Cloud Computing

Network Virtualization

Agenda

- Introduction
- External network virtualization
  - What to be virtualized ?
    - Network device virtualization
    - Network data path virtualization
  - How to be virtualized ?
    - Protocol approach
- Internal network virtualization
  - Traditional approach
  - New techniques
  - Case study
- Best practice with VMware

- Introduction
- External network virtualization
- Internal network virtualization
- Best Practices with VMware

NETWORK VIRTUALIZATION

Related Concepts

- Virtual Private Networks (VPN)
  - Virtual network connecting distributed sites
  - Not customizable enough
- Active and Programmable Networks
  - Customized network functionalities
  - Programmable interfaces and active codes
- Overlay Networks
  - Application layer virtual networks
  - Not flexible enough

Network Virtualization Model

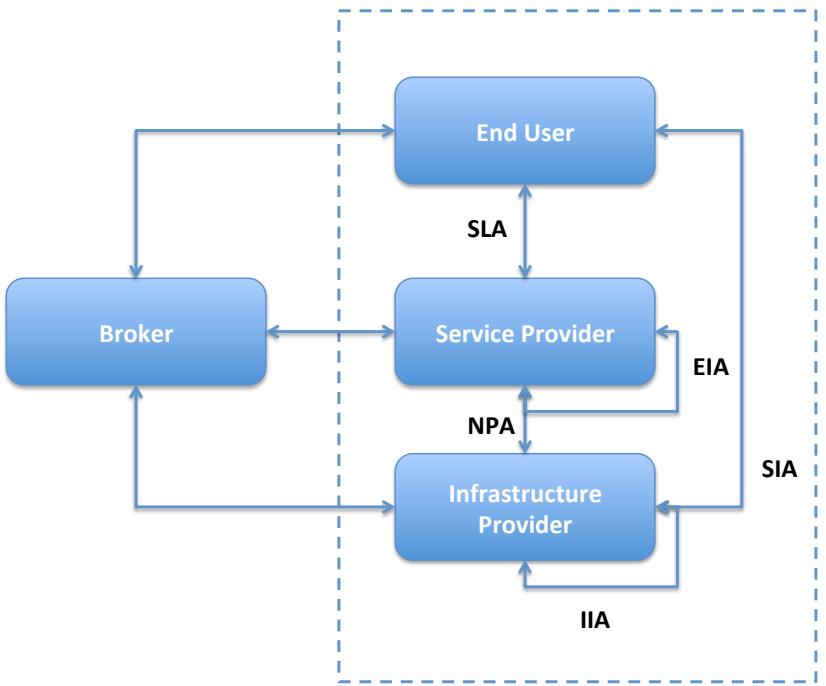
- Business Model
- Architecture
- Design Principles
- Design Goals

Business Model

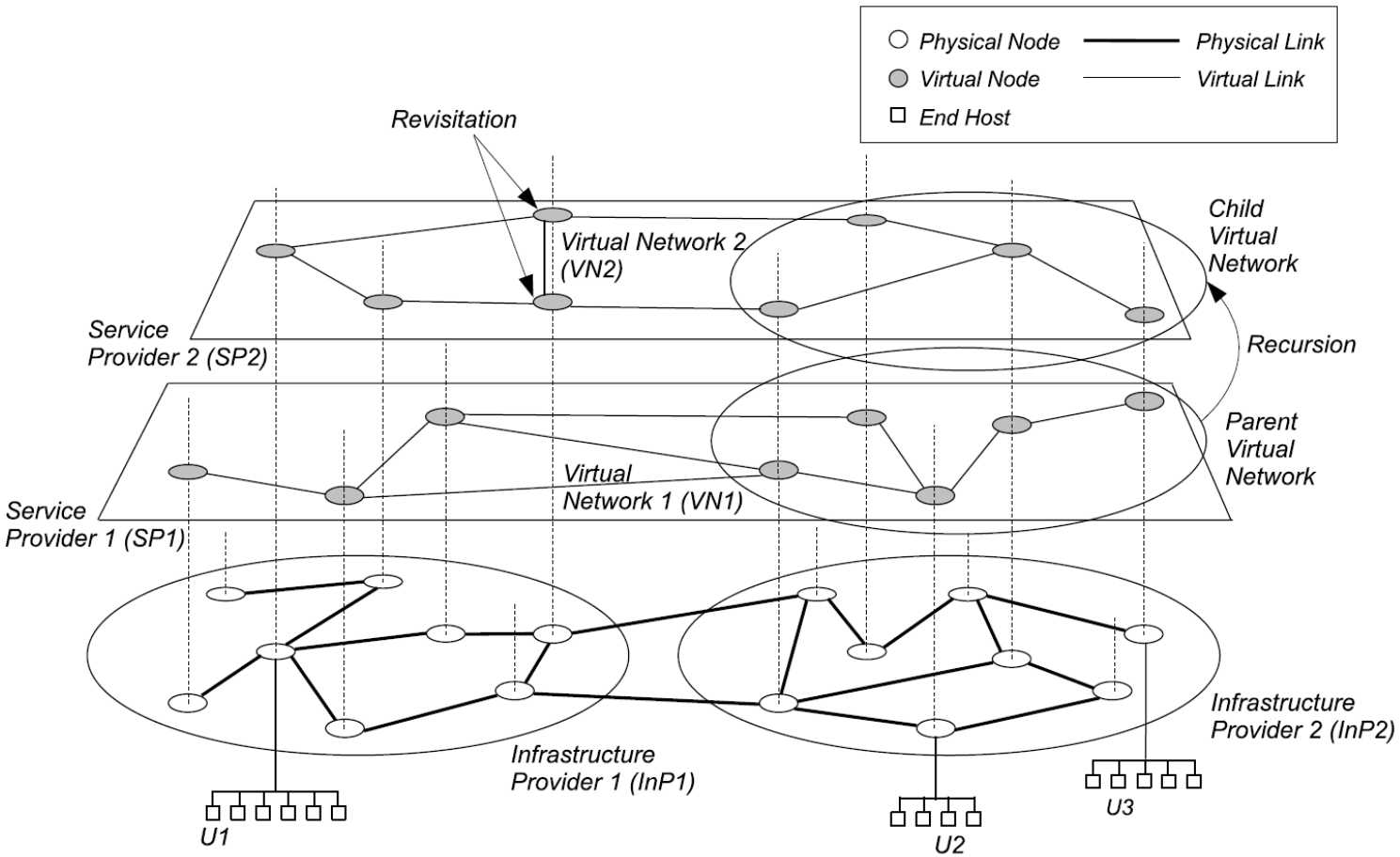
Players

- Infrastructure Providers (*InPs*)
  - Manage underlying physical networks
- Service Providers (*SPs*)
  - Create and manage virtual networks
  - Deploy customized end-to-end services
- End Users
  - Buy and use services from different service providers
- Brokers
  - Mediators/Arbiters

Relationships



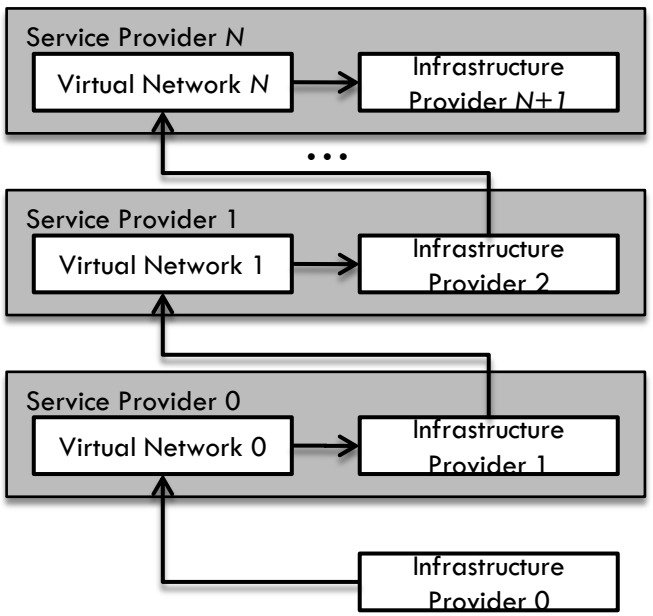
Architecture



Design Principles

- *Concurrence* of multiple heterogeneous virtual networks
  - Introduces diversity
- *Recursion* of virtual networks
  - Opens the door for network virtualization economics
- *Inheritance* of architectural attributes
  - Promotes value-addition
- *Revisitation* of virtual nodes
  - Simplifies network operation and management

Hierarchy of Roles



Design Goals

- Flexibility
  - Service providers can choose
    - arbitrary network topology,
    - routing and forwarding functionalities,
    - customized control and data planes
- Scalability
  - Maximize the number of co-existing virtual networks
  - Increase resource utilization and amortize CAPEX and OPEX
- Security, Privacy, and Isolation
  - Complete isolation between virtual networks
    - *Logical* and *resource*
  - Isolate faults, bugs, and misconfigurations
    - Secured and private

Design Goals (2)

- Programmability
  - Of network elements e.g. routers
  - Answer *"How much"* and *"how"*
  - Easy and effective without being vulnerable to threats

- Heterogeneity
  - Networking technologies
    - Optical, sensor, wireless etc.
  - Virtual networks

Definition (Sort of)

***Network virtualization** is a networking environment that allows multiple service providers to dynamically compose multiple heterogeneous virtual networks that co-exist together in isolation from each other, and to deploy customized end-to-end services on-the-fly as well as manage them on those virtual networks for the end-users by effectively sharing and utilizing underlying network resources leased from multiple infrastructure providers.*

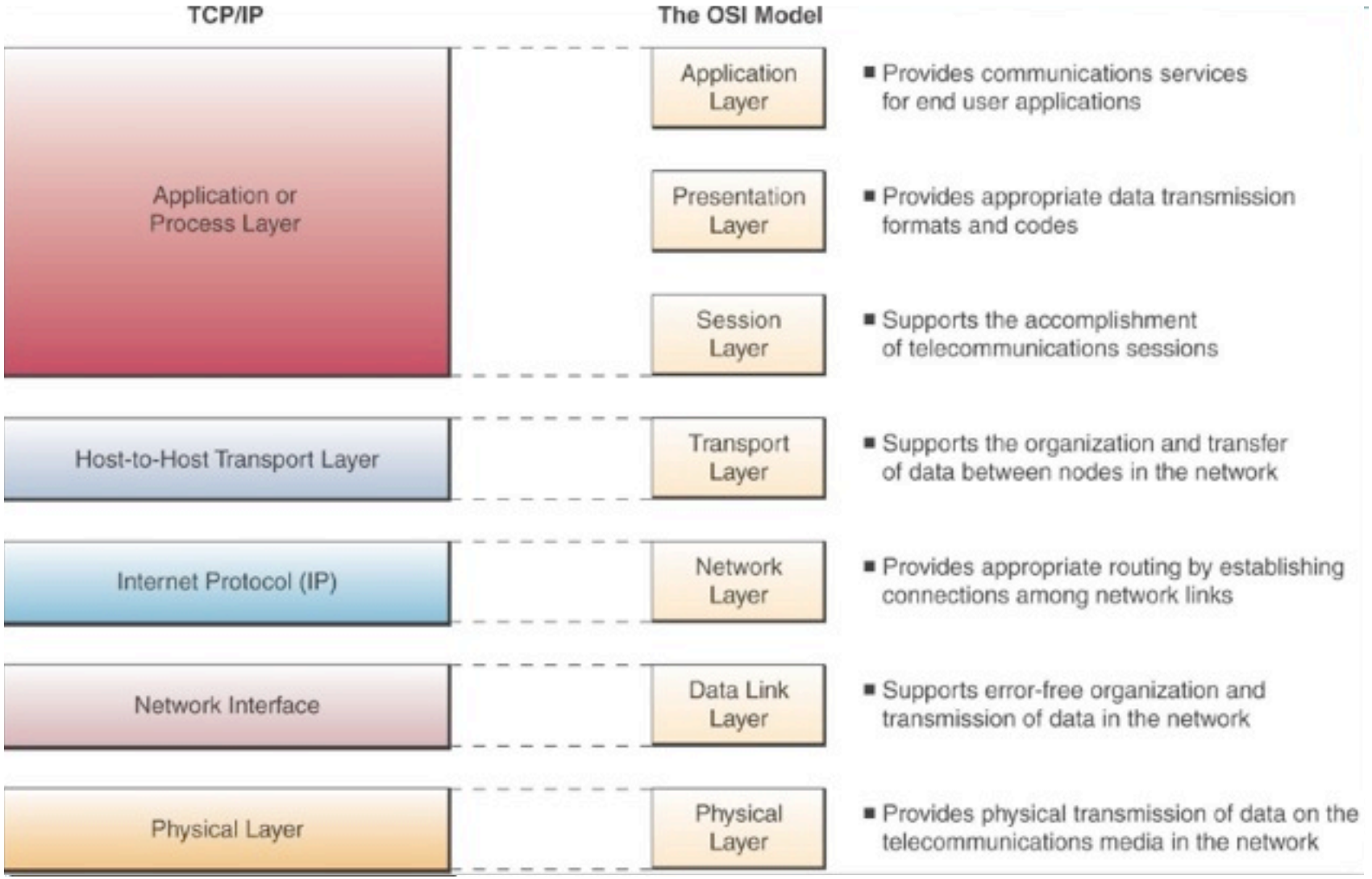
Introduction

- What is computer network ?
  - A computer network, often simply referred to as a network, is a collection of computers and devices interconnected by communications channels that facilitate communications among users and allows users to share resources.
- Why should we study network ?
  - Computer networks are used for communication and coordination, as well as commerce by large as well as small organizations.
  - Computer networks and the Internet is a vital part of business infrastructure.

Network Protocol and Model

- Network protocol
  - Rules and procedures governing transmission between computers
  - Used to identify communicating devices, secure attention of intended recipient, check for errors and re-transmissions
  - All computers using a protocol have to agree on how to code/decode the message, how to identify errors, and steps to take when there are errors or missed communications

Network Protocol and Model

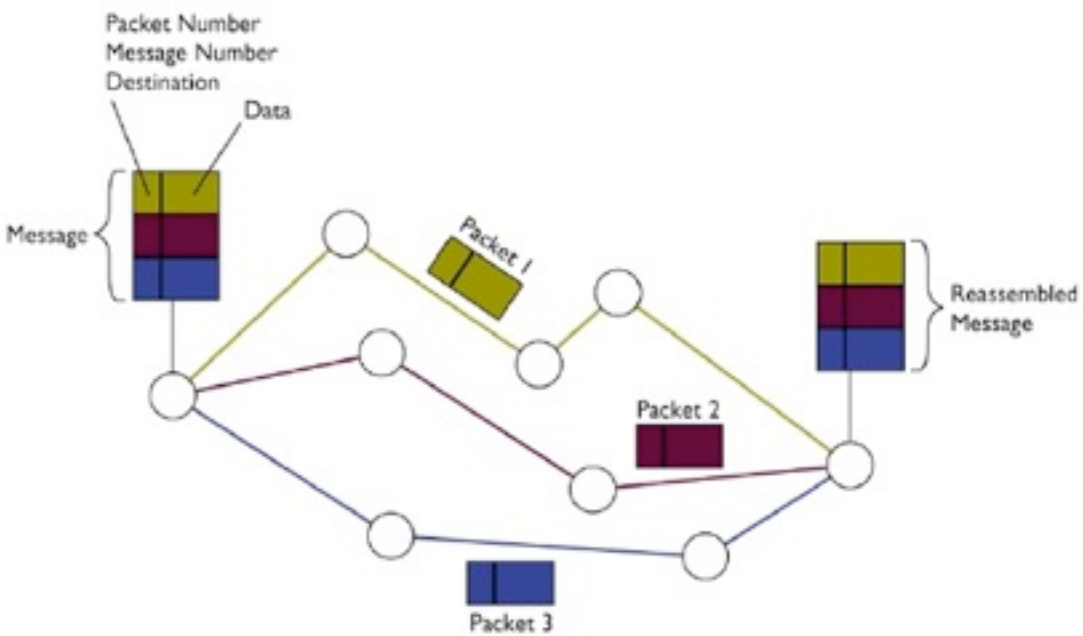


Network Types

- LANs and WANs
  - Local area network
    - Network of computers and other devices within a limited distance
    - Uses star, bus or ring topologies
    - Network interface cards in each device specifies transmission rate, message structure, and topology
    - Network operating system routes and manages communications and coordinates network resources
  - Wide area network
    - Network of computers spanning broad geographical distances
    - Switched or dedicated lines
    - Firms use commercial WANs for communication

Network Architecture

- Packet switching
  - Message/Data is divided into fixed or variable length packets
  - Each packet is numbered and sent along different paths to the destination
  - Packets are assembled at the destination
  - Useful for continued message transmission even when part of the network path is broken

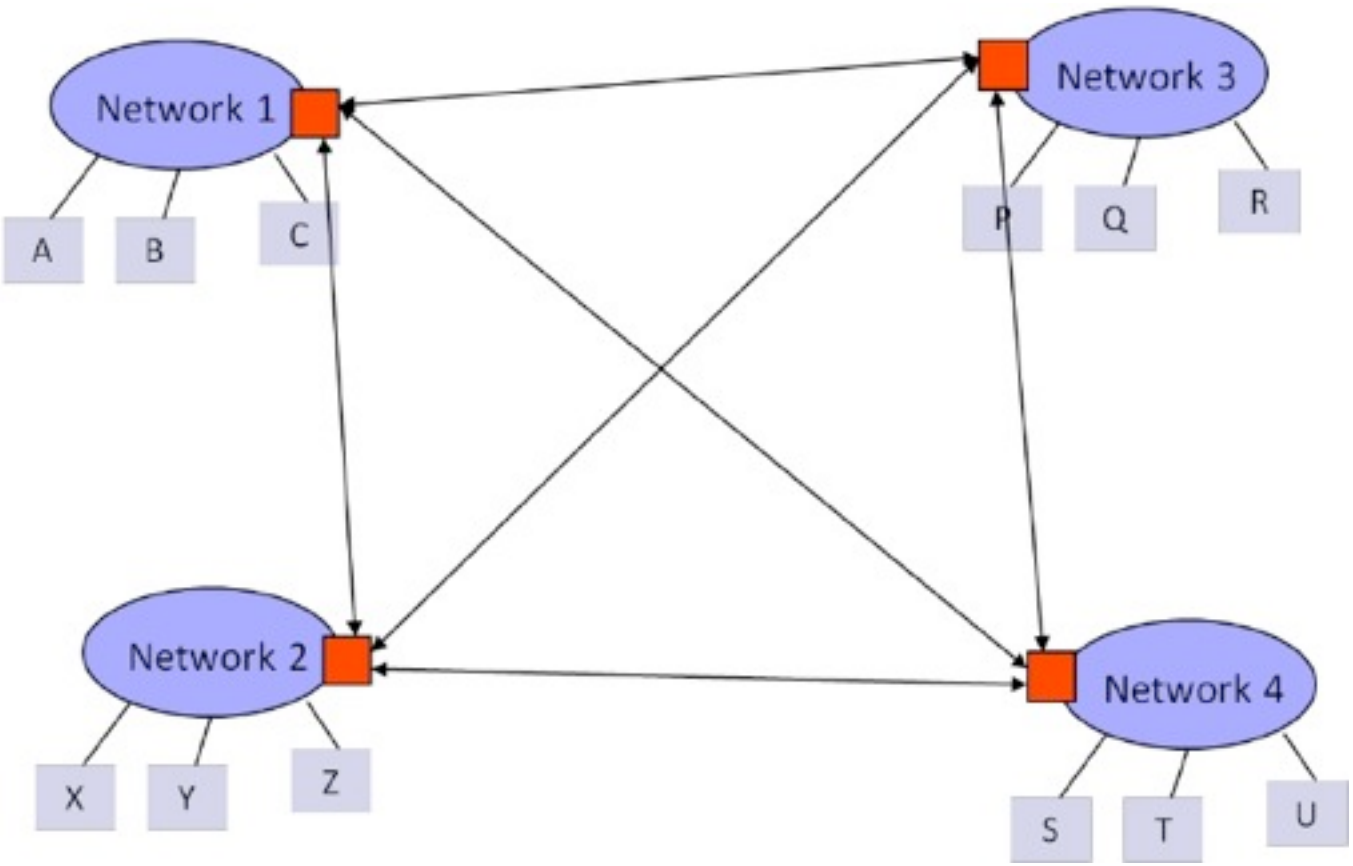


Network Architecture

Connect two networks

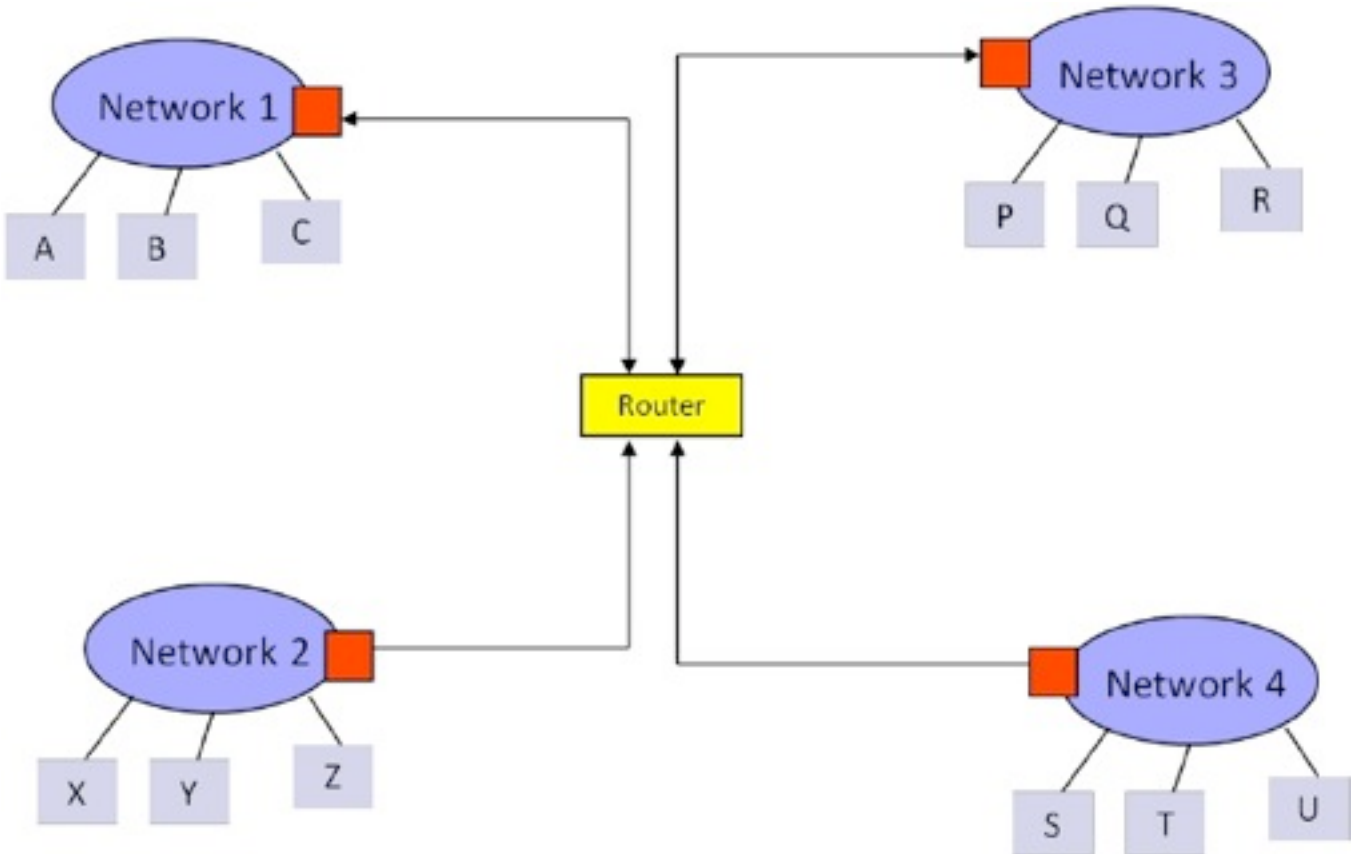
Network Architecture

Connect multiple networks



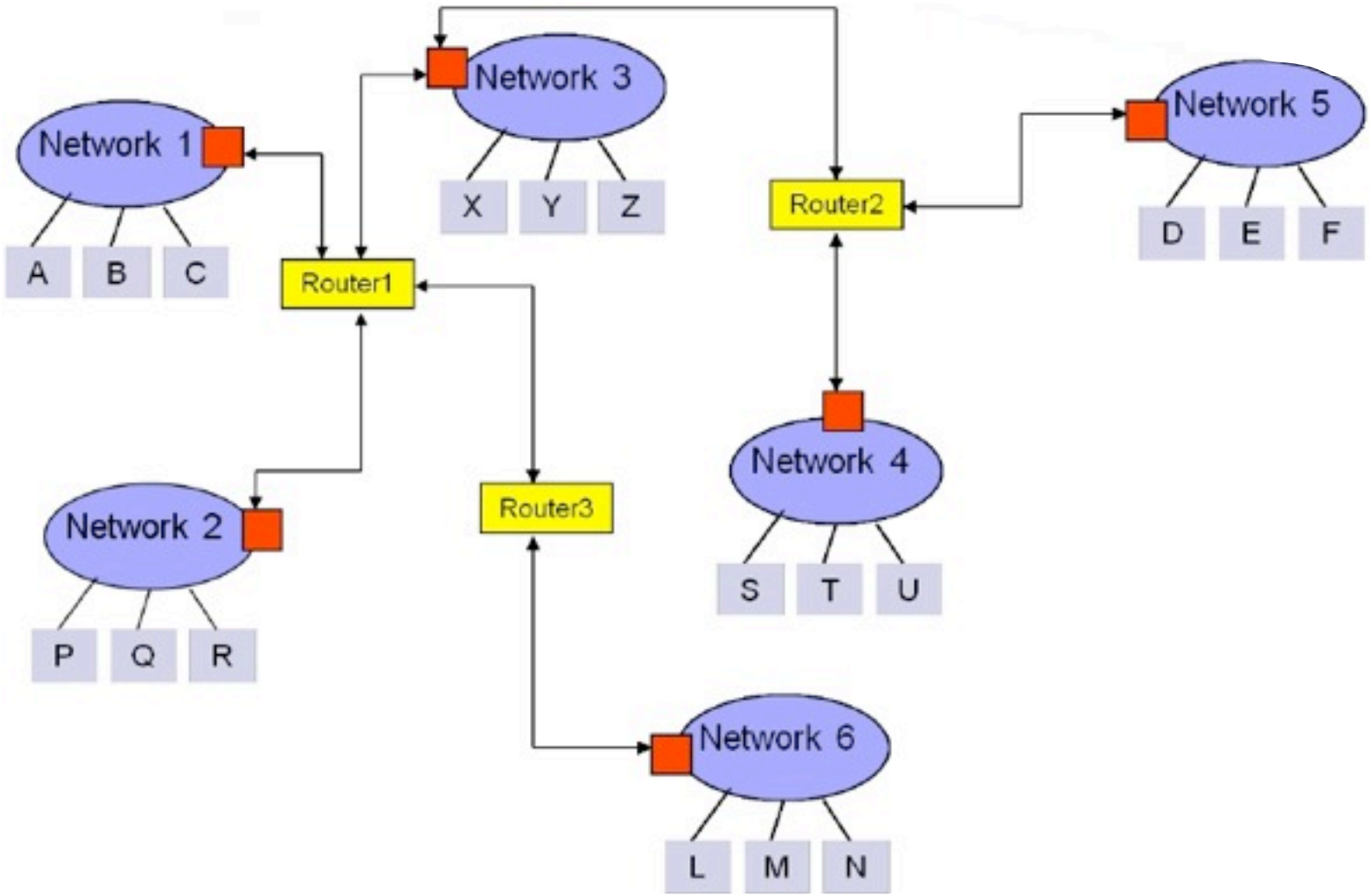
Network Architecture

Connect multiple networks



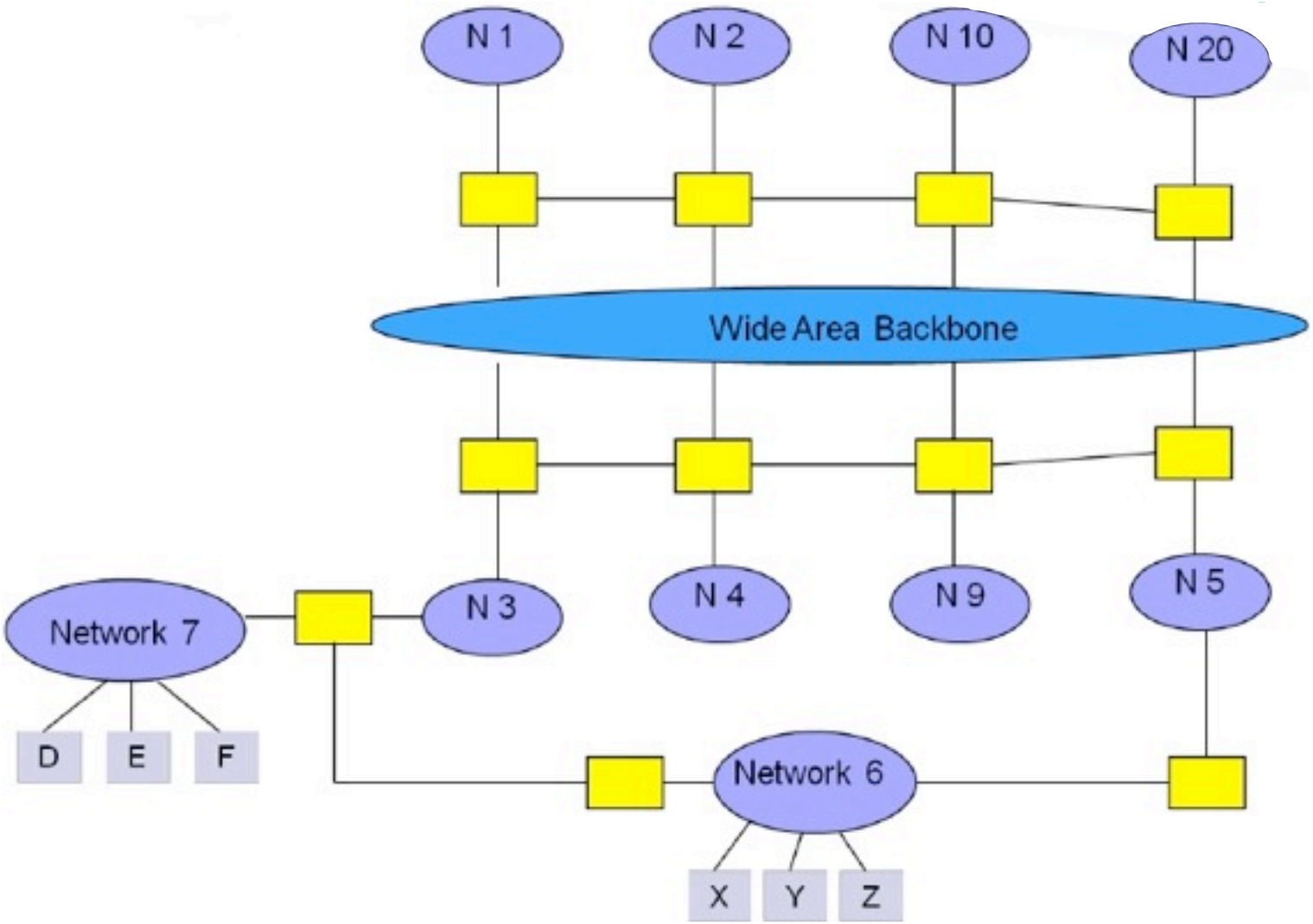
Network Architecture

Connect multiple networks

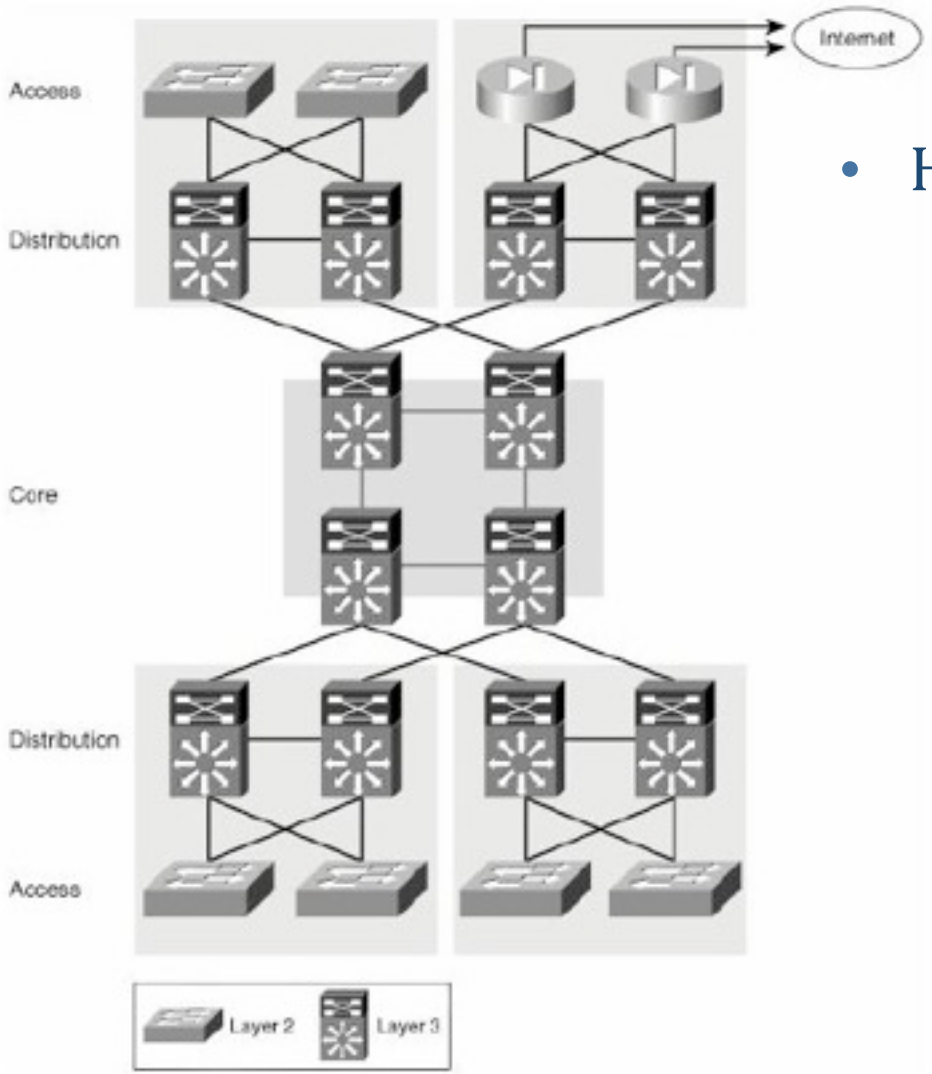


Network Architecture

The simple view of Internet



Network Design Rules



- Hierarchical approach
  - Traffic is aggregated hierarchically from an access layer into a layer of distribution switches and finally onto the network core.
  - A hierarchical approach to network design has proven to deliver the best results in terms of optimizing scalability, improving manageability,and maximizing network availability.

Network Virtualization

- Two categories :
  - External network virtualization
    - Combining many networks, or parts of networks, into a virtual unit.
  - Internal network virtualization
    - Providing network-like functionality to the software containers on a single system.

- Introduction
- External network virtualization
- Internal network virtualization
- Best Practices with VMware

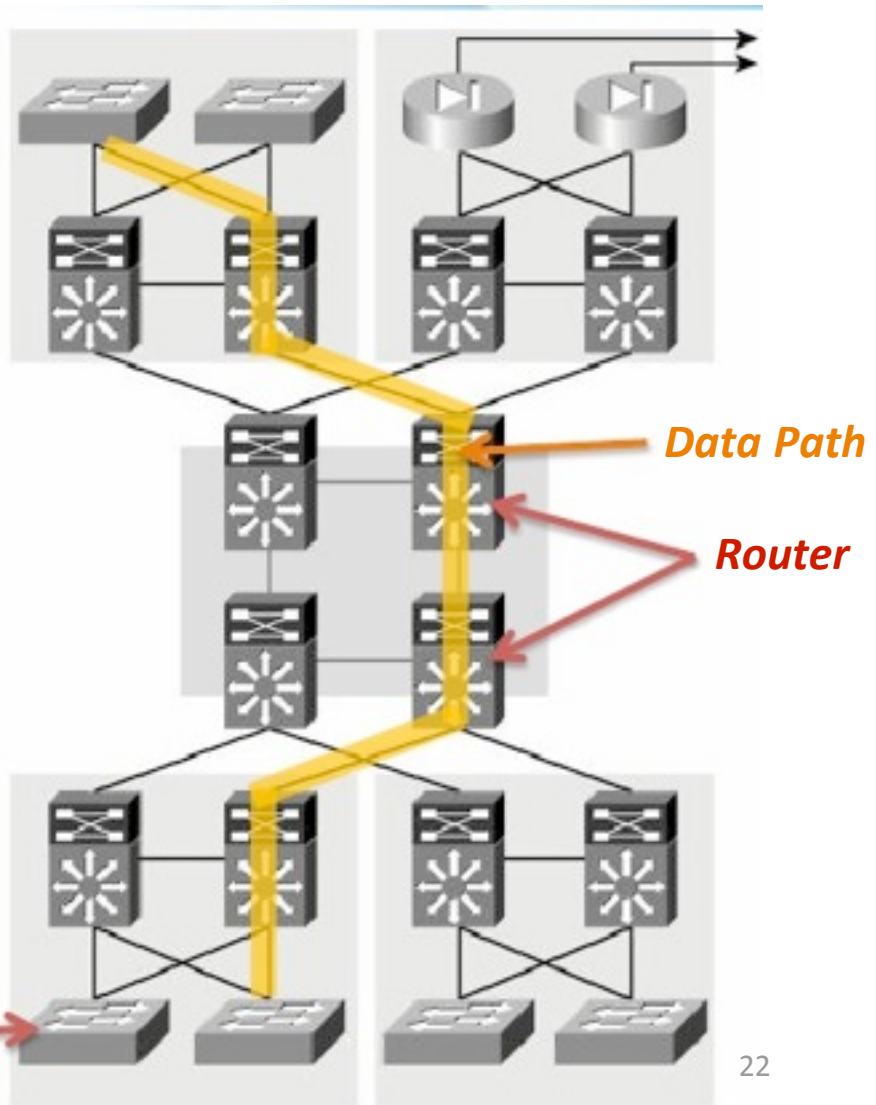
NETWORK VIRTUALIZATION

Network Virtualization

- External network virtualization in different layers :
  - Layer1
    - Seldom virtualization implement in this physical data transmission layer.
  - Layer2
    - Use some tags in MAC address packet to provide virtualization.
    - Example, VLAN.
  - Layer3
    - Use some tunnel techniques to form a virtual network.
    - Example, VPN.
  - Layer4 or higher
    - Build up some overlay network for some application.
    - Example, P2P.

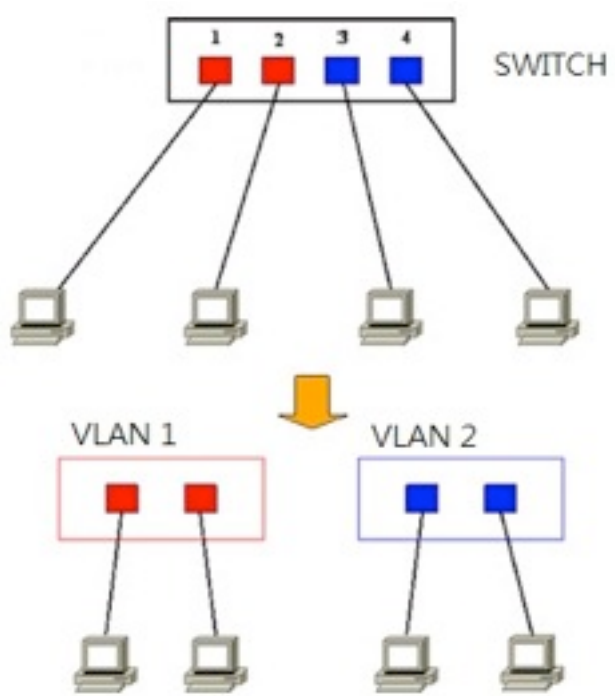
Network Virtualization

- Two virtualization components :
  - Device virtualization
    - Virtualize physical devices in the network
  - Data path virtualization
    - Virtualize communication path between network access points



Network Virtualization

- Device virtualization
  - Layer2 solution
    - Divide physical switch into multiple logical switches.

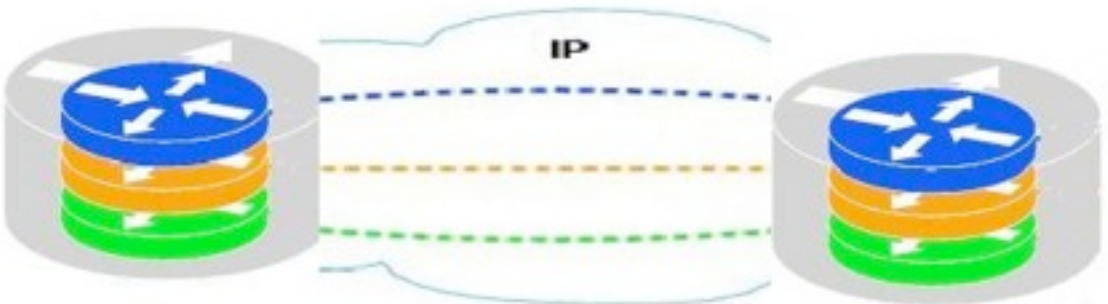


  - Layer 3 solution 3
    - VRF technique
      (Virtual Routing and Forwarding)
    - Emulate isolated routing tables within one physical router.

Network Virtualization

- Data path virtualization
  - Hop-to-hop case
    - Consider the virtualization applied on a single hop data-path.
  - Hop-to-cloud case
    - Consider the virtualization tunnels allow multi-hop data-path.

L2 based labeling allows single hop data path virtualization



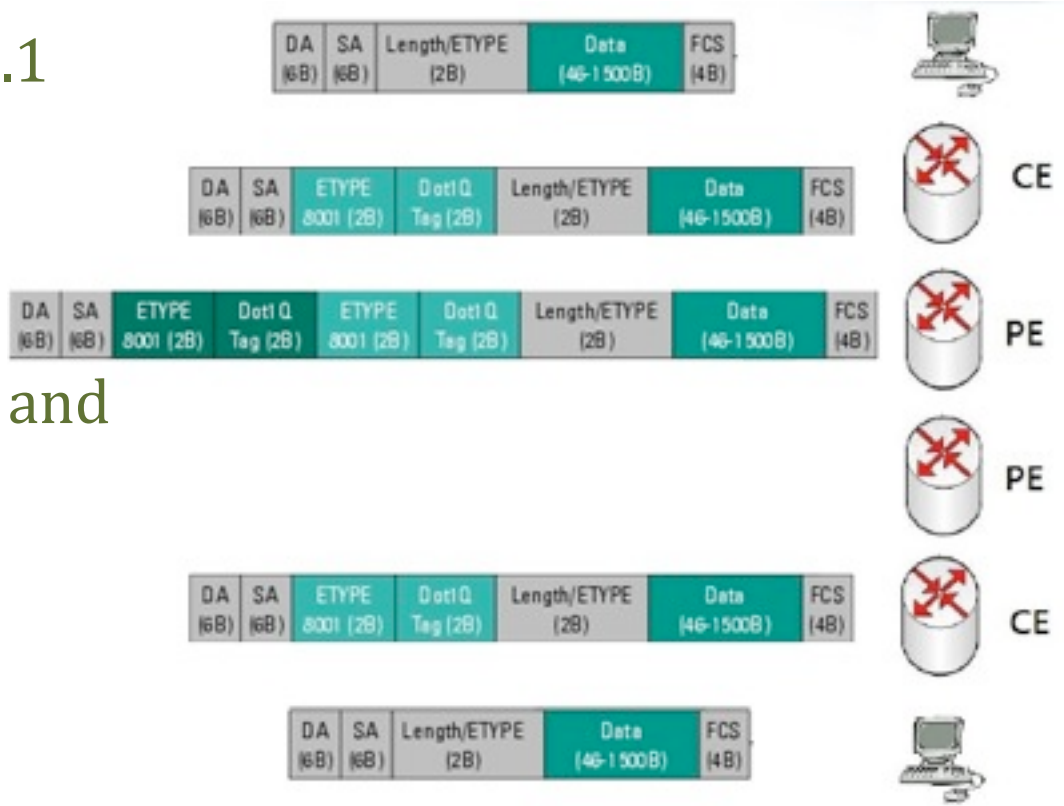
Tunnels allow multi-hop data path virtualization

Network Virtualization

- Protocol approach
  - Protocols usually used to approach data-path virtualization.
  - Three implementations
    - **802.1Q** – implement hop to hop data-path virtualization
    - **MPLS (Multiprotocol Label Switch)** – implement router and switch layer virtualization
    - **GRE (Generic Routing Encapsulation)** – implement virtualization among wide variety of networks with tunneling technique.

Network Virtualization

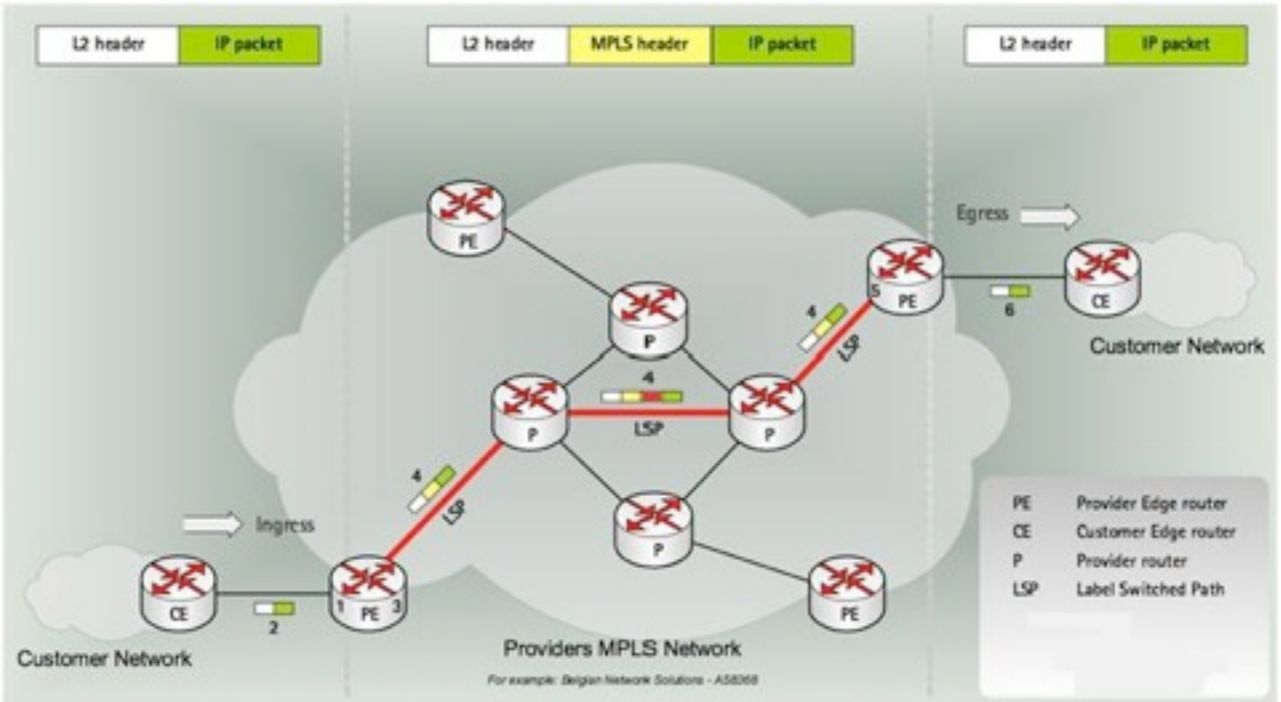
- 802.1Q
  - Standard by IEEE 802.1
  - Not encapsulate the original frame
  - Add a 32-bit field between *MAC address* and *EtherTypes* field
    - ETYPE(2B): Protocol identifier
    - Dot1Q Tag(2B): VLAN number, Priority code



CE: Customer Edge router
PE: Provider Edge router

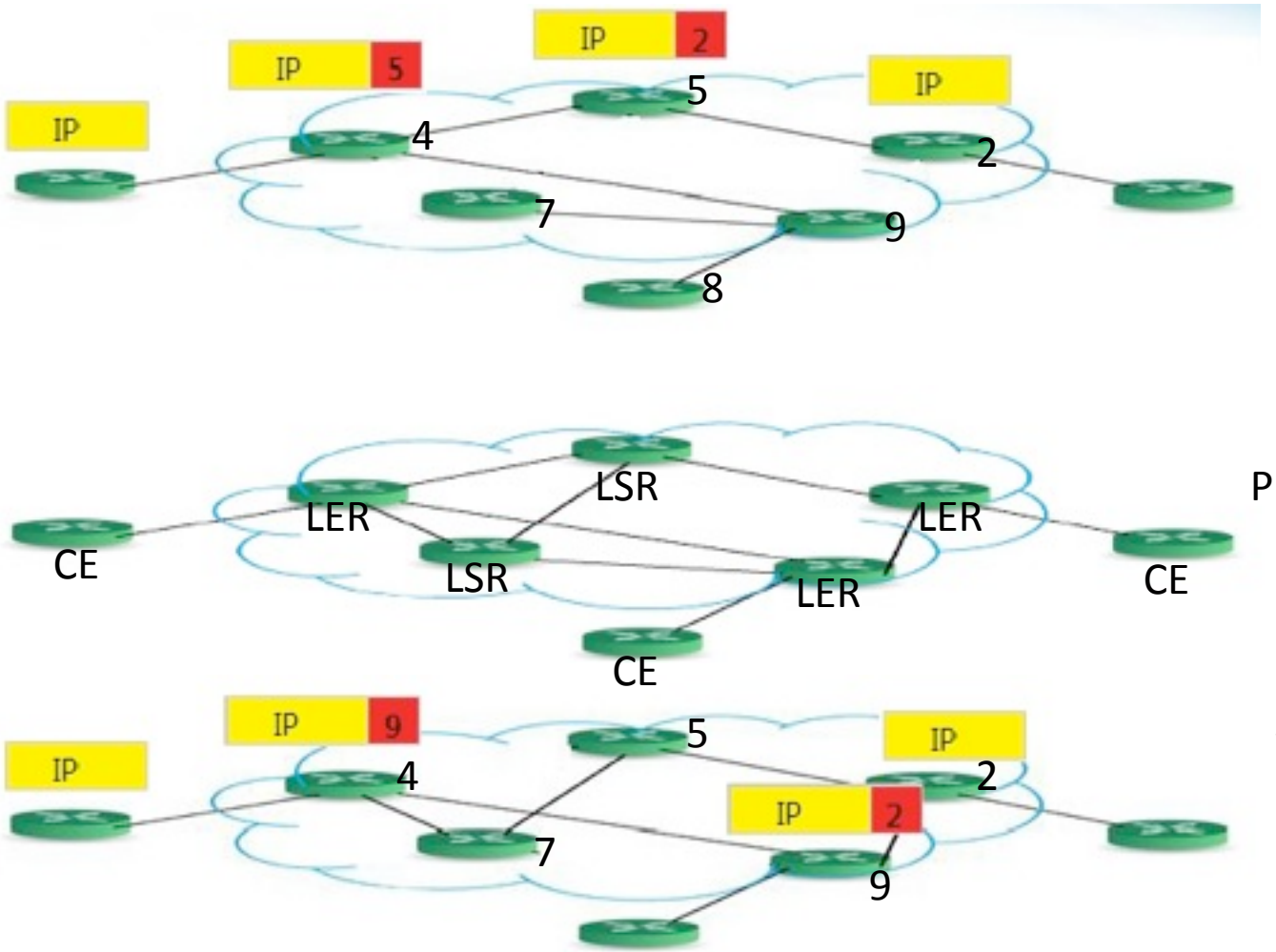
Network Virtualization

- MPLS (Multiprotocol Label Switch)
  - Also classified as layer 2.5 virtualization
  - Add one or more labels into package
  - Need Label Switch Router(LSR)to read MPLS header



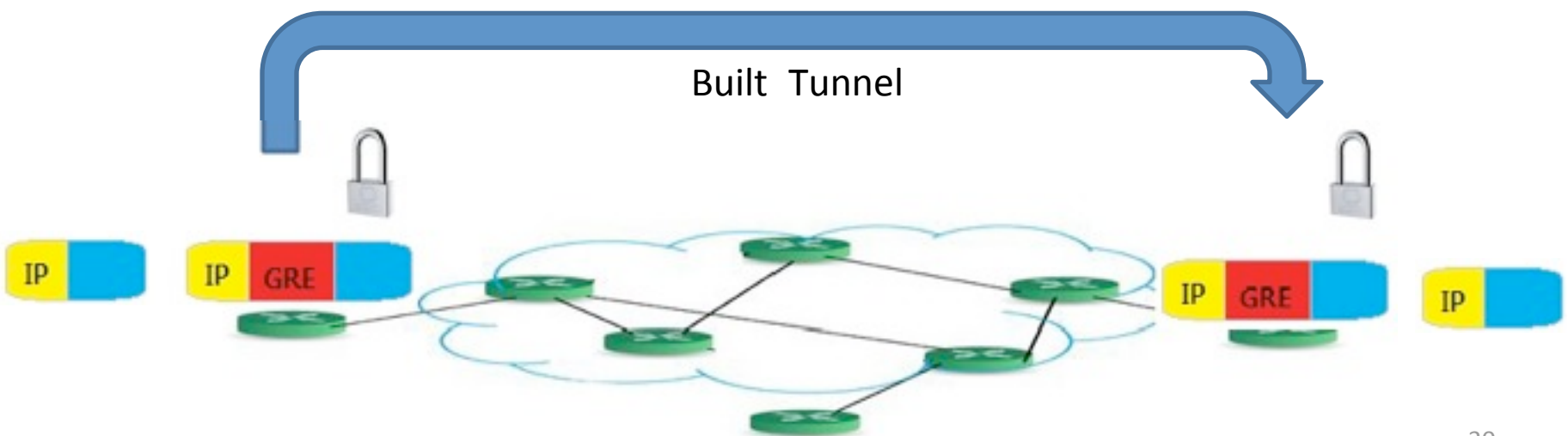
Network Virtualization

- Example of MPLS



Network Virtualization

- GRE (Generic Routing Encapsulation)
  - GRE is a tunnel protocol developed by CISCO
  - Encapsulate a wide variety of network layer protocol
  - Stateless property
    - This means end-point doesn't keep information about the state



- Introduction
- External network virtualization
- Internal network virtualization
- Best Practices with VMware

NETWORK VIRTUALIZATION

Internal Network Virtualization

- Internal network virtualization
  - A single system is configured with containers, such as the Xen domain, combined with hypervisor control programs or pseudo-interfaces such as the VNIC, to create a “network in a box”.
  - This solution improves overall efficiency of a single system by isolating applications into separate containers and/or pseudo interfaces.
  - Virtual machine and virtual switch :
    - The VMs are connected logically to each other so that they can send data to and receive data from each other.
    - Each virtual network is serviced by a single virtual switch.
    - A virtual network can be connected to a physical network by associating one or more network adapters (uplink adapters) with the virtual switch.

Network Virtualization

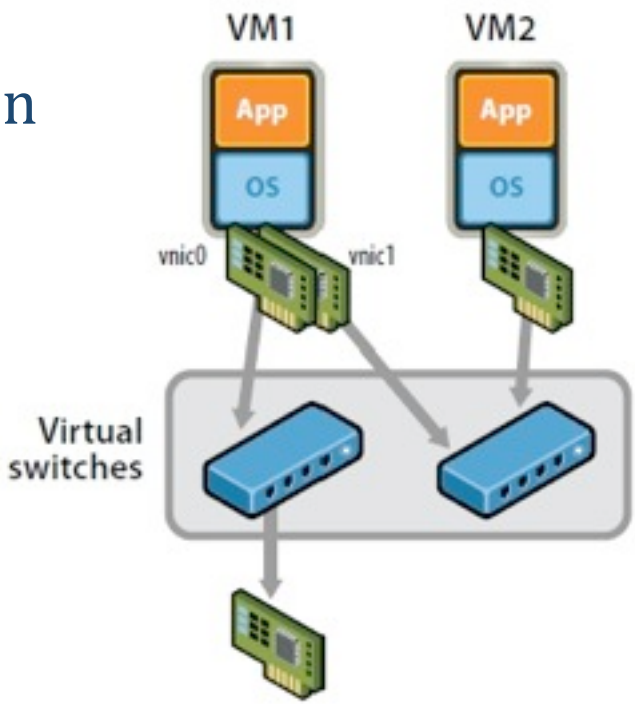
- Internal network virtualization in different layers :
  - Layer1
    - Hypervisor usually do not need to emulate the physical layer.
  - Layer2
    - Implement virtual L2 network devices, such as switch, in hypervisor.
    - Example, Linux TAP driver + Linux bridge.
  - Layer3
    - Implement virtual L3 network devices, such as router, in hypervisor.
    - Example, Linux TUN driver + Linux bridge + IP-tables.
  - Layer4 or higher
    - Layer 4 or higher layers virtualization is usually implemented in guest OS.
    - Applications should make their own choice.

Network Virtualization

- Desirable properties of network virtualization :
  - Scalability
    - Easy to extend resources in need
    - Administrator can dynamically create or delete virtual network connection
  - Resilience
    - Recover from the failures
    - Virtual network will automatically redirect packets by redundant links
  - Security
    - Increased path isolation and user segmentation
    - Virtual network should work with firewall software
  - Availability
    - Access network resource anytime

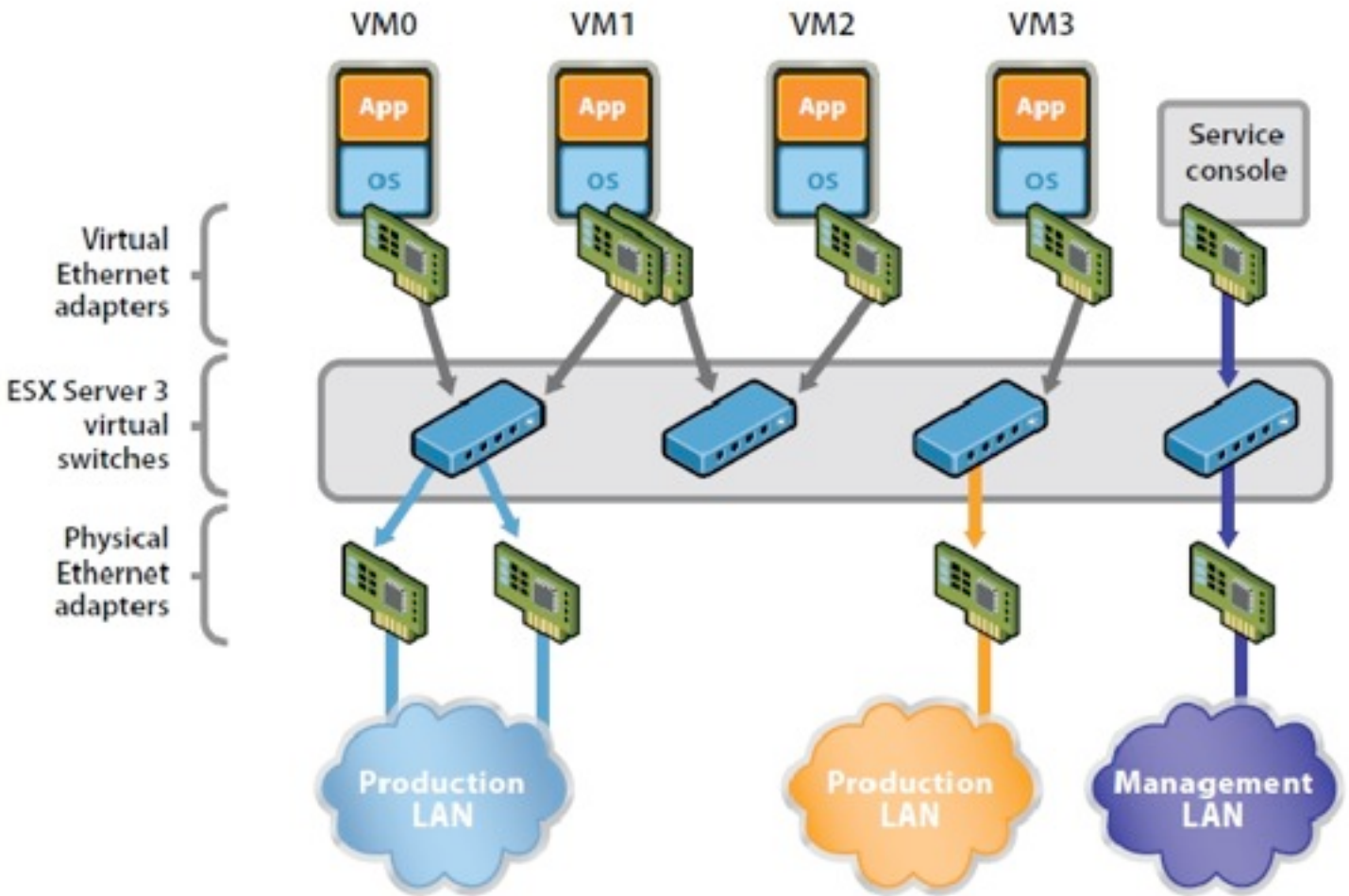
Internal Network Virtualization

- Properties of virtual switch
  - A virtual switch works much like a physical Ethernet switch.
  - It detects which VMs are logically connected to each of its virtual ports and uses that information to forward traffic to the correct virtual machines.
- Typical virtual network configuration
  - Communication network
    - Connect VMs on different hosts
  - Storage network
    - Connect VMs to remote storage system
  - Management network
    - Individual links for system administration



Internal Network Virtualization

Network virtualization example form VMware



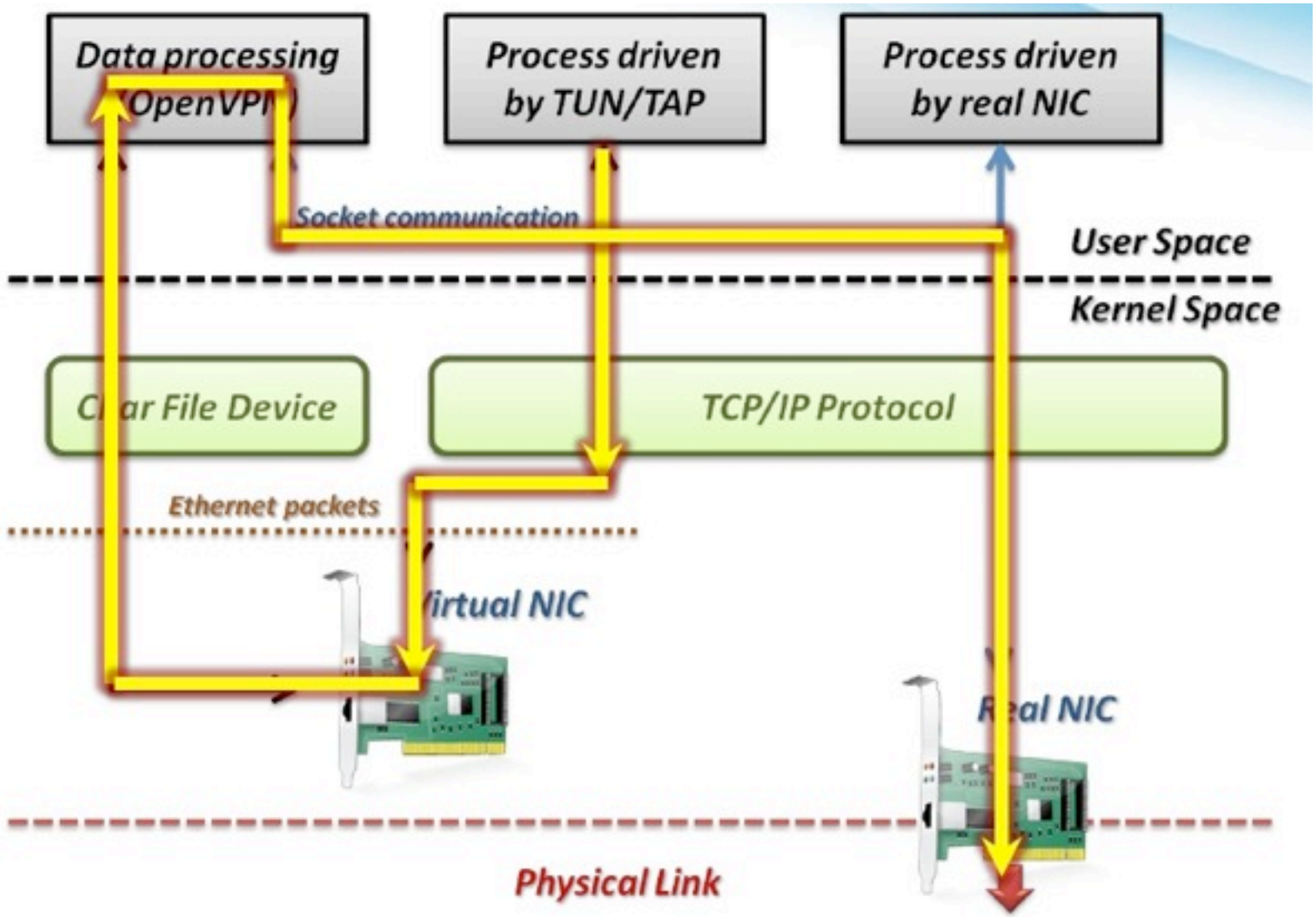
Traditional Approach

- KVM (Kernel-based Virtual Machine) is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (Intel VT or AMD-V). It consists of a loadable kernel module, kvm.ko, that provides the core virtualization infrastructure and a processor specific module, kvm-intel.ko or kvm-amd.ko
  - KVM focus on CPU and memory virtualization, so IO virtualization framework is completed by QEMU project.
  - In QEMU, network interface of virtual machines connect to host by TUN/TAP driver and Linux bridge.
  - Work with TUN/TAP and Linux Bridge :
    - Virtual machines connect to host by a virtual network adapter, which is implemented by TUN/TAP driver.
    - Virtual adapters will connect to Linux bridges, which play the role of virtual switch.

Traditional Approach

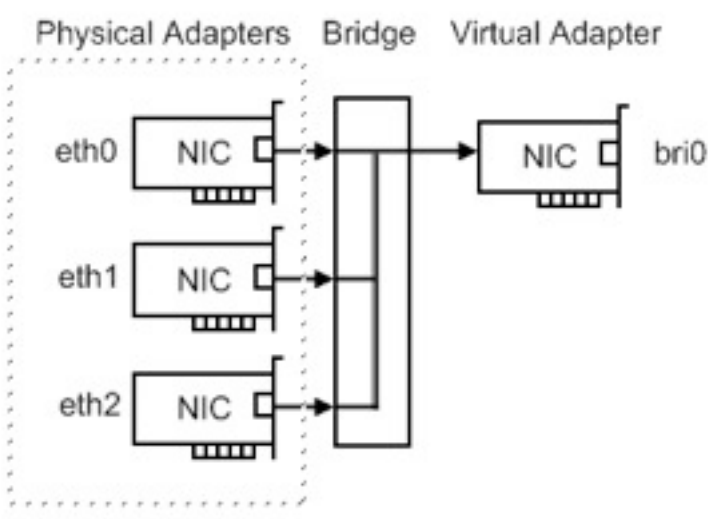
- TUN/TAP driver
  - TUN and TAP are virtual network kernel drivers :
    - TAP (as in network tap) simulates an Ethernet device and it operates with layer 2 packets such as Ethernet frames.
    - TUN (as in network TUNnel) simulates a network layer device and it operates with layer 3 packets such as IP.
  - Data flow of TUN/TAP driver
    - Packets sent by an operating system via a TUN/TAP device are delivered to a user-space program that attaches itself to the device.
    - A user-space program may pass packets into a TUN/TAP device.
    - TUN/TAP device delivers (or "injects") these packets to the operating system network stack thus emulating their reception from an external source.

Traditional Approach



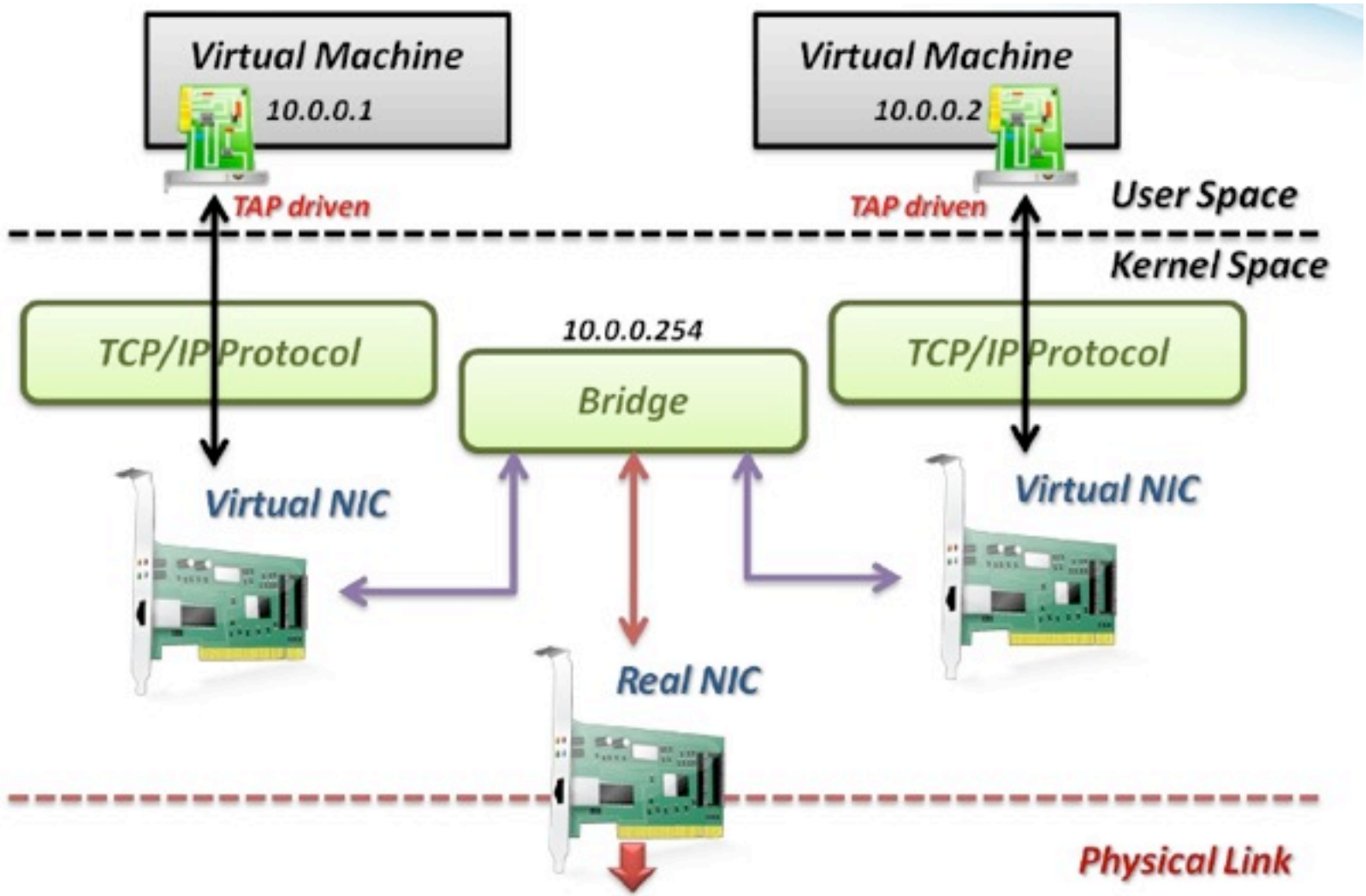
Traditional Approach

- Linux bridge
  - Bridging is a forwarding technique used in packet-switched computer networks.
  - Unlike routing, bridging makes no assumption about where in a network a particular address is located.
  - Bridging depends on flooding and examination of source addresses in received packet headers to locate unknown devices.
  - Bridging connects multiple network segments at the data link layer (Layer 2) of the OSI model.



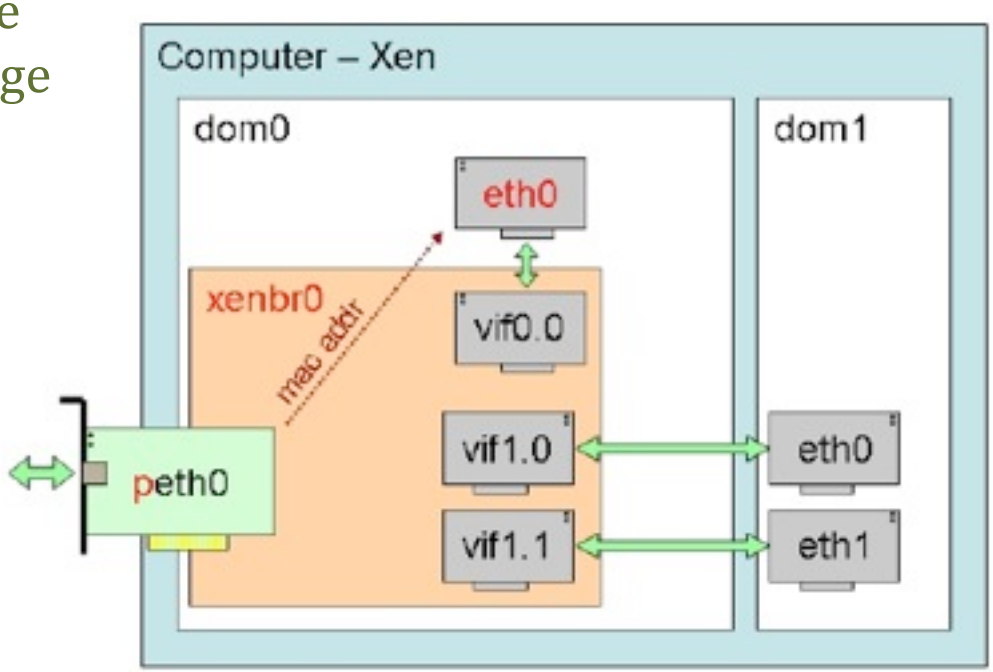
Traditional Approach

TAP/TUN driver + Linux Bridge



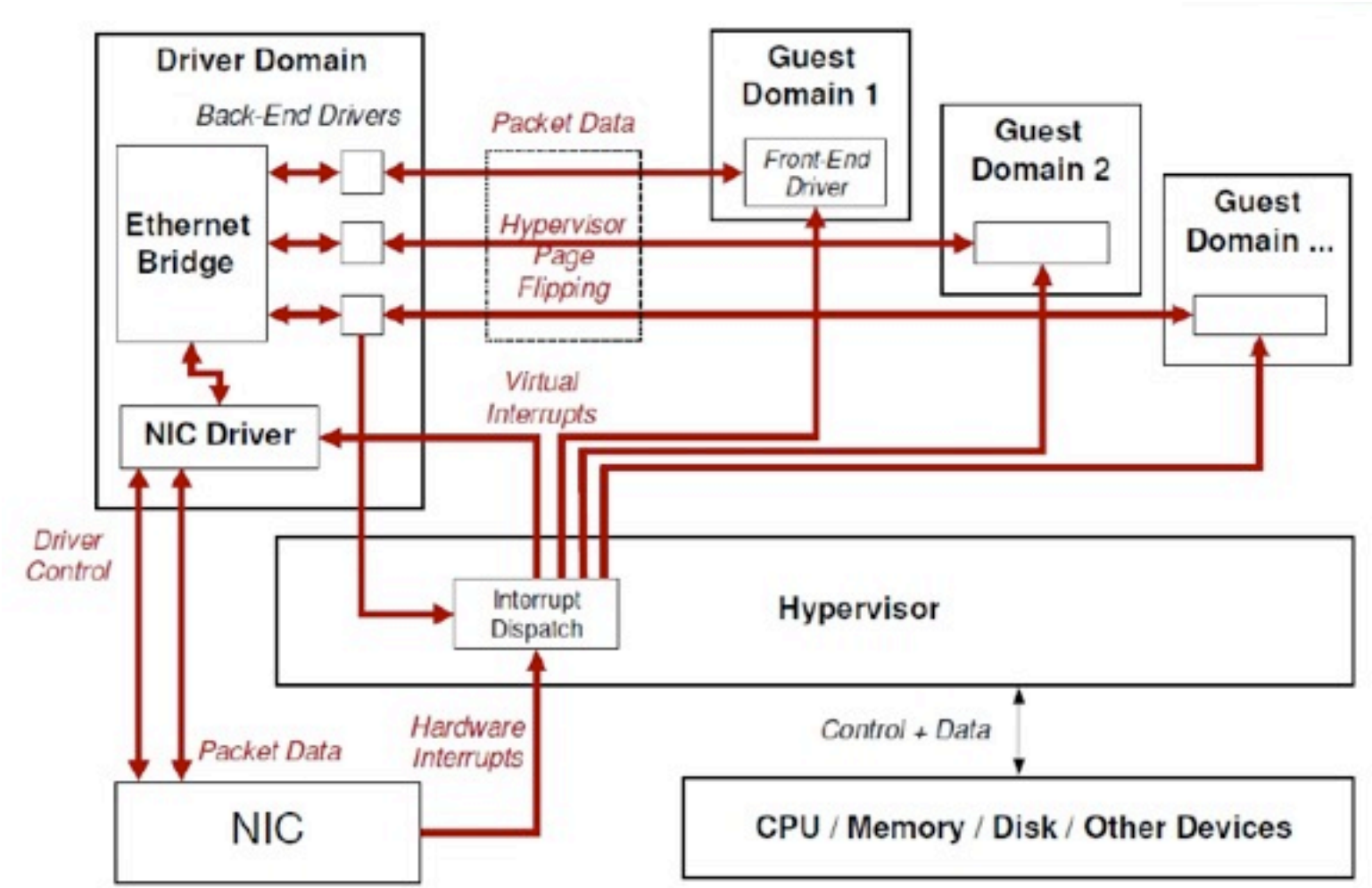
New Techniques

- In Xen system
  - Since implemented with para-virtualization type, guest OS load modified network interface drivers.
  - Modified network interface drivers communicate with virtual switches in Dom0, which act as TAP in traditional approach.
  - Virtual switch in Xen can be implemented by Linux bridge or work with other optimization.



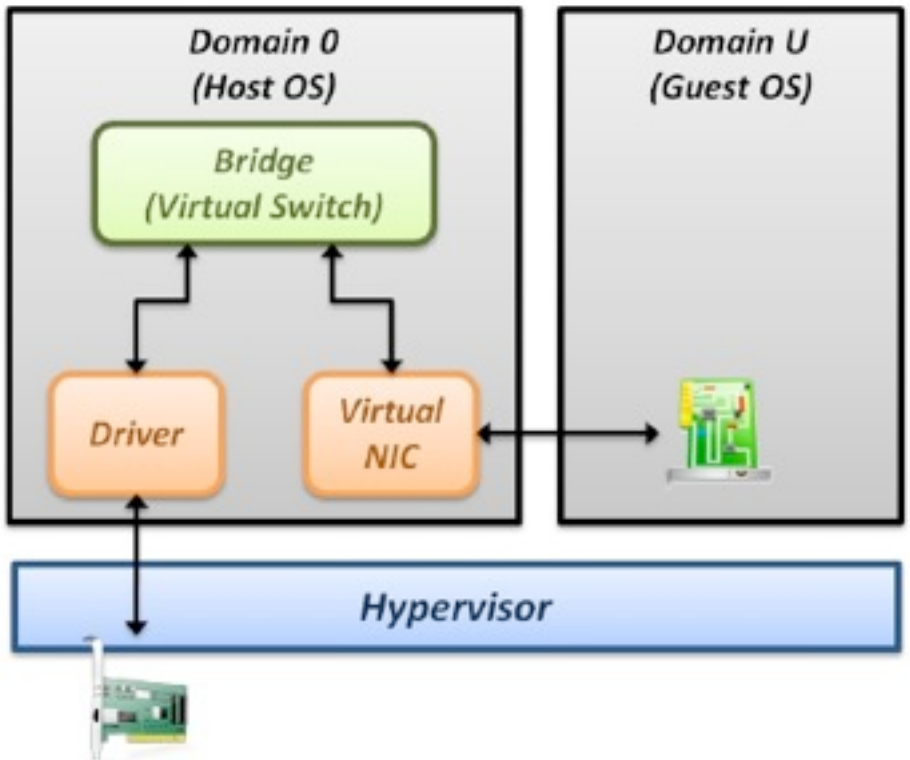
New Techniques

Detail in Xen System



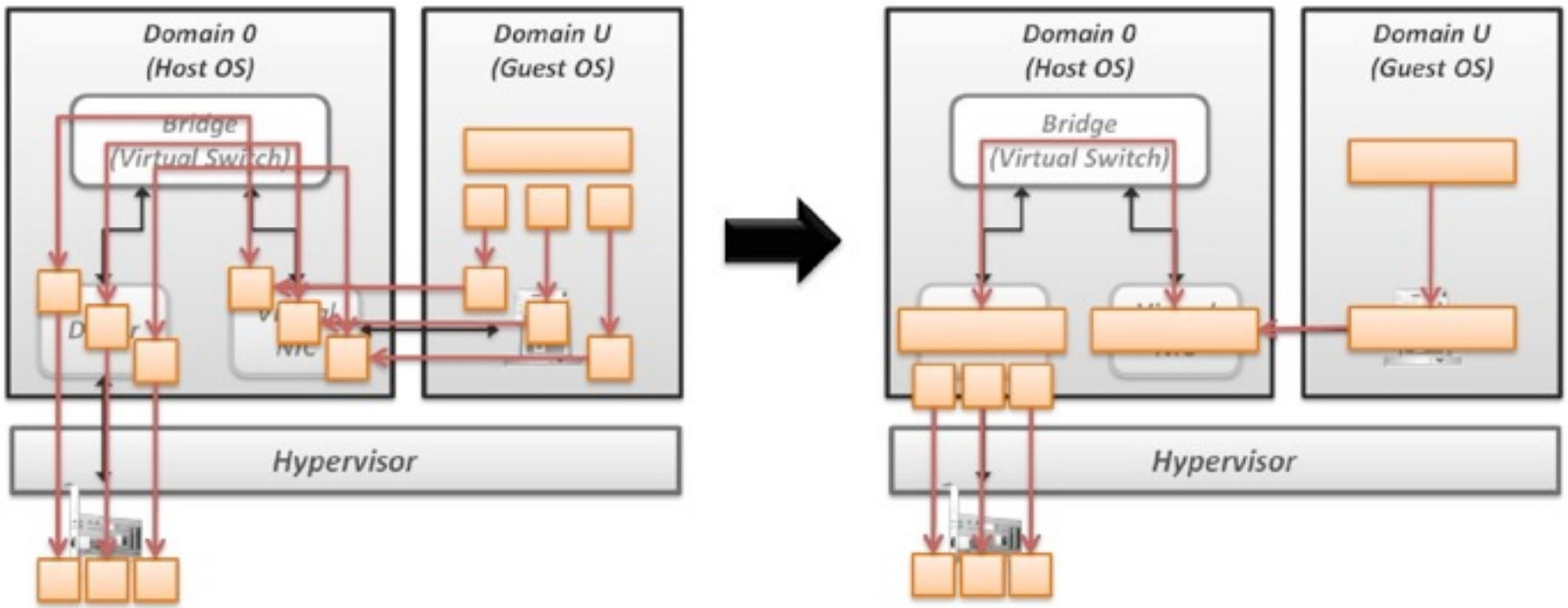
New Techniques

- Some performance issues :
  - Page remapping
    - Hypervisor remap memory page for MMIO.
  - Contexts witching
    - Whenever packets send, induce one context switch from guest to Domain 0 to drive real NIC.
  - Software bridge management
    - Linux bridge is a pure software implementation.
  - Interrupt handling
    - When interrupt occur, induce one context switch again.



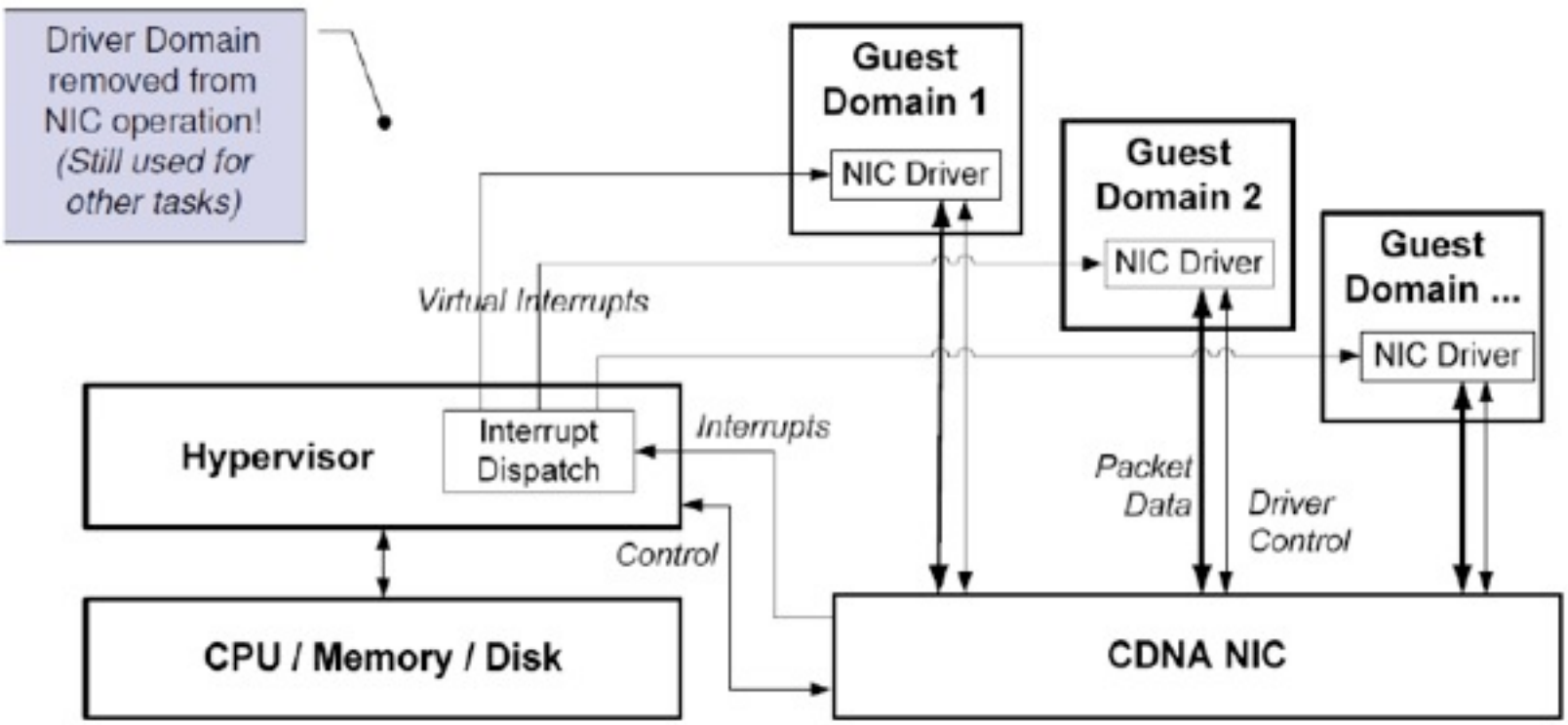
New Techniques

- Improve Xen performance by software
  - Large effective MTU
  - Fewer packets
  - Lower per-byte cost

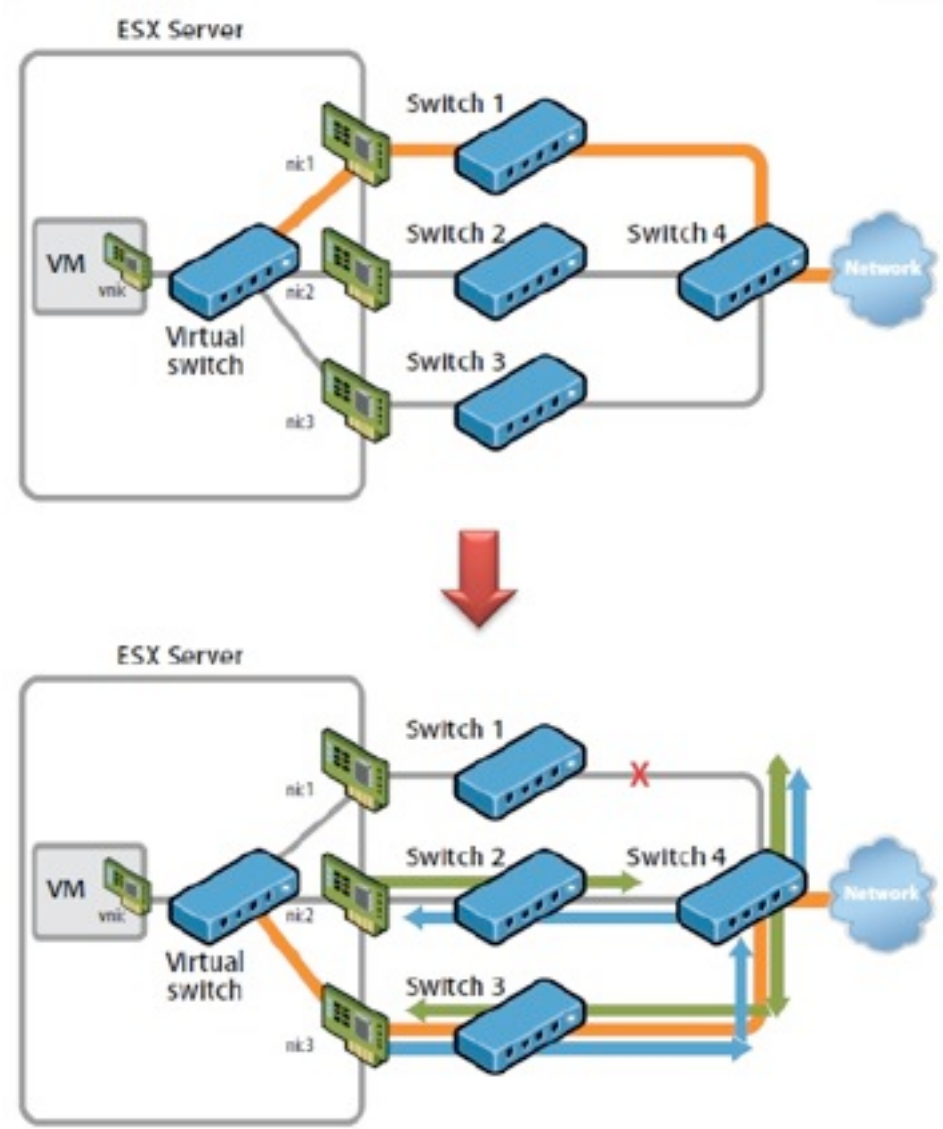


New Techniques

- Improve Xen performance by hardware
  - CDNA (ConcurrentDirect Network Access) hardware adapter
  - Remove driver domain from data and interrupts
  - Hypervisor only responsible for virtual interrupts and assigning context to guest OS



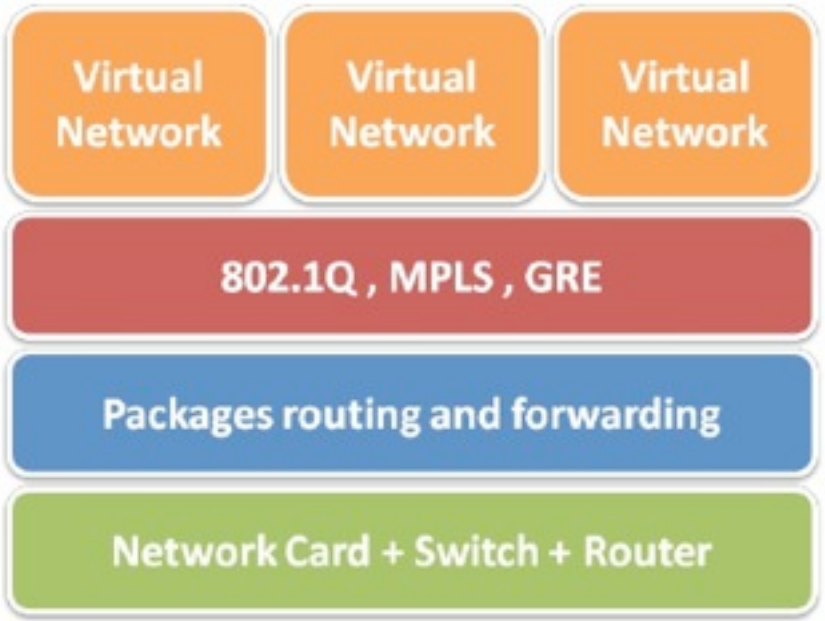
Case Study



- VMware offer a hybrid solution of network virtualization in Cloud.
  - Use redundant links to provide high availability.
  - Virtual switch in host OS will automatically detect link failure and redirect packets to back-up links.

Network Virtualization Summary

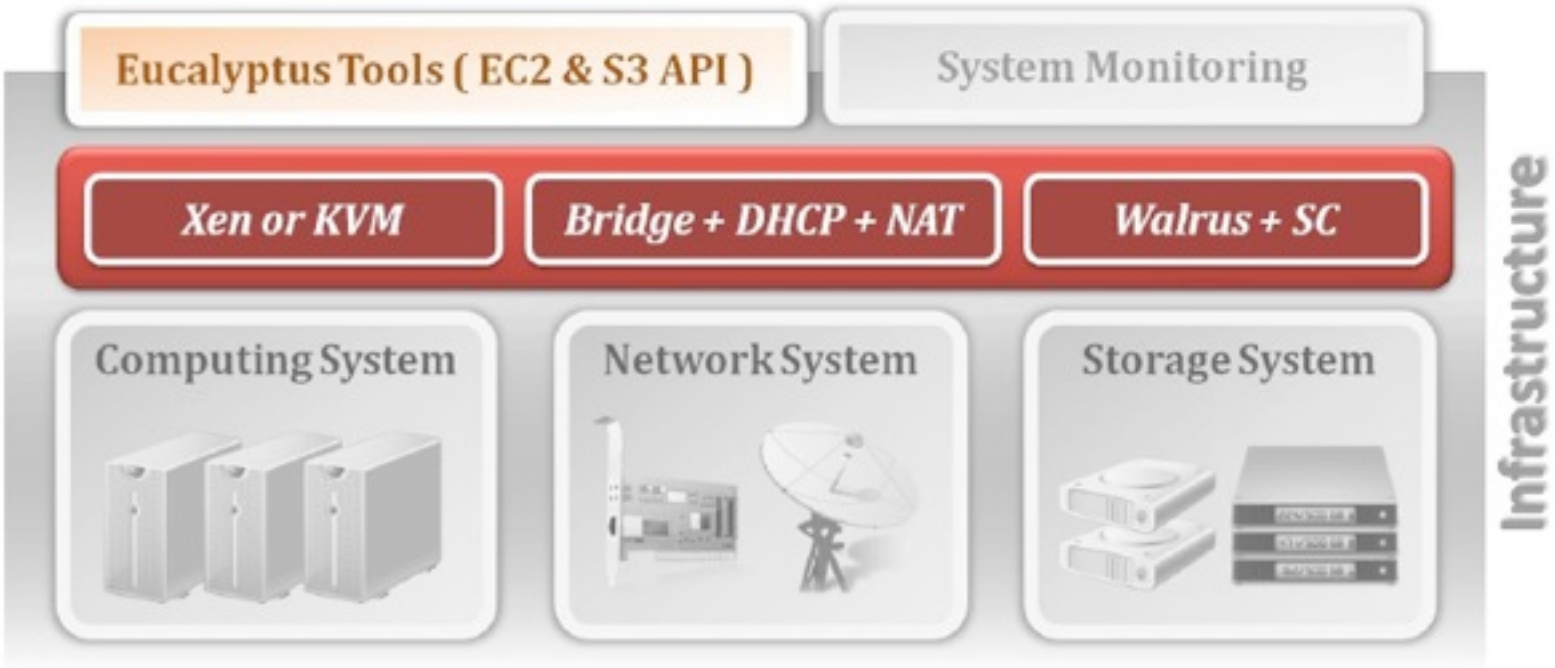
- Virtualization in layers
  - Usually in Layer 2 and Layer 3
- External network virtualization
  - Layer2
    - 802.1q
  - Layer3
    - MPLS, GRE
- Internal network virtualization
  - Layer 2
    - TAP/TUN + Linux bridge
  - Layer3
    - Virtual switch, CDNA



IaaS Case Study

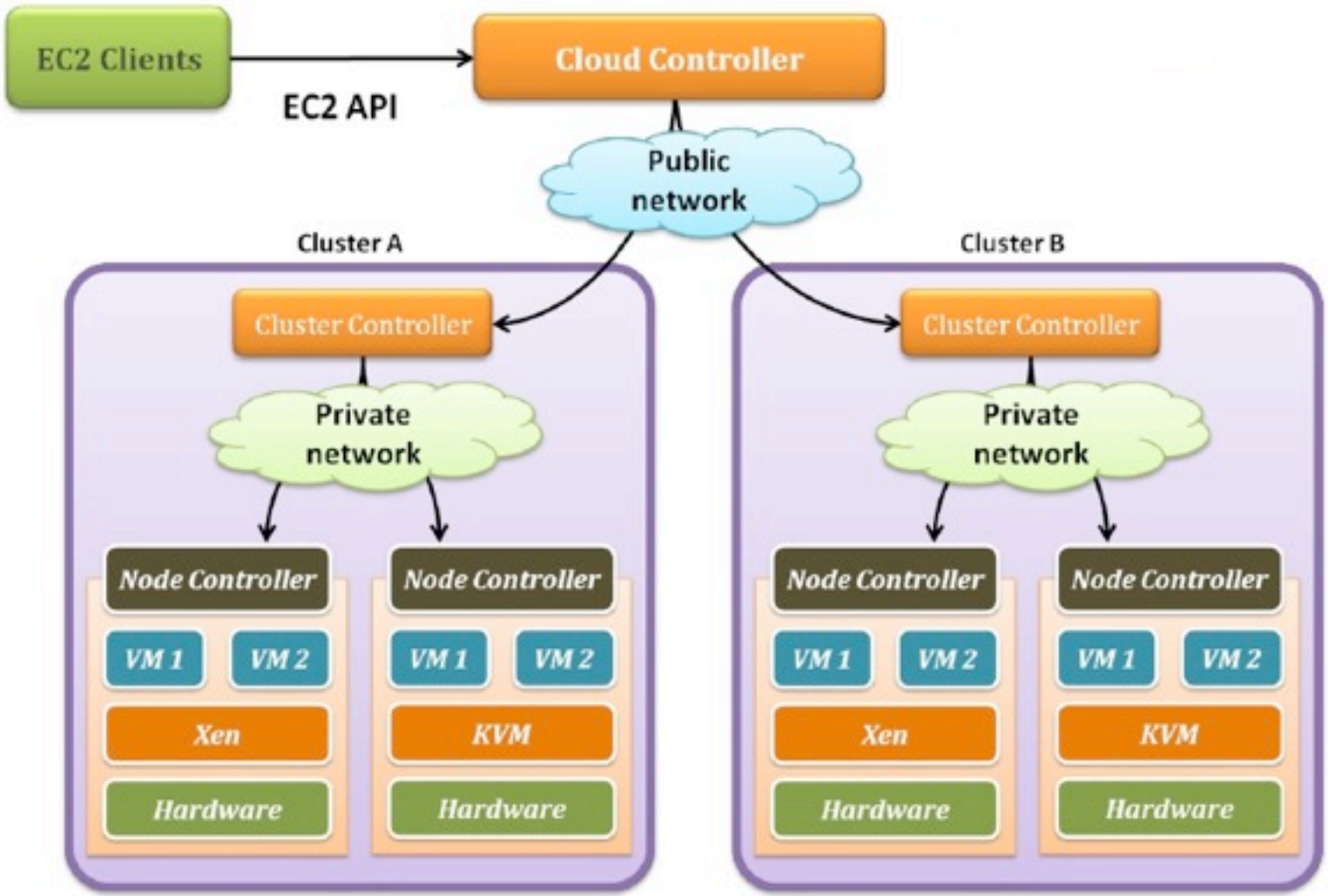
- IaaS open source project – Eucalyptus
  - Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems

IaaS Architecture of Eucalyptus



IaaS Case Study

Server Virtualization

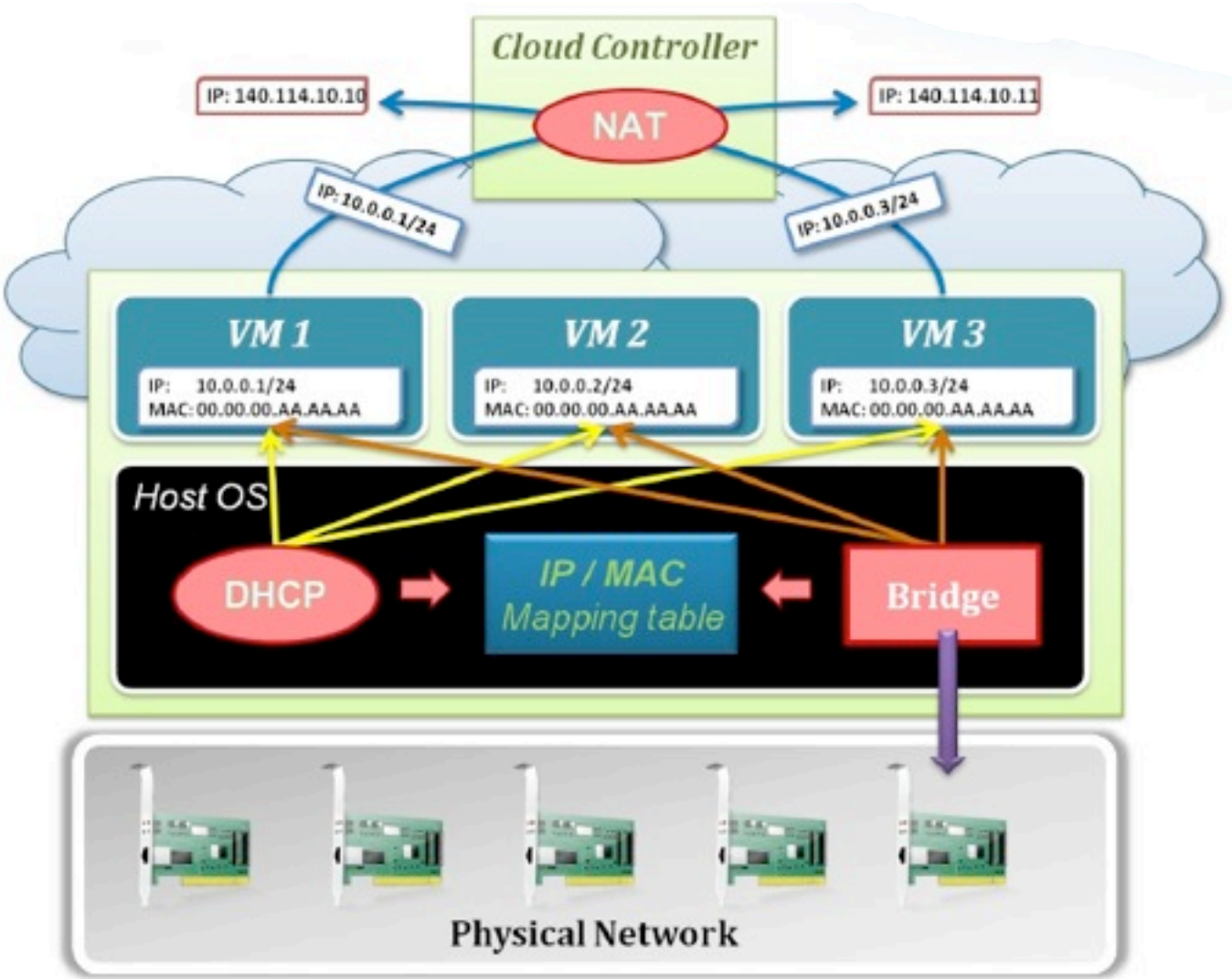


IaaS Case Study

- System Component :
  - Cloud Controller(CLC)
    - Dispatch user request to some clusters.
  - ClusterController(CC)
    - Determine enough resource for virtual machine deployment.
  - Node Controller (NC)
    - Run user's virtual machines.

IaaS Case Study

Network Virtualization



NAT-PT

- Network Address Translation Port Translation or NAT-PT is a technique that allows the translation of local network addresses or ***the internal IP addresses*** (used within an organization) into globally unique IP addresses that help identify an online resource in a unique manner over the Internet.
  - the local network address is in the range of private addresses as defined by RFC 1918 and RFC 4193.

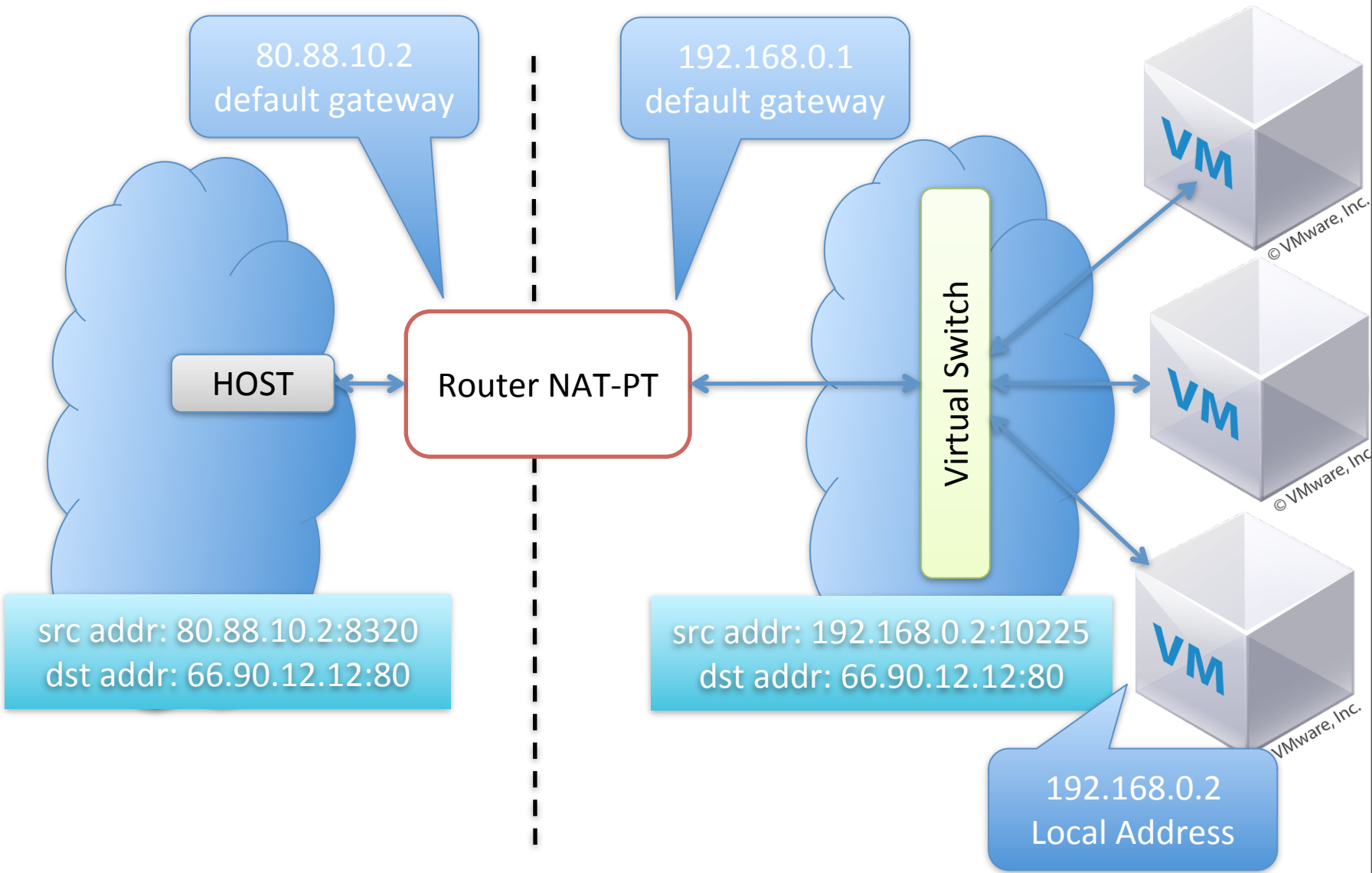
10.0.0.0 – 10.255.255.255	16,777,216
172.16.0.0 – 172.31.255.255	1,048,576 16
192.168.0.0 – 192.168.255.255	65,536 256

  - These addresses are characterized as private because they are not globally delegated, meaning they are not allocated to any specific organization, and IP packets addressed by them cannot be transmitted onto the public Internet. Anyone may use these addresses without approval from a regional Internet registry (RIR).
- The process is also referred to as Network Masquerading or the Native Address Translation. Network Address Translation allows multiple resources within an organization or connected to a local LAN to use a ***single public IP address*** to access the Internet.

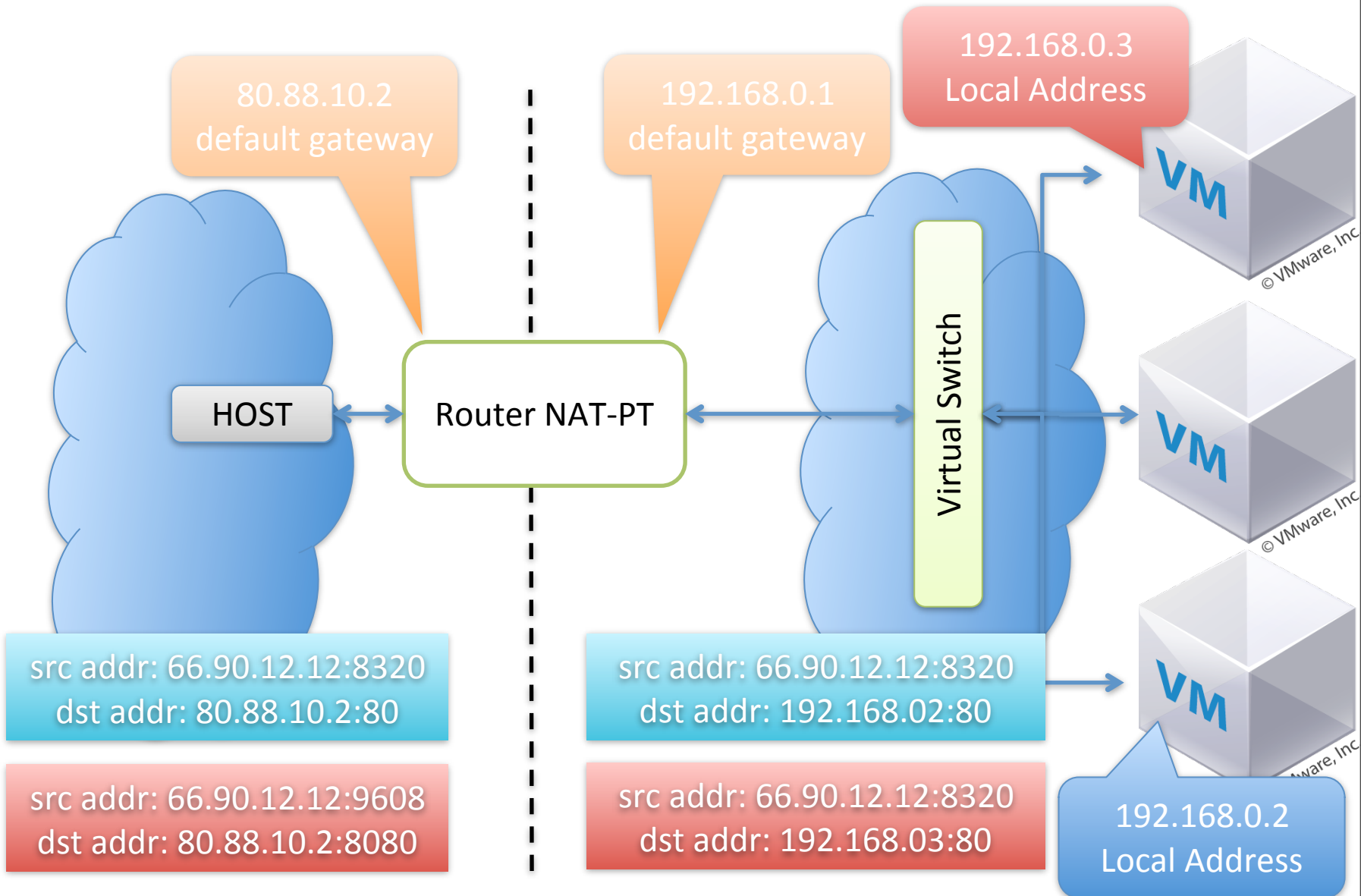
NAT usage

- The idea of Network Address Translation is very simple.
  - It essentially abstracts internal addressing from the global IP addressing used over the Internet. This abstraction allows helps the network resources to get over a shortage of the address space by mapping relatively few real IP addresses to the abundant local IP addresses created locally by the Proxy server for addressing purposes.
  - It allows the use of different addresses over the local and global level and local sharing of IP addresses over the Internet.

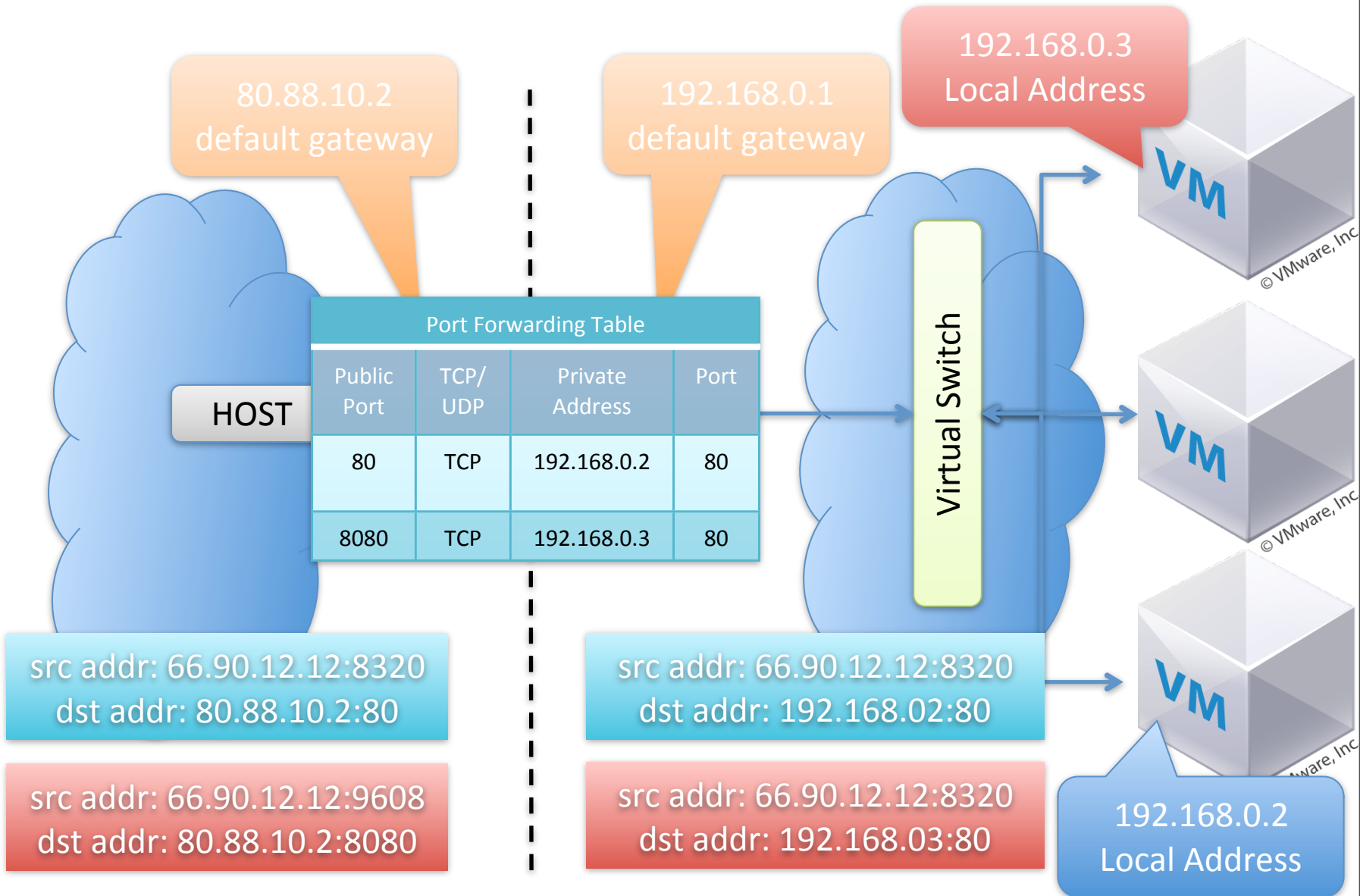
Example 1: when the web service is on the public network



Example 2: when the web services are behind the NAT



Example 2: when the web services are behind the NAT



Drawbacks

1. Network Address Translation **does not allow a true end-to-end connectivity** that is required by some **real time applications**. A number of real-time applications require the creation of a logical tunnel to exchange the data packets quickly in real-time. It requires a fast and seamless connectivity devoid of any intermediaries such as a proxy server that **tends to complicate and slow down** the communications process.

2. NAT creates complications in the functioning of Tunneling protocols. Any communication that is routed through a Proxy server tends to be comparatively slow and prone to disruptions. Certain critical applications offer no room for such inadequacies. Examples include telemedicine and teleconferencing. Such applications find the process of **network address translation as a bottleneck in the communication network** creating avoidable distortions in the end-to-end connectivity.

3. **NAT acts as a redundant channel in the online communication over the Internet**. The twin reasons for the widespread popularity and subsequent adoption of the network address translation process were a shortage of IPv4 address space and the security concerns. Both these issues have been fully addressed in the IPv6 protocol. As the IPv6 slowly replaces the IPv4 protocol, the network address translation process will become redundant and useless while consuming the scarce network resources for providing services that will be no longer required over the IPv6 networks.

IaaS Case Study

- Network architecture :
  - Bridge (Virtual Switch)
    - Make virtual machines on one node share physical NICs.
  - DHCP
    - Map virtual MAC addresses of VMs to private IPs in the LAN.
  - NAT
    - Forward the packages to public network (WAN).
  - IP/MAC mapping table
    - IP addresses are assigned by Eucalyptus.
    - MAC addresses are assigned by hypervisor.
    - This mapping table is maintained by Eucalyptus system.

- Introduction
- External network virtualization
- Internal network virtualization
- Best Practices with VMware **(Guy Brunsdon,** VMware, Inc.)

NETWORK VIRTUALIZATION

Agenda

Virtual Networking Concepts and Best Practices

- Why Virtual Networking?
- Anatomy of Virtual Networking
- Virtual Switch Options and alternatives
- ESX Virtual Switch Capabilities
- Spanning Tree Protocol
- NIC Teaming
- Port Group Configuration
- Traffic types
- VLAN Trunking
- FCoE and 10GigE
- VMotion: network operation under the covers
- Migrating to vDS and/or Nexus 1000V

Networking Design Examples

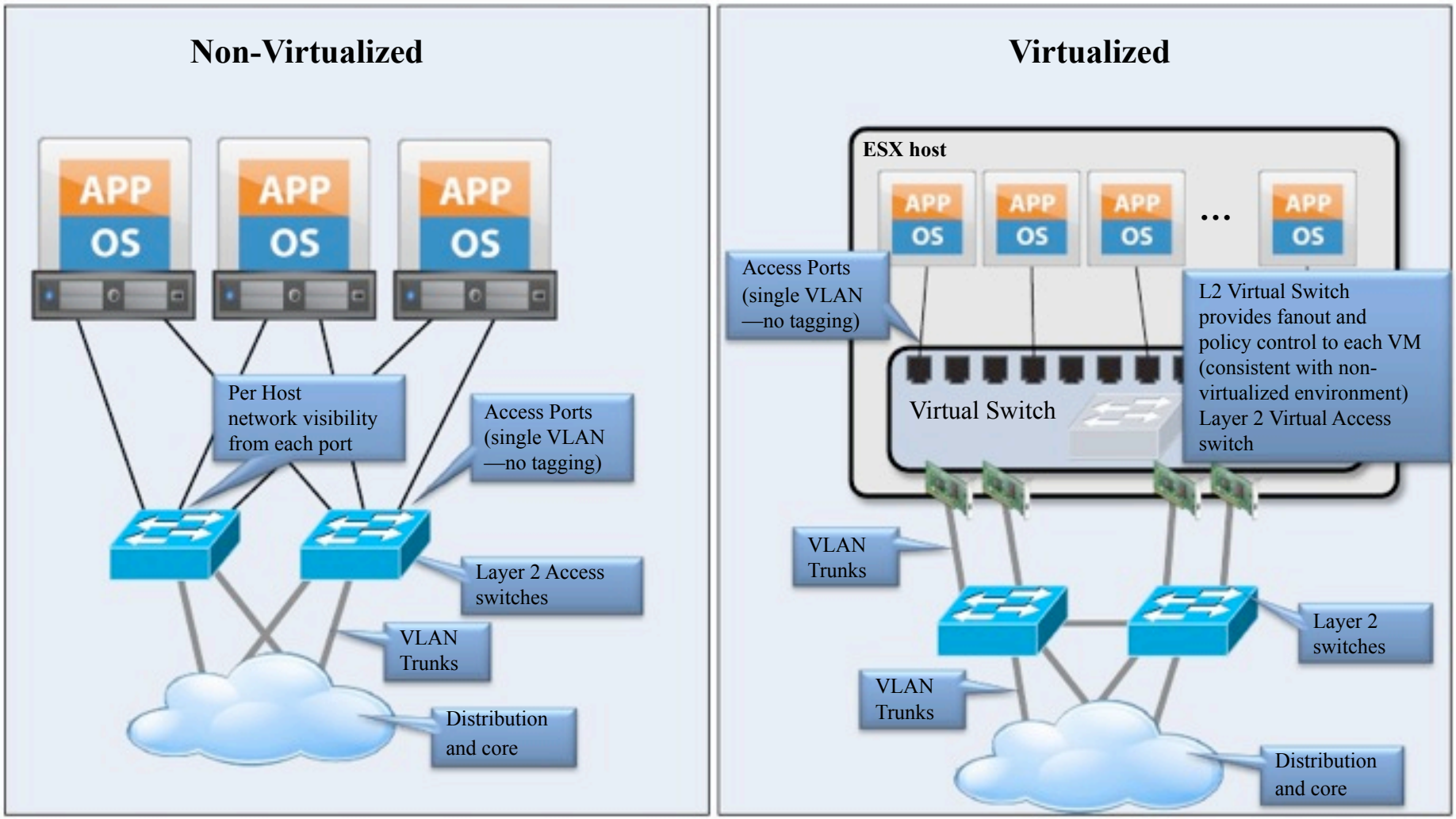
- Example 1: Blade Server with 2 NIC ports
- Example 2: Server with 4 NIC ports
- Example 3: Server with 4 NIC ports (variation)
- Servers with >4 NIC ports

Additional Considerations

- IP Storage considerations
- IPv6
- vNetwork Appliance API
- Further Reading

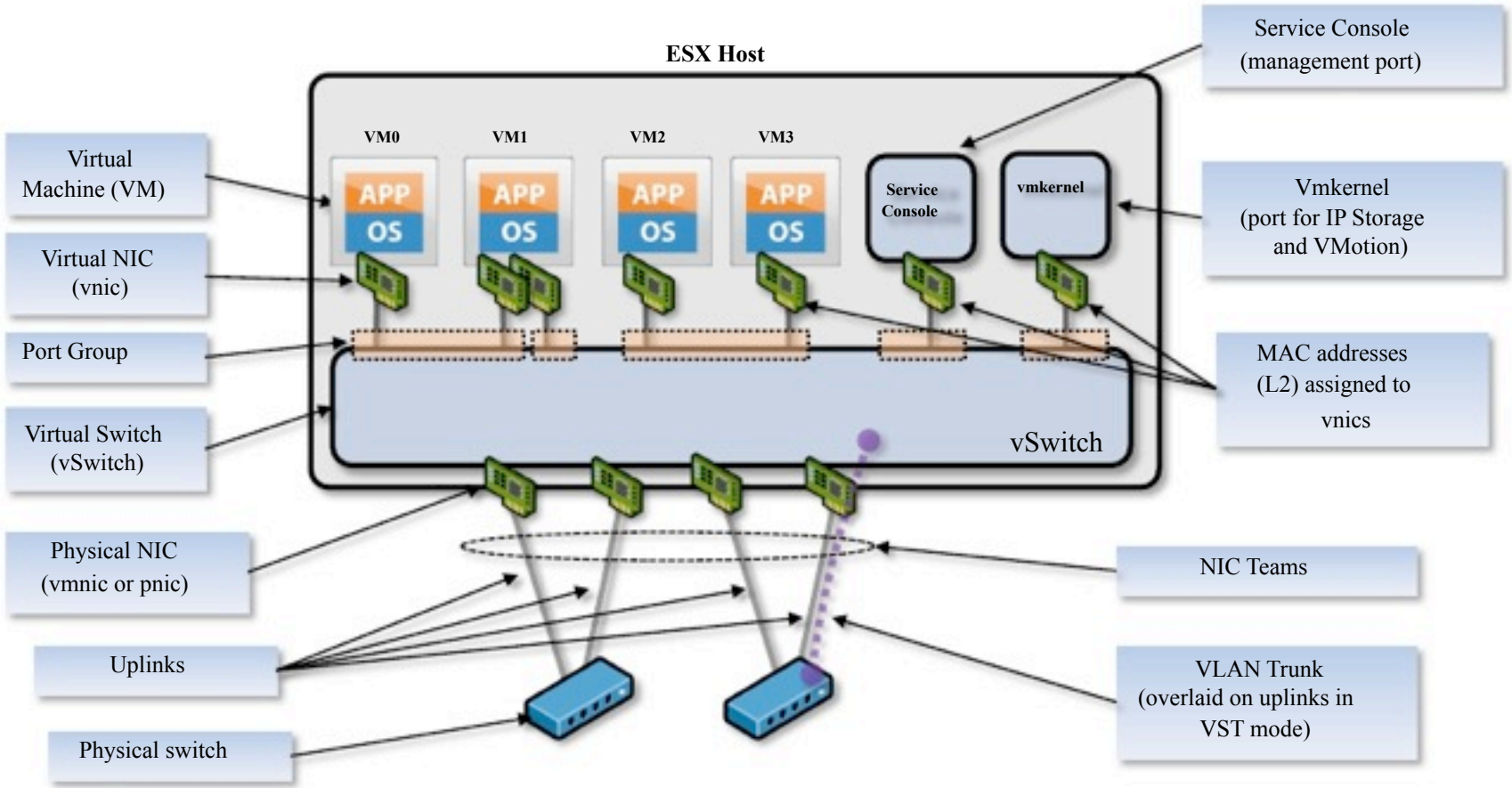
vmware

Why Do We Need a Virtual Switch?

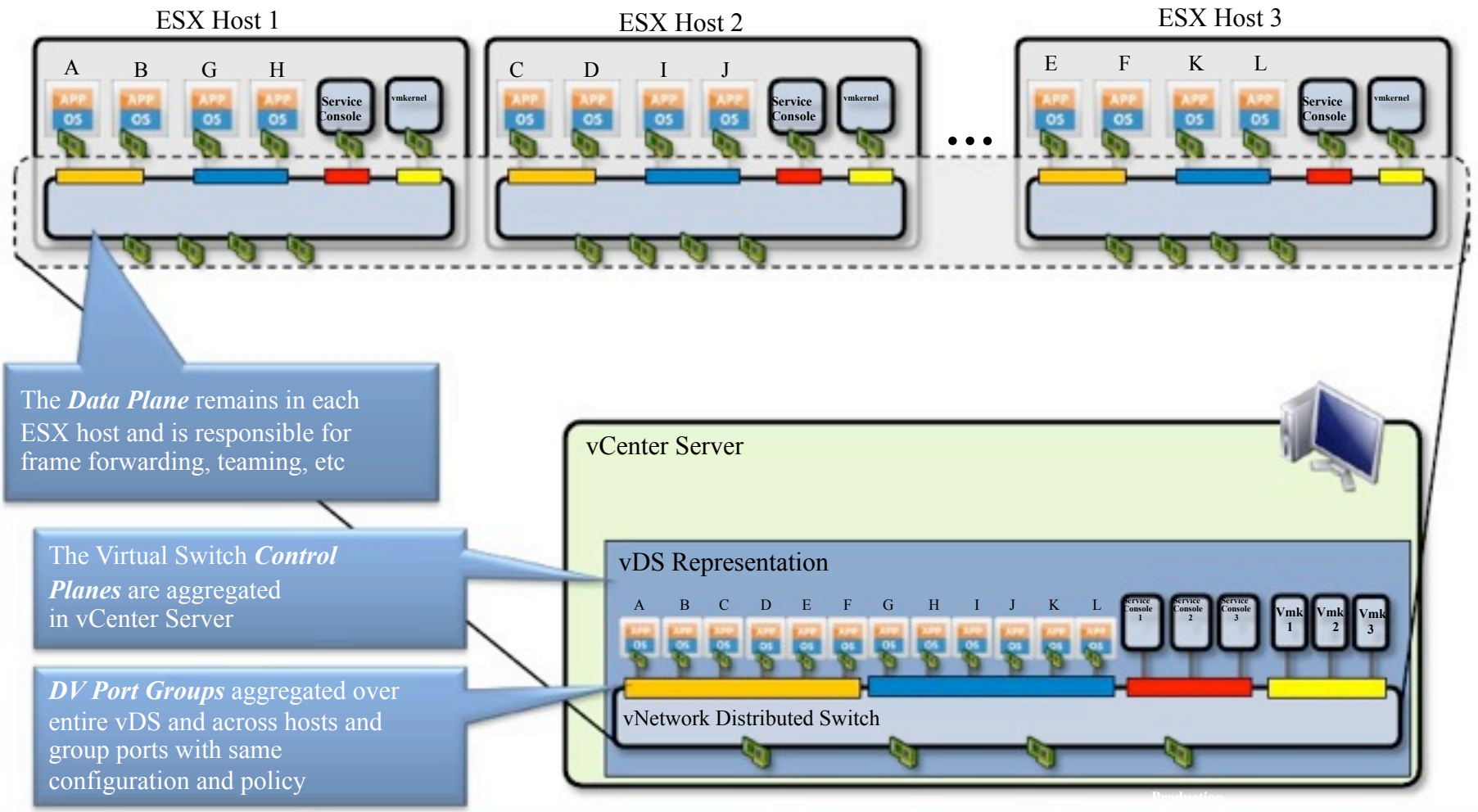


vmware

Anatomy of Virtual Networking



vNetwork Distributed Switch (vDS)



Virtual Switch Options with vSphere 4

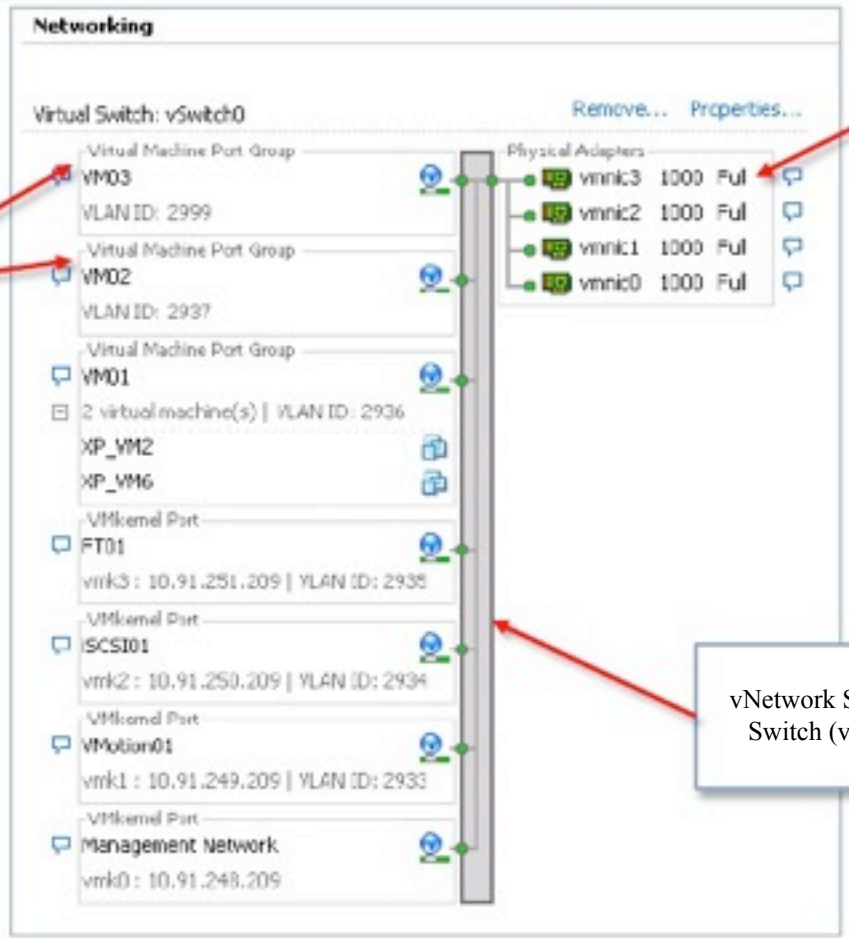
Virtual Switch	Model	Details
vNetwork Standard Switch	Host based: 1 or more per ESX host	- Same as vSwitch in VI3
vNetwork Distributed Switch	Distributed: 1 or more per —Datacenter‖	- Expanded feature set - Private VLANs - Bi-directional traffic shaping - Network Vmotion - Simplified management
Cisco Nexus 1000V	Distributed: 1 or more per —Datacenter‖	- Cisco Catalyst/Nexus feature set - Cisco IOS cli

Virtual networking concepts similar with all virtual switches

vmware

vNetwork Standard Switch: A Closer Look

vSS defined on a per host basis from ***Home → Inventory → Hosts and Clusters***.



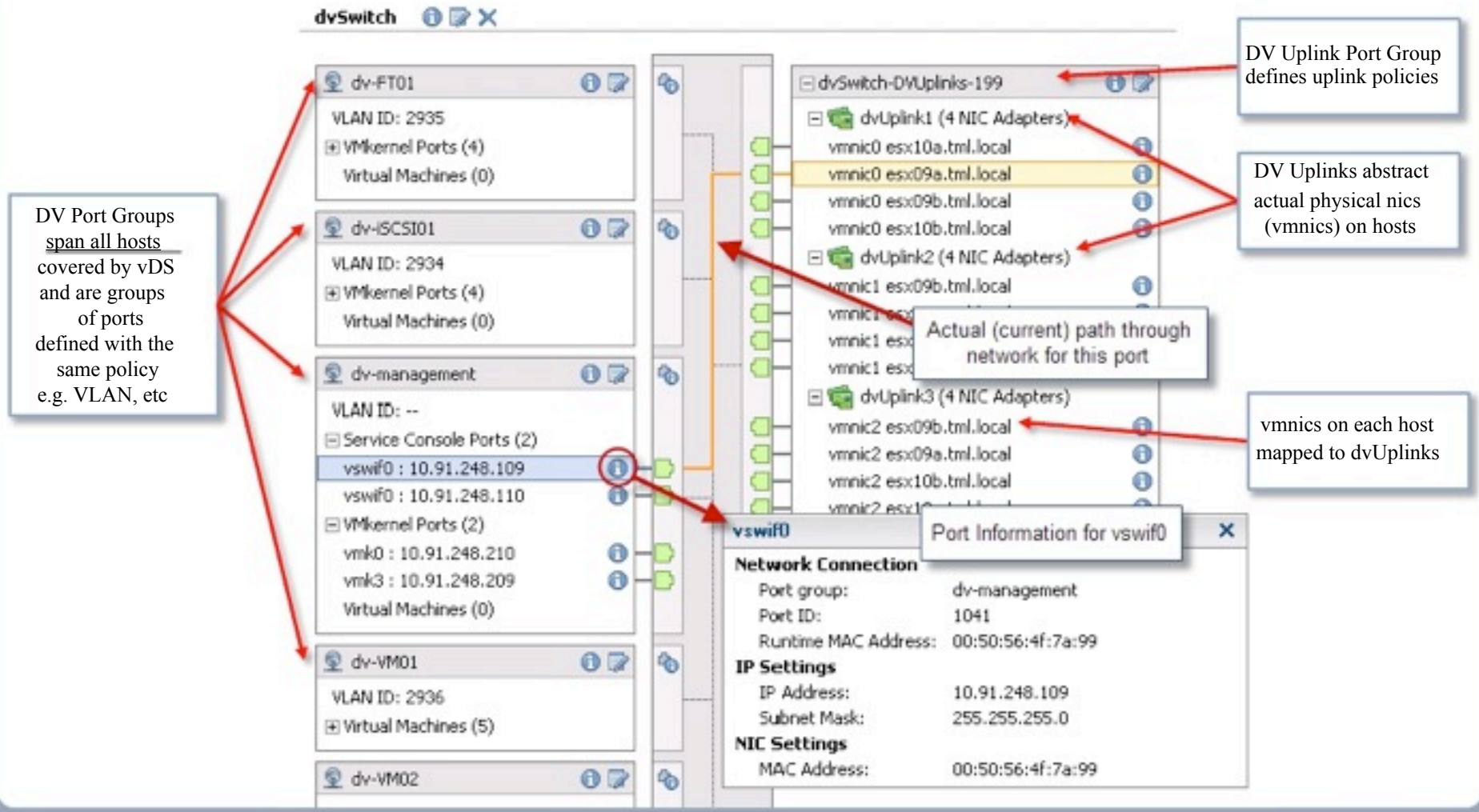
Port Groups are policy definitions for a set or group of ports. e.g. VLAN membership, port security policy, teaming policy, etc

Uplinks (physical NICs) attached to vSwitch.

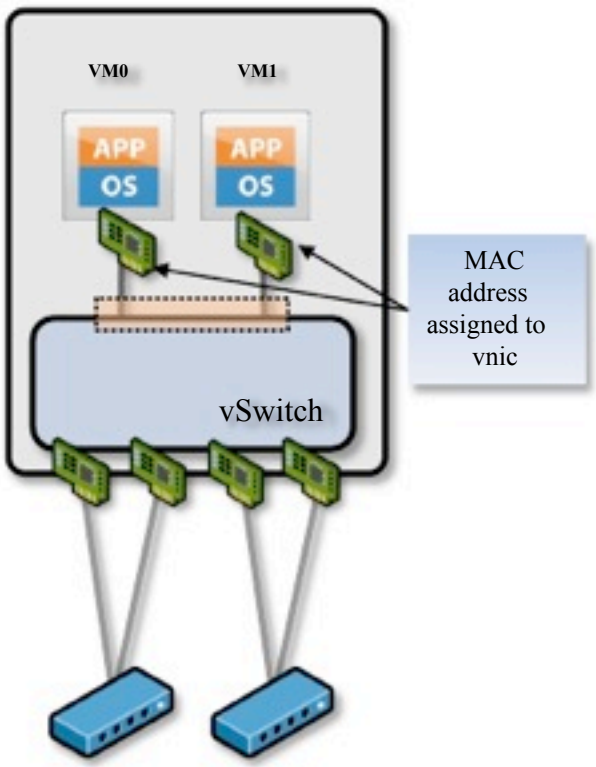
vNetwork Standard Switch (vSwitch)

vmware

vNetwork Distributed Switch: A Closer Look



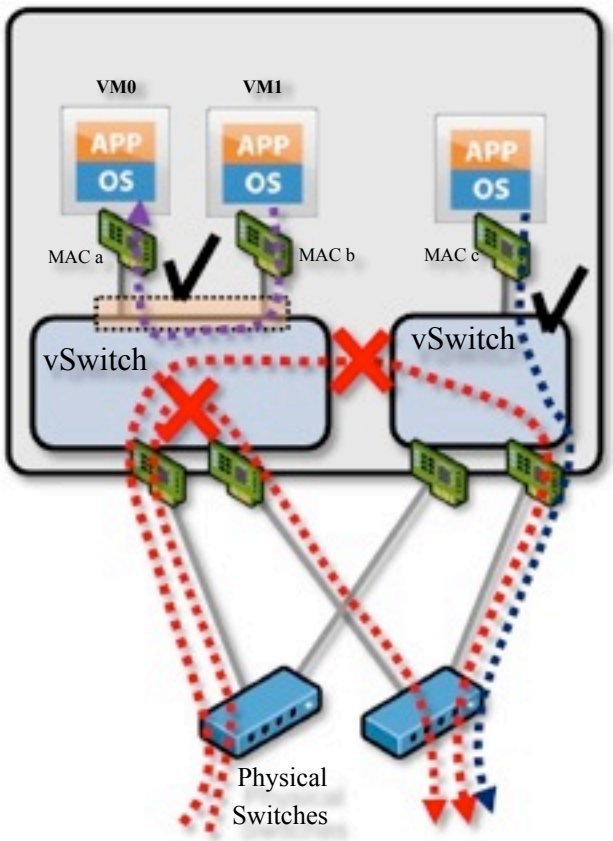
ESX Virtual Switch: Capabilities



- Layer 2 switch—forwards frames based on 48-bit destination MAC address in frame
- MAC address known by registration (it knows its VMs!)—no MAC learning required
- Can terminate VLAN trunks (VST mode) or pass trunk through to VM (VGT mode)
- Physical NICs associated with vSwitches
- NIC teaming (of uplinks)
  - Availability: uplink to multiple physical switches
  - Load sharing: spread load over uplinks

vmware

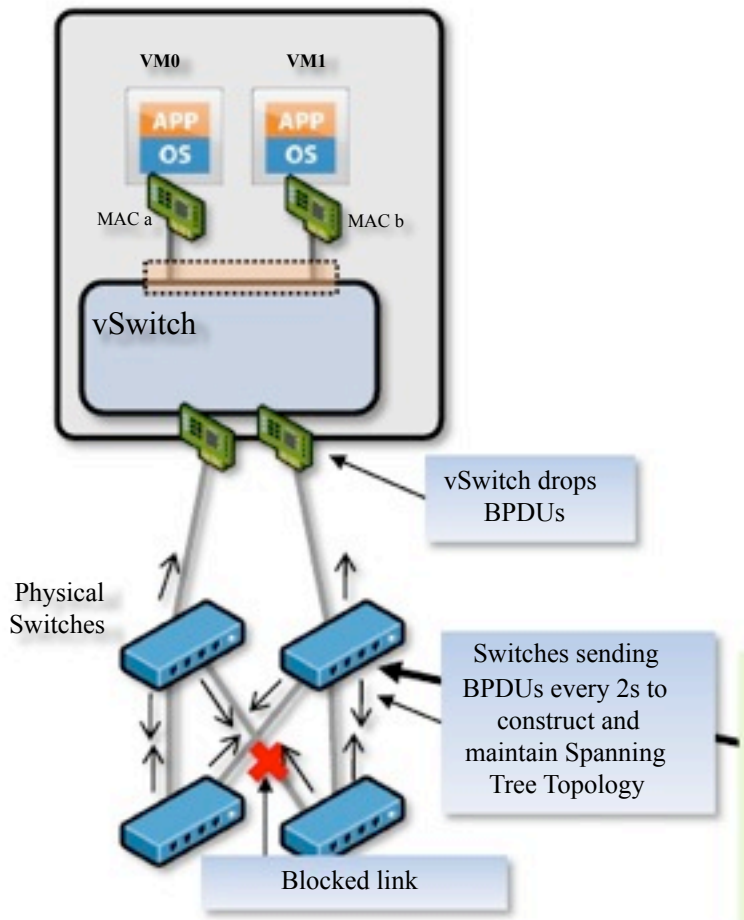
ESX Virtual Switch: Forwarding Rules



- The vSwitch will forward frames
  - VM ←→ VM
  - VM ←→ Uplink
- But not forward
  - vSwitch to vSwitch
  - Uplink to Uplink
- ESX vSwitch will not create loops in the physical network
- And will not affect Spanning Tree (STP) in the physical network

vmware

Spanning Tree Protocol (STP) Considerations



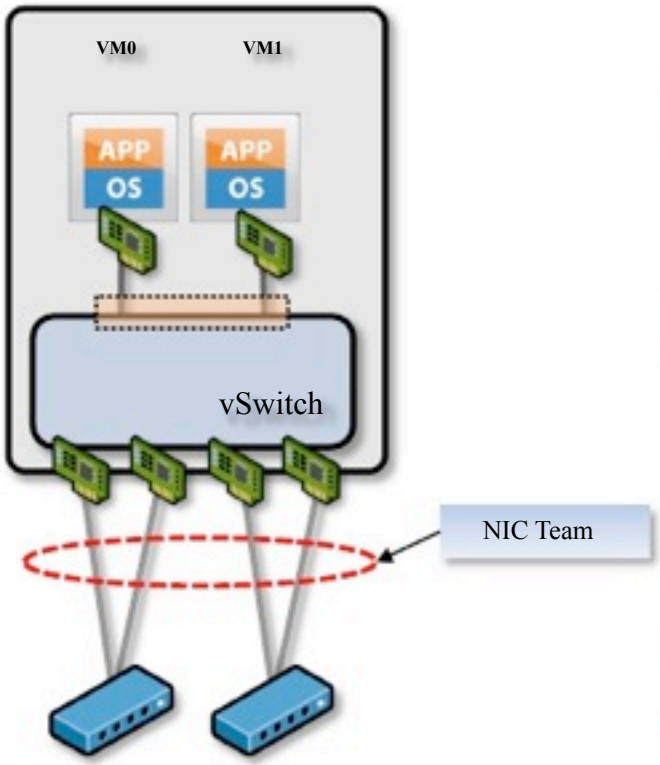
- Spanning Tree Protocol used to create loop-free L2 tree topologies in the physical network
  - Some physical links put in —blocking‖ state to construct loop-free tree
- ESX vSwitch does not participate in Spanning Tree and will not create loops with uplinks
  - →ESX Uplinks will not block and always active (full use of all links)

Recommendations for Physical Network Config:

1. Leave Spanning Tree enabled on physical network and ESX facing ports (i.e. leave it as is!)
2. Use —`portfast`‖ or —`portfast trunk`‖ on ESX facing ports (puts ports in forwarding state immediately)
3. Use —`bpduguard`‖ to enforce STP boundary

vmware

NIC Teaming for Availability and Load Sharing



NIC Teaming aggregates multiple physical uplinks for:

- **Availability**—reduce exposure to single points of failure (NIC, uplink, physical switch)
- **Load Sharing**—distribute load over multiple uplinks (according to selected NIC teaming algorithm)

Requirements:

- Two or more NICs on same vSwitch
- Teamed NICs on same L2 broadcast domain

vmware

NIC Teaming Options

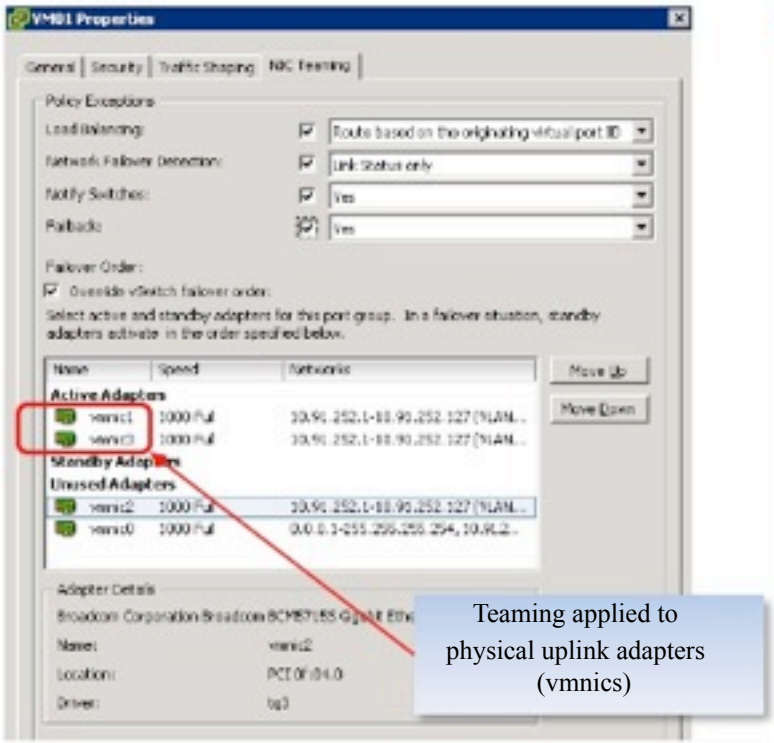
Name	Algorithm—vmnic chosen based upon:	Physical Network Considerations
Originating Virtual Port ID	vnic port	Teamed ports in same L2 domain *(BP: team over two physical switches)*
Source MAC Address	MAC seen on vnic	Teamed ports in same L2 domain *(BP: team over two physical switches)*
IP Hash	Hash(SrcIP, DstIP)	Teamed ports configured in static 802.3ad —Etherchannel‖ - no LACP - Needs MEC to span 2 switches
Explicit Failover Order	Highest order uplink from active list	Teamed ports in same L2 domain *(BP: team over two physical switches)*

Best Practice: Use Originating Virtual PortID for VMs

vmware

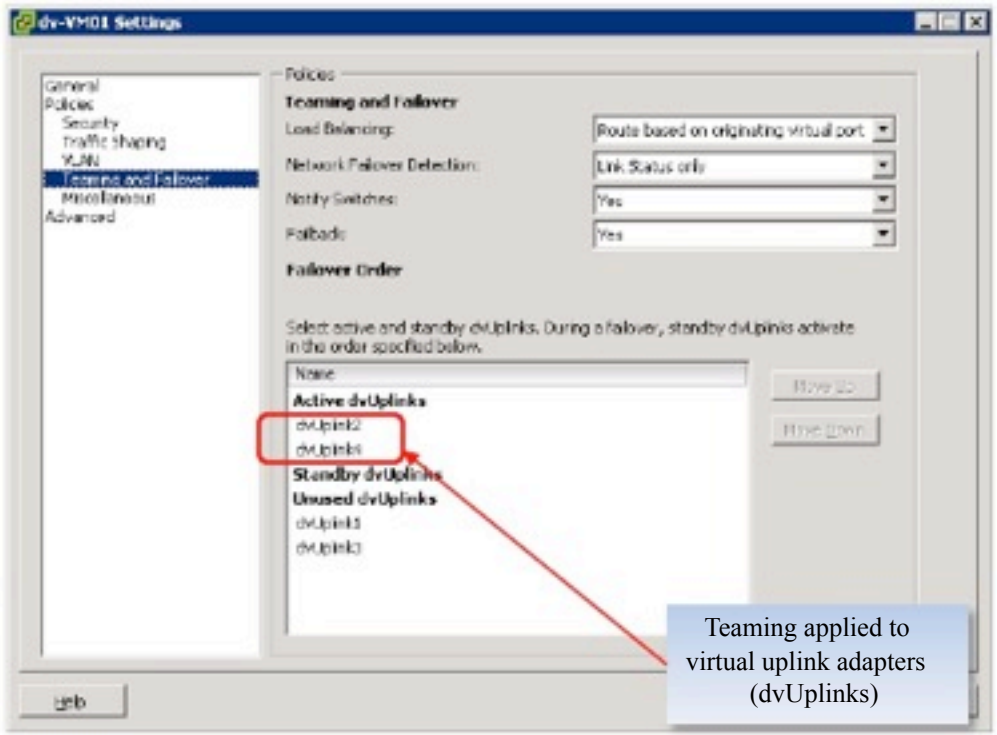
NIC Teaming with vSS and vDS

vNetwork Standard Switch



- Apply NIC Teaming Policy to vSwitch and optionally override on each
- Port Group definition
- Applied to vmnics

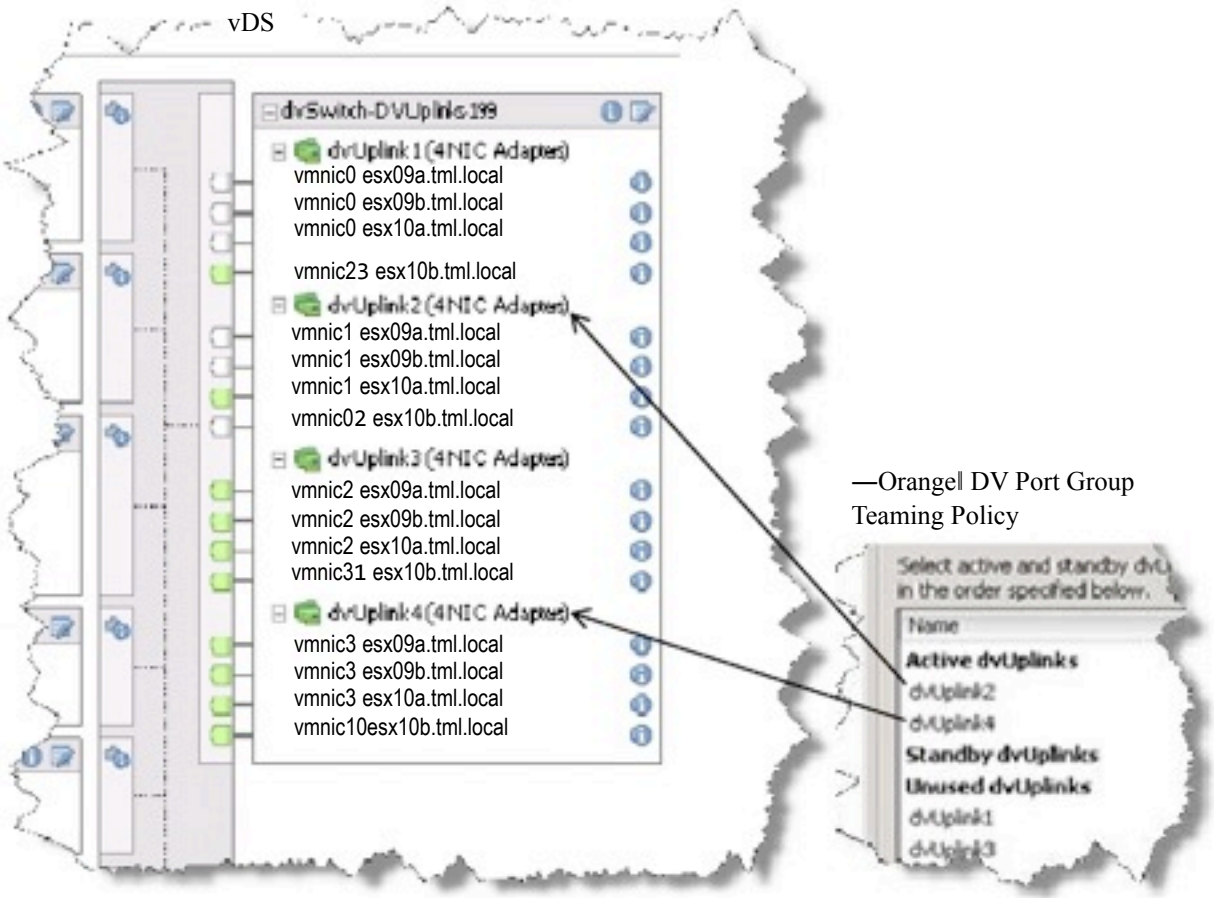
vNetwork Distributed Switch



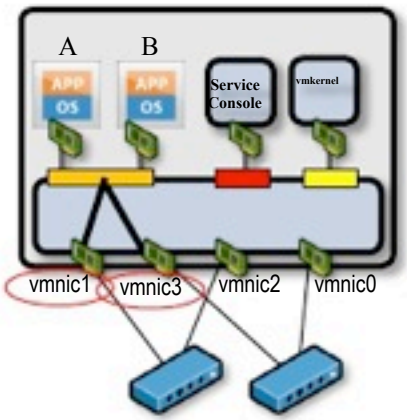
- Apply NIC Teaming Policy on DV Port Groups only
- Applied to dvUplinks (vmnics mapped per host to dvUplinks)

NIC Teaming with vDS

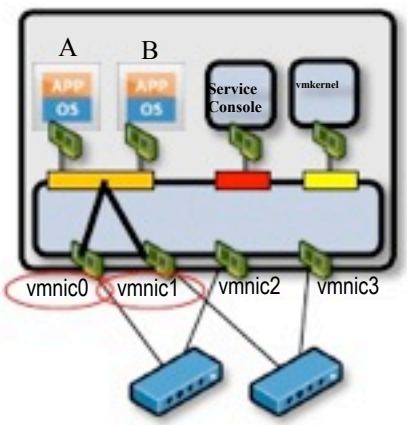
Teaming Policies Are Applied in DV Port Groups to dvUplinks



esx10a.tml.local
esx09a.tml.local
esx09b.tml.local

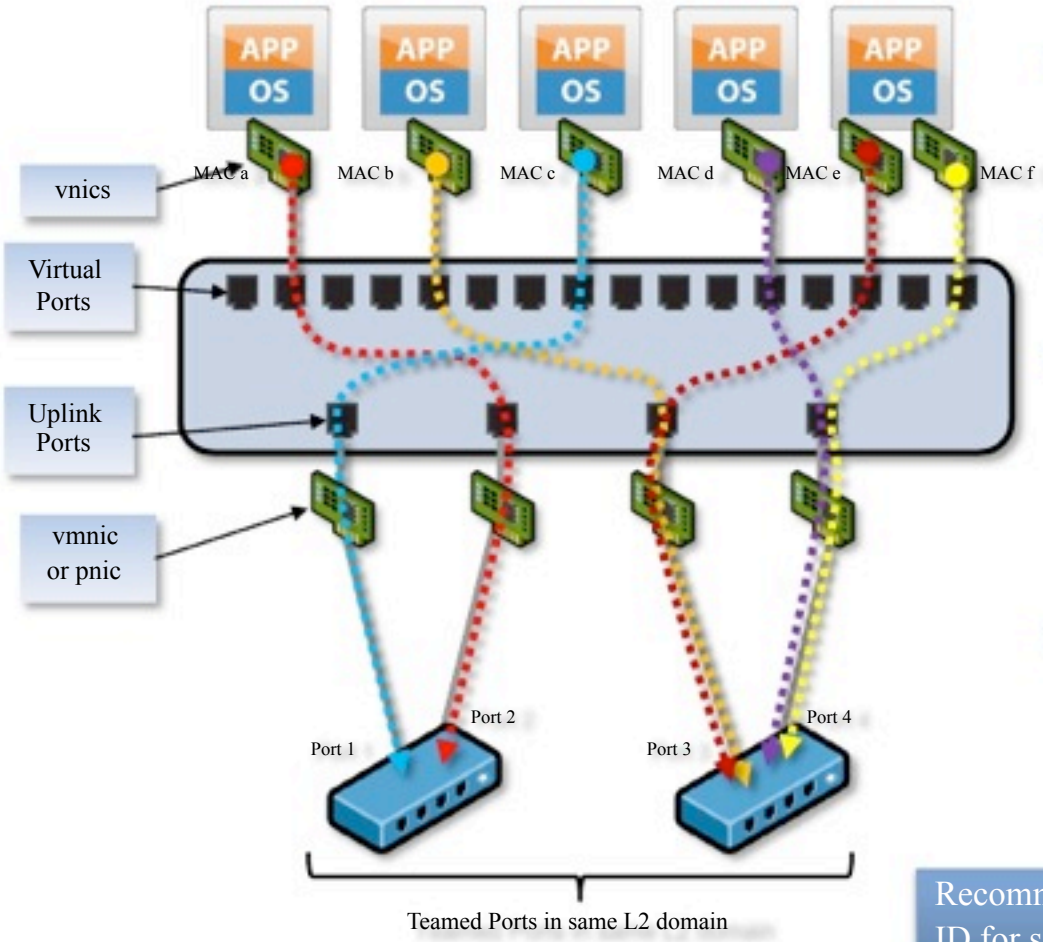


esx10b.tml.local



vmware

NIC Teaming: Originating Virtual Port ID

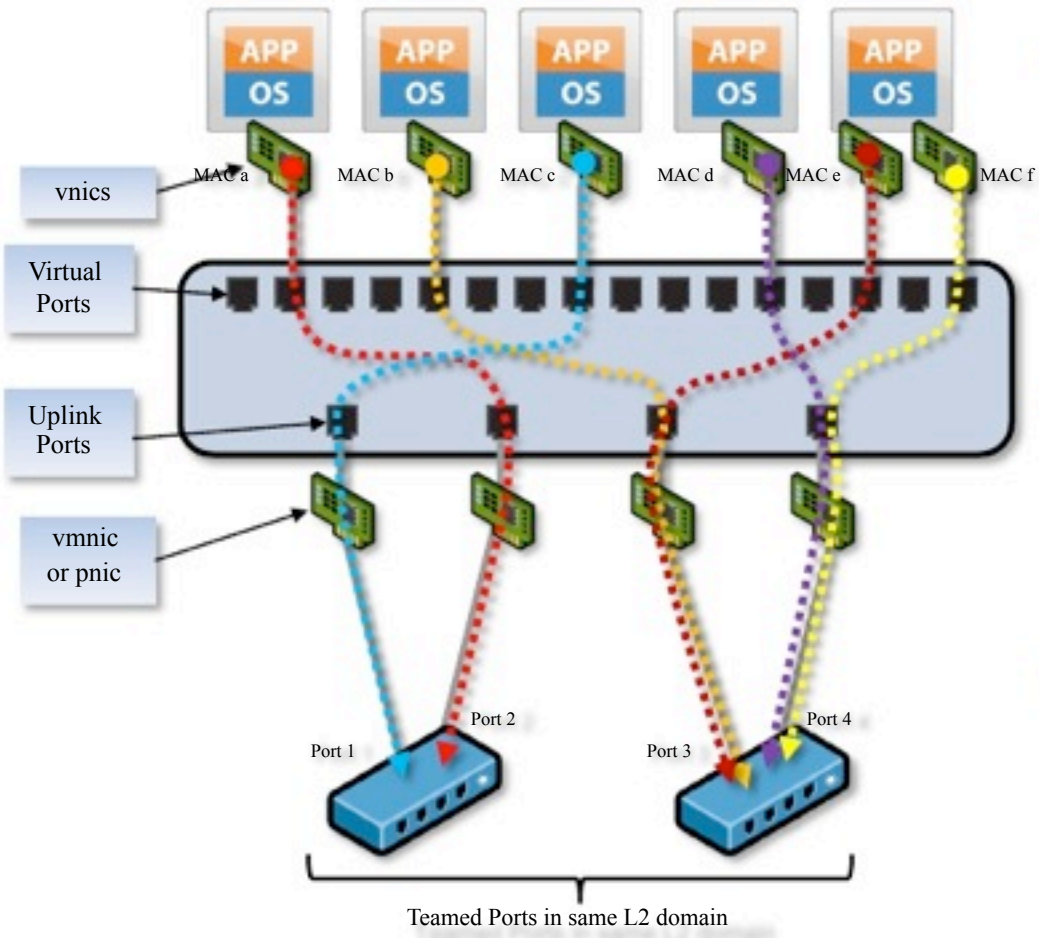


- Outgoing uplink chosen from hash of ―Originating Virtual PortID‖
- All traffic from vnic will hit same vmnic until failover event
- Return traffic will follow same path
  - Physical switch ―learns‖ originating mac address (and populates its CAM table)
- Physical switches unaware of teaming
  - Same L2 domain
  - No special configuration

Recommendation: Use Originating Virtual Port ID for simplicity and multi-switch availability

vmware

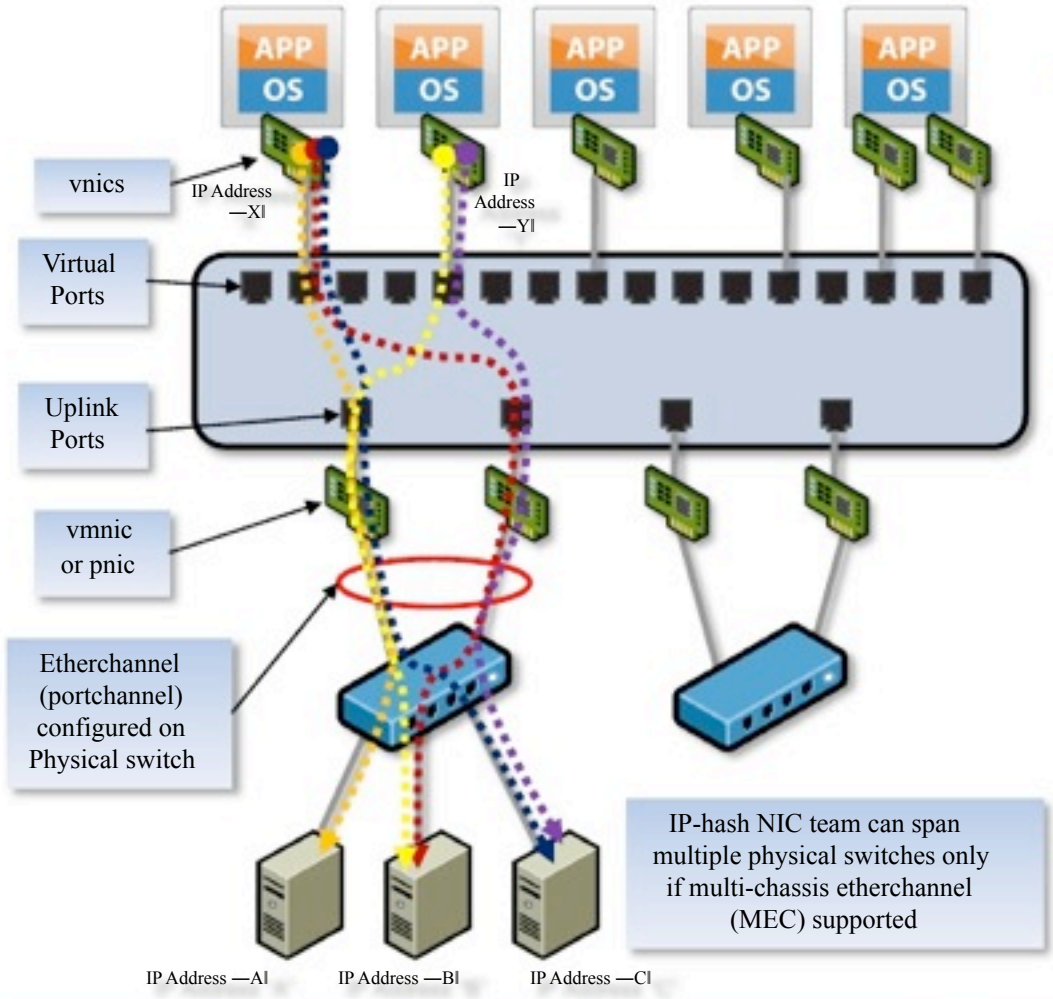
NIC Teaming: MAC Based Teaming



- Outgoing uplink chosen from hash of —Source MAC address‖ from vnic
- All traffic from mac address will hit same vmnic until failover event
- Return traffic will follow same path
  - Physical switch —learns‖ originating mac address (and populates its CAM table)
- Physical switches unaware of teaming
  - Same L2 domain
  - No special configuration

vmware

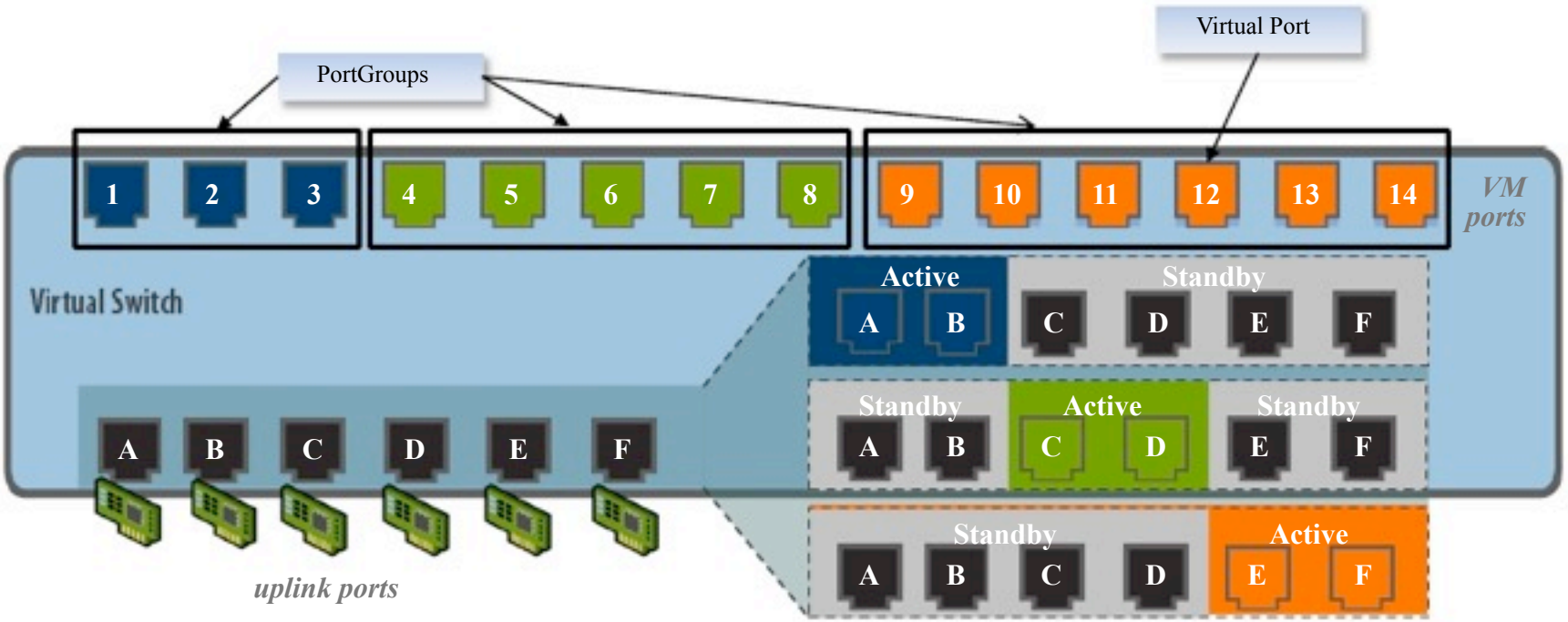
NIC Teaming: IP Hash Based Teaming



- Outgoing uplink chosen from hash of Source IP and Destination IP
- Requires multiple IP Destinations to spread traffic from one source
- Must configure Etherchannel on physical switch
  - Static etherchannel—no LACP
  - Single physical switch unless multi-chassis etherchannel support . e.g.
    - Cisco Cat6500 VSS
    - Catalyst 3750 Cross-Stack Etherchannel
    - Nortel SMLT
- Return path determined by etherchannel hash on physical switch—may use different uplink in NIC team

vmware

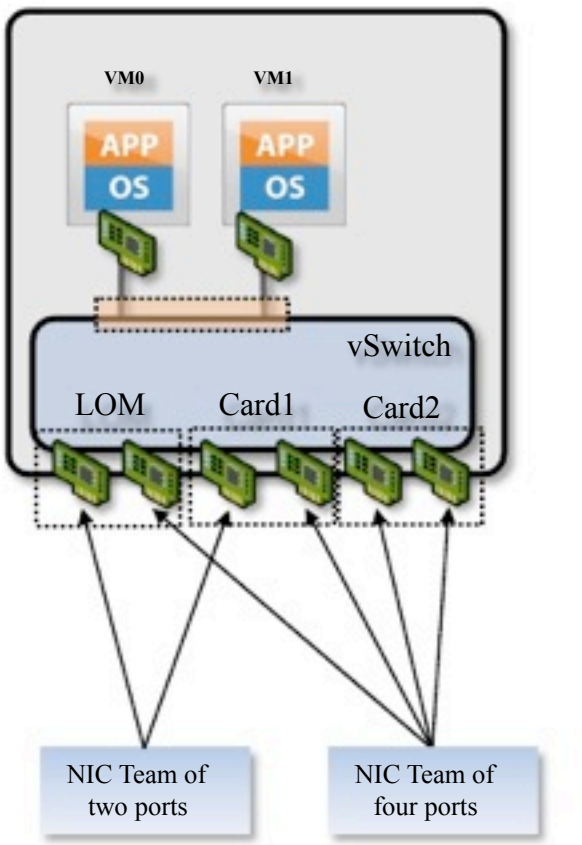
NIC Teaming: Multiple Policies Can Apply Per Team



Port Groups can override failover policy on uplinks for groups of VMs

vmware

NIC Teaming: Assigning Physical NICs



- Mix NIC team with ports from multiple NIC cards and LOM (LAN on Motherboard)
  - Avoid single point of failure from single card or single component

vmware

Port Group Configuration

A Port Group is a template for one or more ports with a common configuration

- Assigns VLAN to port group members
- L2 Security—select ―reject‖ to see only frames for VM mac addr
  - Promiscuous mode/MAC address change/Forged transmits
- Traffic Shaping—limit egress traffic from VM
- Load Balancing—Origin VPID, Src MAC, IP-Hash, Explicit
- Failover Policy— Link Status & Beacon Probing
- Notify Switches—‖yes‖-gratuitously tell switches of mac location
- Failback—‖yes‖ if no fear of blackholing traffic, or, …
- … use Failover Order in ―Active Adapters‖

Distributed Virtual Port Group (vNetwork Distributed Switch)

- All above plus:
  - Bidirectional traffic shaping (ingress and egress)
  - Network VMotion—network port state migrated upon VMotion

vmware

Traffic Types on a Virtual Network

Virtual Machine Traffic

- Traffic sourced and received from virtual machine(s)
- Isolate from each other based on service level

VMotion Traffic

- Traffic sent when moving a virtual machine from one ESX host to another
- Should be isolated

Management Traffic

- Should be isolated from VM traffic (one or two Service Consoles)
- If VMware HA is enabled, includes heartbeats

IP Storage Traffic—NFS and/or iSCSI via vmkernel interface

- Should be isolated from other traffic types

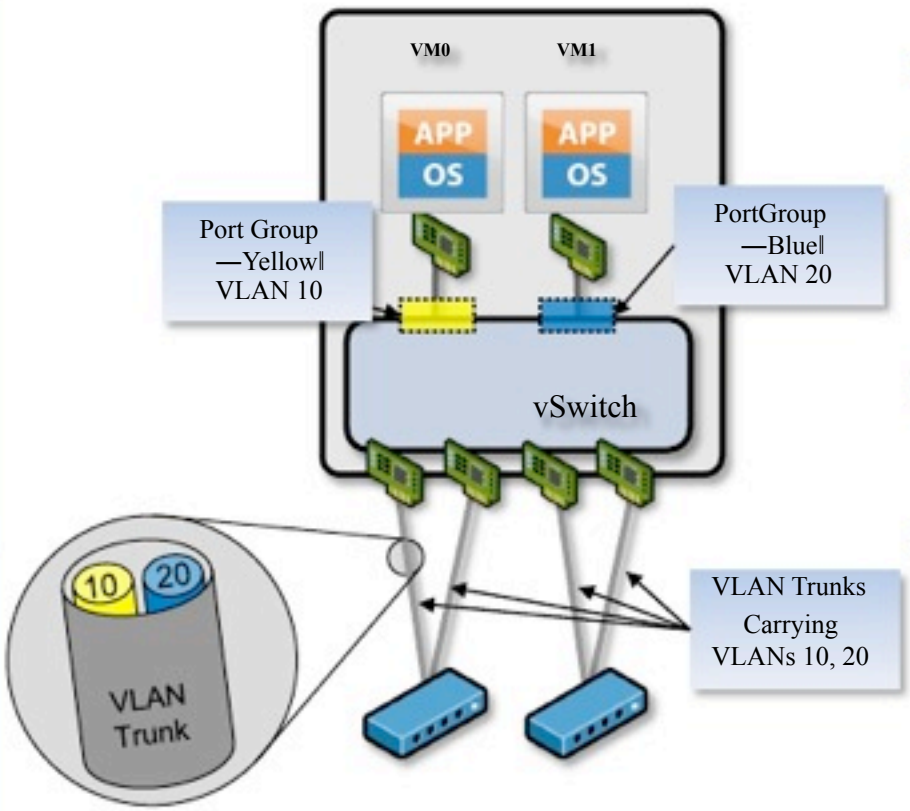
Fault Tolerance (FT) Logging Traffic

- Low latency, high bandwidth
- Should be isolated from other traffic types

How do we maintain traffic isolation without proliferating NICs?

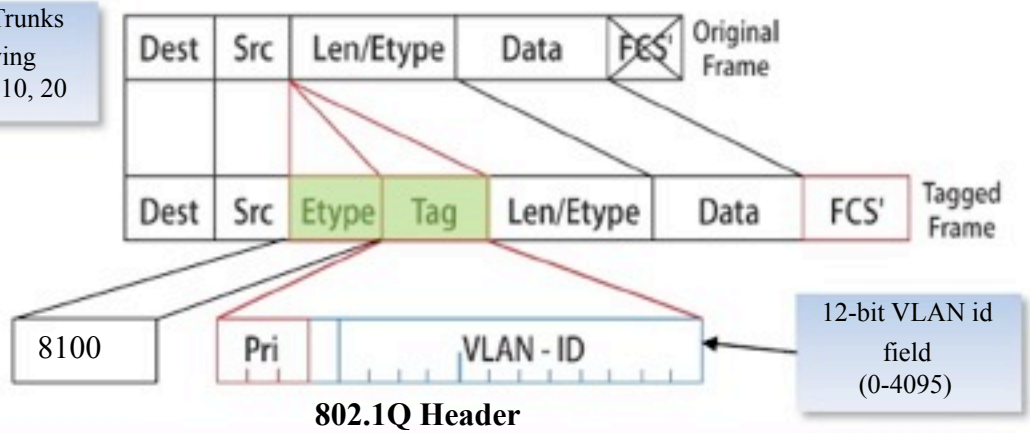
vmware

VLAN Trunking to Server



IEEE 802.1Q VLAN Tagging

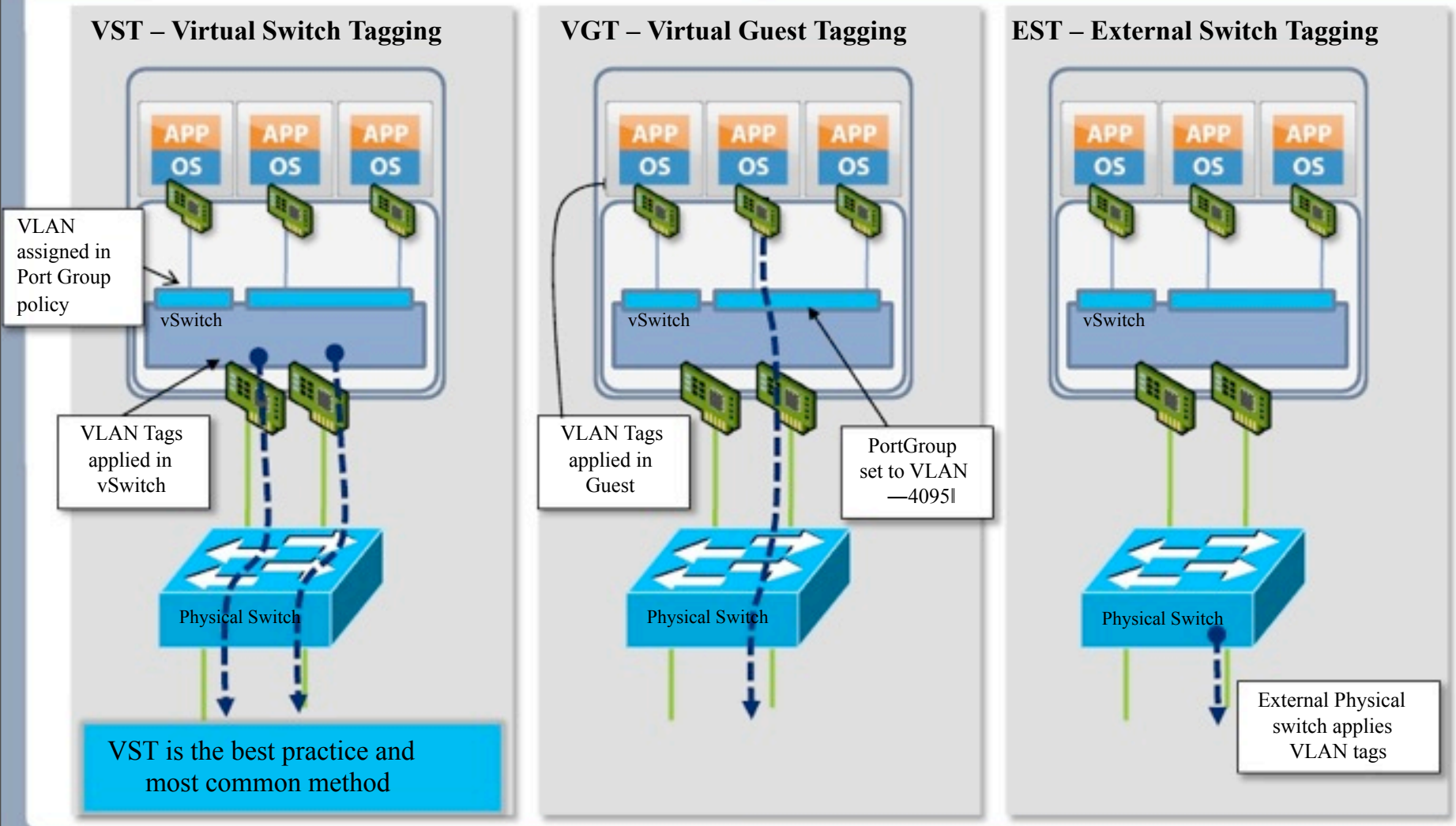
- Enables logical network partitioning (Traffic separation)
- Scale traffic types without scaling physical NICs
- Virtual machines connect to virtual switch ports (like access ports on physical switch)
- Virtual switch ports are associated with a particular VLAN (VST mode)—defined in PortGroup
- Virtual switch tags packets exiting host



802.1Q Header

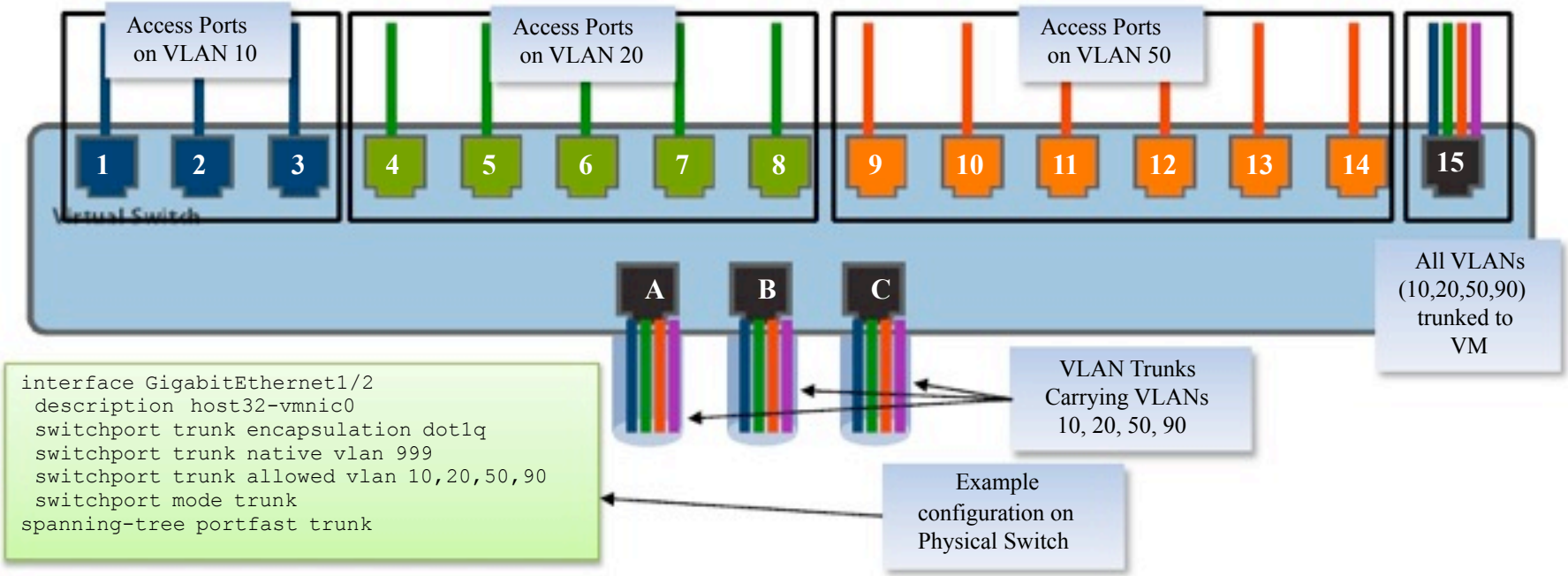
vmware

VLAN Tagging Options



vmware

Virtual Switch VLAN Tagging: Further Example




```
interface GigabitEthernet1/2
 description host32-vmnic0
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 999
 switchport trunk allowed vlan 10,20,50,90
 switchport mode trunk
spanning-tree portfast trunk
```

Uplinks A, B, and C connected to trunk ports on physical switch which carry four VLANs (e.g. VLANs 10, 20, 50, 90)

Ports 1-14 emit *untagged* frames, and only those frames which were tagged with their respective VLAN ID (equivalent to ―access port‖ on physical switch)

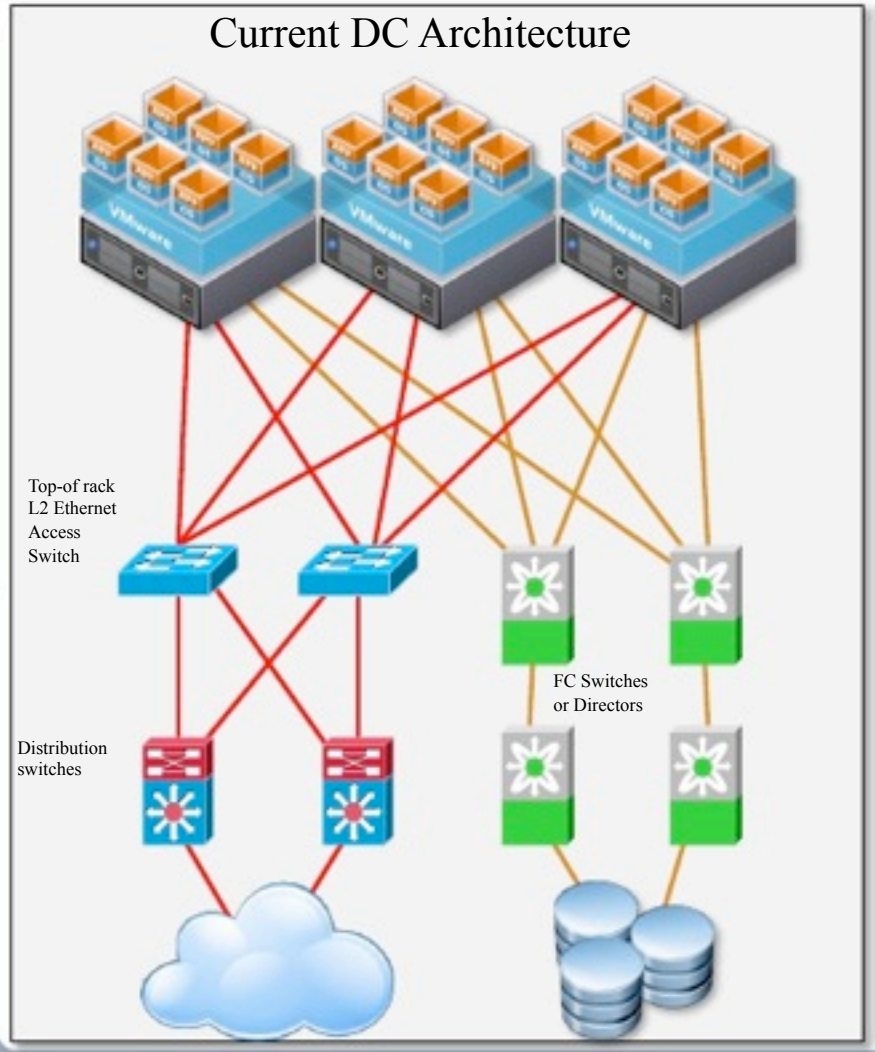
- Port Group VLAN ID set to one of 1-4094

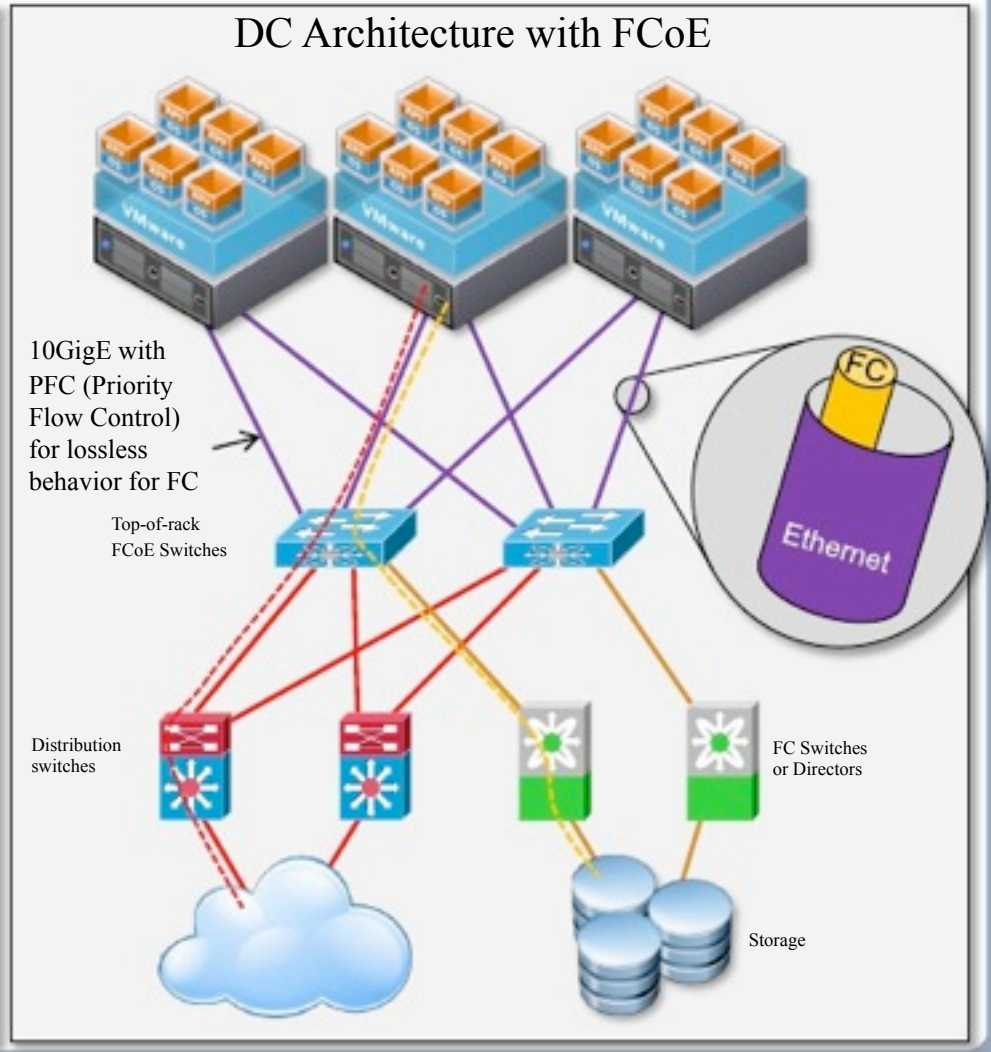
Port 15 emits *tagged* frames for all VLANs.

- Port Group VLAN ID set to 4095 (for vSS) or ―VLAN Trunking‖ on vDS DV Port Group

vmware

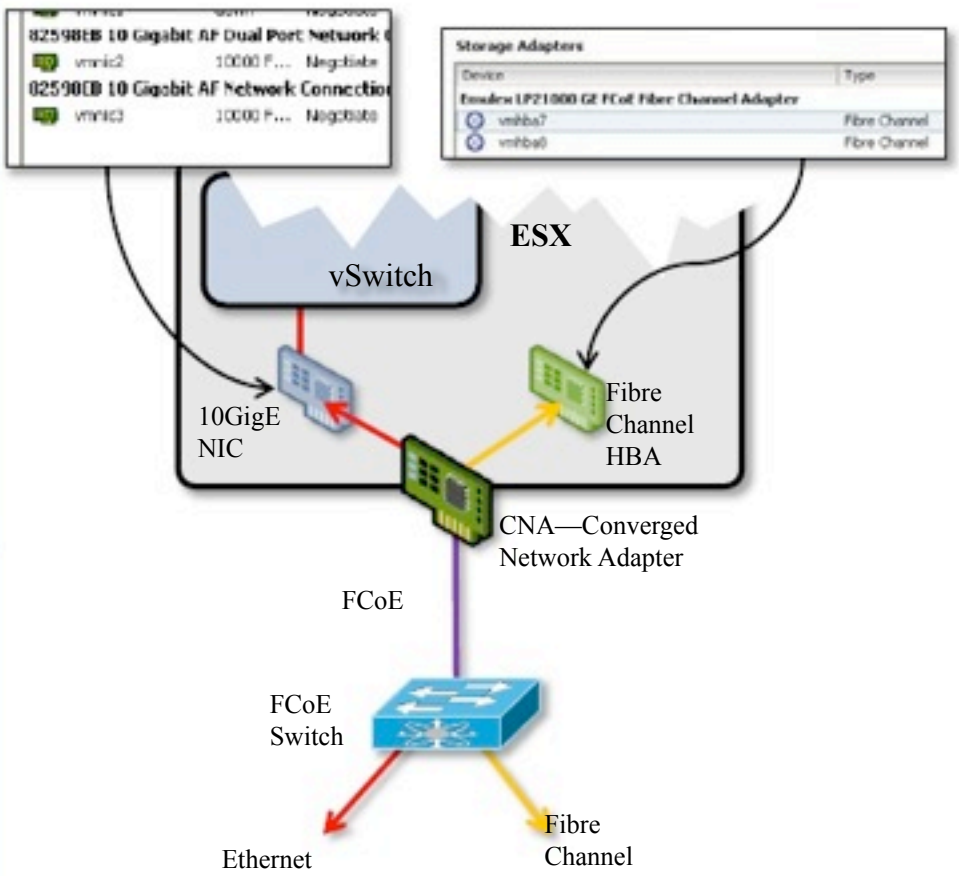
Fibre Channel over Ethernet (FCoE)





vmware

FCoE on ESX

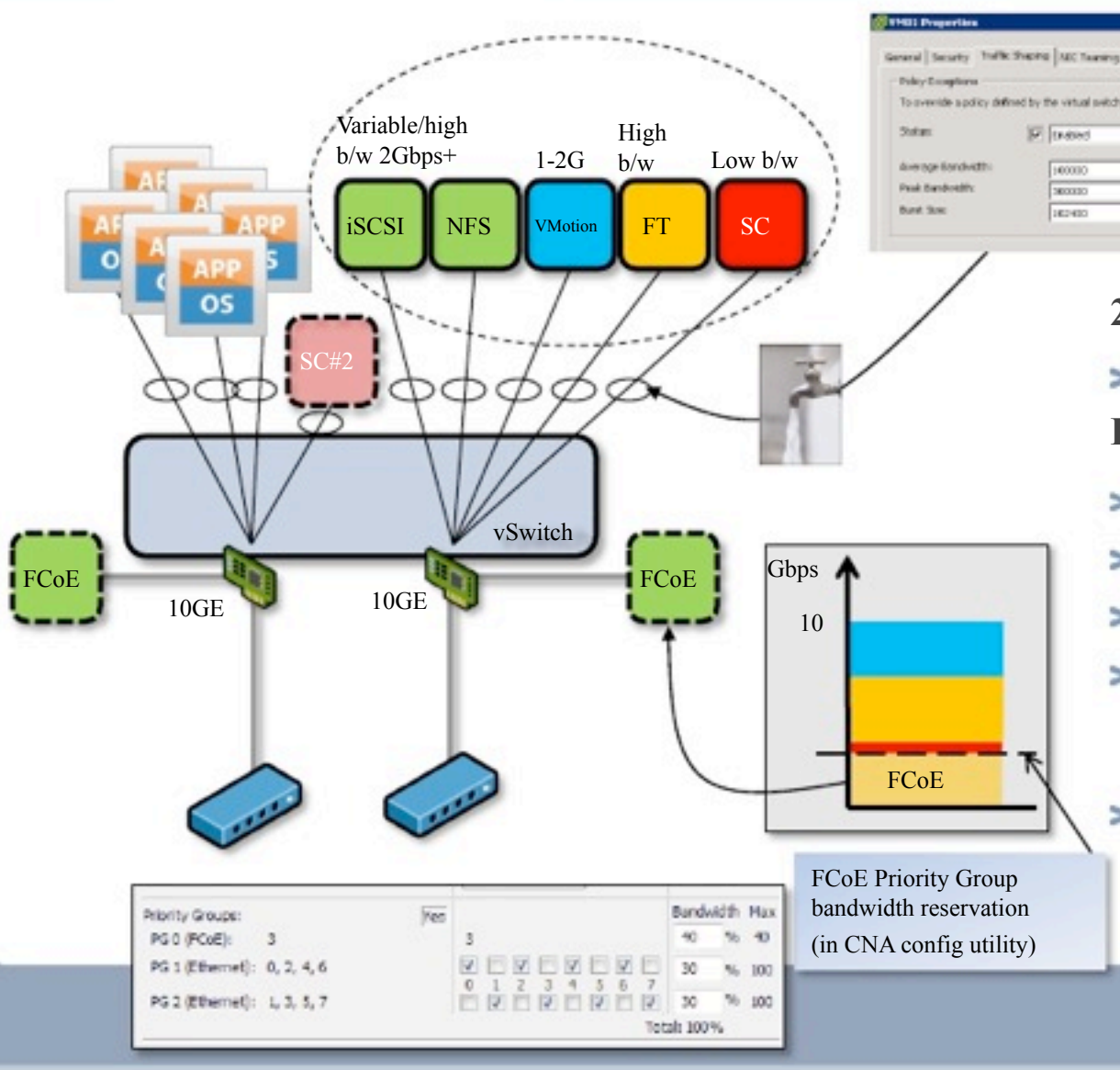


VMware ESX Support

- FCoE supported since ESX 3.5u2
- Requires Converged Network Adapters —CNAs‖—(see HCL) e.g.
  - Emulex LP21000 Series
  - Qlogic QLE8000 Series
- Appears to ESX as:
  - 10GigE NIC
  - FC HBA
- SFP+ pluggable transceivers
  - Copper twin-ax (<10m)
  - Optical

vmware

Using 10GigE



Ingress (into switch) traffic shaping policy control on Port Group

FCoE Priority Group bandwidth reservation (in CNA config utility)

2x 10GigE common/expected

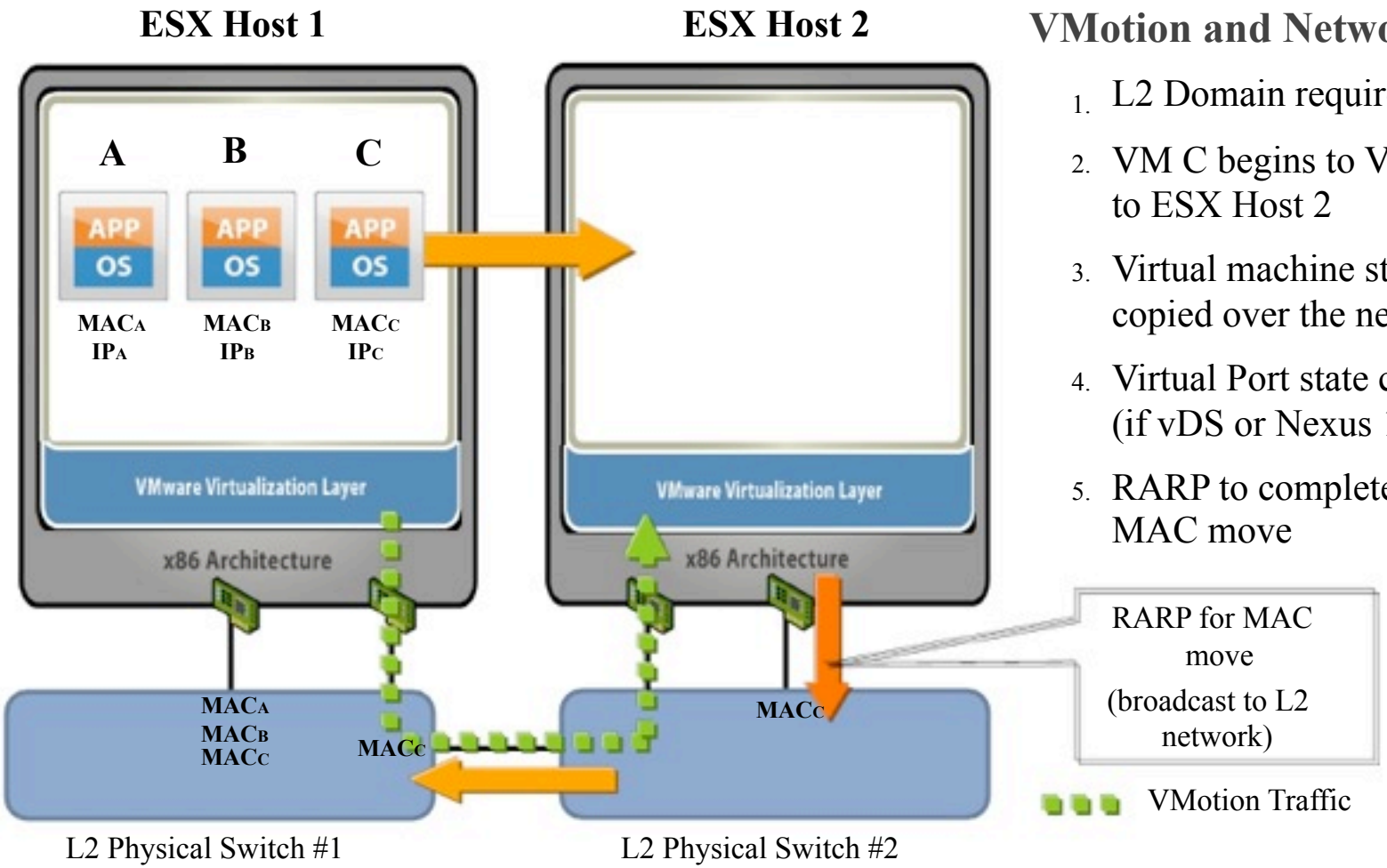
- 10GigE CNAs or NICs

Possible Deployment Method

- Active/Standby on all Portgroups
- VMs ―sticky‖ to one vmnic
- SC/vmk ports sticky to other
- Use Ingress Traffic Shaping to control traffic type per Port Group
- If FCoE, use Priority Group bandwidth reservation (on CNA utility)

vmware

VMotion: How Does It Operate on the Network?

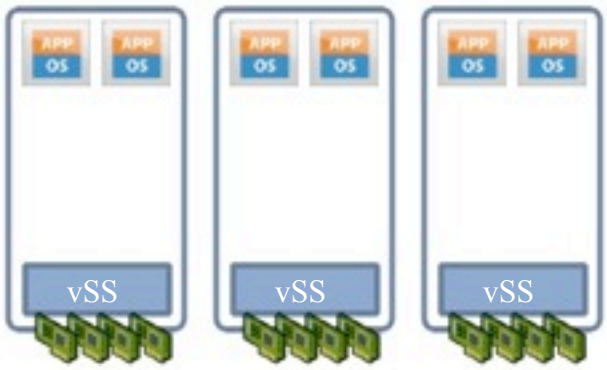


VMotion and Networking

1. L2 Domain required
2. VM C begins to VMotion to ESX Host 2
3. Virtual machine state is copied over the network
4. Virtual Port state copied (if vDS or Nexus 1000V)
5. RARP to complete MAC move

vDS Deployment Options

Original Environment



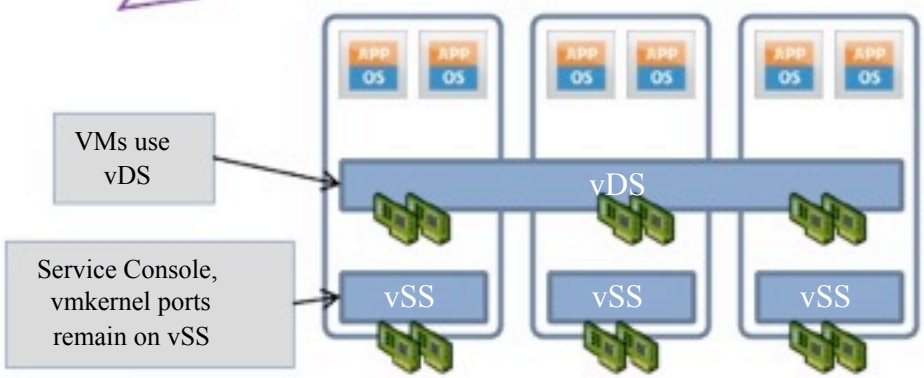
Complete Migration to vDS

vSS, vDS and Nexus Switches can co-exist on same host

Network VMotion only required for Guest VMs

- Optionally leave SC, vmkernel ports on vSS
- Note: enhanced features only on vDS

Partial Migration to vDS

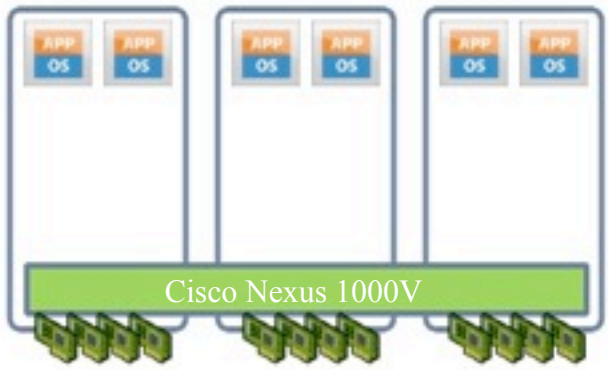


vmware

vDS Deployment Options (Cont.)

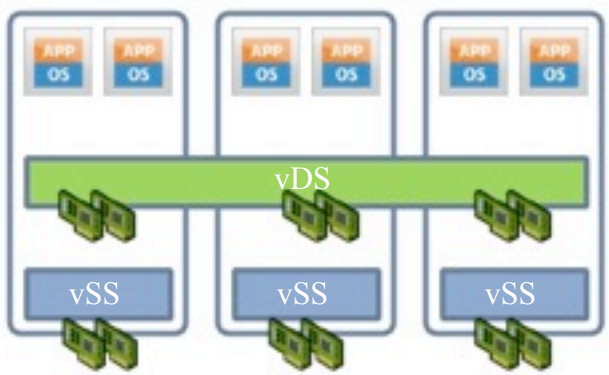
Original Environment

Complete Migration to Nexus 1000V



Multiple vDS

Partial Migration to Nexus 1000V



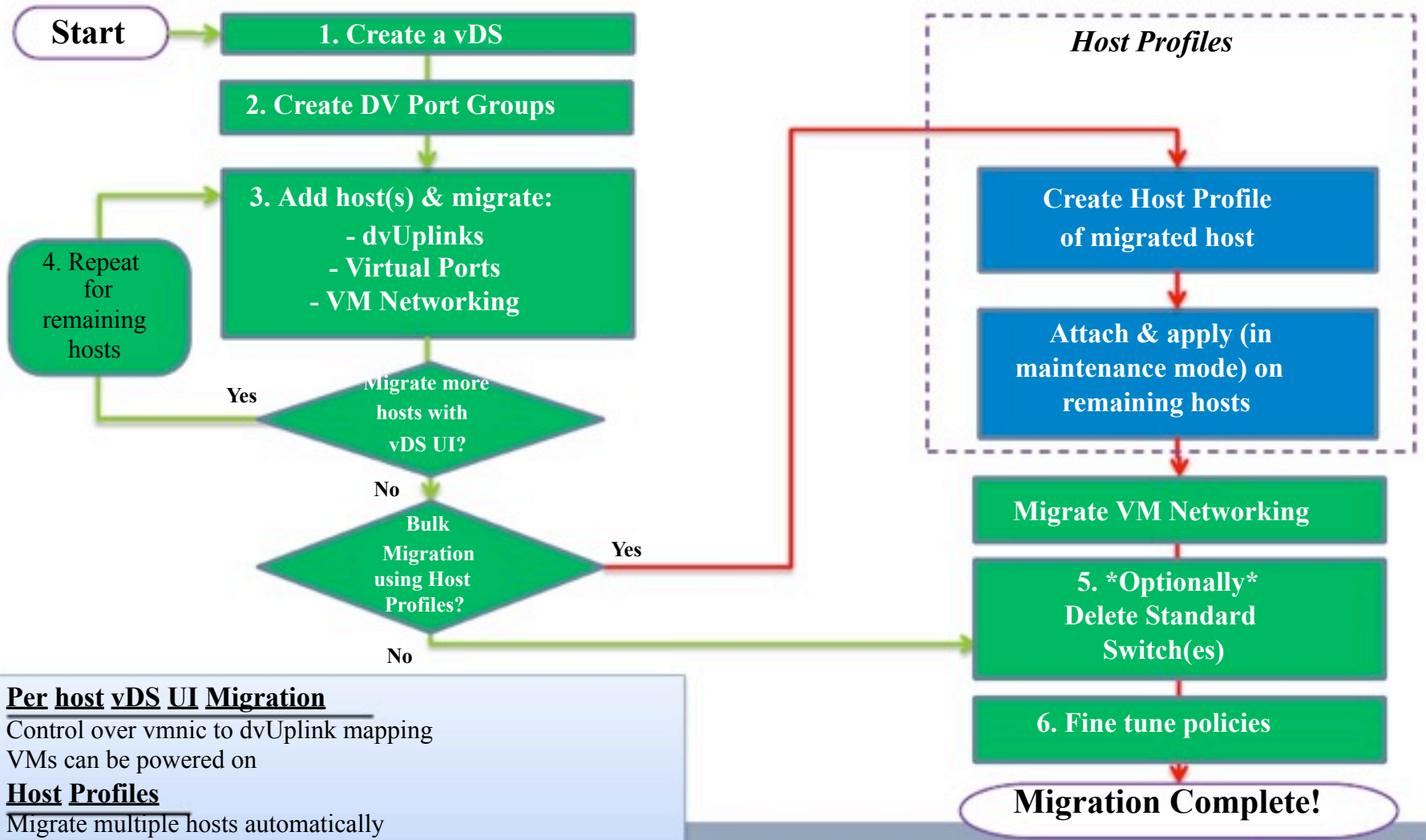
vmware

Deployment Rules

- vSS, vDS, Nexus 1000V can co-exist
  - Multiple vSS and vDS per host
  - Maximum of one Nexus 1000V per host (VEM)
- Take note of deployment limits (subject to change!)
  - Refer to published limits
- pnics (vmnics) can only belong to one virtual switch

vmware

Provisioning vDS using vDS UI and Host Profiles



Per host vDS UI Migration

Control over vmnic to dvUplink mapping

VMs can be powered on

Host Profiles

Migrate multiple hosts automatically

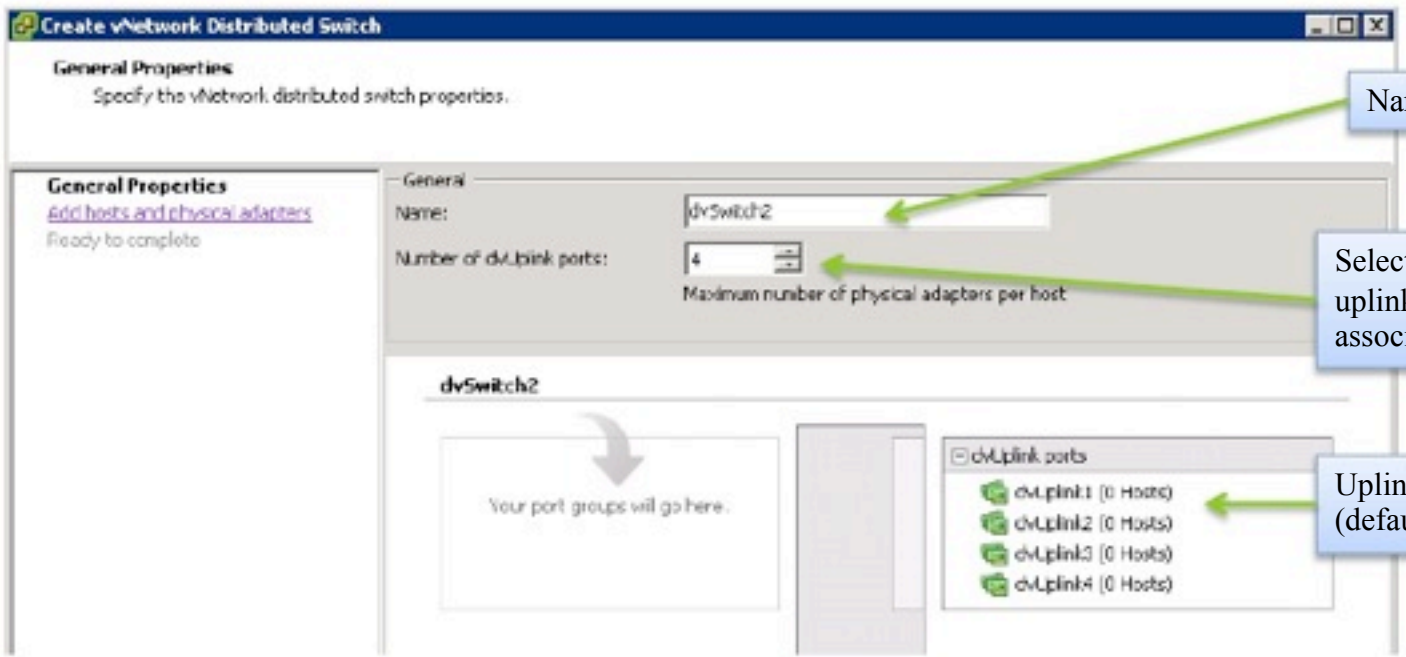
Requires maintenance mode (VMs off or migrated)

vmware

vDS: Step 1: Select General Properties

Home > Inventory > Networking

→Select —New vNetwork Distributed Switch‖



Name the vDS

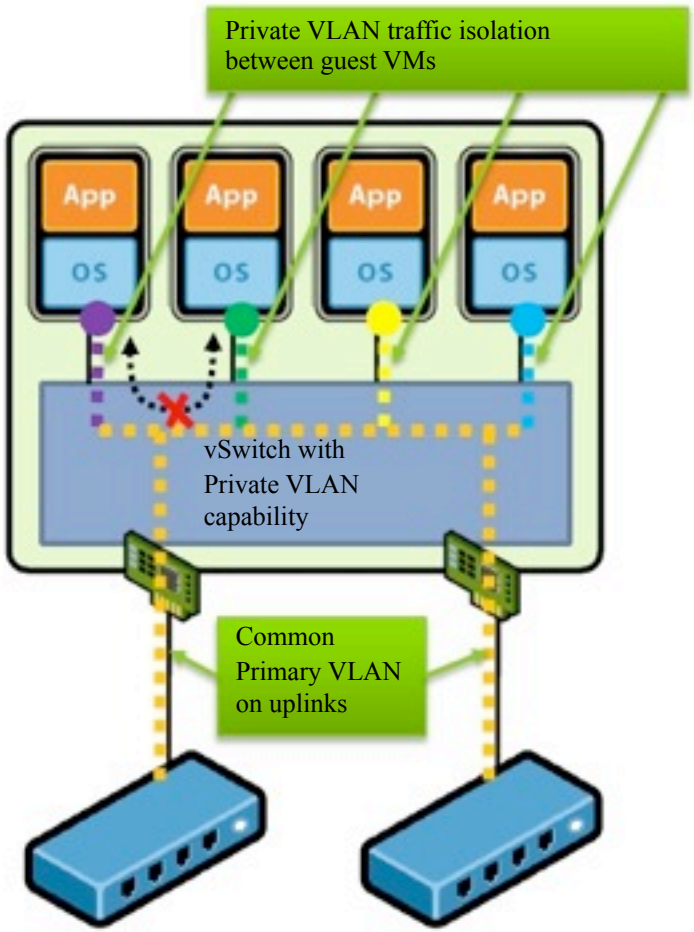
Select the max number of uplink ports (NICs) of any host associated with this vDS

Uplinks show up here (default is four)

vmware

Private VLANs: Traffic Isolation for VMs

- Scale VMs on same subnet but selectivity restrict inter-VM communication
- Avoids scaling and complexity issues from assigning one VLAN and IP subnet per VM



vmware

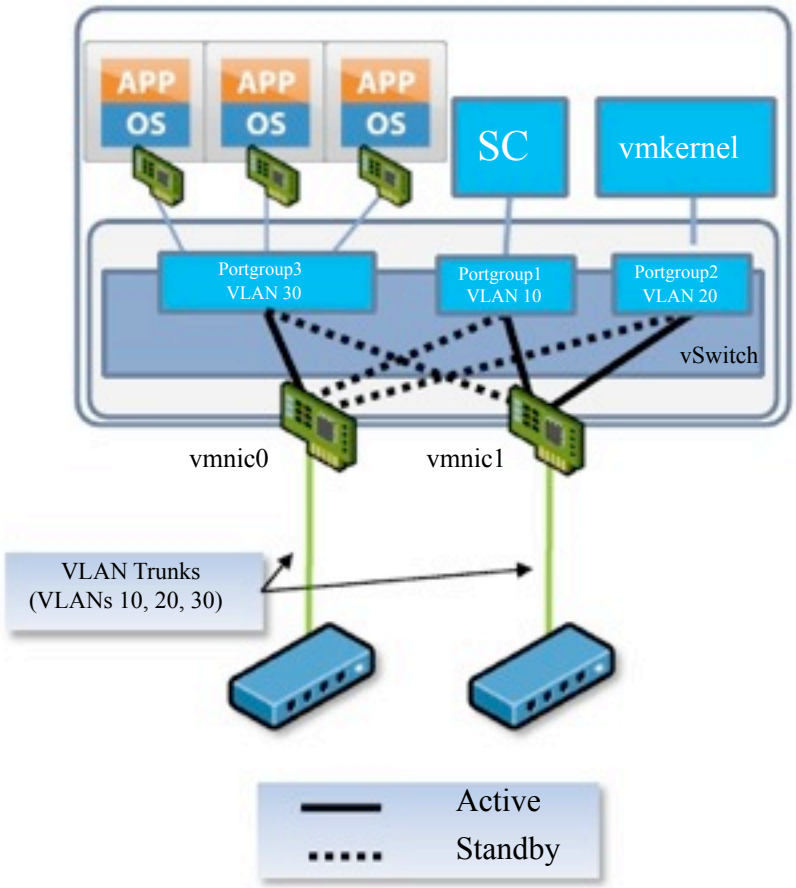
Designing the Network

How do you design the virtual network for performance and availability and but maintain isolation between the various traffic types (e.g. VM traffic, VMotion, and Management)?

- Starting point depends on:
  - Number of available physical ports on server
  - Required traffic types
- 2 NIC minimum for availability, 4+ NICs per server preferred
- 802.1Q VLAN trunking highly recommended for logical scaling (particularly with low NIC port servers)
- Following examples are meant as guidance and do not represent strict requirements in terms of design
- Understand your requirements and resultant traffic types and design accordingly

vmware

Example 1: Blade Server with 2 NIC Ports



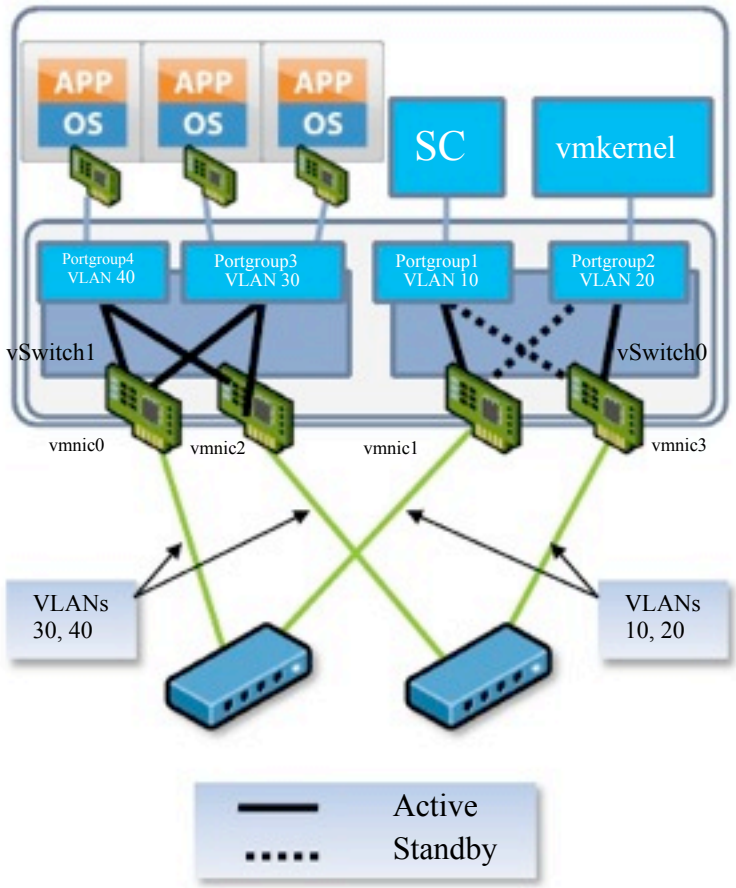
Candidate Design:

- Team both NIC ports
- Create one virtual switch
- Create three port groups:
  - Use Active/Standby policy for each portgroup
  - Portgroup1: Service Console (SC)
  - Portgroup2: VMotion
  - Portgroup3: VM traffic
- Use VLAN trunking
  - Trunk VLANs 10, 20, 30 on each uplink

Note: Team over dvUplinks with vDS

vmware

Example 2: Server with 4 NIC Ports



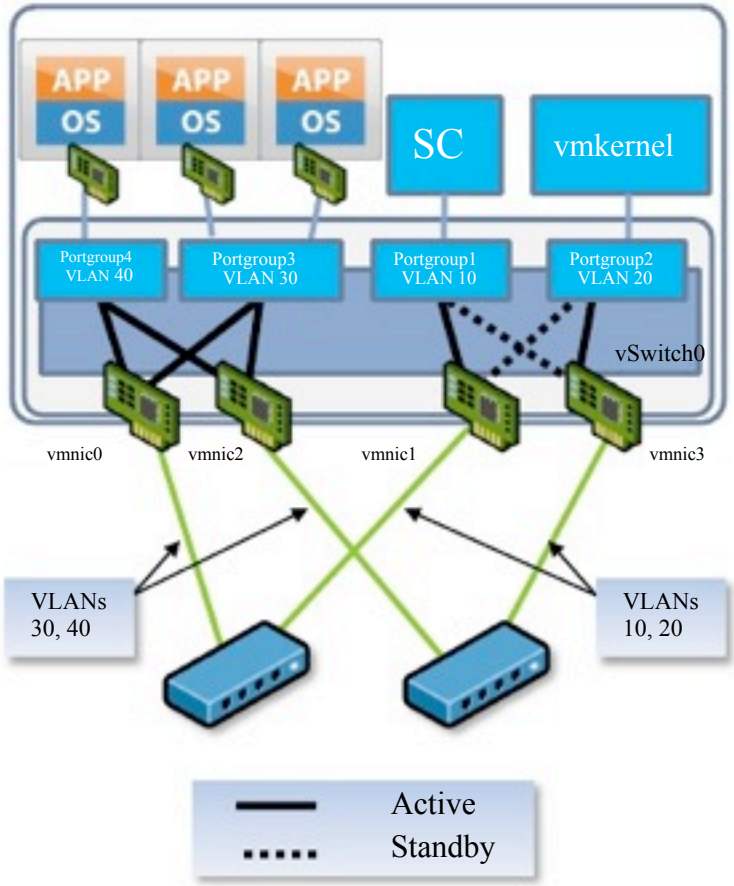
Note: Team over dvUplinks with vDS

Candidate Design:

- Create two virtual switches
- Team two NICs to each vSwitch
- vSwitch0 (use active/standby for each portgroup):
  - Portgroup1: Service Console (SC)
  - Portgroup2: VMotion
- vSwitch1 (use Originating Virtual PortID)
  - Portgroup3: VM traffic #1
  - Portgroup4: VM traffic #2
- Use VLAN trunking
  - vmnic1 and vmnic3: Trunk VLANs 10, 20
  - vmnic0 and vmnic2: Trunk VLANs 30, 40

vmware

Example 3: Server with 4 NIC Ports (Slight Variation)



Note: Team over dvUplinks with vDS

Candidate Design:

- Create one virtual switch
- Create two NIC teams
- vSwitch0 (use active/standby for portgroups 1 & 2):
  - Portgroup1: Service Console (SC)
  - Portgroup2: Vmotion
- Use Originating Virtual PortID for Portgroups 3 & 4
  - Portgroup3: VM traffic #1
  - Portgroup4: VM traffic #2
- Use VLAN trunking
  - vmnic1 and vmnic3: Trunk VLANs 10, 20
  - vmnic0 and vmnic2: Trunk VLANs 30, 40

vmware

Servers with More NIC Ports

More than 4 NIC Ports – Design Considerations

With Trunks (VLAN tagging):

- Use previous approach and scale up to meet additional bandwidth and redundancy requirements
- Add NICs to NIC team supporting VM traffic

VLAN Tagging always recommended, but options if NICs available:

- Dedicated NIC for VMotion
  - At least one NIC
- Dedicated NICs for IP Storage (NFS and/or iSCSI)
  - Usually two teamed NICs (consider IP-hash & etherchannel if multiple destinations and Multi-Chassis Etherchannel employed on physical switches)
- Dedicated NIC(s) for Service Console
  - At least two for availability

Note: easy to consume many physical NICs and switch ports if not using VLAN tagging

vmware

IP Storage: Using iSCSI

Provides SCSI block storage access over IP network

Relevant for VMs using the iSCSI software-based initiator

Design depends on number of NIC ports available on server

General Design Guidance:

- Keep iSCSI traffic on its own dedicated subnet (VLAN)
- Dedicate (if possible) specific NIC(s) to iSCSI traffic
- For redundancy, use at least 2 NICs
- In multi-NIC scenarios, use teaming with:
  - —Virtual Source Port ID‖ setting if all your iSCSI targets share the same IP address
  - —IP Hash‖ setting for other scenarios, including the case for multiple targets

Note: iSCSI Multipath available in ESX 4.0

vmware

iSCSI Design Guide – Specific Examples

Common Case – Server with 6x 1GigE NIC Ports:

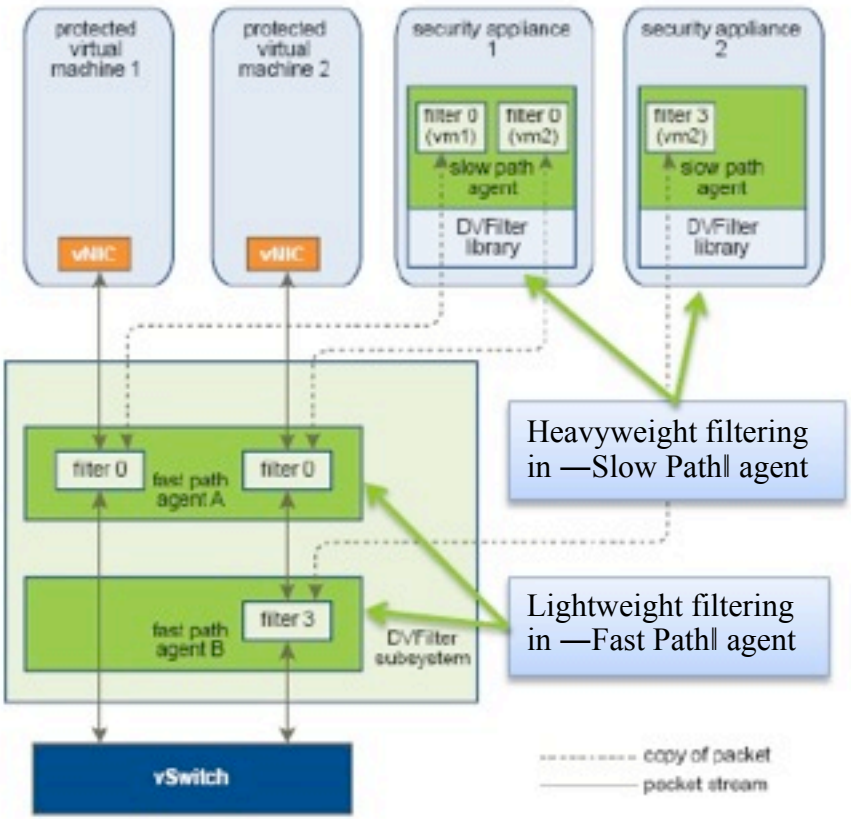
- Follow the 4 port example
- For remaining 2 NIC ports:
  - Create a virtual switch dedicated to iSCSI traffic
  - Create a port group dedicated to iSCSI traffic
  - Team and dedicate both NICs to iSCSI traffic

Uncommon case (some blades) – 2x 1GigE ports:

- Buying additional NIC ports recommended (if possible)
- Follow 2 port example (Port Group 1 and 2)
- For environments with high amount of VM traffic:
  - Create port group 1 – SC + VMotion + iSCSI
  - Create port group 2 – VM traffic
- For environments with low VM traffic:
  - Create port group 1 – SC + VMotion
  - Create port group 2 – VM traffic + iSCSI

vmware

vNetwork Appliance API



- Filter driver in vmkernel to provide security features within ESX networking layer
- vNetwork Appliance APIs available to partners
- Clients of this API may inspect/alter/drop/inject any frame on a given port:
  - Either directly in the IO path (fast path agent)
  - Or by punting frames up to an appliance VM (slow path agent)
- State mobility for data in fast path agent *and* slow path agent
- Communication between slow path and fast path agents
- Bind to VM's vNIC or to dvswitch port

vmware

IPv6 in vSphere 4

- IPv6 guests supported since ESX 3.5
- IPv6 support for
  - ESX 4
  - vSphere Client
  - vCenter Server
  - Vmotion
  - IP Storage (iSCSI, NFS)—experimental
- Not supported for vSphere vCLI, HA, FT

vmware

Further Reading…

VMware Networking Technology

vmware.com/go/networking

Networking Blog

blogs.vmware.com/networking

Reference

- Books :
  - Kumar Reddy & Victor Moreno, ***Network Virtualization***, Cisco Press 2006
- Web resources :
  - Linux Bridge *http://www.ibm.com/developerworks/cn/linux/l-tuntap/index.html*
  - Xen networking *http://wiki.xensource.com/xenwiki/XenNetworking*
  - VMware Virtual Networking Concepts
    http://www.vmware.com/files/pdf/virtual_networking_concepts.pdf
  - TUN/TAP wiki *http://en.wikipedia.org/wiki/TUN/TAP*
  - Network Virtualization wiki *http://en.wikipedia.org/wiki/Network_virtualization*
  - *VMware Networking Technology vmware.com/go/networking*
- Papers :
  - A. Menon, A. Cox, and W. Zwaenepoel. Optimizing Network Virtualization in Xen. USENIX Annual Technical Conference (USENIX 2006), pages 15–28, 2006.
  - N.M. Mosharaf Kabir Chowdhury, Raouf Boutaba, “A Survey of Network Virtualization”, University of Waterloo Technical Report CS-2008-25, Oct. 2008.