscription Example for Leaf-Spine Topology

As discussed in the previous section, depending on the rack's function, more or less bandwidth can be made available to a rack by virtue of provisioning more or fewer uplinks. In other words, the level of oversubscription can vary on a per-rack basis.

From an architecture standpoint, one rule must be obeyed: The number of uplinks from a leaf switch to each spine switch must be the same; that is, having two uplinks to spine switch A and only one uplink to spine switches B, C and D would be a poor design because “more” traffic would be sent to the leaf switch via spine switch A, potentially creating a hot spot.

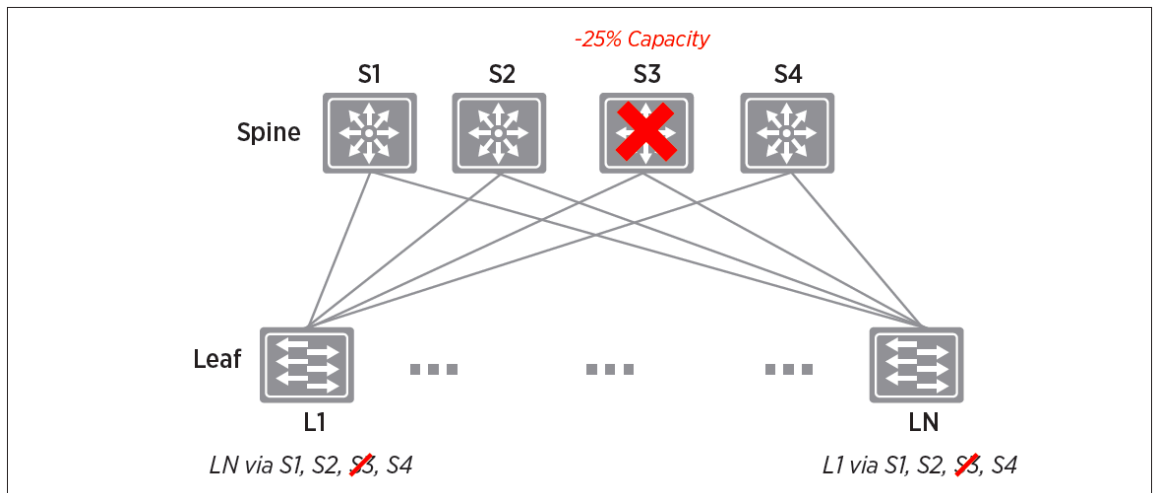
### Fault Tolerant

The larger the environment, the more switches that make up the overall fabric and the greater the possibility for one component of the data center switching fabric to fail. The reason for building a resilient fabric is that it can sustain individual link or box failures without a widespread impact.



**Figure 9.** Link Failure Scenario in Leaf-Spine Topology

For example, if one of the spine switches were to fail, traffic between racks would continue to be routed in an L3 fabric across the remaining spine switches. For L3 fabrics, the routing protocol ensures that only remaining paths would be chosen. And because more than two spine switches can be installed, the impact of a spine switch failure is reduced.

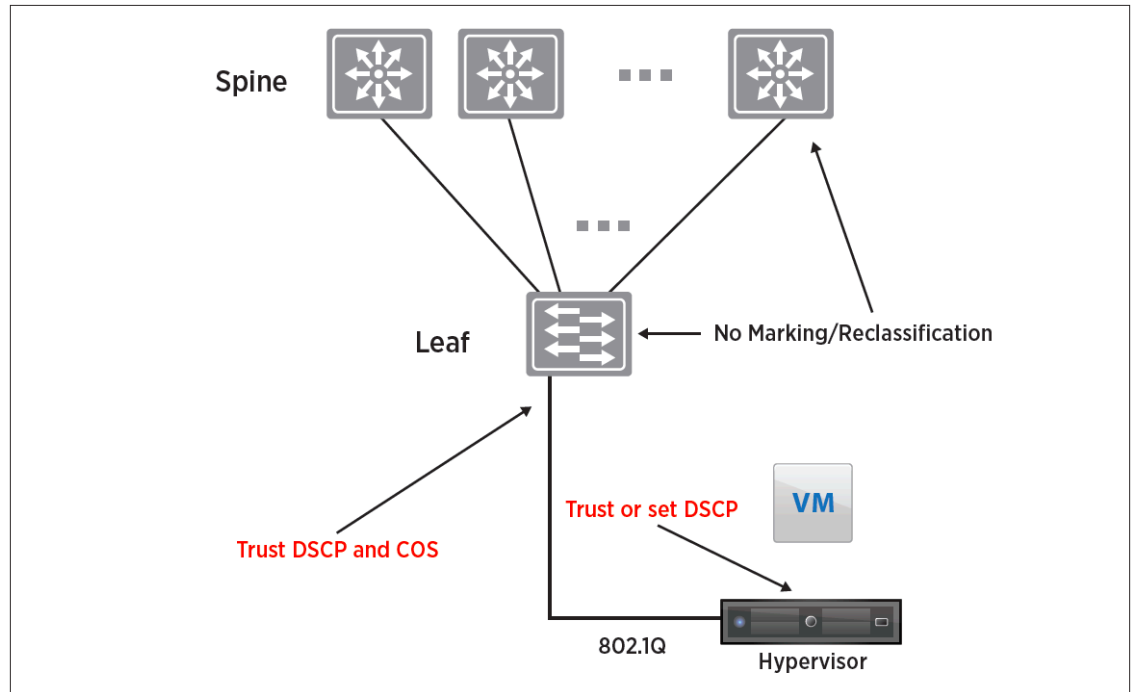


**Figure 10.** Spine Switch Failure Scenario and Impact on Bandwidth

Multipathing-capable fabrics handle box or link failures, reducing the need for manual network maintenance and operations. If a software upgrade must be performed on a fabric switch, the node can be taken out of service gracefully by changing routing protocol metrics; quickly, the traffic through that switch will be drained, freeing it up for maintenance. Depending on the width of the spine—that is, how many switches are in the aggregation or spine layer—the additional load the remaining switches must carry is not as significant as if there were only two switches in the aggregation layer.

### Differentiated Services – Quality of Service

Virtualized environments must carry various types of traffic—including tenant, storage and management—across the switching infrastructure. Each traffic type has different characteristics and applies different demands on the physical switching infrastructure. Although management traffic typically is low in volume, it can be critical for controlling physical and virtual network state. IP storage traffic typically is high in volume and generally stays within a data center. The cloud operator might be offering various levels of service for tenants. Different tenants' traffic carries different quality of service (QoS) values across the fabric.



**Figure 11.** Quality of Service (QoS) Tagging

For virtualized environments, the hypervisor presents the trusted boundary, meaning it sets the respective QoS values for the different traffic types. In this case, the physical switching infrastructure is expected to “trust” these values. No reclassification is necessary at the server-facing port of a leaf switch. If there were a congestion point in the physical switching infrastructure, the QoS values would be looked at to determine how traffic should be sequenced—and potentially dropped—or prioritized.

There are two types of QoS configuration supported in the physical switching infrastructure; one is handled at L2 and the other at L3 or IP layer. The L2 QoS is sometimes referred to as “class of service” and the L3 QoS as “DSCP marking.” With the VMware vSphere 5.5 release, both class of service and DSCP marking are supported, and users can mark the traffic based on the traffic type or packet classification method. When the virtual machines are connected to the VXLAN-based logical switches or networks, the QoS values from the internal packet headers are copied to the VXLAN-encapsulated header. This enables the external physical network to prioritize the traffic based on the tags on the external header.

## Data Center Access Layer Deployment Scenarios

This section discusses how network virtualization is placed on top of a scalable network fabric. Network virtualization consists of three major aspects: decouple, reproduce and automate. All three aspects are vital in achieving the desired efficiencies. This section focuses on decoupling, which is key to simplifying and scaling the physical infrastructure. The network virtualization solution can use connectivity only as provided by a scalable fabric—specifically, the network virtualization solution cannot span VLANs beyond a single rack inside the switching infrastructure.

We are considering a Layer 3 design in the data center access layer for the consideration of this design guide.

### Layer 3 in the Data Center Access Layer

When building a new environment, it is essential to choose an architecture that allows for future growth. The approach discussed here works for deployments that begin small but grow to large-scale ones while keeping the same overall architecture.

The guiding principle for such deployments is that the network virtualization solution does not imply any spanning of VLANs beyond a single rack. Although this appears to be a simple requirement, it has widespread impact on how a physical switching infrastructure can be built and on how it scales.

We differentiate between the following three types of racks within the infrastructure:

- Compute
- Edge
- Infrastructure

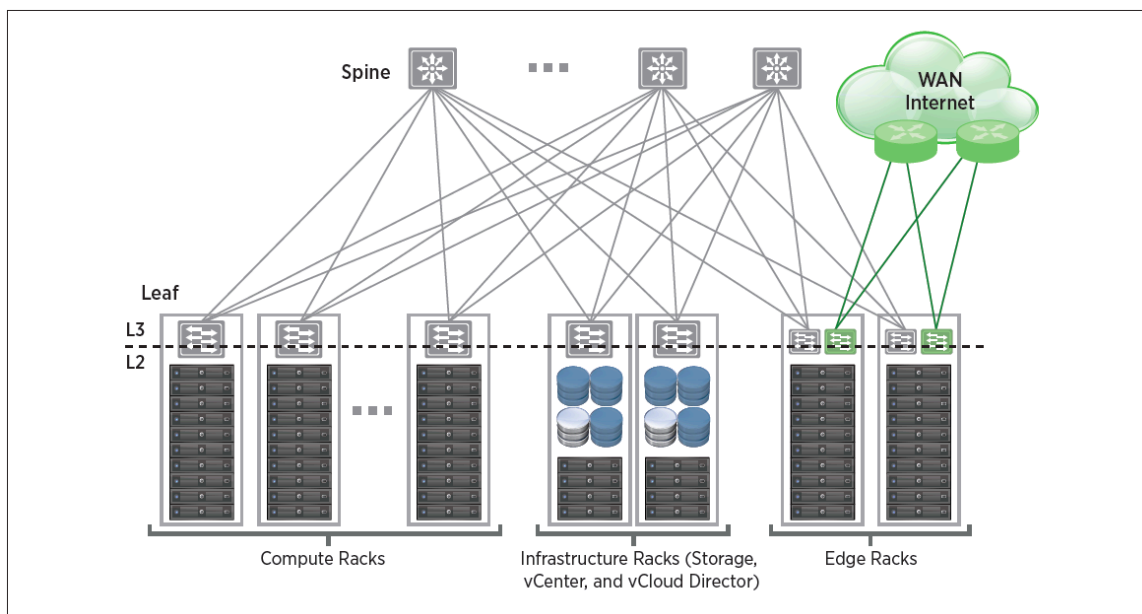


Figure 12. Data Center Design – L3 in Access Layer

#### Compute Racks

Compute racks make up the part of the infrastructure where tenant virtual machines are hosted. They should have the following design characteristics:

- Interoperate with an existing network
- For new deployments or redesigns
  - Should not require VLANs for virtual machines

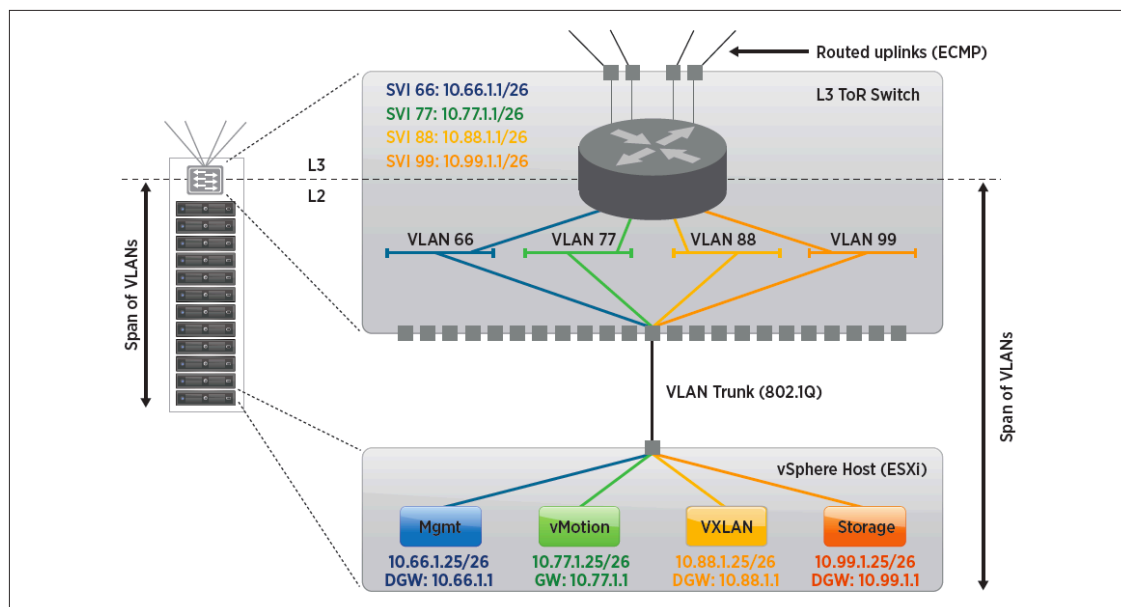
- Should not require VLANs to extend beyond a compute rack
- Provide a repeatable-rack design

A hypervisor typically sources three or more types of traffic. In the following, we look at VXLAN, management, vSphere vMotion and storage traffic. The VXLAN traffic is a new traffic type that carries all the virtual machine communication and encapsulates it in the UDP frame. The following section will discuss how the hypervisors connect to the external network and how these different traffic types are commonly configured.

### Connecting Hypervisors

The servers in the rack are connected to the access layer switch via a number of one Gigabit Ethernet (1GbE) interfaces or a limited number of 10GbE interfaces. The server physical NICs are connected to the virtual switch on the other end. For best practices on how to connect the NICs to the virtual and physical switches, refer to the VMware vSphere Distributed Switch Best Practices technical white paper <http://www.vmware.com/files/pdf/techpaper/vsphere-distributed-switch-best-practices.pdf>.

Different traffic types can be segregated via VLANs, enabling clear separation from an IP addressing standpoint. The various VMkernel NICs are assigned discrete VLANs and IP addresses. Each VLAN terminates at the leaf switch, so the leaf switch will provide an L3 interface for each VLAN. Such interfaces are also known as SVIs or RVIs.



**Figure 13.** Example – Host and Leaf Switch Configuration in a Rack

Because the hypervisor might have multiple routed interfaces, we will look more closely at how to configure them. Different VMkernel NICs can be assigned different gateways via DHCP, or IP addresses can be assigned statically. For the latter option, only one default gateway can be configured. That mandates the static routing configuration, which is not required for the management network VMkernel NIC but is required for the other VMkernel NICs.

We will now look more closely at some host traffic types:

#### VXLAN Traffic

After the vSphere hosts have been prepared for network virtualization using VXLAN, a new traffic type gets enabled on the hosts. Virtual machines connected to one of the VXLAN-based logical L2 networks use this traffic type to communicate. The traffic from the virtual machine is encapsulated and sent out as VXLAN traffic. The external physical fabric never detects the virtual machine IP and MAC address. The virtual tunnel endpoint (VTEP) IP address is used to transport the frame across the fabric. In the case of VXLAN, the

tunnels are initiated and terminated by a VTEP. Traffic that flows between virtual machines in the same data center is typically referred to as east–west traffic. For this type of traffic, both the source and destination VTEP are situated in hypervisors located in compute racks. Traffic leaving the data center will flow between a tenant virtual machine and an NSX edge, for example. This traffic is referred to as north–south traffic.

VXLAN configuration requires a NSX vSwitch. Because a VDS can span across hundreds of hypervisors, it can reach beyond a single leaf switch. Therefore, the host VTEPs—even if they are on the same VDS—must be able to be situated in different subnets. One of the requirements of a single-VDS–based design is that a single VLAN is defined for VXLAN transport network.

#### *Management Traffic*

Management traffic can be categorized into two types: one is sourced and terminated by the management VMkernel interface on the host; the other is involved with the communication between the various NSX components. The traffic that is carried over the management VMkernel interface of a host includes the communication between vCenter Server and hosts as well as communication with other management tools such as NSX Manager. The communication between the NSX components involves the heartbeat between active and standby edge appliances.

Management traffic stays inside the data center. A single VDS can span multiple hypervisors that are deployed beyond a single leaf switch. Because no VLANs can be extended beyond a leaf switch, the management interfaces of hypervisors participating in a common VDS are in separate subnets.

#### *vSphere vMotion Traffic*

During the vSphere vMotion migration process, running virtual machine state is transferred over the network to another host. The vSphere vMotion VMkernel interface on each host is used to move this virtual machine state. Each vSphere vMotion VMkernel interface on the host is assigned an IP address. Depending on the speed of the physical NIC, the number of simultaneous virtual machine vSphere vMotion migrations is decided. On a 10GbE NIC, eight simultaneous vSphere vMotion migrations can be performed. From the support point of view, having the VMkernel interfaces in the same subnet is recommended. However, while designing the network for network virtualization using L3 in the access layer, users can select different subnets in different racks for vSphere vMotion VMkernel interface. For ongoing support, it is recommended that users go through the RPQ process so VMware will validate the design.

Unless a long-distance vSphere vMotion migration across different sites is involved, vSphere vMotion traffic stays primarily within the data center. As with a management VMkernel interface, based on which rack the host is situated on, the vSphere vMotion VMkernel interface on that host is in a separate subnet.

#### *Storage Traffic*

A VMkernel interface is used to provide features such as shared or nondirectly attached storage. Typically, we refer to storage that can be attached via an IP connection—NAS or iSCSI, for example—rather than FC or FCoE. From an IP addressing standpoint, the same rules that apply to management traffic apply to storage VMkernel interfaces. The storage VMkernel interface of servers inside a rack—that is, connected to a leaf switch—is part of the same subnet. This subnet, however, cannot span beyond this leaf switch. Therefore, the storage VMkernel interface IP of a host in a different rack is in a different subnet. For an example of the IP address for these VMkernel interfaces, refer to the “VLAN Provisioning” section.

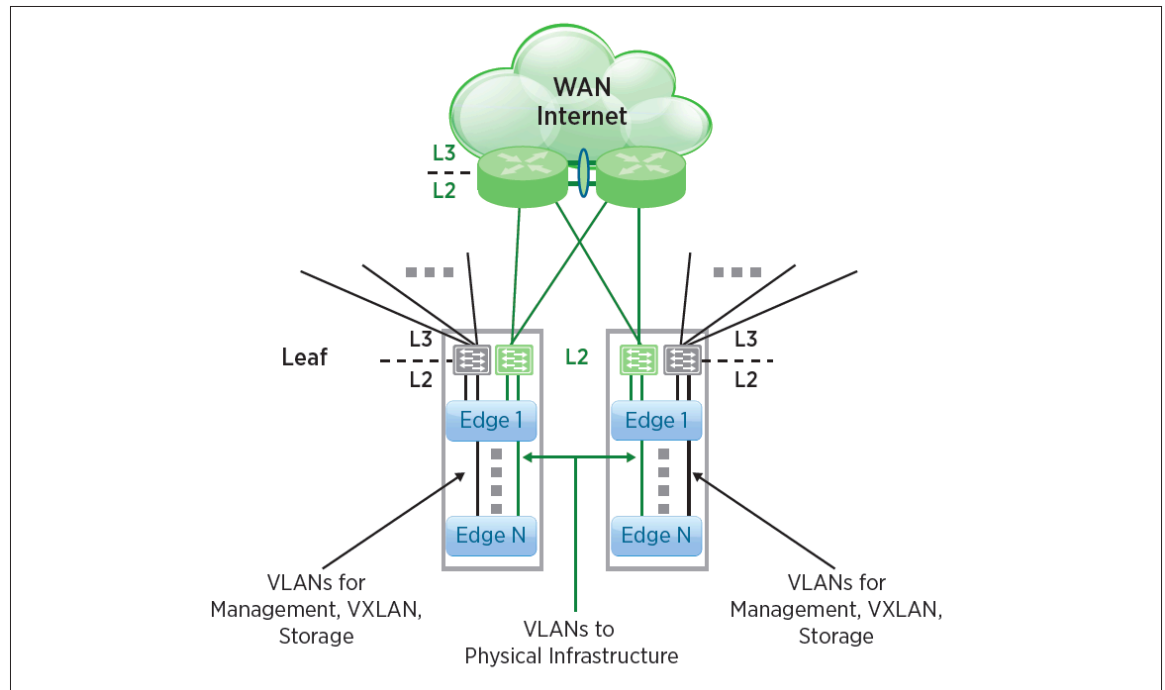
#### *Edge Racks*

Tighter interaction with the physical infrastructure occurs while bridging between the overlay world and the physical infrastructure. The following are main functions provided by an edge rack:

- Provide on-ramp and off-ramp connectivity to physical networks
- Connect with VLANs in the physical world
- Host centralized physical services

Tenant-specific addressing is exposed to the physical infrastructure in cases where traffic is not encapsulated in VXLAN—for example, where NAT is not being used at the edge. In the case of an L3 edge, the IP addresses within the overlay are exposed to the physical fabric. The guiding principle in these cases

is to separate VXLAN (overlay) traffic from the un-encapsulated (native) traffic. As shown in Figure 14, VXLAN traffic hits the data center internal Ethernet switching infrastructure. Native traffic traverses a dedicated switching and routing infrastructure facing the WAN or Internet and is completely decoupled from the data center internal network.

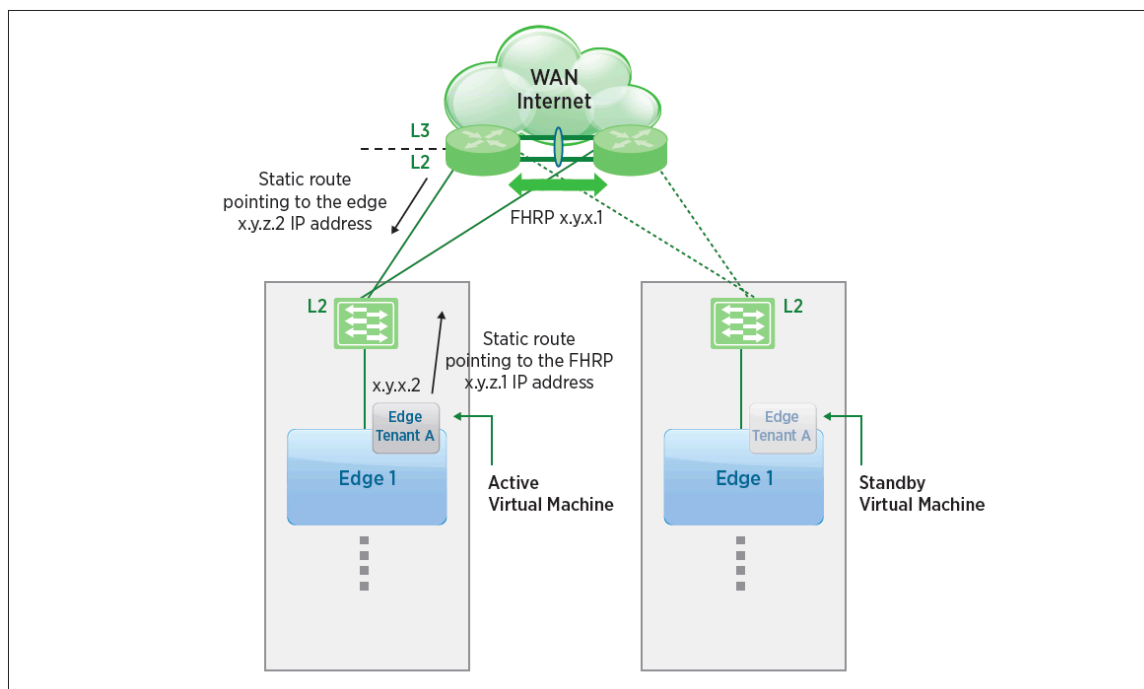


**Figure 14.** VXLAN Traffic and the Data Center Internal Ethernet Switching Infrastructure

To maintain the separation, NSX Edge virtual machines can be placed in edge racks, assuming the NSX Edge has at least one native interface. For routing and high availability, the two interface types—overlay and native—must be examined individually. The failover mechanism is based on the active standby model, where the standby edge takes over after detecting the failure of the active edge.

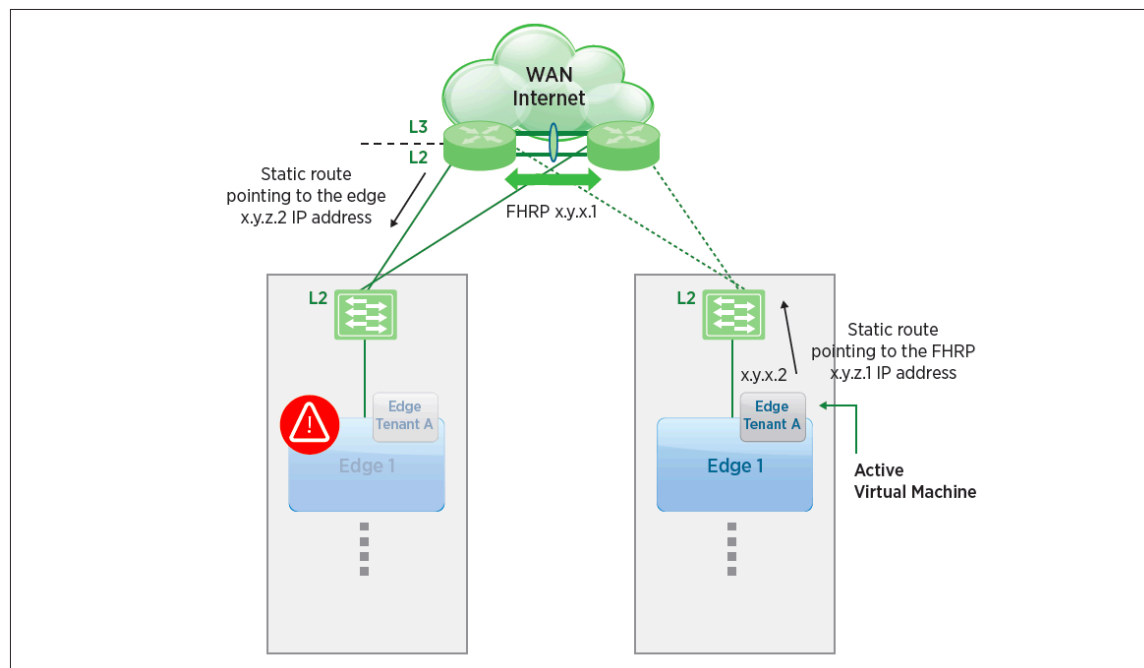
### Layer 3 Edge

In this case, the edge terminates all logical networks and presents an L3 hop between the physical and the logical world. Depending on the use case, users can decide to employ either the NAT or the static routing option to provide connectivity to the external network.



**Figure 15.** High Availability – Active Standby Edge Topology

In case the edge fails, the standby takes over and assumes the outside IP address of the previously active edge. To notify the upstream infrastructure—that is, L2 switches that potentially interconnect the Edge and the first physical router—a GARP message is sent out. For this mechanism to work, a VLAN must be extended between the edge racks. Tunnel interfaces connecting the VXLAN endpoints do not have to extend any VLAN. Before the failover, the hypervisor VTEPs sent traffic to the VTEP of the hypervisor hosting the edge. After failover, that traffic is sent to the VTEP of the hypervisor that hosts the newly active edge.



**Figure 16.** Failure of Active Edge



### Infrastructure Racks

Infrastructure racks host the management components, including vCenter Server, NSX Manager, NSX Controller, CMP and other shared IP storage–related components. It is key that this portion of the infrastructure does not have any tenant-specific addressing. If bandwidth-intense infrastructure services are placed in these racks—IP-based storage, for example—bandwidth of these racks can be dynamically scaled, as discussed in the “**High Bandwidth**” subsection of the “**Data Center Fabric Attributes**” section.

### VLAN Provisioning

As discussed previously, each compute rack has four different subnets supporting four different traffic types: tenant, management, vSphere vMotion and storage traffic. In this section, we will discuss how to automate the provisioning of IP addresses to the VMkernel NICs of each traffic type, using the vSphere Host Profile method.

The Host Profile feature enables users to create a reference host with properties that are shared across the deployment. After this host has been identified and required sample configuration has been performed, a Host Profile can be created from that host and applied across the other hosts in the deployment. With this approach, users can quickly configure large numbers of hosts.

Before discussing how the Host Profile method can be used during configuration of whole compute racks, we will look at the type of sample configuration required on a host in a rack. As shown in Figure 17, the same set of four VLANs—storage, vSphere vMotion, VXLAN, management—is provided in each rack. The following are among the configurations required per host:

- 1) vmknic IP configuration per traffic type in the respective subnet or VLAN
- 2) Static route configuration per subnet, to handle proper traffic routing to the respective gateways

Static routes are required because one TCP/IP stack support on the VMware ESXi™ host will limit the number of default gateway configurations to one.

For example, in rack 1, host 1 has the following vmknic configuration:

- Storage vmknic with IP address 10.66.1.10
- vSphere vMotion vmknic with IP address 10.77.1.10
- VXLAN vmknic with IP address 10.88.1.10
- Management vmknic with IP address 10.99.1.10

The default gateway configuration on host 1 is in the management vmknic subnet 10.99.1.0/26. To support proper routing for other subnets, the following static routes are configured as part of the host 1 preparation:

- Storage network route – `esxcli network ip route ipv4 add -n 10.66.0.0/26 -g 10.66.1.1`
- vSphere vMotion network route – `esxcli network ip route ipv4 add -n 10.77.0.0/26 -g 10.77.1.1`

After host 1 of rack 1 has been configured, a Host Profile is created and then applied to other hosts in the rack. While applying the profile to the hosts, new vmknics are created and the static routes are added, simplifying the deployment.

In the case of a vSphere Auto Deploy environment, the PXE boot infrastructure, along with the Auto Deploy server and vCenter Server, supports the host-booting process and helps automate the deployment as well as the upgrades of the ESXi hosts.

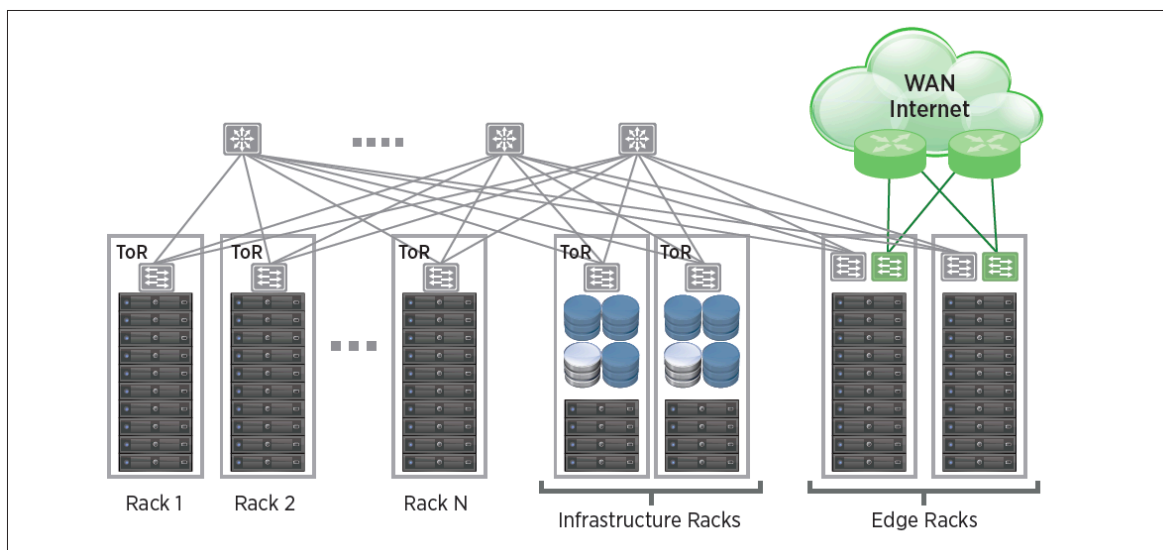


Figure 17. Host Infrastructure Traffic Types and IP Address Assignment

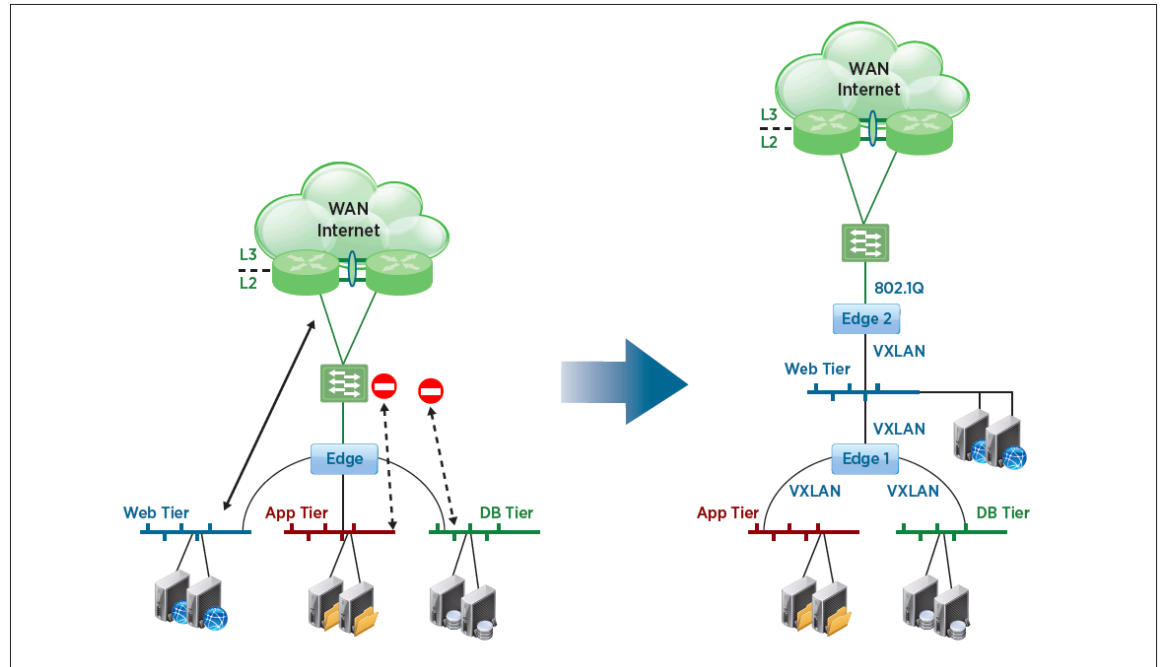
IP ADDRESS MANAGEMENT AND VLANs <sup>1</sup>		
Function	Global VLAN ID	IP Address
Storage	66	10.66.R_id.x/26
vMotion	77	10.77.R_id.x/26
VXLAN/VTEP	88	10.88.R_id.x/26
Management	99	10.99.R_id.x/26

<sup>1</sup> Values of VLANs, IP addresses, and masks are an example (not prescriptive to the design)

Table 1. IP Address Management and VLANs

### Multitier Edges and Multitier Application Design Considerations

Classical multitier compute architectures have functions that are logically separated, where each function has different requirements in terms of resource access, data segregation and security. A classical three-tier architecture typically comprises a presentation tier, an application or data access tier and a database tier. Communication between the application tier and the database tier should be allowed while an external user has access to only the presentation tier, which is typically a Web-based service. The recommended solution to comply with data-access policies is to deploy a two-tier edge design. The inner edge enables VXLAN-to-VXLAN east-west traffic among the presentation, database and application tiers, represented by different logical networks. The outer edge connects the presentation tier with the outer world for on-ramp and off-ramp traffic. Communication within a specific virtual wire enables virtual machines to span across multiple racks to achieve optimal utilization of the compute rack infrastructure. At the current stage, a logical network can span only within a single vCenter domain. Figure 18 shows the placement of the logical elements of this architecture.



**Figure 18.** Two Options for Logical Element Placement in a Multitier Application

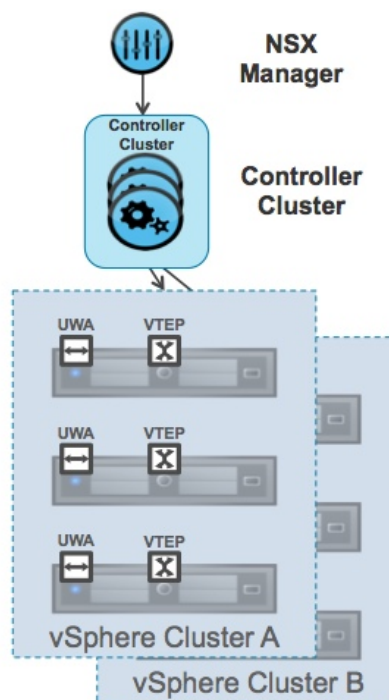
It is preferable that the outer edges be physically placed in the edge racks. Inner edges can be centralized in the edge racks or distributed across the compute racks, where Web and application compute resources are located.

## Logical Switching

The logical switching capability in the NSX platform provides customers the ability to spin up isolated logical L2 networks with the same flexibility and agility, as it is to spin up virtual machines. This section describes the various components in logical switching and the communication among those components

### Components

As shown in the figure 19, there are three main components that help decouple the underline physical network fabric and provide network abstraction. For more technical explanation, this decoupling is achieved by encapsulating the virtual machine traffic using VXLAN or STT protocols. Let's take a look at the function of each component used in logical switching.



**Figure 19.** Logical Switching Components

#### *NSX Manager*

The NSX manager is the management plane component that helps configure logical switches and connect virtual machines to these logical switches. It also provides API interface, which helps automate deployment and management of these switches through a Cloud management platform.

#### *Controller Cluster*

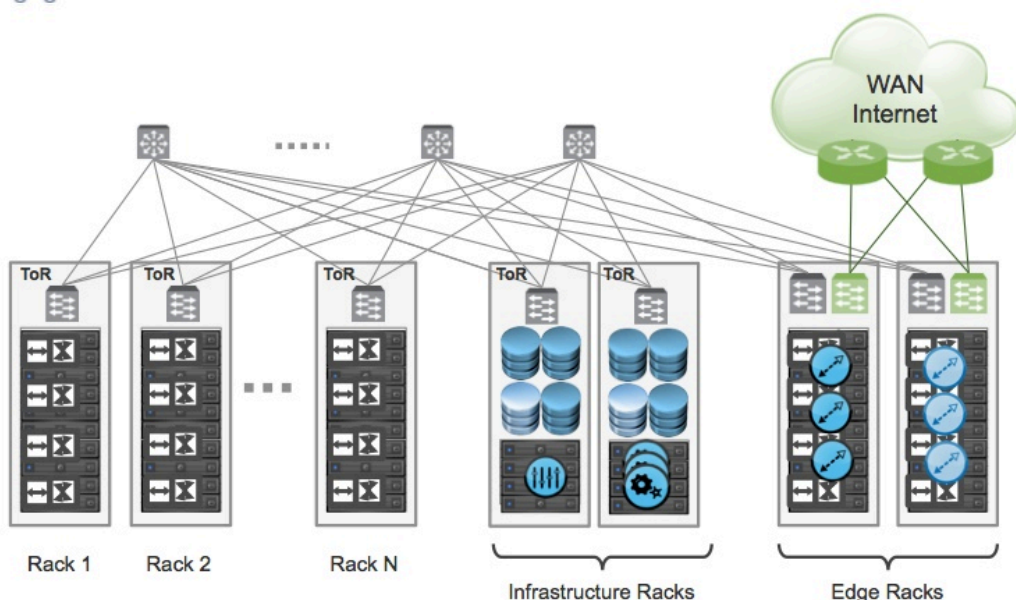
The Controller cluster in the NSX platform is the control plane component that is responsible in managing the switching and routing modules in the hypervisors. The controller cluster consists of controller nodes that manage specific logical switches. The use of controller cluster in managing VXLAN based logical switches eliminates the need for multicast support from the physical network infrastructure. Customers now don't have to provision multicast group IP addresses and also don't need to enable PIM routing or IGMP snooping features on physical switches or routers. Selecting the "Unicast" check box while creating the logical switch can enable this mode of VXLAN operation.

#### *User World Agent (UWA) and VXLAN Tunnel Endpoint (VTEP)*

There are two data plane components on the hypervisor that provides communication paths between the controller cluster and other hypervisors. They also perform the data path function of the logical switches.

The User world Agent is used to establish communication with the Controller cluster while the VTEP provides the ability to create tunnels between hypervisors.

As part of the preparation process, the controller cluster and the hypervisor modules are deployed and configured through the NSX Manager. After logical switching components are configured, the next step is to define the span of logical switches. The span of logical switches is defined by creating a transport zone. In a transport zone customers can add a set of clusters. For example, if there are 10 clusters in the data center a transport zone can include all those 10 clusters. In this scenario a logical switch can span the whole data center. The diagram below shows a deployment after the NSX components are installed to provide logical switching. The Edge Services router in the edge rack provides the logical switches access to the WAN and other network services.



**Figure 20.** Logical Switching Components in the racks

#### *Logical Switch Addressing*

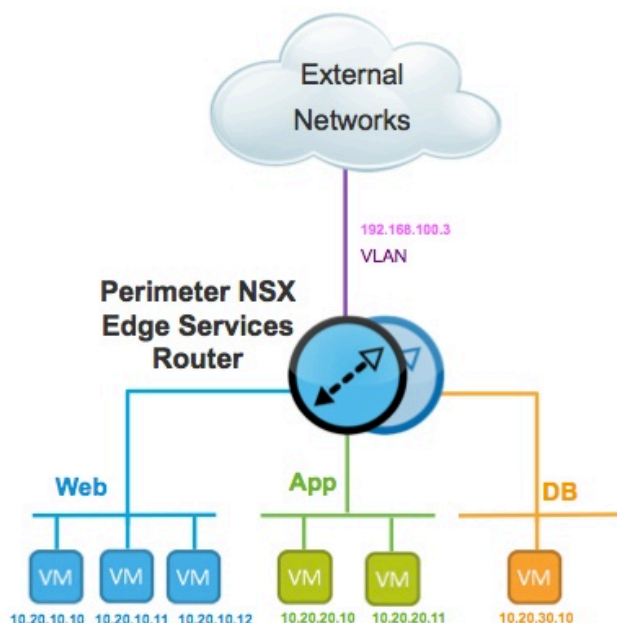
In a large cloud environment with multiple tenants or big enterprises with multiple organizations and applications, IP address management is a critical task. In this section, we will focus on IP address management of the virtual machines deployed on the logical switches. Each logical switch created is a separate L2 broadcast domain that can be associated with a separate subnet using a private IP space or public IP space. Depending on whether private IP space or public IP space is used for the assignment to the logical networks, users must choose either the NAT or the non-NAT option on the NSX Edge services router. So the IP address assignment depends on whether the virtual machine is connected to a logical switch through a NAT or a non-NAT configuration. We will look at examples of each of the following two deployments:

- 1) With NAT services of the Edge services router
- 2) Without NAT services of the Edge service router

#### *With Network Address Translation*

In the deployments where organizations have limited IP address space, NAT is used to provide address translation from private IP space to the limited public IP addresses. By utilizing Edge services router, users can provide individual tenants with the ability to create their own pool of private IP addresses, which ultimately get mapped to the publicly routable external IP address of the external Edge services router interface.

Figure 21 shows a three-tier app deployment, with each tier virtual machine connected to separate logical switch. The Web, App and DB logical switches are connected to the three internal interfaces of the Edge services router; the external interface of the Edge services router is connected to the Internet via a external data center router.

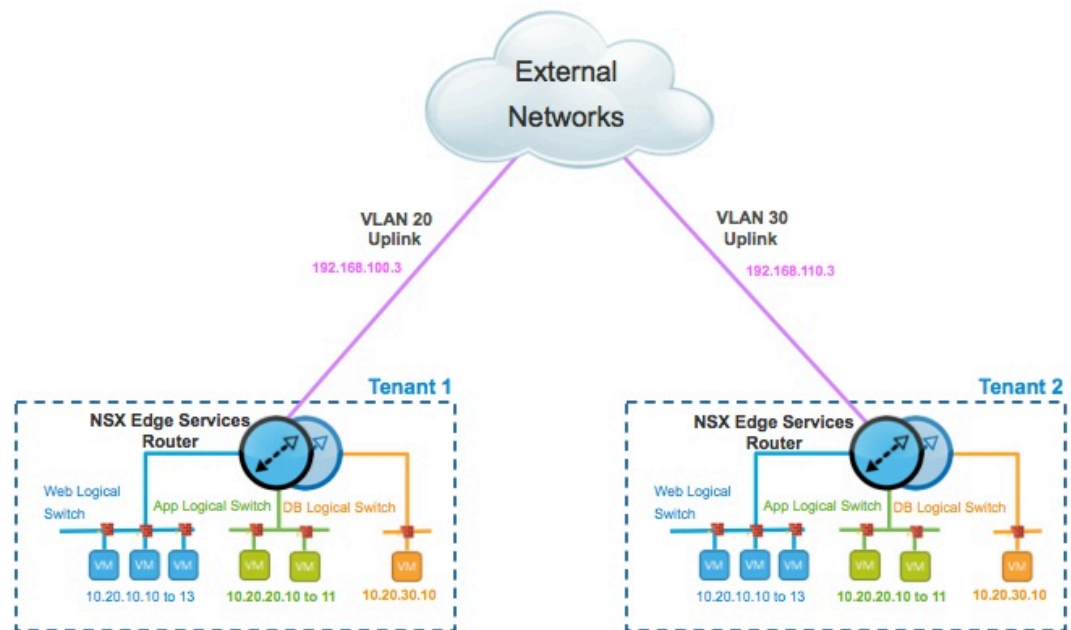


**Figure 21.** NAT and DHCP Configuration on NSX Edge Service Router

The following are configuration details of the NSX Edge Services Router:

- Web, App and DB logical switches are connected to the Internal interfaces of the NSX edge services router.
- The Uplink interface of the NSX edge services router is connected to the VLAN port group that is in subnet 192.168.100.0/24
- Enable DHCP service on this internal interface of by providing a pool of IP addresses. For example, 10.20.10.10 to 10.20.10.50.
- The NAT configuration on the external interface of the vCloud Networking and Security Edge gateway enables virtual machines on a logical switch to communicate with devices on the external network. This communication is allowed only when the requests are initiated by the virtual machines connected to the internal interface of the Edge services router.

In situations where overlapping IP and MAC address support is required, one Edge services router per tenant is recommended. Figure 22 shows an overlapping IP address deployment with two tenants and two separate NSX Edge Services router.



**Figure 22.** Overlapping IP and MAC addresses

#### *No Network Address Translation*

Organizations that are not limited by routable IP addresses, have virtual machines with public IP addresses or do not want to deploy NAT can use static and dynamic routing features available with the NSX platform. In the NSX platform two different modes of logical routing is supported. One is called distributed routing and the other one as centralized routing. The distributed routing provides better throughput and performance for the East West traffic while the centralized routing handles the North South traffic. The next section provides more details on logical routing. For the additional network services required for the applications in the datacenter please refer to the logical Firewall and Logical load balancer sections.

# Logical Routing

As mentioned in earlier section, two modes of routing are supported in the NSX platform. This section will provide more details on the two modes and also describe some common routing topologies that can be built in the customer's environment.

## Distributed Routing

The distributed routing capability in the NSX platform provides an optimized and scalable way of handling East - West traffic within a data center. The communication between virtual machines or resources within the datacenter is referred to as East-West traffic. The amount of East West traffic in the data center is growing. The new collaborative, distributed, and service oriented application architecture demands higher BW for server-to-server communication.

If these servers are virtual machines running on a hypervisor, and they are connected to different subnets, the communication between these servers has to go through a router. Also, if Physical router is used to provide routing services the virtual machine communication has to go out to the physical router and get back in to the server after routing decision. This un-optimal traffic flow is sometimes called as "hair pinning".

The distributed routing on the NSX platform prevents the "hair-pinning" by providing hypervisor level routing functionality. Each hypervisor has a routing kernel module that performs routing between the logical interfaces (LIFs) defined on that distributed router instance. The components section below describes the various modules in distributed routing and the communication among those modules.

## Centralized Routing

The NSX Edge services router provides the traditional centralized routing support in the NSX platform. Along with the routing services NSX Edge also supports other network services that includes DHCP, NAT, Load balancing etc.

## Components

As shown in the figure 23, there are multiple components for the logical routing. Some of the components are related to distributed routing and some others to centralized routing. Let's take a look at the each component and its function in the distributed or centralized routing.



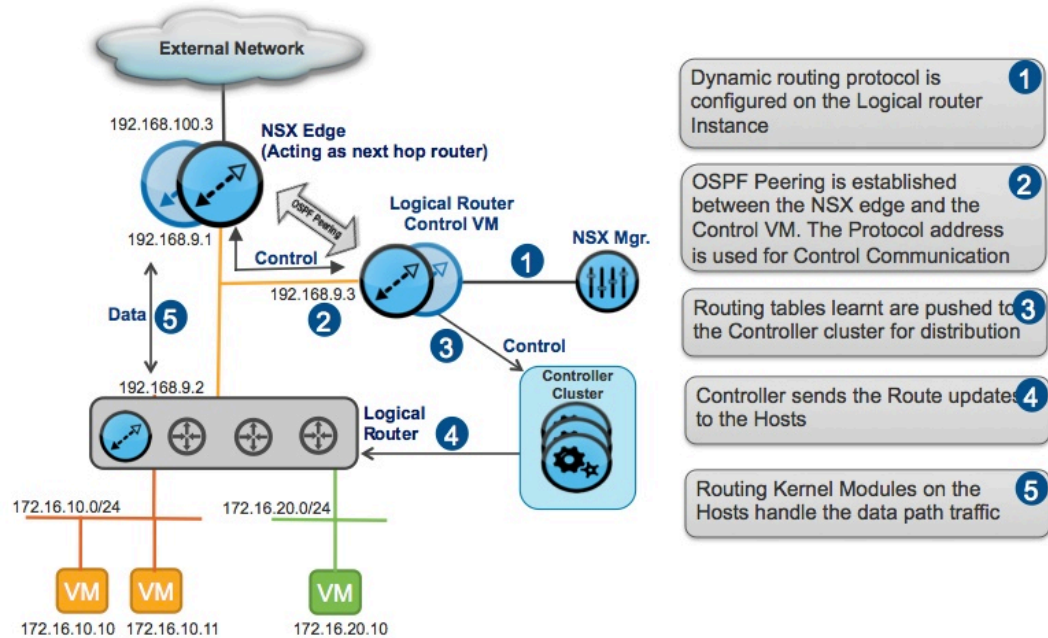


Figure 23. Logical Routing Components

### NSX Manager

The NSX manager also helps configure and manage logical routing services. During the configuration process customers have a choice to deploy a distributed or centralized logical router. If distributed router is selected, the NSX manager deploys the logical router control VM and pushes the Logical Interface configurations to each host through the controller cluster. In the case of centralized routing, NSX manager just deploys the NSX Edge services router VM. The API interface of the NSX manager helps automate deployment and management of these logical routers through a Cloud management platform.

### Logical Router Control VM

The logical router control VM is the control plane component of the routing process. It supports the following dynamic routing protocol:

- 1) OSPF
- 2) BGP

The logical router control VM communicates with the next hop router using the dynamic routing protocol, and it pushes the learned routes to the Hypervisors through the controller cluster. Customers can enable High Availability (HA) while deploying the control VM. Two VMs are deployed in Active-Standby mode when HA mode is selected.

### Logical Router Kernel Module

The logical router kernel module is configured as part of the preparation process through the NSX manager. The kernel modules are similar to the line cards in a modular chassis supporting layer 3 routing. The kernel modules have routing information base (RIB) that is pushed through the controller cluster. All the data plane function of route lookup, ARP entry lookup is performed by the kernel modules.

### Controller Cluster

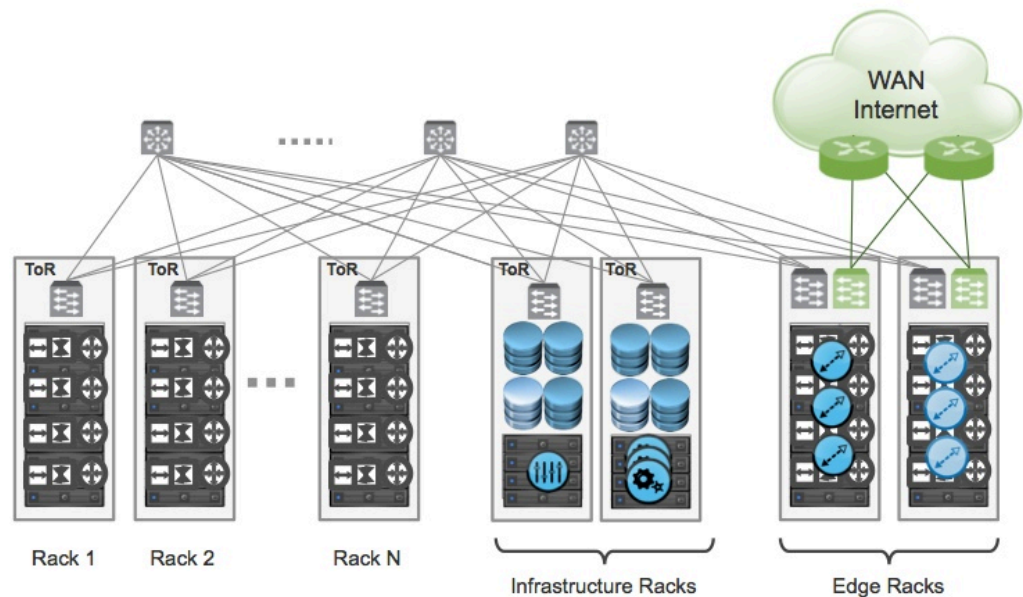
The Controller cluster is responsible for distributing routes learned from the control VM across the hypervisors. Each controller node in the cluster takes responsibility of distributing the information for a particular logical router instance. In a deployment where there are multiple logical router instance deployed the load is distributed across the controller nodes.

**NSX Edge Services Router**

This is the centralized services router that provides other network services along with the support for the following routing protocols:

- 1) BGP
- 2) OSPF
- 3) IS-IS

The other services include DHCP, NAT, Firewall, Load Balancing, and VPN capabilities.



**Figure 24.** Logical Routing Components in the Racks

**Routing Deployments**

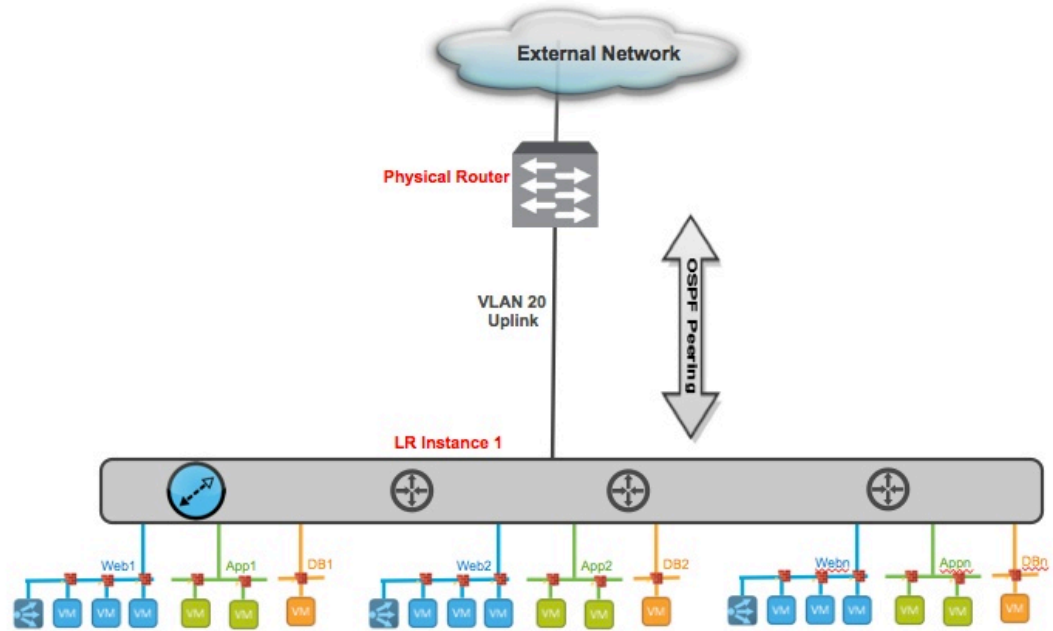
Depending on the customer requirements various topologies can be built using logical switching and logical routing features of the NSX platform. In this section, we will cover the following two routing topologies that utilizes both distributed and centralized logical routing capabilities:

- 1) Physical Router as Next Hop
- 2) Edge Services Router as Next Hop

**Physical Router as Next Hop**

As shown in the diagram below, an organization is hosting multiple applications and wants to provide connectivity among the different tiers of the application as well as connectivity to the external network. In this topology separate logical switches provide layer 2 network connectivity for the VMs in the particular tier. The distributed logical routing configuration allows the VMs on two different tiers to communicate with each other. Similarly, the dynamic routing protocol support on logical router enables the exchange of routes with the physical next hop router. This in turn allows external users to access the applications connected to the logical switches in the data center.

In this topology the East-West and North-South routing decision happens at the hypervisor level in a distributed fashion.



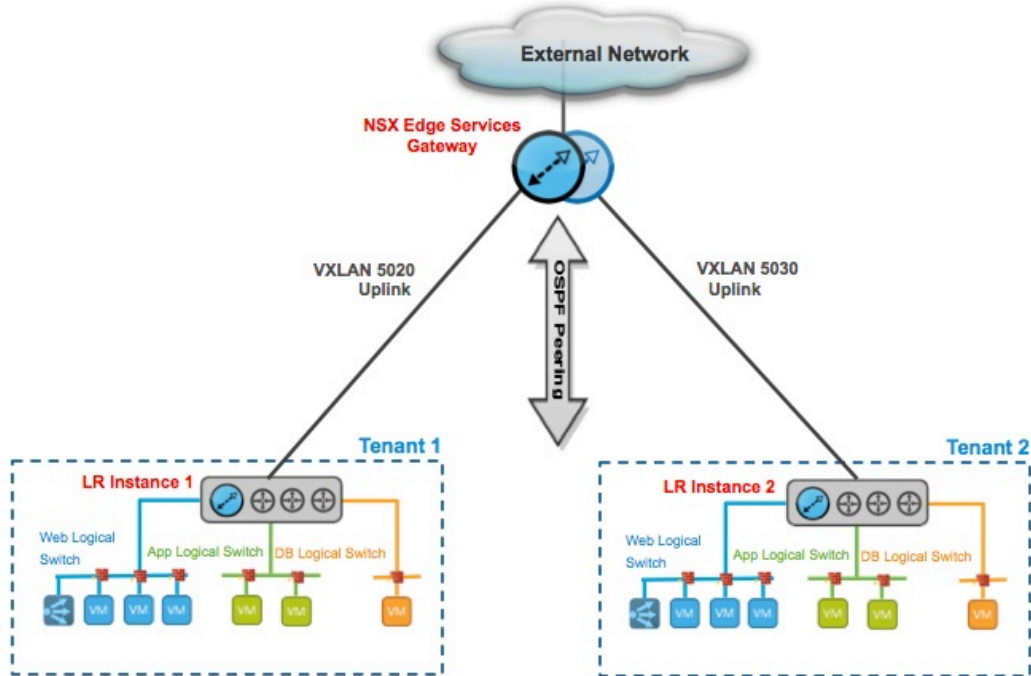
**Figure 25.** Physical Router as Next hop

### Edge Services Router as Next Hop

In a service provider environment there are multiple tenants, and each tenant can have different requirements in terms of number of isolated logical networks and other network services such as LB, Firewall, and VPN etc. In such deployments, NSX Edge services router provides network services capabilities along with dynamic routing protocol support.

As shown in the diagram below, the two tenants are connected to the external network through the NSX Edge services Router. Each tenant has its logical router instance that provides routing within the tenant. Also, the dynamic routing protocol configuration between the tenant logical router and the NSX Edge services router provides the connectivity to the tenant VMs to the external network.

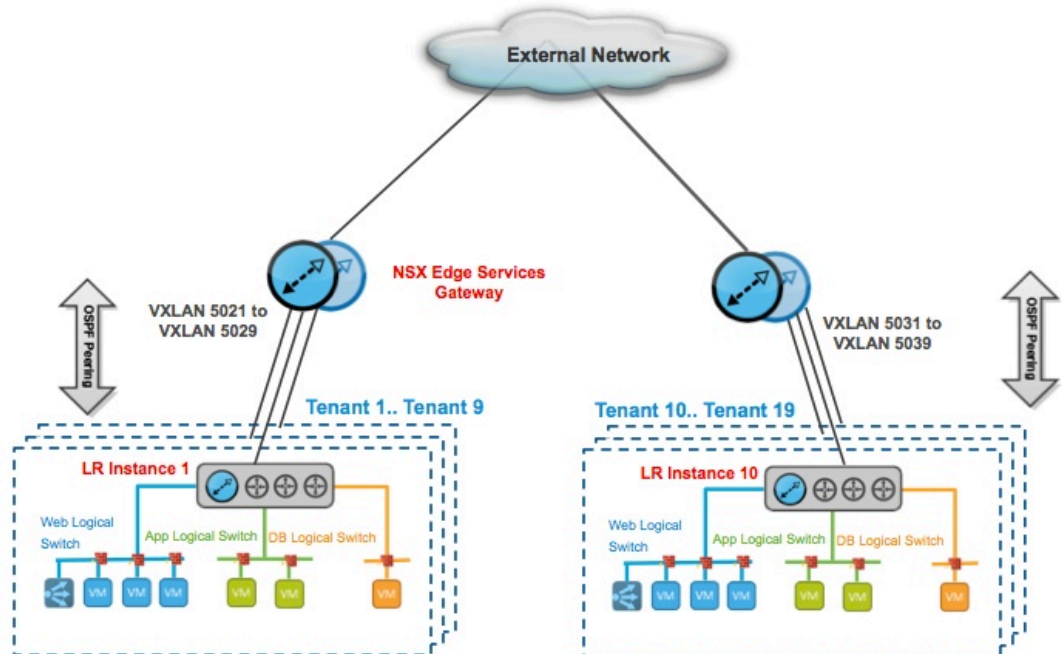
In this topology the East-West traffic routing is handled by the distributed router in the hypervisor and the North-South traffic flows through the NSX Edge Services router.



**Figure 26.** NSX Edge services Router as next hop and also providing network services

### Scalable Topology

The service provider topology described in the earlier section can be scaled out as shown in the figure 27. The diagram shows nine tenants served by NSX Edge on the left and the other nine by the Edge on the right. Service provider can easily provision another NSX edge to serve additional tenants.



**Figure 27.** Scalable Topology

# Logical Firewalling

The VMware NSX platform includes distributed kernel enabled firewalling with line rate performance, virtualization and identity aware with activity monitoring, among other network security features native to network virtualization.

## Network Isolation

Isolation is the foundation of most network security, whether for compliance, containment or simply keeping development, test and production environments from interacting. While manually configured and maintained routing, ACLs and/or firewall rules on physical devices have traditionally been used to establish and enforce isolation, isolation and multi-tenancy are inherent to network virtualization.

Virtual networks are isolated from any other virtual network and from the underlying physical network by default, delivering the security principle of least privilege. Virtual networks are created in isolation and remain isolated unless specifically connected together. No physical subnets, no VLANs, no ACLs, no firewall rules are required to enable this isolation.

Any isolated virtual network can be made up of workloads distributed anywhere in the data center. Workloads in the same virtual network can reside on the same or separate hypervisors. Additionally, workloads in several multiple isolated virtual networks can reside on the same hypervisor. Case in point, isolation between virtual networks allows for overlapping IP addresses, making it possible to have isolated development, test and production virtual networks, each with different application versions, but with the same IP addresses, all operating at the same time, all on the same underlying physical infrastructure.

Virtual networks are also isolated from the underlying physical infrastructure. Because traffic between hypervisors is encapsulated, physical network devices operate in a completely different address space than the workloads connected to the virtual networks. For example, a virtual network could support IPv6 application workloads on top of an IPv4 physical network. This isolation protects the underlying physical infrastructure from any possible attack initiated by workloads in any virtual network. Again, independent from any VLANs, ACLs, or firewall rules that would traditionally be required to create this isolation.

## Network Segmentation

Segmentation is easy with network virtualization. Related to isolation, but applied within a multi-tier virtual network, is segmentation. Traditionally, network segmentation is a function of a physical firewall or router, designed to allow or deny traffic between network segments or tiers. For example, segmenting traffic between a web tier, application tier and database tier. Traditional processes for defining and configuring segmentation are time consuming and highly prone to human error, resulting in a large percentage of security breaches. Implementation requires deep and specific expertise in device configuration syntax, network addressing, application ports and protocols.

Network segmentation, like isolation, is a core capability of VMware NSX network virtualization. A virtual network can support a multi-tier network environment, meaning multiple L2 segments with L3 segmentation or micro-segmentation on a single L2 segment using distributed firewall rules. As in the example above, these could represent a web tier, application tier and database tier. Physical firewalls and access control lists deliver a proven segmentation function, trusted by network security teams and compliance auditors. Confidence in this approach for cloud data centers, however, has been shaken, as more and more attacks, breaches and downtime are attributed to human error in to antiquated, manual network security provisioning and change management processes.

In a virtual network, network services (L2, L3, ACL, Firewall, QoS etc.) that are provisioned with a workload are programmatically created and distributed to the hypervisor vSwitch. Network services, including L3 segmentation and firewalling, are enforced at the virtual interface. Communication within a virtual network never leaves the virtual environment, removing the requirement for network segmentation to be configured and maintained in the physical network or firewall.

### Taking advantage of abstraction

Traditionally, network security required the security team to have a deep understanding of network addressing, application ports, protocols, all bound to network hardware, workload location and topology. Network virtualization abstracts application workload communication from the physical network hardware and topology, allowing network security to break free from these physical constraints and apply network security based on user, application and business context.

### Advanced Security Service Insertion, chaining and steering

The base VMware NSX network virtualization platform provides basic stateful firewalling features to deliver segmentation within virtual networks. In some environments, there is a requirement for more advanced network security capabilities. In these instances, customers can leverage VMware NSX to distribute, enable and enforce advanced network security services in a virtualized network environment. NSX distributes network services into the vSwitch to form a logical pipeline of services applied to virtual network traffic. Third party network services can be inserted into this logical pipeline, allowing physical or virtual services to be consumed in the logical pipeline.

Every security team uses a unique combination of network security products to meet the needs of their environment. The VMware NSX platform is being leveraged by VMware's entire ecosystem of security solution providers (<http://www.vmware.com/products/nsx/resources.html>). Network security teams are often challenged to coordinate network security services from multiple vendors in relationship to each other. Another powerful benefit of the NSX approach is its ability to build policies that leverage NSX service insertion, chaining and steering to drive service execution in the logical services pipeline, based on the result of other services, making it possible to coordinate otherwise completely unrelated network security services from multiple vendors.

For example, our integration with Palo Alto Networks (<http://researchcenter.paloaltonetworks.com/2013/11/palo-alto-networks-vmware-milestone-software-defined-data-center-security/>) will leverage the VMware NSX platform to distribute the Palo Alto Networks VM-Series next generation firewall, making the advanced features locally available on each hypervisor. Network security policies, defined for applications workloads provisioned or moved to that hypervisor, are inserted into the virtual network's logical pipeline. At runtime, the service insertion leverages the locally available Palo Alto Networks next-generation firewall feature set to deliver and enforce application, user, context-based controls policies at the workloads virtual interface.

### Consistent visibility and security model across both physical and virtual infrastructure

VMware NSX allows automated provisioning and context-sharing across virtual and physical security platforms. Combined with traffic steering and policy enforcement at the virtual interface, partner services, traditionally deployed in a physical network environment, are easily provisioned and enforced in a virtual network environment, VMware NSX delivers customers a consistent model of visibility and security across applications residing on both physical or virtual workloads.

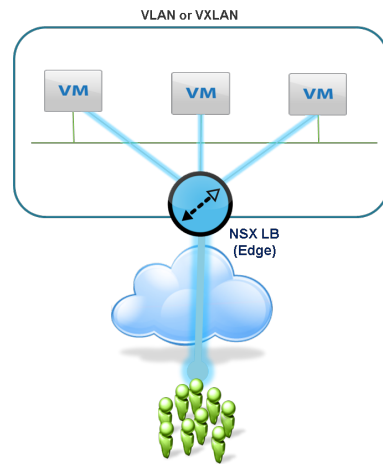
1. **Existing tools and processes.** Dramatically increase provisioning speed, operational efficiency and service quality while maintaining separation of duties between server, network and security teams.
2. **Control closer to the application, without downside.** Traditionally, this level of network security would have forced network and security teams to choose between performance and features. Leveraging the ability to distribute and enforce the advanced feature set at the applications virtual interface delivers the best of both.
3. **Reduce human error in the equation.** The infrastructure maintains policy, allowing workloads to be placed and moved anywhere in the data center, without any manual intervention. Pre-approved application security policies can be applied programmatically, enabling self-service deployment of even complex network security services.



## Logical Load Balancing

Load Balancing is another network service available within NSX.

This service offers **distribution workload** across multiple servers, as well as **high-availability of applications**:



**Figure 28.** NSX Load Balancing

The NSX load balancing service is specially **designed for cloud** with:

- Fully programmable via API
- Same single central point of management/monitoring as other NSX network services

The NSX load balancing service replies to customer applications load balancing needs with:

- **Multiple architectures support**
  - 1-arm mode (called proxy mode)
  - 2-arms mode (called transparent mode)
- **Large feature set to support an extensive number of applications**
  - Support any TCP applications
  - Including, but not limited to, LDAP, FTP, HTTP, HTTPS
  - Multiple load balancing distribution algorithms
  - Round-robin, least connections, source IP hash, URI
  - Multiple health checks
  - TCP, HTTP, HTTPS including content inspection
  - Persistence
  - Source IP, MSRD, cookie, ssl session-id
  - Connection throttling
  - Max connections and connections/sec
  - L7 manipulation
  - Including, but not limited to, URL block, URL rewrite, content rewrite
  - Optimization
  - SSL offload

The NSX load balancing service **scales very high to support very demanding applications**.

Each NSX edge scales up to:

- Throughput: 9Gbps
- Concurrent connections: 1 million
- New connections per sec: 131k

## Conclusion

The VMware network virtualization solution addresses current challenges with physical network infrastructure and brings flexibility, agility and scale through VXLAN-based logical networks. Along with the ability to create on-demand logical networks using VXLAN, the vCloud Networking and Security Edge gateway helps users deploy various logical network services such as firewall, DHCP, NAT and load balancing on these networks. This is possible due to its ability to decouple the virtual network from the physical network and then reproduce the properties and services in the virtual environment.

### References

[1] *What's New in VMware vSphere 5.5*

<http://www.vmware.com/files/pdf/vsphere/VMware-vSphere-Platform-Whats-New.pdf>

[2] *vSphere 5.5 Configuration Maximums*

<http://www.vmware.com/pdf/vsphere5/r55/vsphere-55-configuration-maximums.pdf>