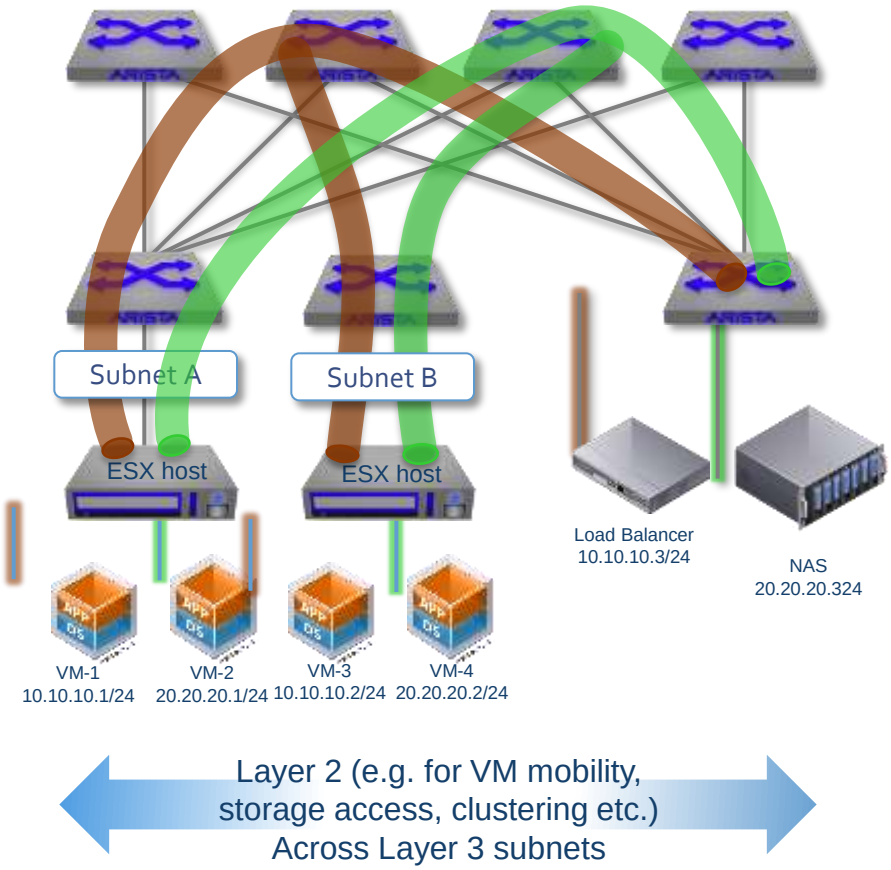
VXLAN Bridging & Routing

Darrin Machay
darrin@arista.com

CHI-NOG 05
May 2015

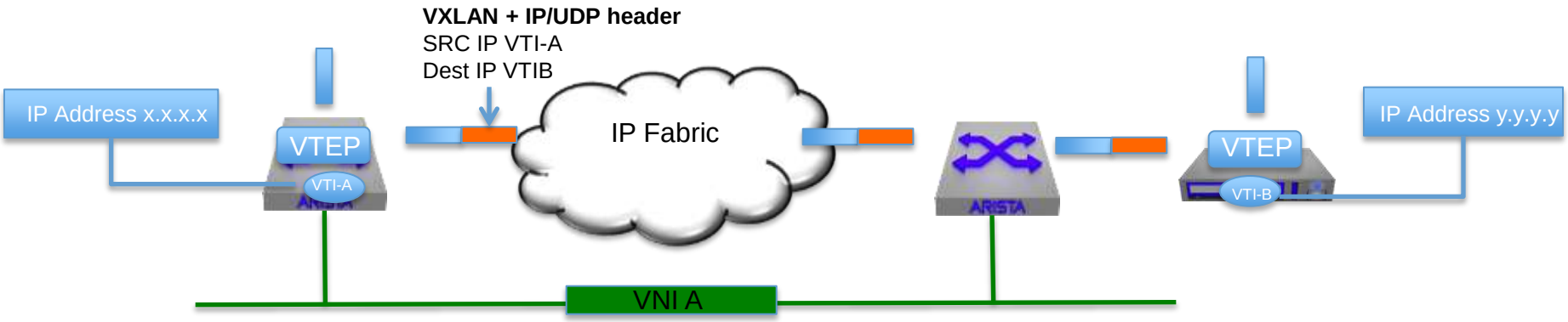
VXLAN



- **Virtual eXtensible LAN (VXLAN)**
  - IETF framework proposal, co-authored by:
    - Arista, Broadcom, Cisco, Citrix Red Hat & VMware
- **Provides Layer 2 "Overlay Networks" on top of a Layer 3 network**
  - "MAC in IP" Encapsulation
  - Layer 2 multi-point tunneling over IP UDP
- **Enables Layer 2 interconnection across Layer 3 boundaries**
  - Transparent to the physical IP network
  - Provides Layer 2 scale across the Layer 3 IP fabric
  - Abstracts the Virtual connectivity from the physical IP infrastructure
  - e.g. Enables VMotion, L2 clusters etc. across standards based IP fabrics

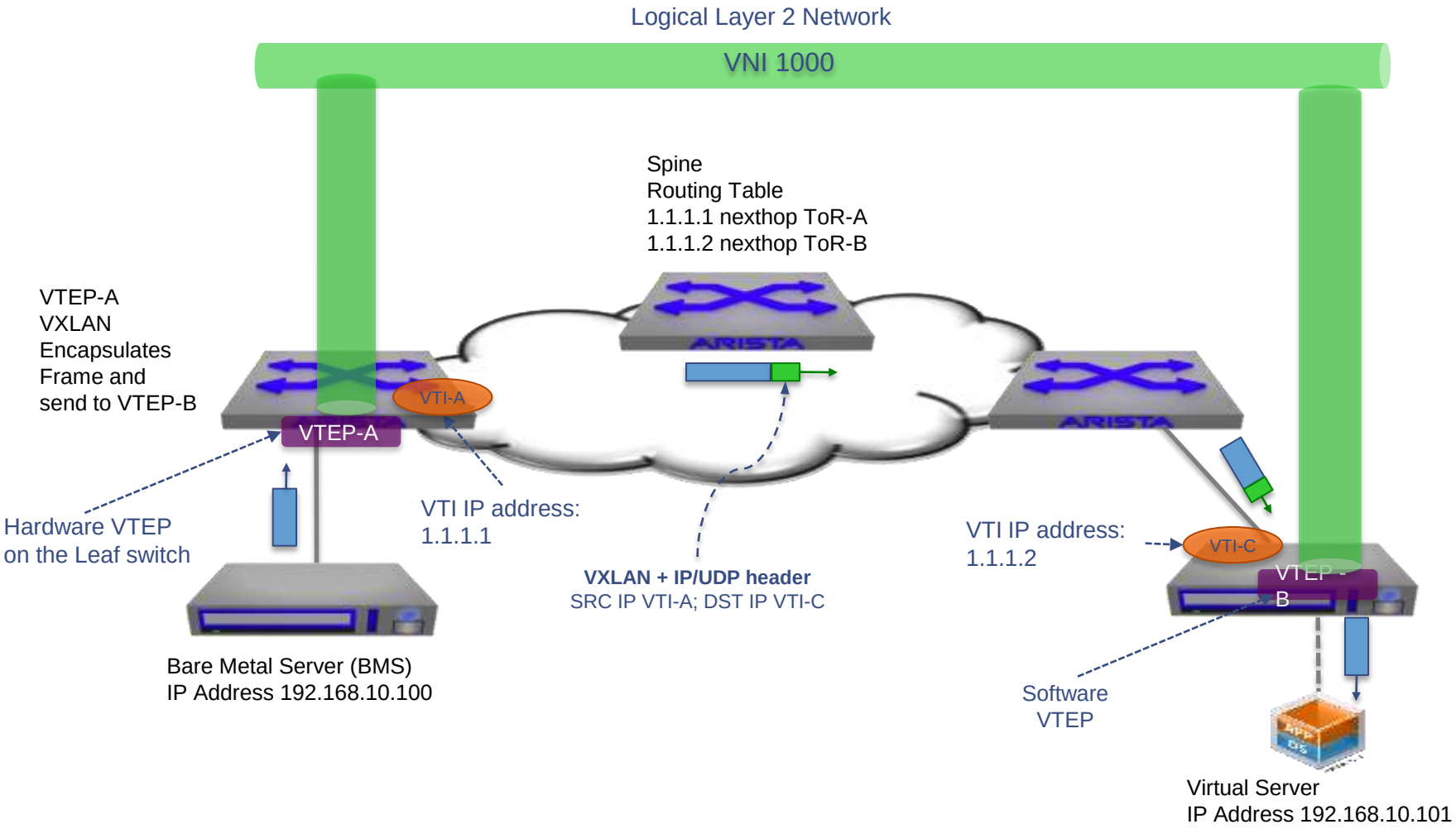
VXLAN Terminology

- **Virtual Tunnel End-point (VTEP).**
  - The VTEP acts as the entry point for connecting hosts into the VXLAN overlay network.
  - The task of the VTEP is to encap/decap with the appropriate VXLAN header.
  - The VTEP component can reside either a software virtual switch or a physical switch.
- **Virtual Tunnel Identifier (VTI)**
  - An IP interface used as the Source IP address for the encapsulated VXLAN traffic
- **Virtual Network Identifier (VNI)**
  - A 24-bit field added within the VXLAN header.
  - Identifies the Layer 2 segment of the encapsulated Ethernet frame
- **VXLAN Header**
  - The IP/UDP and VXLAN header added by the VTEP
  - The SRC UDP port of the header is a hash of the inner frame to create entropy for ECMP



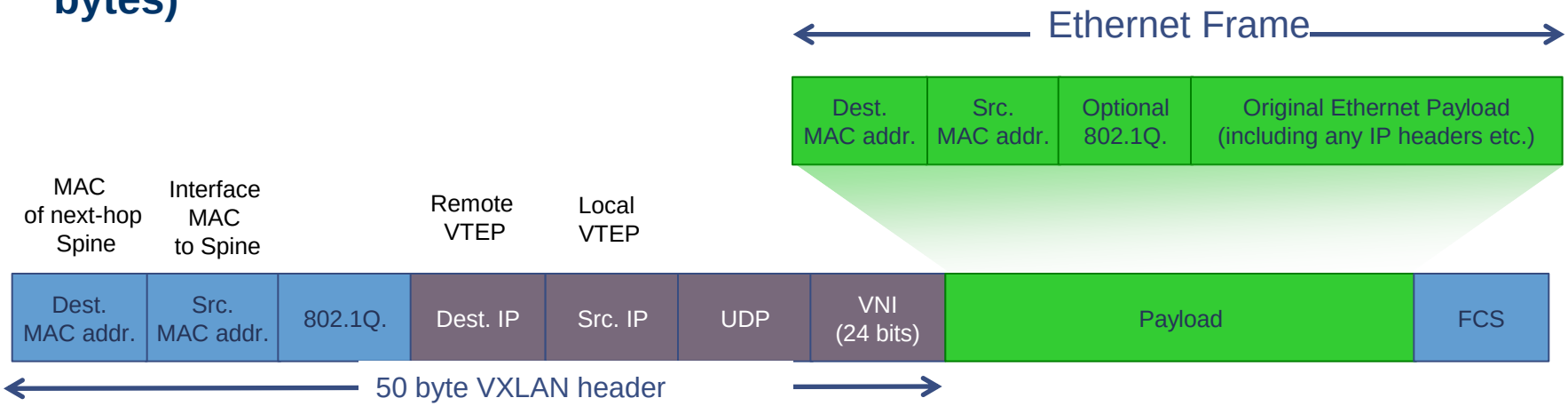
VXLAN Logical Layer 2

VXLAN Components



VXLAN Encapsulated Frame Format

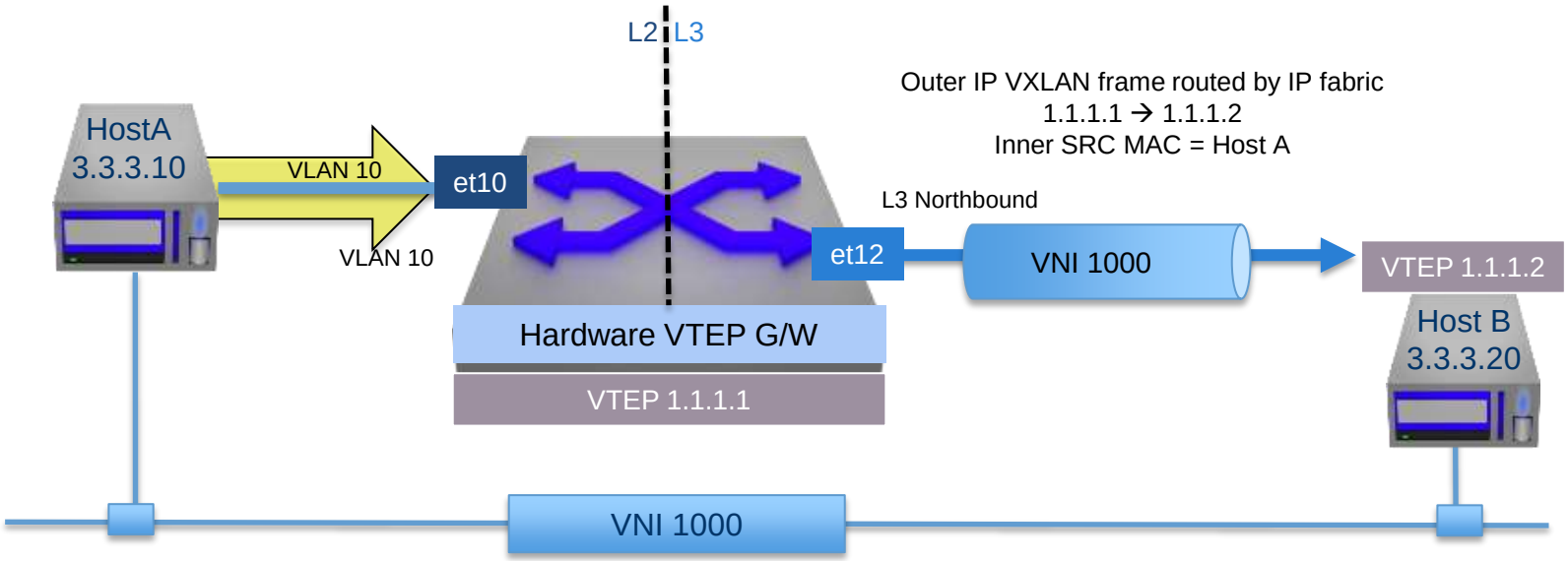
- **Ethernet header uses local VTEP MAC and default router MAC (14 bytes plus 4 optional 802.1Q header)**
- **The VXLAN encapsulation source/destination IP addresses are those of local/remote VTI (20 bytes)**
- **UDP header, with SRC port hash of the inner Ethernets header, destination port IANA defined (8 bytes)**
  - Allows for ECMP load-balancing across the network core which is VXLAN unaware.
- **24-bit VNI to scale up to 16 million for the Layer 2 domain/ vWires (8 bytes)**



VXLAN Control Plane

- **The VXLAN control plane is used for MAC learning and packet flooding**
  - Learning what remote VTEP a host resides behind
  - Mapping the remote MAC to a the VTI of the remote VTEP
  - Allowing traffic destined to the remote MAC via unicast
  - Forwarding of the Broadcast and multicast traffic within the Layer 2 segment (VNI)
- **Typically flood-and-learn using head-end replication (HER)**

VXLAN Bridging

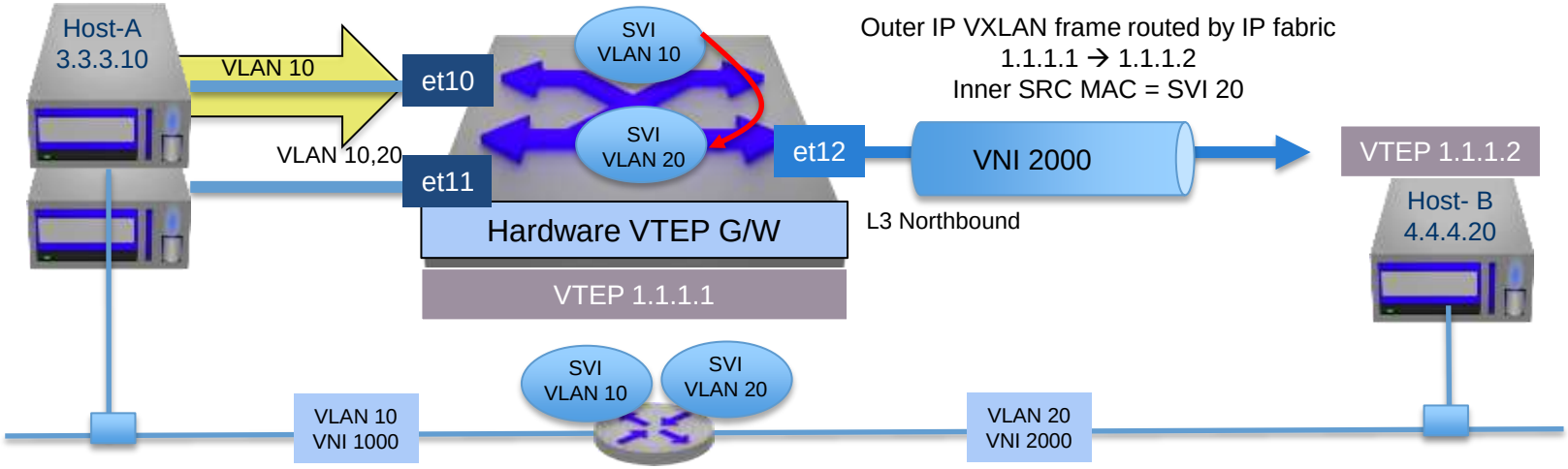



```
interface ethernet10
  switchport
  switchport mode access
  switchport access vlan 10
!
interface ethernet11
  switchport
  switchport mode trunk
  switchport trunk allow vlan 10,20
!
interface ethernet12
  no switchport
  ip address 2.2.2.1/30
```

```
interface loopback0
  ip address 1.1.1.1/32
!
interface vxlan1
  vxlan vlan 10 vni 1000
  vxlan vlan 20 vni 2000
  vxlan source-interface loopback0
!
ip route 0.0.0.0/0 2.2.2.2
```

VXLAN Routing

- **Route and the VXLAN Encap**
  - Local host with a DG on the local VTEP forwarding to Remote host in a different subnet

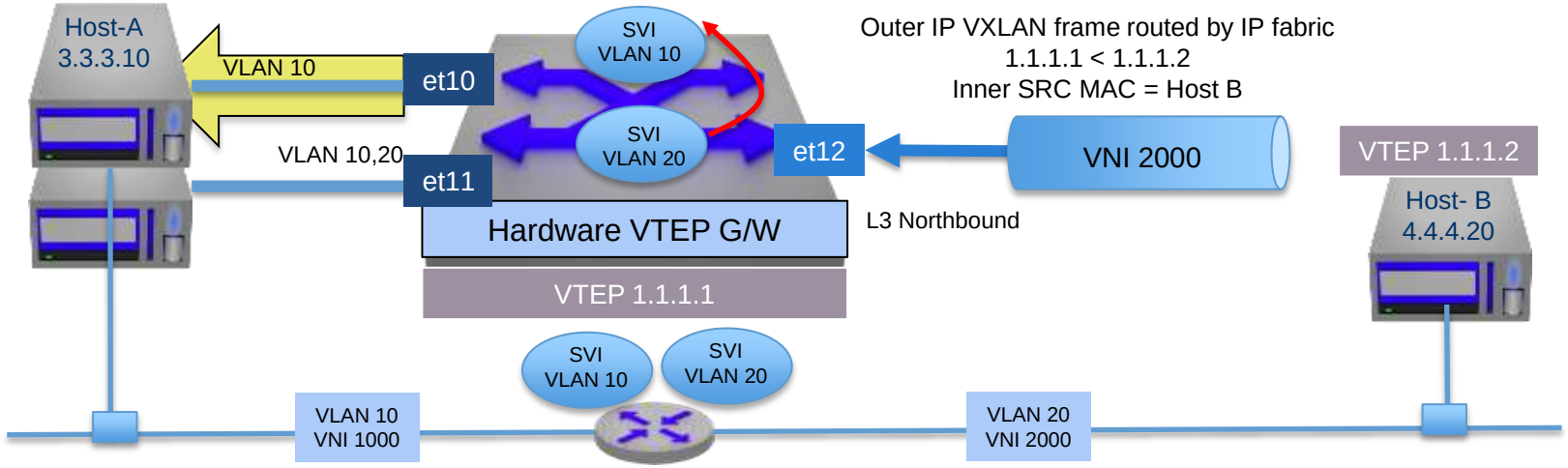



```
interface ethernet10
  switchport
  switchport mode access
  switchport access vlan 10
!
interface ethernet11
  switchport
  switchport mode trunk
  switchport trunk allow vlan 10,20
!
interface ethernet12
  no switchport
  ip address 2.2.2.1/30
```

```
interface vlan 10
  ip address 3.3.3.1/24
!
Interface vlan 20
  ip address 4.4.4.1/24
!
interface loopback0
  ip address 1.1.1.1/32
!
interface vxlan1
  vxlan vlan 10 vni 1000
  vxlan vlan 20 vni 2000
  vxlan source-interface loopback0
!
ip route 0.0.0.0/0 2.2.2.2
```

VXLAN Routing

- **VXLAN Decap and then Route**
  - Host with a DG on a remote VTEP, where the destination host also locally resides

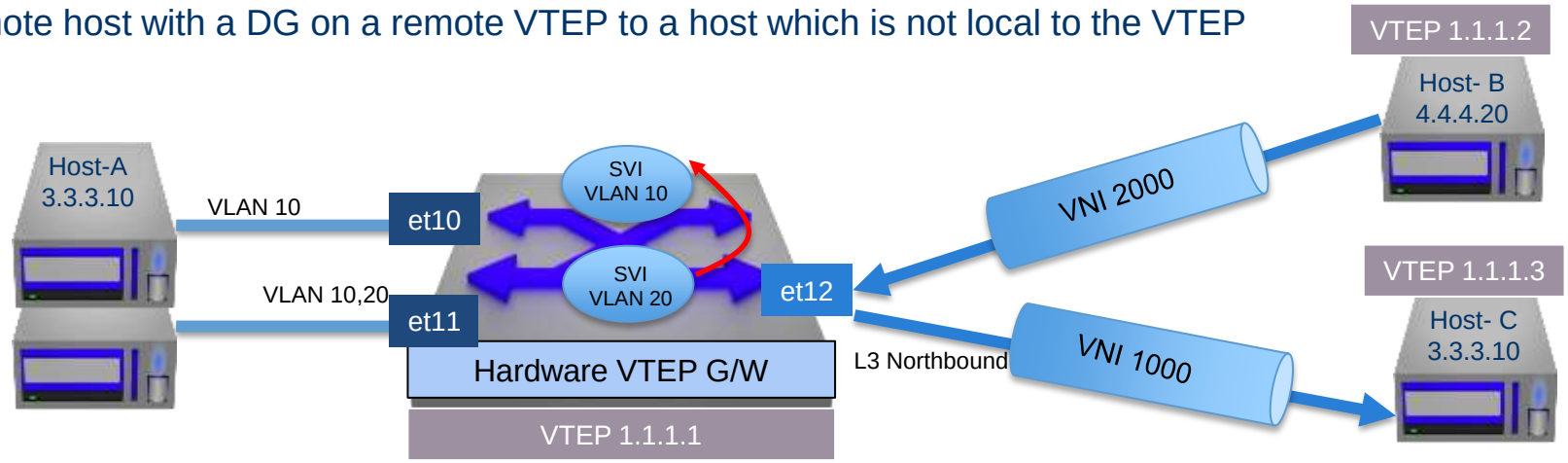



```
interface ethernet10
  switchport
  switchport mode access
  switchport access vlan 10
!
interface ethernet11
  switchport
  switchport mode trunk
  switchport trunk allow vlan 10,20
!
interface ethernet12
  no switchport
  ip address 2.2.2.1/30
```

```
interface vlan 10
  ip address 3.3.3.1/24
!
Interface vlan 20
  ip address 4.4.4.1/24
!
interface loopback0
  ip address 1.1.1.1/32
!
interface vxlan1
  vxlan vlan 10 vni 1000
  vxlan vlan 20 vni 2000
  vxlan source-interface loopback0
!
ip route 0.0.0.0/0 2.2.2.2
```

VXLAN Routing

- **VXLAN Decap, Route and then Encap**
  - Remote host with a DG on a remote VTEP to a host which is not local to the VTEP



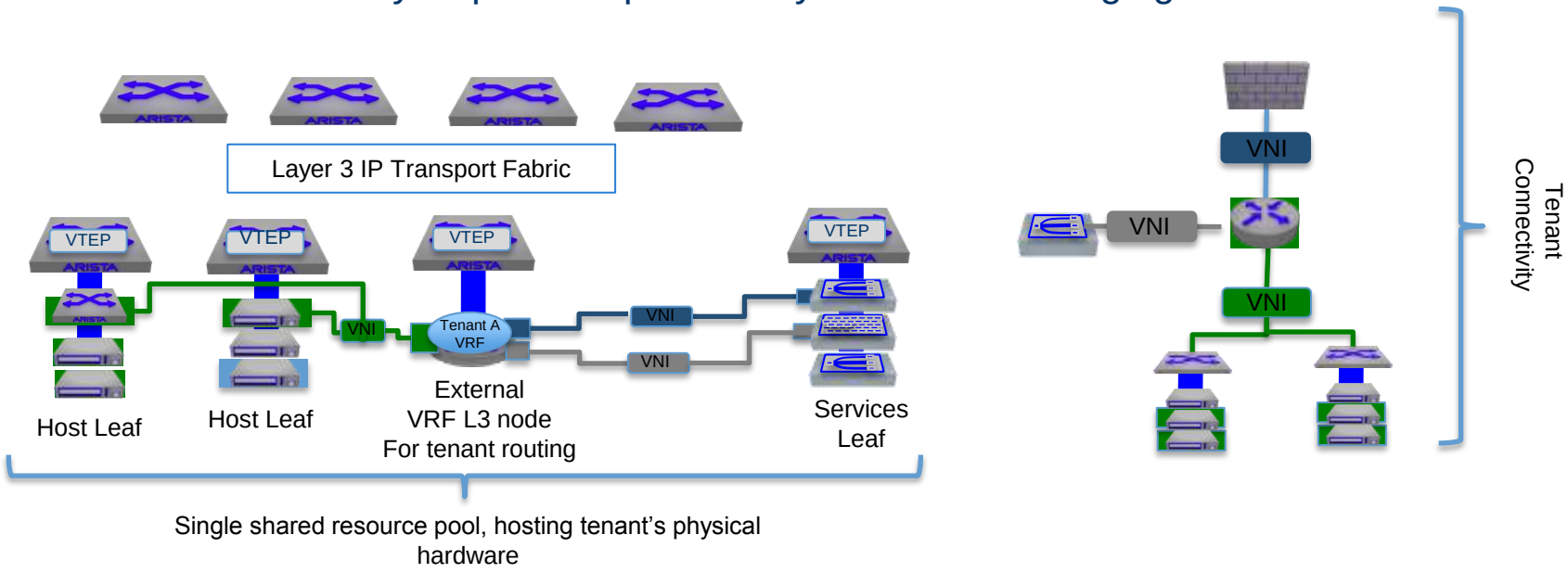

```
interface ethernet10
  switchport
  switchport mode access
  switchport access vlan 10
!
interface ethernet11
  switchport
  switchport mode trunk
  switchport trunk allow vlan 10,20
!
interface ethernet12
  no switchport
  ip address 2.2.2.1/30
```

```
interface vlan 10
  ip address 3.3.3.1/24
!
Interface vlan 20
  ip address 4.4.4.1/24
!
interface loopback0
  ip address 1.1.1.1/32
!
interface vxlan1
  vxlan vlan 10 vni 1000
  vxlan vlan 20 vni 2000
  vxlan source-interface loopback0
!
ip route 0.0.0.0/0 2.2.2.2
```

Bridging & Routing Use Cases

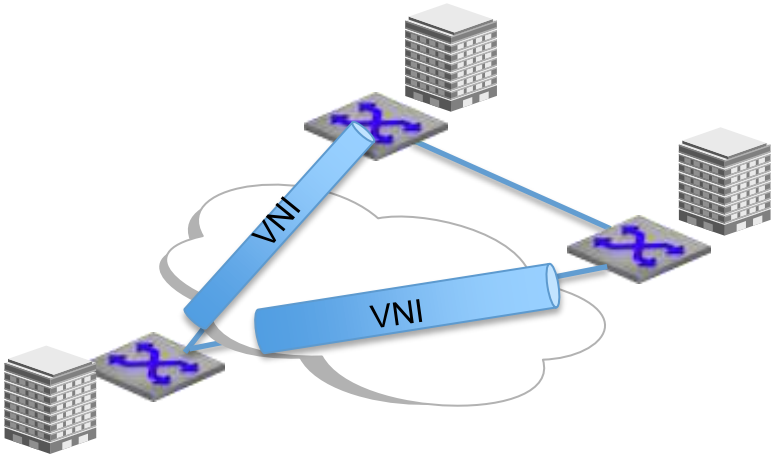
VXLAN- Hosting/Cloud Provider

- **Provider looking to support multiple customers over a shared L3 infrastructure.**
  - Wants the flexibility to deploy tenant resources across racks.
  - Layer 2 (VXLAN bridging) required to stitch the tenant's resources/appliances together across racks .
  - Tenant VRF's required for security or overlapping private IP address space
  - Large scale VRF required, tenant routing achieved using dedicated router
  - Fabric VTEP thus only required to provide layer 2 VXLAN bridging service

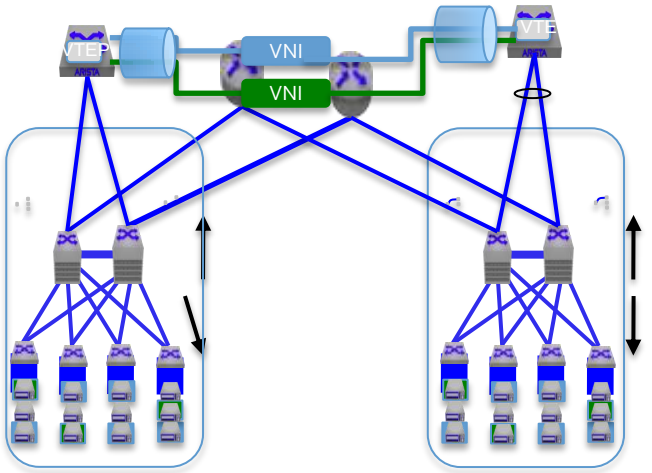


VXLAN- Data Center Interconnect (DCI)

- **Enterprises looking to interconnect DCs across geographically disperse sites**
  - Layer 2 connectivity between sites, providing VM mobility between sites
  - Within the DC for server migration between PODs, for integrating new infrastructure
  - Drop in VXLAN bridging only service, no requirement for VXLAN routing



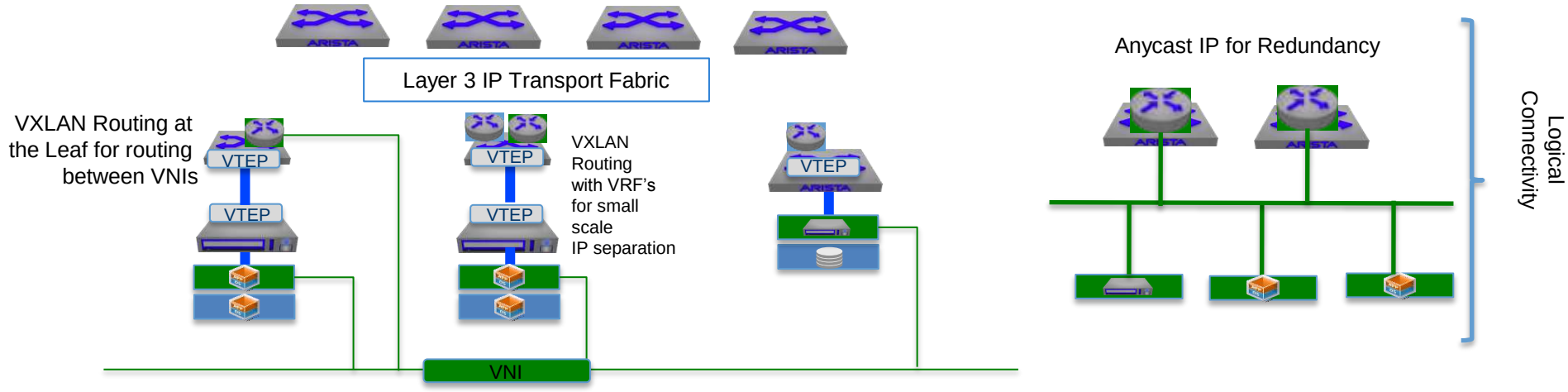
DCI to provide Layer 2 connectivity between geographically disperse sites



Server migration POD interconnect for connectivity between DC's PODs

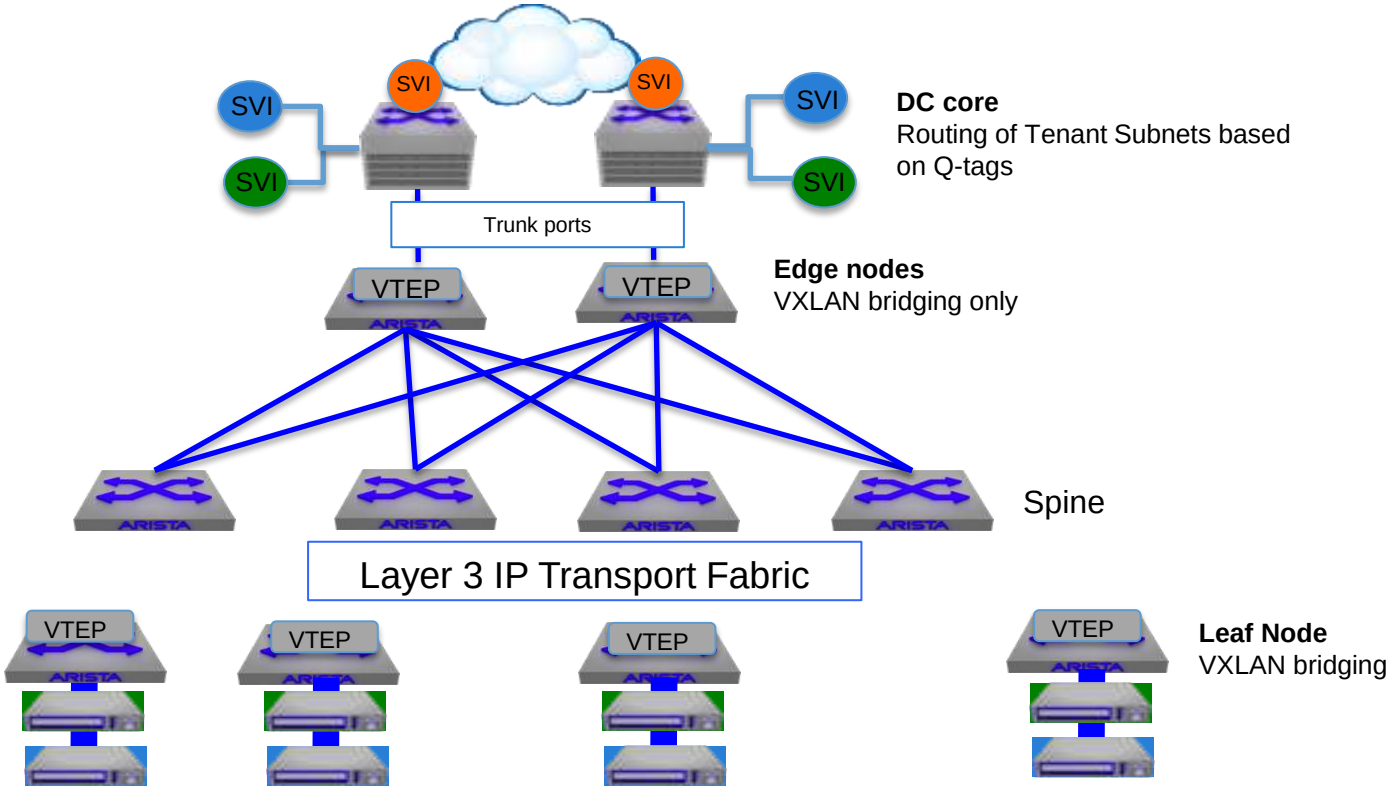
VXLAN – Enterprise Cloud

- **Enterprise Cloud deploying multiple BU's applications across a single shared L3 infrastructure**
  - Virtual Machines and BMS dynamically deployed across available racks
  - VXLAN bridging deployed to provide L2 connectivity across racks
  - VXLAN routing at the leaf layer to provide L3 connectivity between different BU VNIs
  - Single Enterprise so limited need for Layer 3 separation and scaled VRF
  - May need Layer 3 VRF separation for Production and Develops applications (ease migration process)



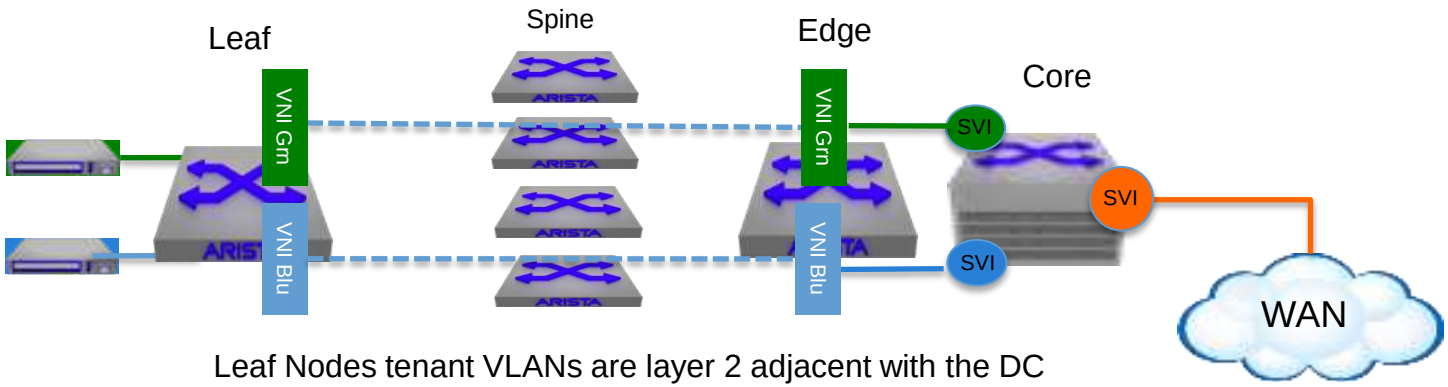
VXLAN Routing Topologies

Centralized Routing



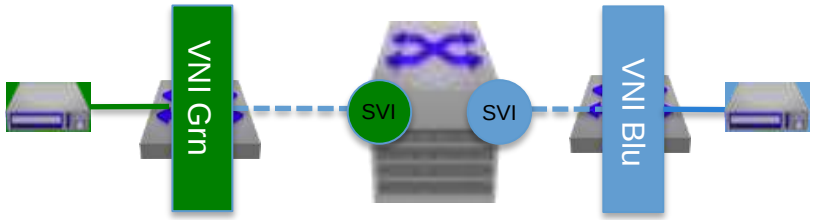
- **Centralized Routing**
  - Leaf Nodes are configured for VXLAN bridging only.
  - The DC core has a VLAN and SVI for each of the tenant subnets – pair for redundancy and a route to the WAN
  - Edge Node provides VXLAN Bridging between the DC core (mapping Q-tags to VNIs) to each leaf VTEP node.
  - Spine nodes are transparent to the tenant overlay networks, only routing the underlay network

Centralized Routing



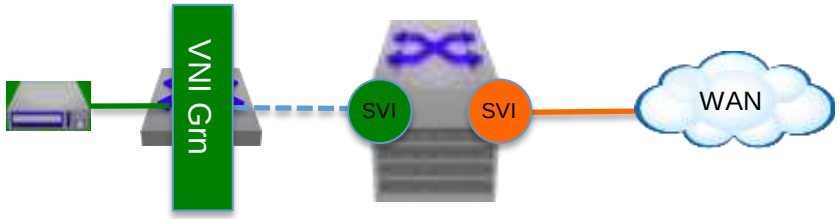
Leaf Nodes tenant VLANs are layer 2 adjacent with the DC Core via VXLAN bridging between the Leaf and Edge nodes. The Spine layer is transparent to tenant VLANs/Subnets

East to West Traffic



SVI on DC Core only
Routing between Tenant subnet occurs on the DC core

North to South Traffic



DC Core is the exit point to the WAN, via peering with the DC border router

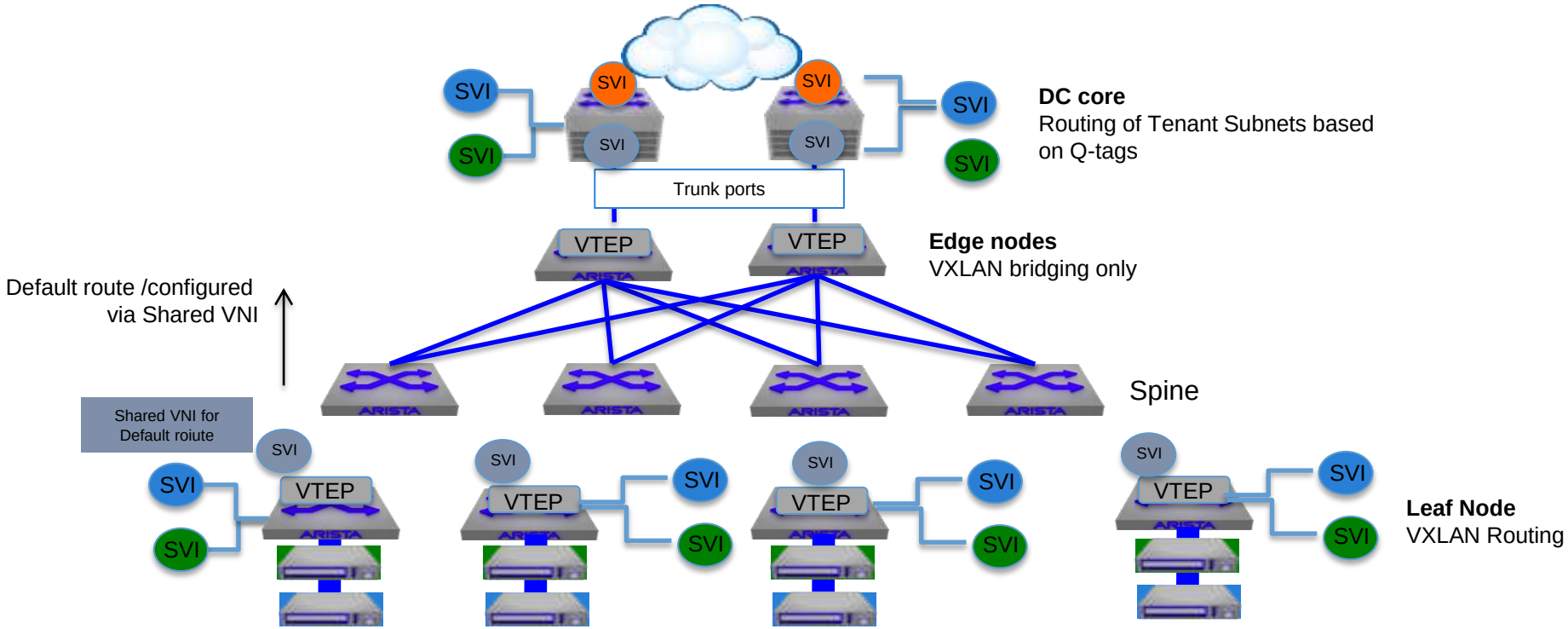
Centralized Routing

- **Routed traffic flow between tenant subnets**
  - Default gateway for all tenant subnets reside on the DC core
  - Traffic is VXLAN bridged at the Leaf to the Edge Node – Spine is routing the outer frame header
  - Edge Node decap the frame and forwards as a Q-tag to the DC core
  - DC Core routes the frame into the Dst VLAN, Dst tenant host learnt on the link to the Edge node.
  - The Edge node maps the Q-tag into and VNI and VXLAN bridges directly to the host's Leaf node where it is VXLAN decap.
- **Traffic Flow between tenant host and external host**
  - Default gateway for all tenant subnets reside on the DC core
  - Traffic VXLAN bridged by the first hop Leaf node to the Edge node and onto the DC core
  - The DC core routes the frame into the WAN.
  - Return traffic from the external host follows the same path
- **Use Case**
  - SP Cloud and Hosting due to the potential to provide Layer 3 tenant separation at scale with VRF's on the the DC core

Centralized Routing

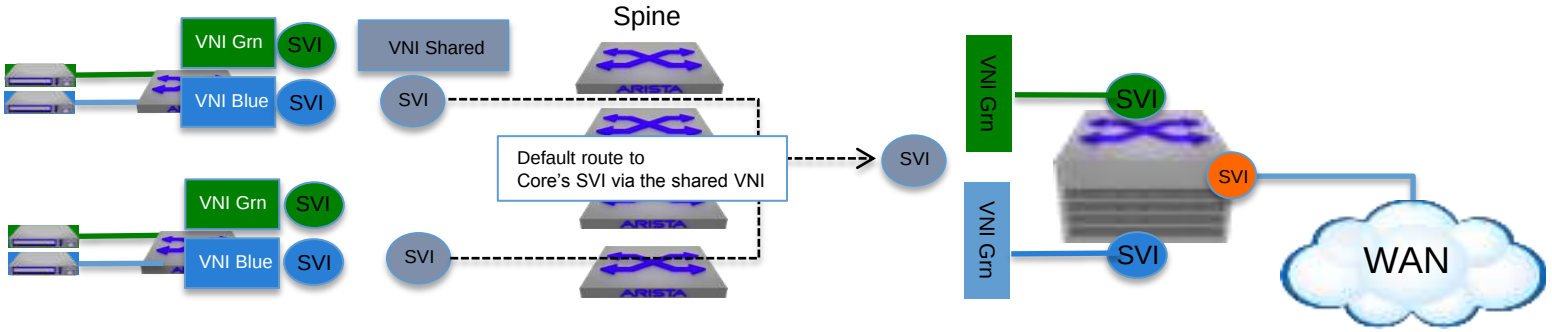
- **Pros**
  - Separation of the Underlay and Overlay networks - Spine is transparent to tenant
  - Leaf + Spine have no SVI or routes for the overlay, therefore do not need to learn tenant host routes / ARPs, significantly increasing scale at the Leaf.
  - Optimal forwarding for North to South traffic – Core VTEP is layer 2 adjacent to all host,
  - Simple design, with predictable forwarding.
  - It's very easy to draw ☺
- **Cons**
  - All routing takes place on a single central point, therefore forwarding bandwidth is limited by the forwarding capacity of a single device/pair of devices.
  - Central point means the DC core device needs to learn all host routes/ARP's for all the devices within the DC.
  - Centralized point means the Edge node need to learn remote-mac's for all tenant hosts in the DC
  - With a single Edge Node pair, would only provide support for 4k VLANs/VNIs
  - Traffic traverses the IP Fabric twice for routing – VXLAN bridged to Core + routed + VXLAN Bridge Dst Host

Direct Routing



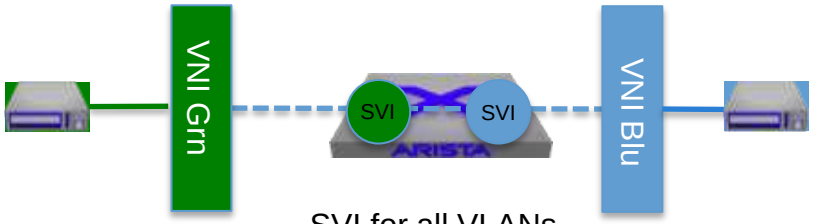
- **Direct Routing**
  - Leaf Nodes are configured with an SVI/VARP in ALL of the tenant subnets
  - The configured VARP address on the leaf acts as the default gateway for the local tenant hosts in the rack
  - The DC core has a VLAN and also a SVI for each of the tenant subnets
  - Edge Node provides VXLAN Bridging between the DC core and the leaf nodes.
  - Leaf nodes are configured with a default route to the DC core for routing traffic out the DC
  - Spine nodes are transparent to the tenant overlay networks

Direct Routing



East to West Traffic

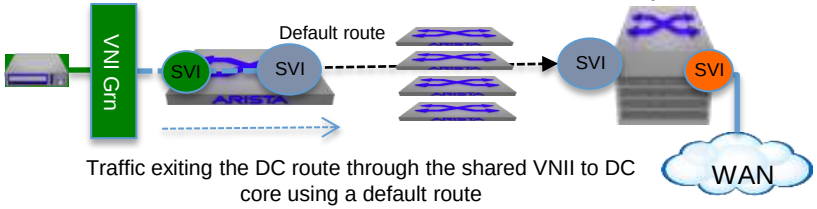
Routing directly at the Leaf Node



SVI for all VLANs
on each leaf node.
Routing between Tenant subnet
occurs at the FH Leaf node

South to North Traffic

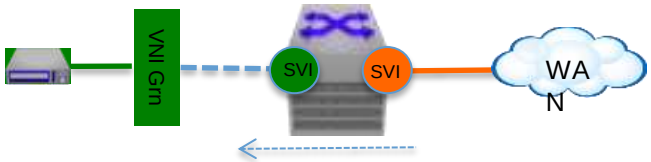
Traffic routed via default route in the underlay



Traffic exiting the DC route through the shared VNII to DC core using a default route

North to South Traffic

Traffic routed through the shared VNI to the DC core



DC Core has an SVI on the local subnets

Traffic routed directly at the core

Direct Routing

- **In the Direct routing model, each VTEP needs to have an SVI in all subnets**
  - This could result in IP address space of the tenant subnet being consumed by SVIs
  - With a /24 subnet and 253 VTEPs, there would be no space left for host address.
- **Potential problems when a VTEP sources a frame with shared SVI-VARP address**
  - For example an SSH session from the switch to a host, the remote VTEP connected to the host would also own the source IP
  - To avoid this when a packet sent from the switch to a remote host, the source IP address is NATed to the highest IP address of the loopback interfaces in the switch.
  - No loopback interface is not present the highest IP address of vlan interfaces

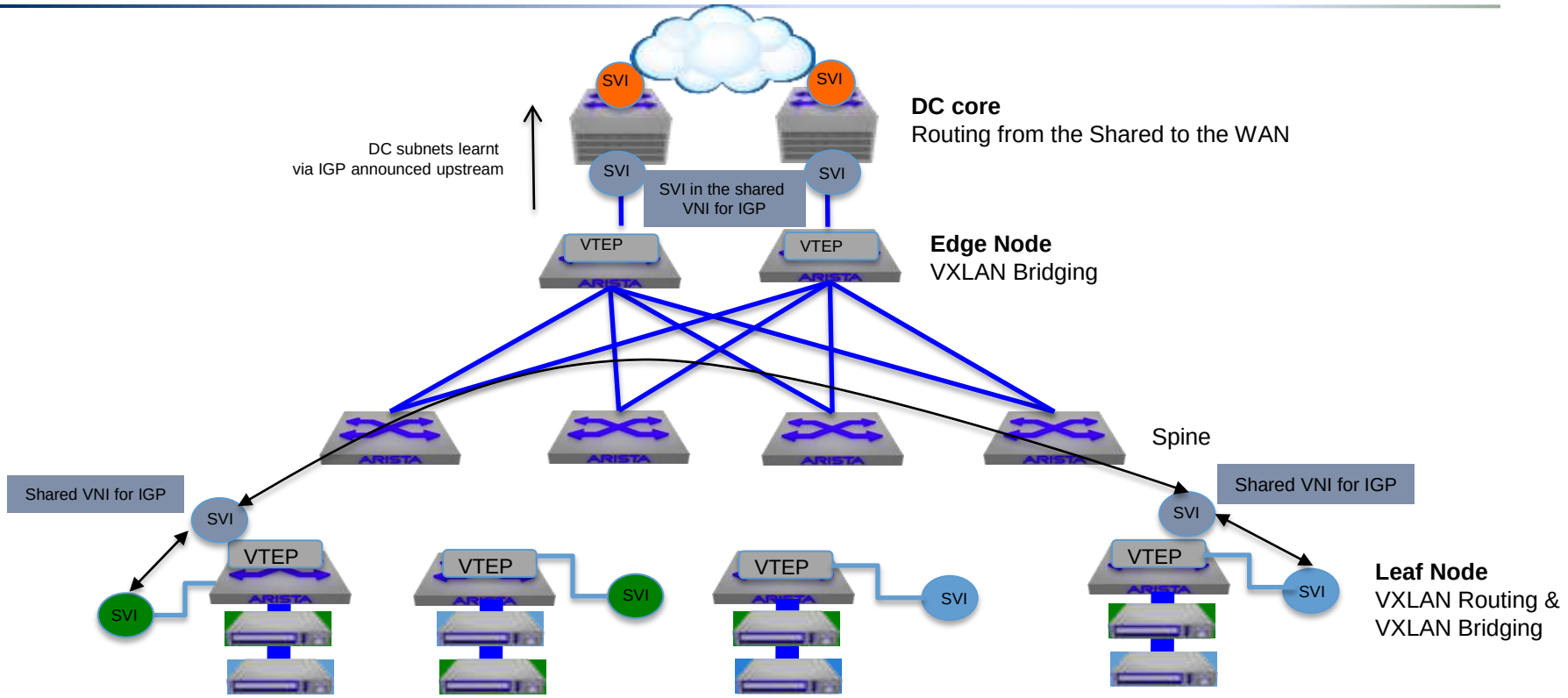
Direct Routing

- **Traffic flow between tenant subnets**
  - Traffic routed into the Dst tenant subnet at the first hop Leaf node
  - Host local then directly switched to the locally attached host
  - Remote host (learnt behind a VTEP), VXLAN bridged across the Spine and de-encapsulated at the remote VTEP
- **Traffic Flow between tenant host and external host (South to North)**
  - Traffic routed by the first hop leaf node to the DC core via the default route in the shared VNI – Spine transparent
- **Traffic Flow between external host and tenant host (North to South)**
  - Traffic routed at the DC core into the tenant subnet, and switched into the host's tenant VLAN on the DC core.
  - The VLAN is mapped to a VNI on the Edge node and VXLAN bridged directly to the host's Leaf node for VXLAN decap
- **Use Case**
  - Enterprise Cloud as tenant routing is being done on the Leaf Nodes level of Layer 3 tenant separation is limited – Dev/Test/Prod VRFs probable all that is required

Direct Routing

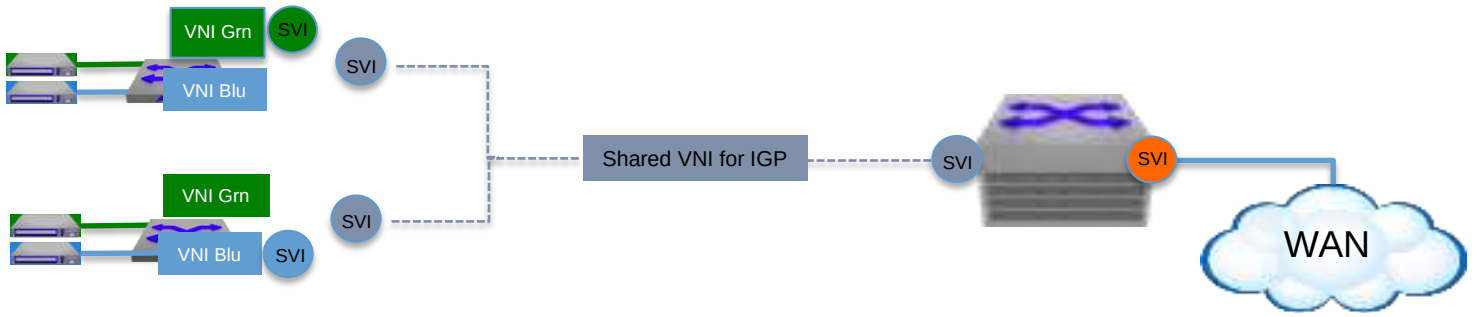
- **Pros**
  - Retains full separation of the Underlay and Overlay networks.
  - Traffic routed between two hosts within the same rack always routed locally by the FH leaf
  - Traffic routed between racks follows the optimal path, routed at the FH leaf and VXLAN bridged to the remote host
  - North to South traffic, is always bridged directly to the host as the DC Core is layer 2 adjacent (via VXLAN) with all hosts
- **Cons**
  - Every Leaf node and the Core switches require an SVI for all tenant subnets
  - In addition to learning all MAC addresses in the VXLAN, Leaf switches also need to learn all ARP/Host routes.
  - As all devices learn all L2 and L3 state the size of the deployment is limited by the lowest common denominator (typically the Leaf node)
  - Traffic routing is asymmetric when exiting and entering the Data Center – exiting the Date uses the Default route path

Indirect Routing



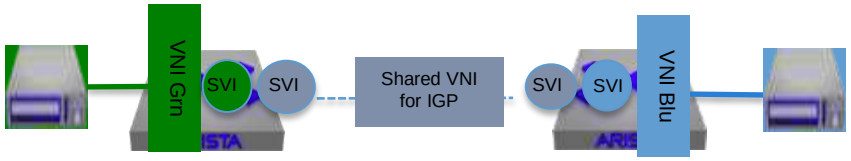
- **Indirect Routing**
  - Leaf nodes are configured with SVIs for a subset of the tenant subnets, SVI's deployed in pairs for redundancy
  - All Leaf nodes are members of a shared VNI, which runs an IGP
  - The shared VNI is used to learn the tenant subnets of neighboring Leafs and routes for external connectivity.
  - The DC core has an SVI in the shared VLAN/VNI only
  - Edge Node provides VXLAN Bridging between the DC core and the leaf nodes within the shared VNI

Indirect Routing



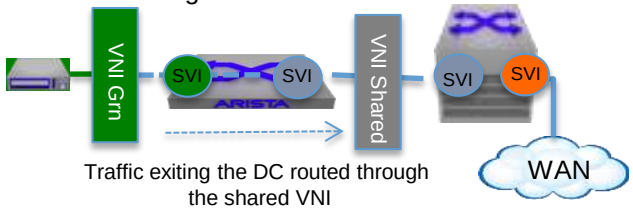
East to West Traffic

Non local subnets learned via the IGP and routed through the shared VNI



South to North Traffic

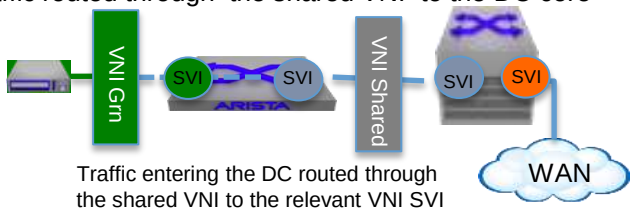
Traffic routed through the shared VNI to the DC core



Traffic exiting the DC routed through the shared VNI

North to South Traffic

Traffic routed through the shared VNI to the DC core



Traffic entering the DC routed through the shared VNI to the relevant VNI SVI

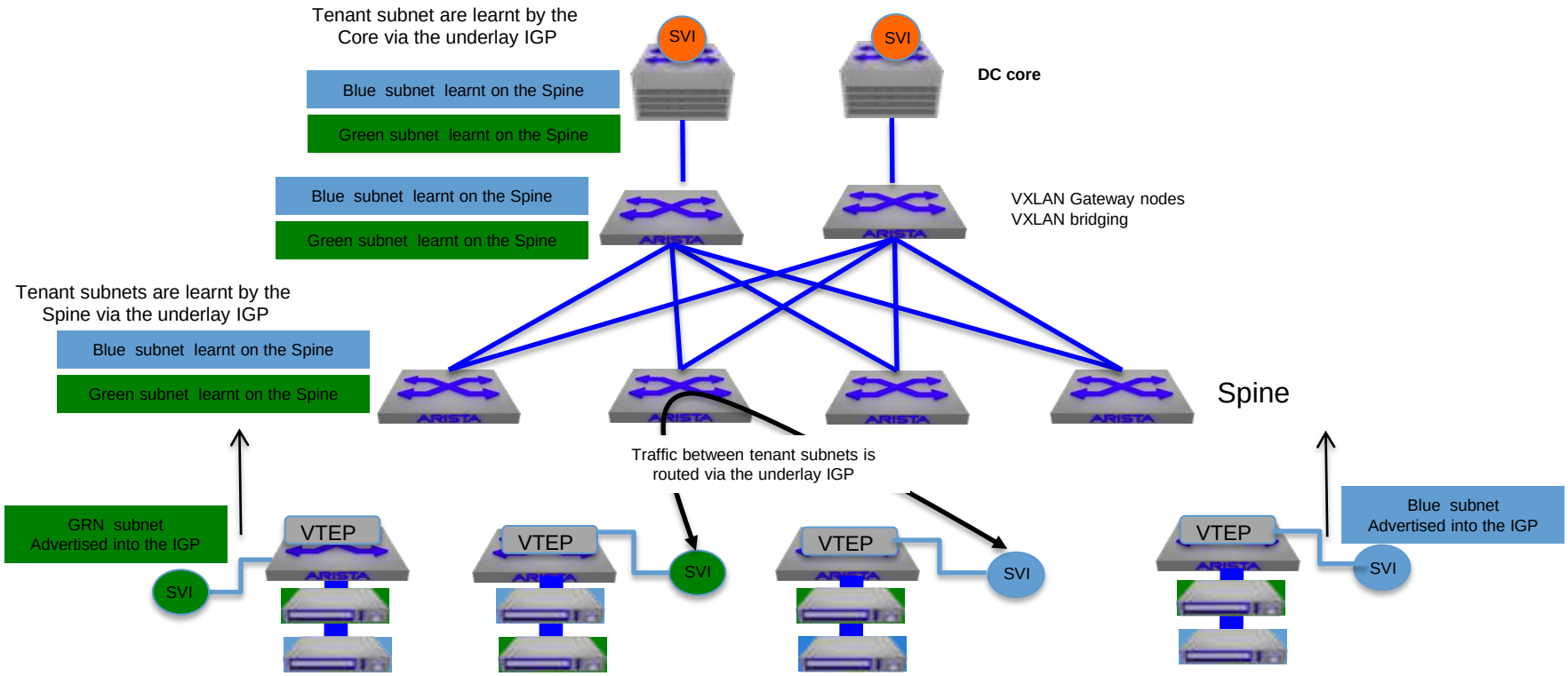
Indirect Routing

- **Traffic flow between tenant subnets - DFG on local FH leaf**
  - Routed at the first hop leaf into the share VNI
  - VXLAN bridged across the shared VNI to the VTEP announcing the Dst tenant subnet
  - Remote VTEP, VXLAN routes the frame into the tenant subnet and switches it local or VXLAN bridges if the host is remote.
- **Traffic flow between tenant subnets – DFG not local FH leaf**
  - Traffic would first be VXLAN bridged to the VTEP owning the DFG for the tenant subnet.
- **Traffic Flow between tenant host and external host (South to North)**
  - Traffic routed by the VTEP owning the SVI for the host's tenant subnet into the shared VNI
  - Bridged via the Shared VNI to the DC core for routing into the WAN
- **Traffic Flow between external host and tenant host (North to South)**
  - Traffic routed at the DC core into the shared VLAN/ VNI
  - Edge Node then VXLAN bridges to the VTEP owning the SVI for the host via the shared VNI
- **Use Case**
  - Enterprise Cloud as tenant routing is being done on the Leaf Nodes level of Layer 3 tenant separation is limited – Dev/Test/Prod VRFs probable all that is required

Indirect Routing

- **Pros**
  - Retains full separation of the Underlay and Overlay networks.
  - Reduces the requirement on the Leaf nodes, improving the overall scale
  - Symmetrical routing for North to South traffic via the shared VNI
- **Cons**
  - Sub optimal routing and non-deterministic forwarding for tenant to tenant routing.
  - Tenant to tenant traffic will in the majority of cases, traverse the spine multiple times even for host within a single rack

Naked Routing



- **Naked Routing**
  - Leaf nodes are configured with SVIs for a subset of the tenant subnets, SVI's deployed in pairs for redundancy
  - Leaf nodes learn and announce tenant subnets to neighboring Leafs via the underlay IGP
  - The DC core announces external connectivity to the Leaf nodes via the underlay IGP

Naked Routing

- **Traffic flow between tenant subnets - DFG on local FH leaf**
  - Routed at the first hop leaf into the underlay, which has a next-hop for remote tenant subnet
  - Traffic routed naked via the Spine which is the next-hop to the remote Leaf Node
  - Remote Leaf if host is local, switches traffic to the host, if remote VXLAN encaps the frame to the remote VTEP for the host
- **Traffic flow between tenant subnets – DFG not local FH leaf**
  - Traffic would first be VXLAN bridged to the VTEP owning the DFG for the tenant subnet and then follow the above behavior
- **Traffic Flow between tenant host and external host (South to North)**
  - Traffic routed by the VTEP owning the SVI for the host's tenant subnet, into the underlay and routed naked to ta Spine switch which would be the next-hop
- **Traffic Flow between external host and tenant host (North to South)**
  - Traffic routed at the DC core into underlay and forwarded to next-hop Spine switch
  - Spine switch forwards to the Leaf announcing the SVI for the tenant subnet.
- **Use Case**
  - Enterprise Cloud as the Spine is involved in the tenant's routing and it is no longer transparent.

Naked Routing

- **Pros**
  - North to South traffic is simplified and doesn't' require an Edge VTEP
  - Reduces the number of routing adjacencies in comparison to the indirect routing model
- **Cons**
  - The underlay is not transparent to the overlay network. All routes in the overlay network are now visible to the Spine/underlay
  - As tenant subnets scale, the routing table of the Spine nodes also need to scale.
  - Limited support for Layer 3 tenant separation or overlapping tenant IP addresses.

Questions?