We’re ready.
Are you?

Evolution of Network Overlays in Data Centre Clouds

Victor Moreno, Distinguished Engineer

Cisco live!

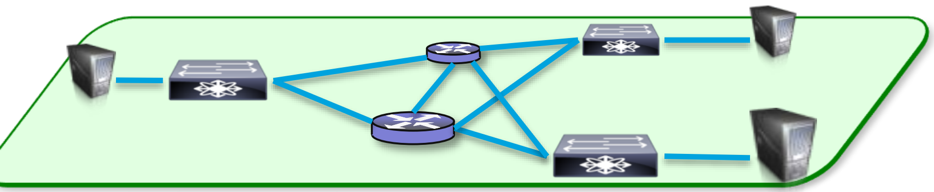
Agenda

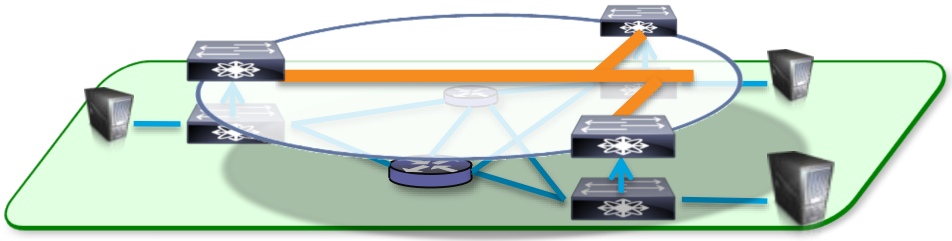
- Overlay Foundational Principles and evolution
- Mapping overlay technologies to the network
- The role of the underlay
- Management and orchestration

Foundational Principles of Network Overlays

Why Overlays?

Seek well integrated best in class Overlays and Underlays





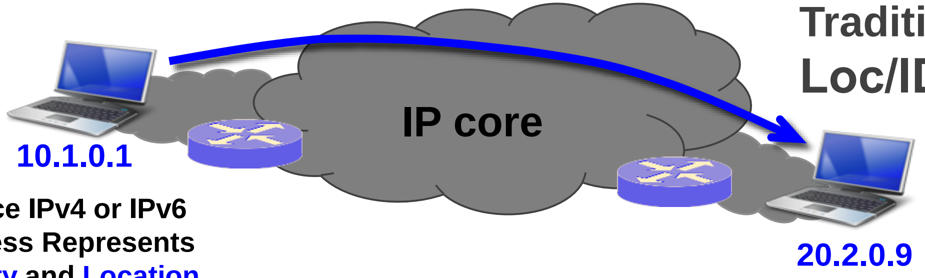
Robust Underlay/Fabric

- High Capacity Resilient Fabric
- Intelligent Packet Handling
- Programmable & Manageable

Flexible Overlay Virtual Network

- Mobility – Track end-point attach at edges
- Scale – Reduce core state
  - Distribute and partition state to network edge
- Flexibility/Programmability
  - Reduced number of touch points

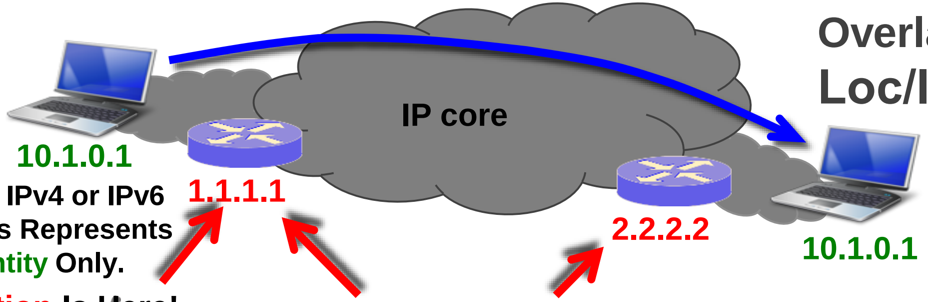
Seminal Idea: Location and Identity Separation



Device IPv4 or IPv6 Address Represents Identity and Location

Traditional Behaviour
Loc/ID “Overloaded” Semantic

When the Device Moves, It Gets a New IPv4 or IPv6 Address for Its New Identity and Location



Device IPv4 or IPv6 Address Represents Identity Only.

Its Location Is Here!

Only the Location Changes

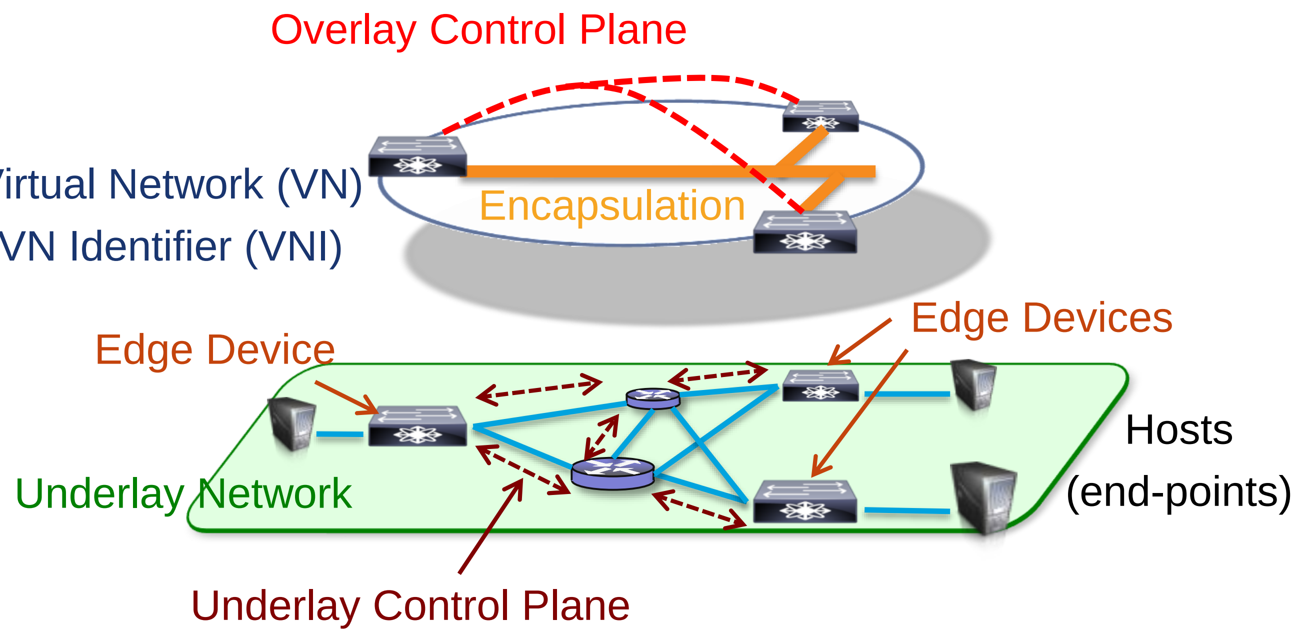
Overlay Behaviour
Loc/ID “Split”

When the Device Moves, Keeps Its IPv4 or IPv6 Address.
It Has the Same Identity

Overlay Taxonomy

Service = Virtual Network (VN)

Identifier = VN Identifier (VNI)



Overlay Attributes

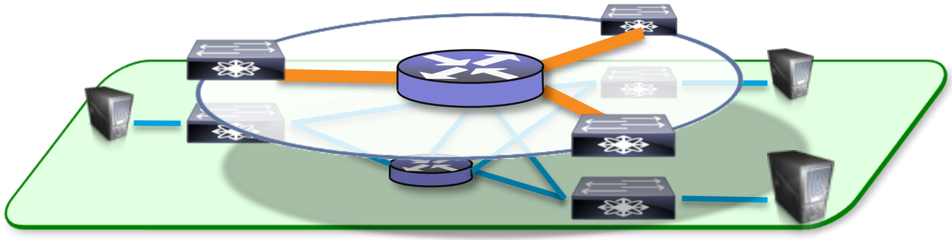
Service	Edge Device	Signalling
Layer 2 Service Layer 3 Service	Host Overlays Network Overlays	Data Plane Learning Control Plane Learning

Overlay Service Type Evolution

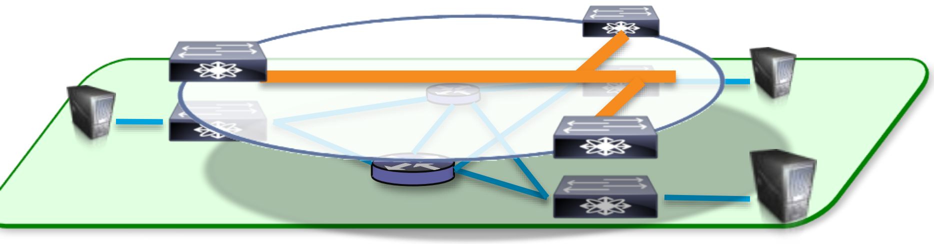
Service

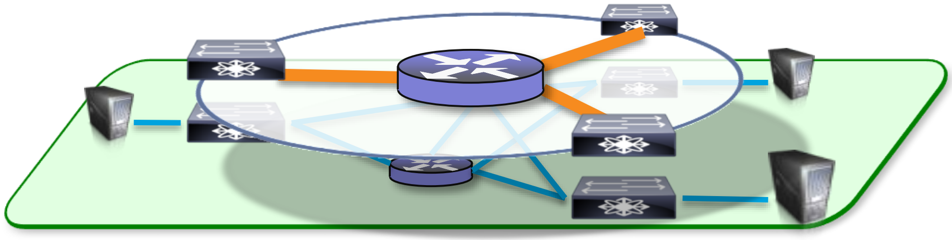
Layer 2 Service

Layer 3 Service



Types of Overlay Service





Layer 2 Overlays

- Emulate a LAN segment
- Transport Ethernet Frames (IP and non-IP)
- Single subnet mobility (L2 domain)
- Exposure to open L2 flooding
- Useful in emulating physical topologies

Layer 3 Overlays

- Abstract IP based connectivity
- Transport IP Packets
- Full mobility regardless of subnets
- Contain network related failures (floods)
- Useful in abstracting connectivity and policy

Hybrid L2/L3 Overlays offer the best of both domains

Layer 2 Overlay Considerations

Solved with …

- **Scale** of the edge devices → Layer 3 Overlays
  - L2 addresses in Ethernet (MACs) use a flat space which cannot be summarised
- **L2/L3 boundary** scaling → Layer 3 Overlays
  - Large L2 domains require a large capacity L3 gateway to handle large ARP and MAC tables at a frequent rate of refresh
- **Multi-homing** sites can induce loops in the network → Network Overlays
- **Flooding** of L2 protocols, unknown unicasts and broadcast in general can propagate failures across the entire L2 domain → MAC routing

Multi-homing in L2 Overlays

Source learning assumes single attached sites

But network overlays involve edge resiliency

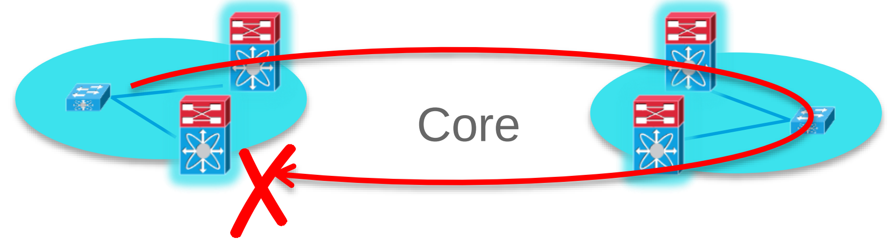
Enhancements are required to address:

- Loop resolution
- Multi-pathing
- Broadcast/Multicast de-duplication

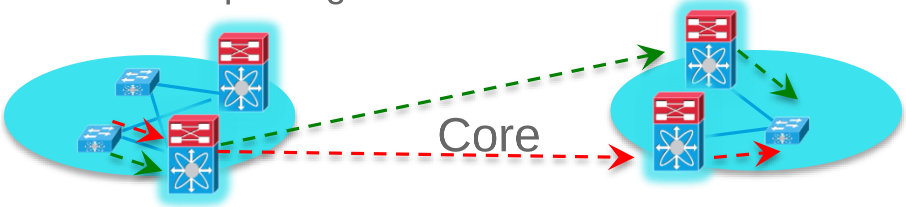
Two Approaches:

- Active-Standby (Data Plane or Control Plane)
  - One active device per VLAN (single attached site)
  - VLAN based load balancing
- Active-Active (Control Plane only)
  - One active device for multi-destination traffic
  - Intra-VLAN load balancing for unicast

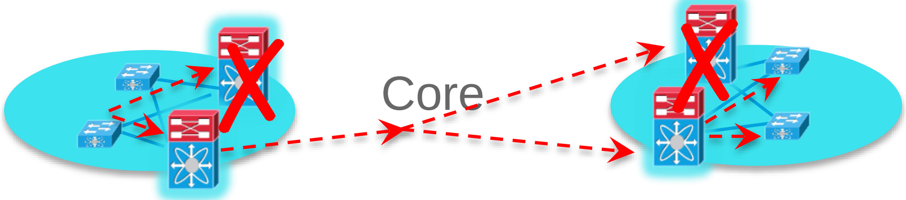
- Loop resolution



- Multi-pathing



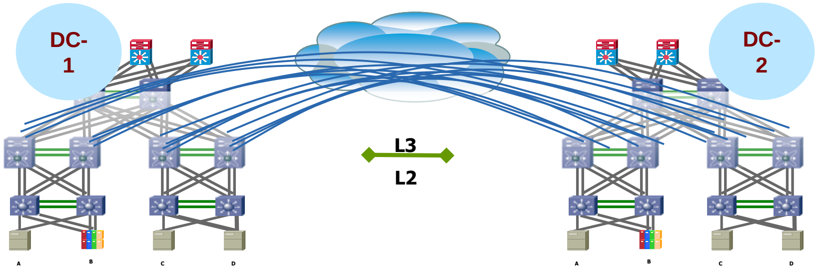
- Broadcast/Multicast de-duplication



Flooding in L2 Overlays

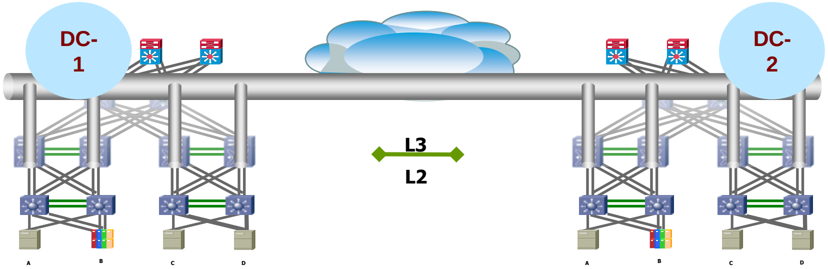
- Control Plane Signalling eliminates the need for floods

Data Plane Learning



- Pre-set flood facility
- MAC learning based on flooding
- Flood L2 protocols and unknown unicast
  - ➔ Failure propagation
- Fail Open
- Suitable for small domains (failure scope)

Control Protocol



- No predetermined flood tree
- MAC learning by control protocol
  - ➔ Contain Failures and L2 protocols
  - ➔ Rich information
- Fail Closed
- Better suited for broad scope

Flooded L2 Overlays

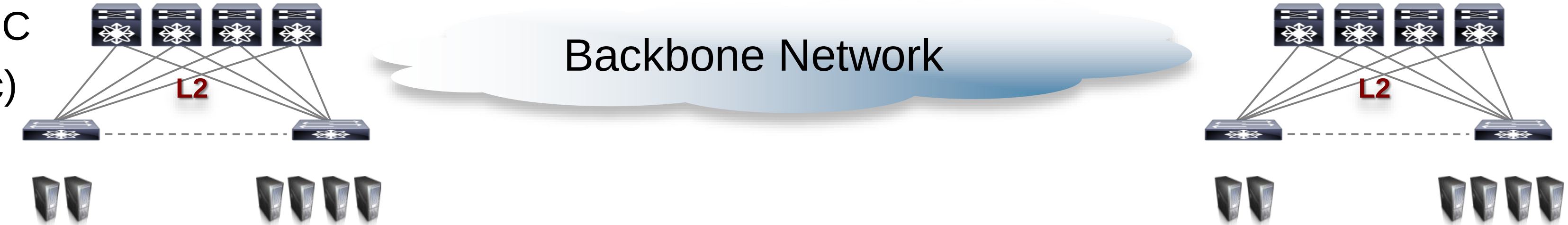
MAC Routing

L2 Overlay Evolution

Inter-DC (DCI)

	VPLS	OTV / EVPN
Underlay Control Plane	MPLS	IP or MPLS
Overlay Control Plane	Flood and Learn	IS-IS / BGP
Encapsulation	MAC in MPLS	MAC in IP
Locator	MPLS PE	NV Edge IP

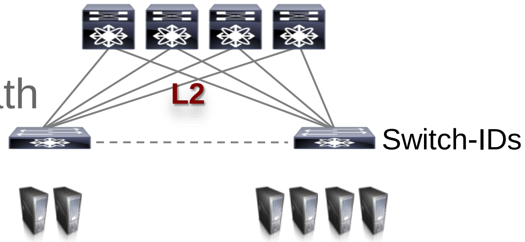
Intra-DC (Fabric)



	Fabric Path	VXLAN ➔ EVPN
Underlay Control Plane	IS-IS	Any IP routing protocol
Overlay Control Plane	Flood and Learn	Flood and Learn ➔ BGP
Encapsulation	MAC in MAC	MAC in IP
Locator	Access Switch-ID	Access IP

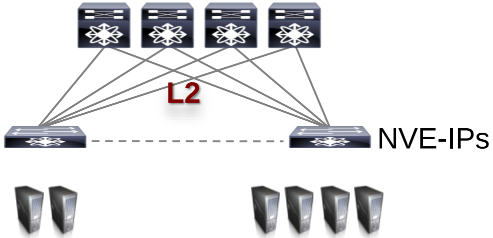
L2 Overlay Flood/Learn Implementations

Fabric Path



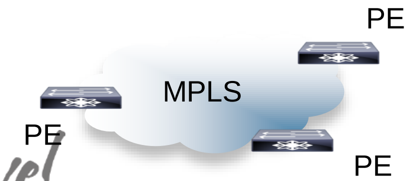
1. Underlay Control Plane: IS-IS calculates all possible paths between switch-IDs (Locators)
2. IS-IS calculates a multicast distribution tree for floods
3. BUM traffic flooded over multicast tree
4. Locators for each host learnt by gleaning Floods

VXLAN



1. Underlay Control Plane: IP calculates all possible paths between NVE-IPs (Locators)
2. IP multicast distribution tree for floods
3. BUM traffic flooded over multicast tree
4. Locators for each host learnt by gleaning Floods

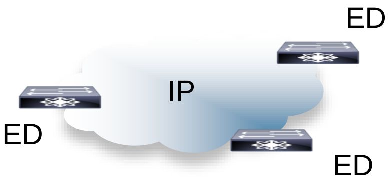
VPLS



1. Underlay Control Plane: MPLS calculates all possible LSPs between PEs
2. Pre-determined group of pseudo-wires for flooding
3. BUM traffic flooded over multicast tree
4. PEs for each host gleaned from Floods

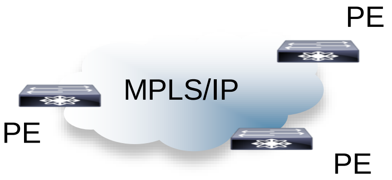
L2 Overlay Control Plane Implementations

OTV



1. Underlay Control Plane: IP calculates all possible paths between Edge Devices (Locators)
2. Overlay Control Plane: IS-IS adjacencies amongst Edge Devices
3. Locators for each host advertised in IS-IS
4. No Floods, integrated multi-homing

EVPN



1. Underlay Control Plane: MPLS calculates all possible LSPs between PEs or IP underlay
2. Overlay Control Plane: BGP adjacencies amongst Edge Devices
3. Locators for each host advertised in BGP
4. No Floods, integrated multi-homing

Layer 3 Overlay Considerations

Addressed with …

- **Scale** of the edge devices
  - Can be improved further by using an on-demand pull model

  → On-demand Pull

- **IP Mobility** for subnet disaggregation
  - Members of a subnet may be distributed across locations
  - Any host anywhere

  → Layer 2 Semantics with IP routing

- **Broadcast & Link-local multicast** traffic to be handled as a special case
  - Potentially without even learning MAC addresses

  → Combined L2/L3 overlay

L3 Overlay Evolution

- Edge Device Scale

Push Protocol Model

- IP/BGP MPLS VPNs are highly scalable today
- PE routers must:
  - Hold a large number of prefixes
  - Maintain multiple routing protocol adjacencies
- Mobility and cloud will add pressure in terms of:
  - Prefix granularity and volume
  - Increased number of PEs

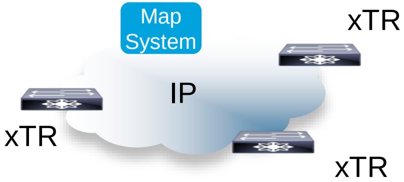
Pull Protocol (on-demand) Model

- LISP deployments and footprint are increasing rapidly
- On-demand caching models ease the requirements on the edge devices:
  - Only prefixes being utilised are cached
  - No routing adjacencies are maintained
- A pull model is expected to provide global scalability to enable pervasive cloud models

L3 Overlay Implementations

LISP

(pull)



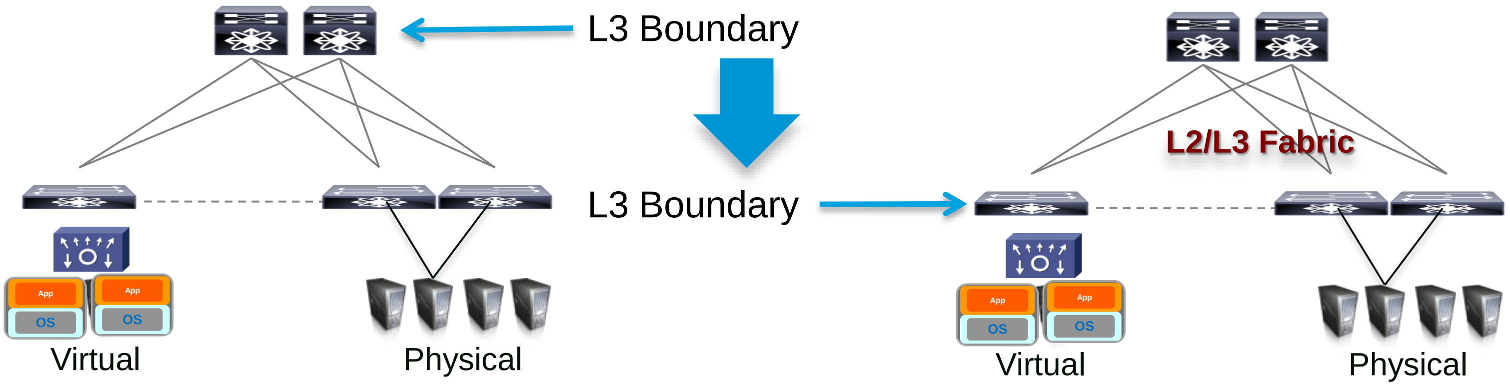
1. Underlay Control Plane: IP calculates all possible paths between Edge Devices (Locators)
2. Overlay Control Plane: All mappings registered with Mapping System by xTRs
3. xTRs “pull” mappings on demand

BGP VPNs

(push)

1. Underlay Control Plane: MPLS calculates all possible LSPs between PEs or IP Multipath Routing
2. Overlay Control Plane: BGP adjacencies amongst PEs
3. Locators for each host pushed in BGP to all PEs

Distributed Gateway Function in L3 Overlays



Traditional L2 - centralised L2/L3 boundary

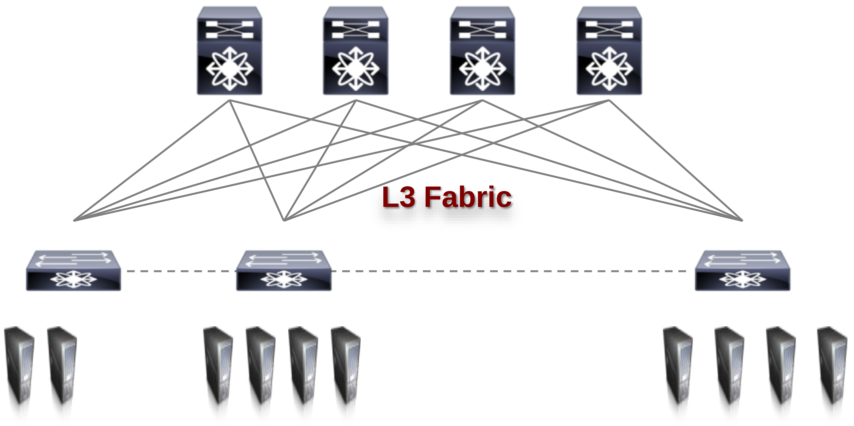
- Always bridge, route only at an aggregation point
- Large amounts of state converge
- Scale problem for large# of L2 segments
- Traditional L2 and L2 overlays

L2/L3 fabric (or overlay)

- Always route (at the leaves), bridge when necessary
- Distribute and disaggregate necessary state
- Optimal scalability
- Enhanced forwarding and L3 overlays

IP Mobility with L3 Overlays

- Granular location information (host routes)
  - Allow subnet members to move anywhere
- Layer 2 semantics
  - ARP proxy
  - Consistent default Gateway presence
- L3 at the Access
  - Access switch replies to all ARPs with the same MAC address
  - Host routing for all traffic within the fabric
  - Summary prefix outside the fabric

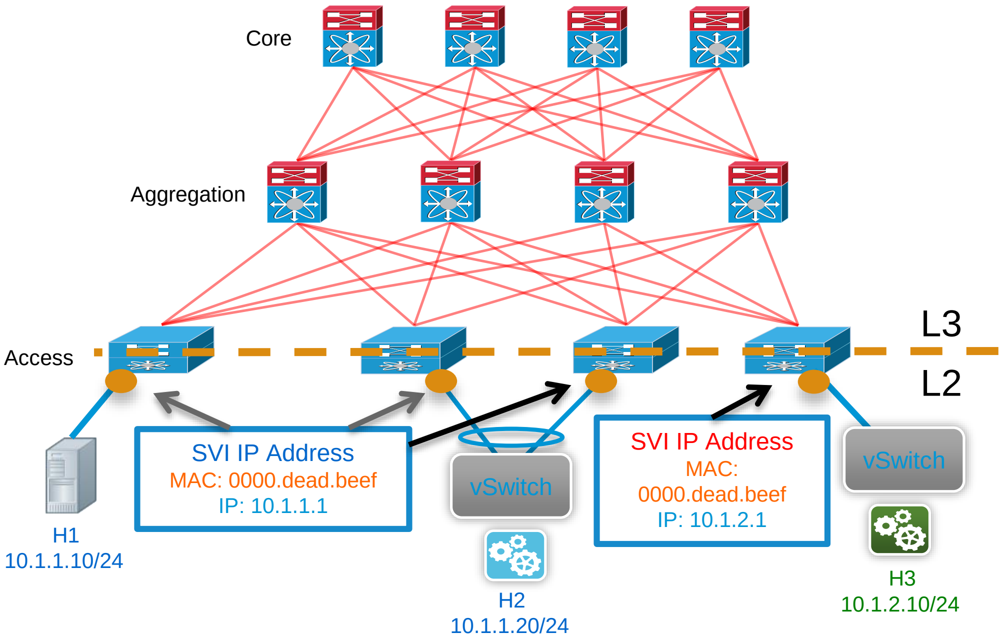


Cisco live!

L3 Overlay First Hop Routing

Routing on the Leaf Nodes

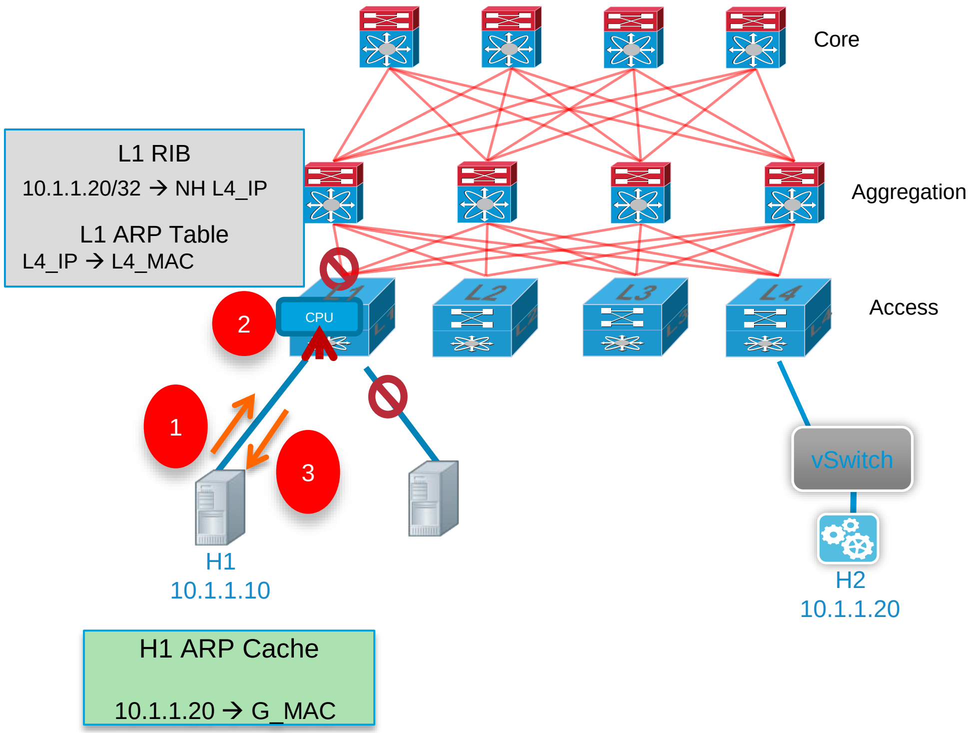
- A leaf switch is assigned an IP address and a gateway MAC address for each locally defined subnet with a connected host → IP address of the SVIs
- The same anycast IP address is assigned to all leaves supporting attached hosts in the same subnet
- The same gateway MAC address can be used across all subnets supported on all the leaves



L3 Overlays – ARP and Intra-subnet Forwarding

ARP Handling

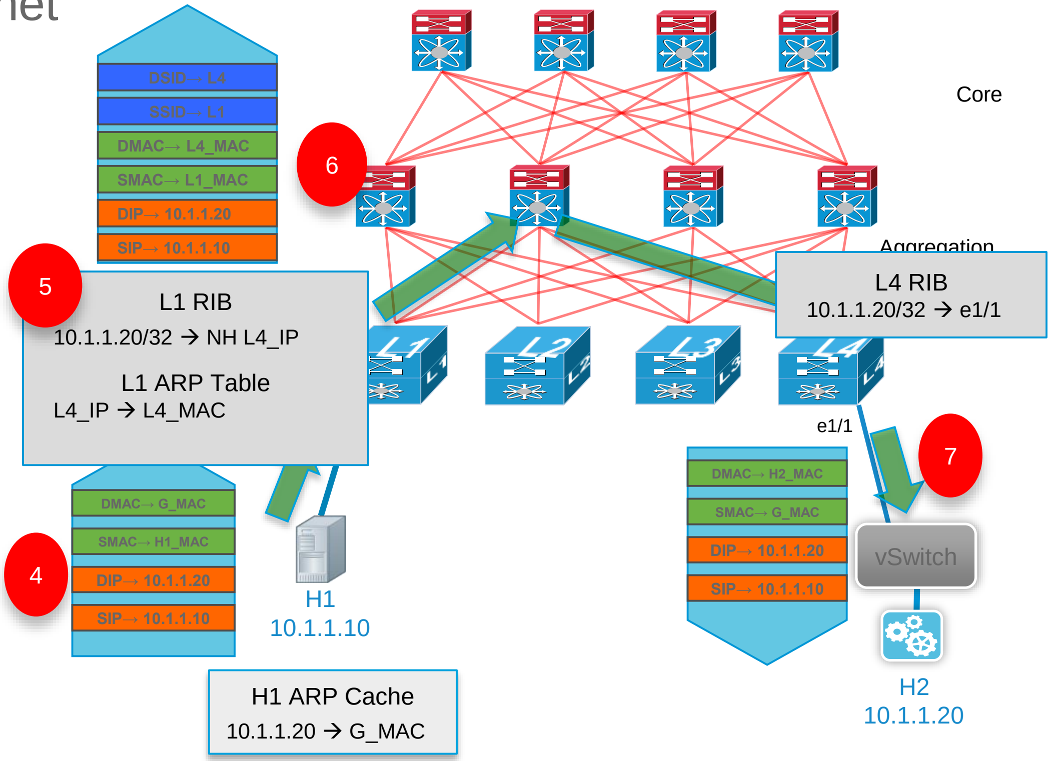
1. H1 sends an ARP request for H2 – 10.10.10.20
2. The ARP request is intercepted at the leaf L1 and punted to the Sup
3. A few options:
    1. If L1 has a valid route to H2, L1 may ARP reply with its own G_MAC
    2. If L1 has a MAC-IP binding for H2, L1 may ARP-reply on behalf of H2 with H2's MAC
    3. L1 may unicast the ARP request to the leaf where H2 is attached
    4. L1 may simply flood the ARP request



L3 Overlays – ARP and Intra-subnet Forwarding

IP Forwarding within the Same Subnet

- If H1 generates a data packet destined to G_MAC, then a MAC re-write, TTL decrement and host IP forwarding takes place
- If H1 generates a data packet destined to H2_MAC, then overlay forwarding can be done without TTL decrement based on either H2_MAC or H2_IP depending on the overlay implementation.



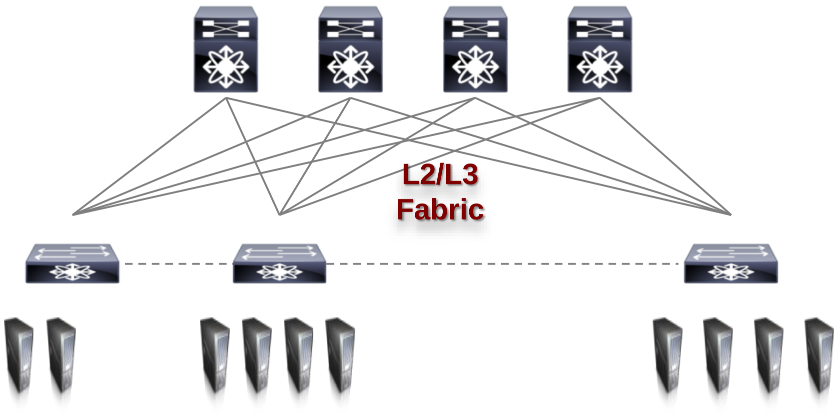
Combined L2/L3 Overlays

Enhanced Forwarding Mode:

- Route all IP traffic including Intra-subnet
- Bridge only:
  - Non-IP / Broadcast / Link-local multicast
- Assumption is that most traffic is IP

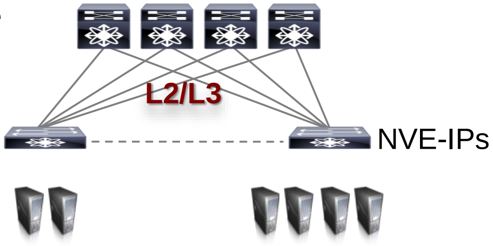
Traditional Forwarding Mode:

- Route inter-subnet traffic
- Bridge intra-subnet and non-IP traffic



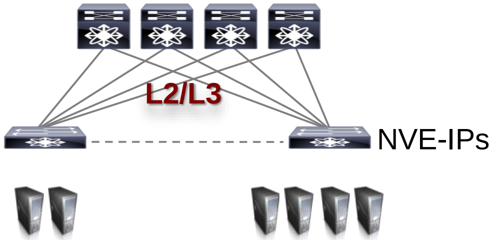
Combined L2/L3 Overlay Service Implementations

Programmable Fabric with VXLAN EVPN



1. Underlay Control Plane: IP calculates all possible paths between NVE-IPs (Locators)
2. L2+L3: MP-BGP advertisement of host locations
3. Route inter-subnet, bridge intra-subnet

Application Centric Infrastructure



1. Underlay Control Plane: IP calculates all possible paths between NVE-IPs (Locators)
2. Overlay Control Plane: Demand protocol
    1. Register both IP and MACs for every host
    2. Leaf nodes “pull” IP and/or MAC mappings on demand
3. Forward on L3 information unless data is non-IP

26

Overlay Edge Device and Data Plane Evolution

Service

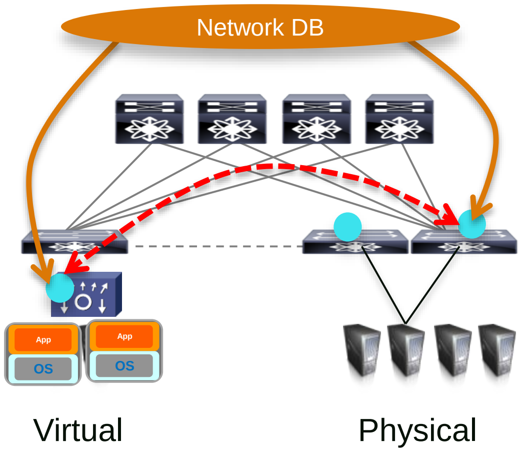
Layer 2 Service

Layer 3 Service

Edge Device

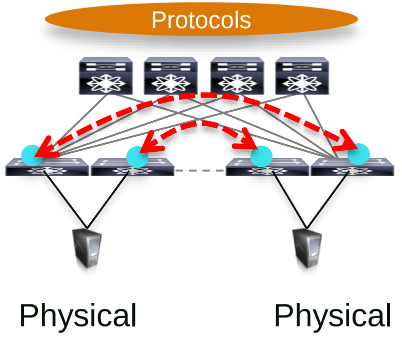
Host Overlays

Network Overlays



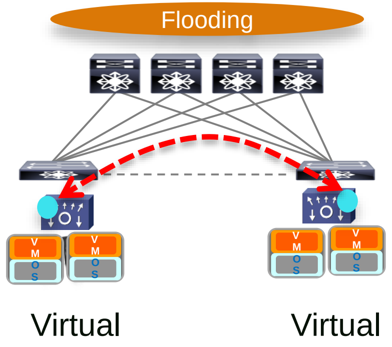
Overlay Network Evolution: Edge Devices

Network Overlays



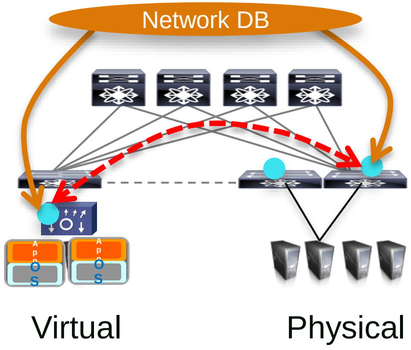
- Router/switch end-points
- Protocols for resiliency/loops
- Traditional VPNs
- OTV, VPLS, LISP, FP

Host Overlays



- Virtual end-points only
- Single admin domain
- VXLAN, NVGRE, STT

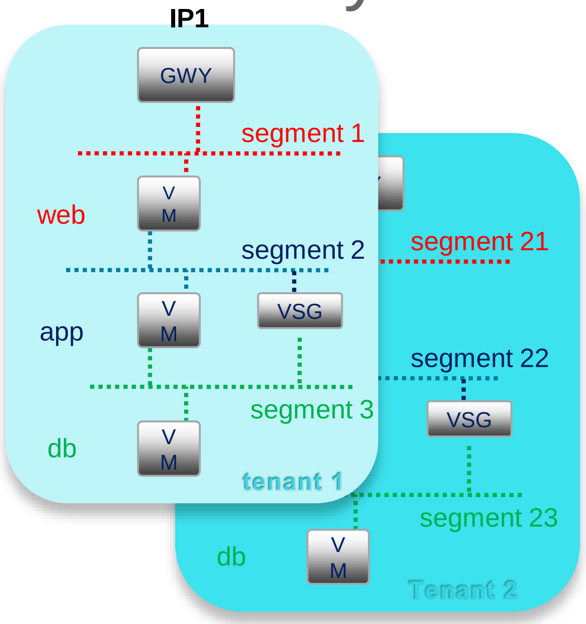
Hybrid Overlays



- Physical and Virtual
- Resiliency + Scale
- x-organisations/federation
- Open Standards

Tunnel End-points

Host Overlays



Multi-tier Virtual App = VMs + vSegments + GWY

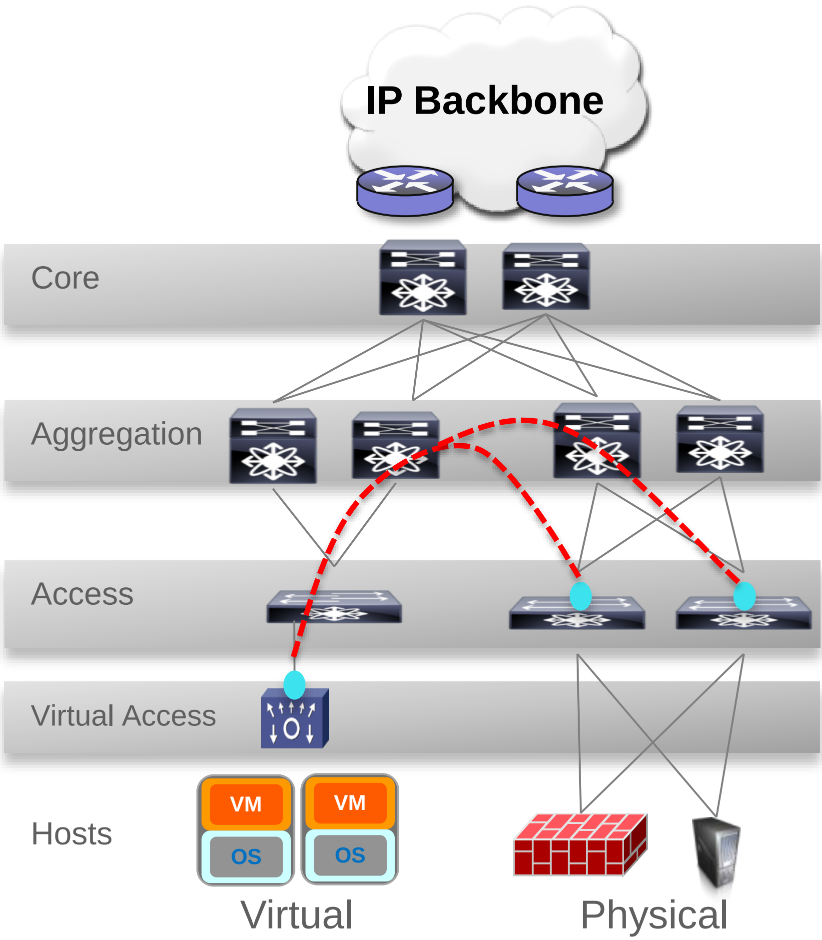
Application: Cloud Services

Elastic creation of **virtual Segments**

- Mobile: Can be instantiated anywhere
  - Move along with VMs as necessary
- Very large number of segments
  - Do not consume resources in the network core
- Isolated, not reachable from the IP network
  - Front-end segment must be handled by the fabric
- Host overlays are initiated at the hypervisor virtual switch ➔ Virtual hosts only
- GWY to connect to the non-virtualised world
- Variants: VXLAN, NVGRE, STT

Hybrid Overlays

- Hypervisors introduce an additional tier in the network: The virtual Access (virtual Switch)
- **VMs** connect to the virtual Access
  - **Host overlays** start at the virtual Access
  - Virtualisation based resiliency: **Single attached sites**
- **Physical hosts** connect to the physical Access
  - **Network overlays** start at the physical Access
  - Network resiliency: **Site multi-homing**
- A hybrid overlay allows the combination of physical and virtual resources

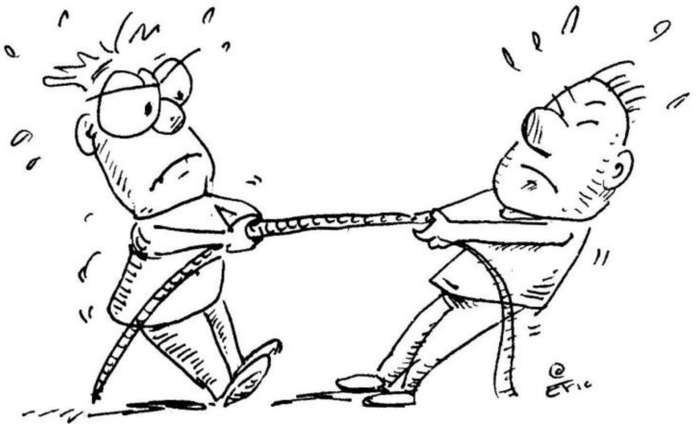


Which Encapsulation?

VXLAN

NVGRE

LISP



MPLS

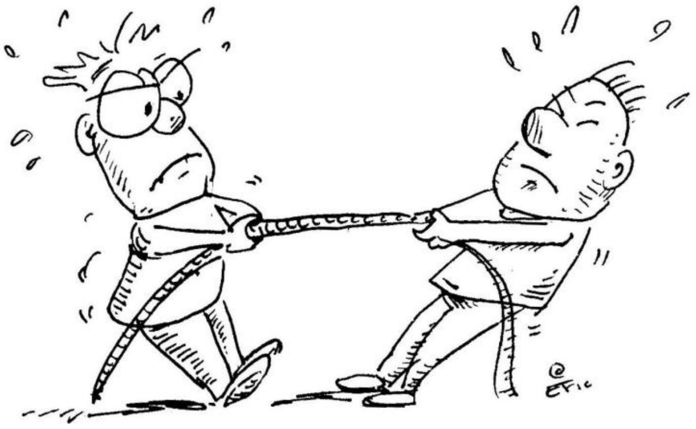
STT

FabricPath

The Multi-protocol Router

TCP/IP

SPX/IPX



AppleTalk

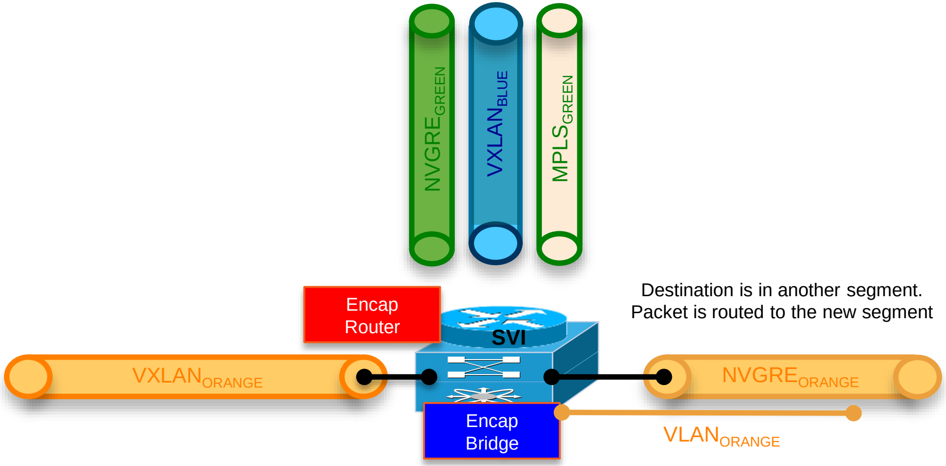
Token Ring

ATM

DECNet

The Multi-encapsulation Gateway

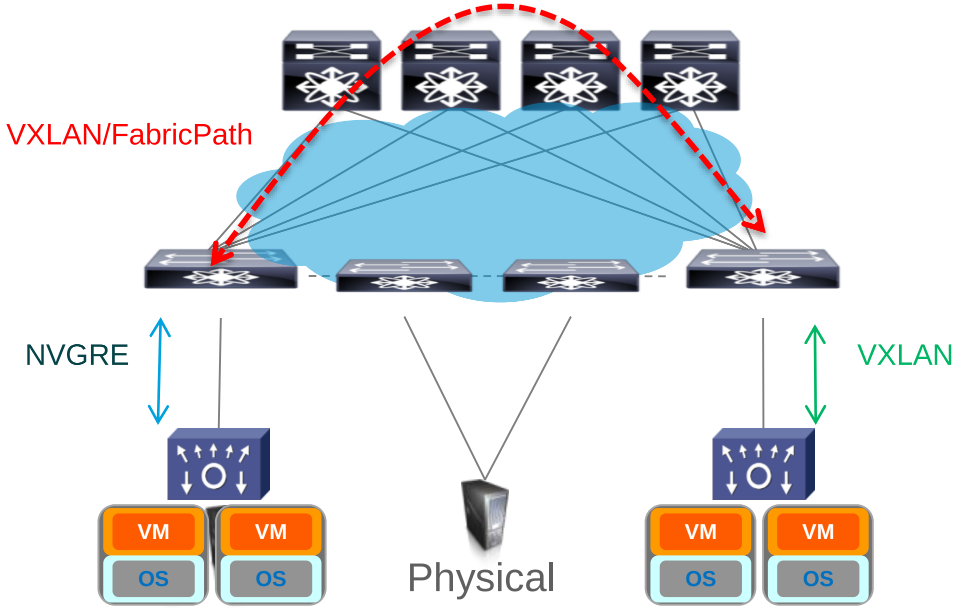
- Multi-encapsulation Gateway:
  - VXLAN, NVGRE, MPLS, LISP, VLAN, OTV, Geneve, etc.
- Bridging (L2 Gateway)
- Routing (L3 Gateway)



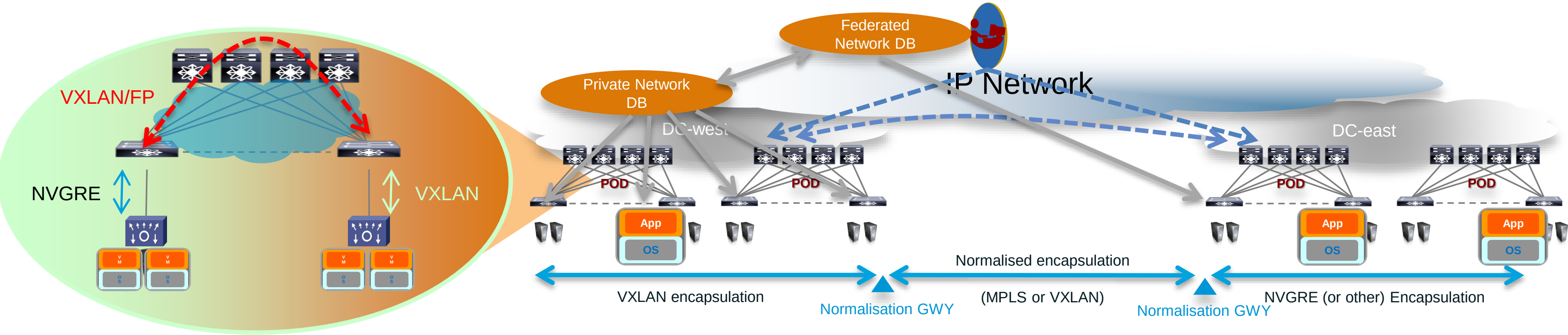
- Multiple TEPs in independent VRFs
- Nesting of IP overlays into MPLS VPNs
- Available across the product line

Normalisation: The Encapsulation Doesn’t Matter

- Intelligence in the Control Plane
- Capabilities Exchange in Control Plane (negotiate encapsulation)
- Normalise to common encapsulation
- Pervasive Multi-encap Gateways for optimal traffic patterns



Data Plane and Control Plane Normalisation



- Multi-encapsulation Hardware Gateways
- Normalise to a common encapsulation in the Fabric and/or between Data Centres
- Terminate and map multiple types of encapsulation
  - VXLAN, NVGRE, MPLS, OTV, LISP
- Terminate and re-distribute information between overlay control protocols
  - Controllers, BGP, LISP

Encapsulation HW Offload

Host Overlays

- Current forwarding penalty for SW encap is about 50% throughput
- STT leverages TCP offload engine in existing NICs
  - TCP violation, short lived workaround
  - P2P only, no routing of flows
- VXLAN/NVGRE offload on NICs
  - The way forward for host overlays
  - Disruptive, many touch points
  - Static as ASICs: headers still in flux

Network Overlays

- ASIC acceleration of overlay encapsulations
  - Cisco ASICs with parser programmability
  - Fast enablement of incremental functions in header reserved fields without replacing HW
- Minimal disruption at the network access
  - Manageable number of touch points
- Encapsulation Normalisation
- Maximise throughput

LISP and VXLAN Headers Today

```
       0                   1                   2                   3
       0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |Version|  IHL  |Type of Service|          Total Length         |
   /  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   |         Identification        |Flags|      Fragment Offset    |
  |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  OH  |  Time to Live | Protocol = 17 |         Header Checksum       |
  |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   |                    Source Routing Locator                     |
   \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \ |                 Destination Routing Locator                   |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |       Source Port = xxxx      |       Dest Port = 4341        |
  UDP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \ |           UDP Length          |        UDP Checksum           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  L   |N|L|E|V|I|flags|            Nonce/Map-Version                  |
  I \ +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  S / |                 Instance ID/Locator Status Bits               |
  P   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |Version|  IHL  |Type of Service|          Total Length         |
   /  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   |         Identification        |Flags|      Fragment Offset    |
  |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  IH  |  Time to Live |    Protocol   |         Header Checksum       |
  |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   |                           Source EID                          |
   \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \ |                         Destination EID                       |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

LISP: IP only today

```
       0                   1                   2                   3
       0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |Version|  IHL  |Type of Service|          Total Length         |
   /  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   |         Identification        |Flags|      Fragment Offset    |
  |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  OH  |  Time to Live | Protocol = 17 |         Header Checksum       |
  |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   |                    Source Routing Locator                     |
   \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \ |                 Destination Routing Locator                   |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |       Source Port = xxxx      |       Dest Port =  8472       |
  UDP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \ |           UDP Length          |        UDP Checksum           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |R|R|R|R|I|R|R|R|            Reserved                           |
VXLAN +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \ |       VXLAN Network Identifier (VNI) |   Reserved             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |             Inner Destination MAC Address                     |
   /  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   | Inner Destination MAC Address | Inner Source MAC Address      |
 Eth  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   |                 Inner Source MAC Address                      |
   \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
Optional Ethertype = C-Tag [802.1Q]   | Inner.VLAN Tag Information    |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / | Ethertype of Original Payload |                               |
   /  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               |
  |   |                   Original Ethernet Payload                   |
Payload|                                                              |
  |   |(Note that the original Ethernet Frame's FCS is not included)  |
  |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   \  |  New FCS (Frame Check Sequence) for Outer Ethernet Frame      |
    \ +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

VXLAN: Ethernet only today

LISP, OTV and VXLAN Normalisation with Generic Protocol Extension (gpe)

draft-ietf-nvo3-vxlan-gpe

```
           0                   1                   2                   3
           0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
          +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        / |Version|  IHL  |Type of Service|          Total Length         |
       /  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |   |         Identification        |Flags|      Fragment Offset    |
      |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    OH    |  Time to Live | Protocol = 17 |         Header Checksum       |
      |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |   |                    Source Routing Locator                     |
       \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        \ |                 Destination Routing Locator                   |
          +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        / |       Source Port = xxxx      |       Dest Port = 4789        |
    UDP   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        \ |           UDP Length          |        UDP Checksum           |
          +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        / |R|R|R|R|I|P|R|R|    Reserved   |        Protocol Type          |
  VXLAN   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        \ |                VXLAN Network Identifier (VNI) |   Reserved    |
          +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Ethernet or IP Payload: Defined in the Protocol Type
Common encapsulation for LISP and VXLAN
L2 and L3 Payloads in both LISP and VXLAN

Header Evolution: Metadata and Overlay Headers

- Segmentation (VRFs, VPNs, Instances, Segments)
- L2 and L3 Payloads
- Policy (End-Point-Groups, Scalable Group Tags)
- Service Chaining (Network Services Header)
- Underlay integration (load balancing, traffic engineering)

LISP, OTV and VXLAN GPE Plus Network Service Header

draft-ietf-sfc-nsh

Protocol Type = 0xNSH

Protocol Type = IP

```
Base Service Header:

 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Ver|O|C|R|R|R|R|R|R|    Length   |  MD-type=0x1  | Next Protocol |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Service Path ID                      | Service Index |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

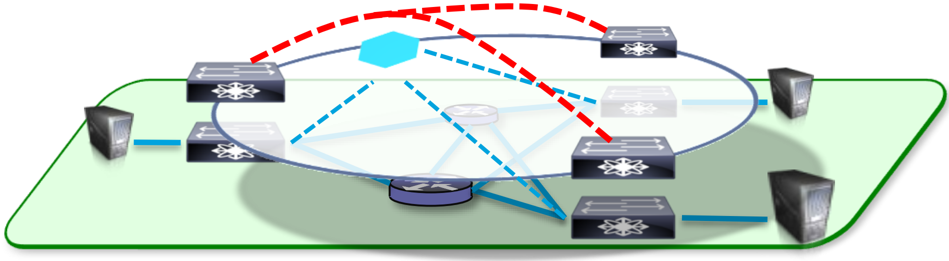
```
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     / |Version|  IHL  |Type of Service|          Total Length         |
    /  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   |         Identification        |Flags|      Fragment Offset    |
   |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   OH  |  Time to Live | Protocol = 17 |         Header Checksum       |
   |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   |                    Source Routing Locator                     |
    \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     \ |                 Destination Routing Locator                   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     / |       Source Port = xxxx      |       Dest Port = 4341        |
   UDP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     \ |           UDP Length          |        UDP Checksum           |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     / |N|L|E|V|I|P|R|R|   Reserved    |Nonce/Map-Version/Protocol-Type|
  LISP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     \ |                 Instance ID/Locator-Status-Bits               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     / |                                                               |
    /  |                          Base Header                          |
   |   |                                                               |
   |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   |                         Context Header                        |
  NSH  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   |                         Context Header                        |
   |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   |                         Context Header                        |
    \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     \ |                         Context Header                        |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     / |Version|  IHL  |Type of Service|          Total Length         |
    /  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   |         Identification        |Flags|      Fragment Offset    |
   |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   IH  |  Time to Live |    Protocol   |         Header Checksum       |
   |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   |                           Source EID                          |
    \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     \ |                         Destination EID                       |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Overlay Signalling Evolution

Service	Edge Device	Signalling
Layer 2 Service Layer 3 Service	Host Overlays Network Overlays	Data Plane Learning Control Plane Learning

Overlay Signalling

- Service Discovery
  - Edge devices in an overlay need to discover each other
- Address Advertising and Tunnel Mapping
  - Edge devices must exchange host reachability information
  - Map end-point to location
- Tunnel Management
  - Maintain and manage connections between edge devices

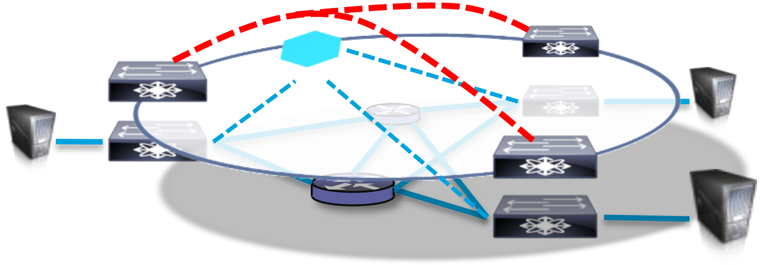


Overlay Signalling

Data Plane Learning

- Based on gleaning information from data plane events
  - Example: Source Learning on bridges
- Provides the following:
  - Address advertisement/mapping (very effectively)
  - Some tunnel management is possible
  - Does not provide Service Auto-discovery
- Requires a flood facility for data plane events to propagate:
  - Multicast tree
  - Unicast replication group at the head-end
- Flood facility can be manually configured on every device (e.g. join a mcast group or configure a list of unicast destinations)
- Usually is supplemented with a control protocol for Service Discovery (specially if using unicast replication)

Overlay Signalling
Control Plane



- Provides:
  - Service Discovery
  - Address Advertising/Mapping
  - Tunnel Management
  - Extensions for multi-homing and advanced services can be provided

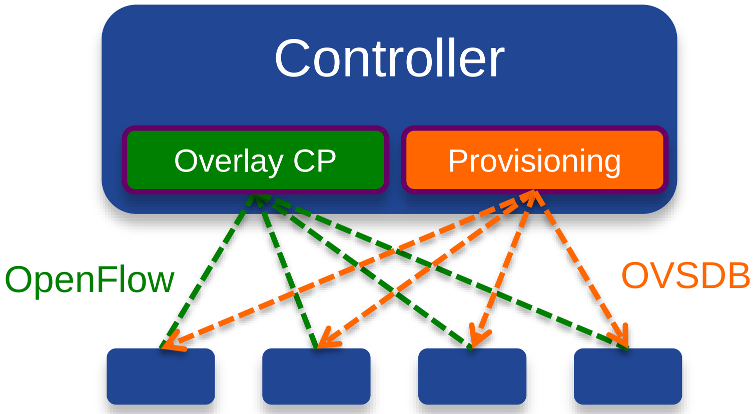
Protocol or Controller:

- **Routing Protocol** amongst Edge Devices
  - BGP, IS-IS, LISP
- Central database on a **Controller**
  - Distributed Virtual Switches (OVS, N1Kv/VSM)

Push or Pull:

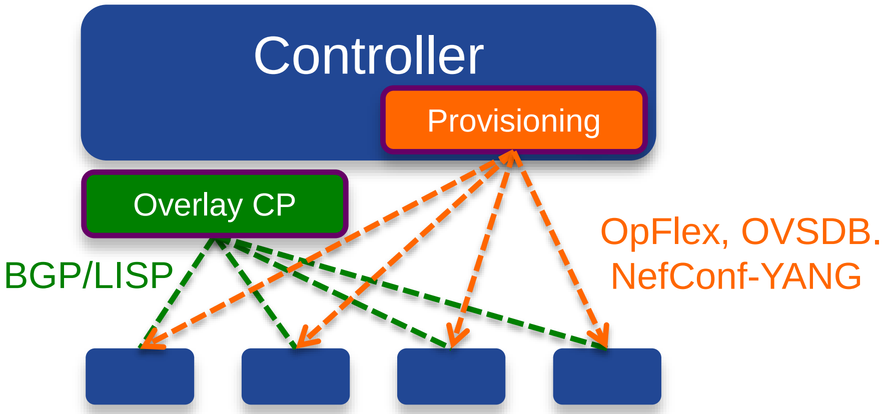
- **Push** all information to all Edge Devices
  - BGP, IS-IS, Controllers
- **Pull** and cache on demand @ ED
  - LISP, DNS, Controllers

Control and Management Planes



Centralised - Database

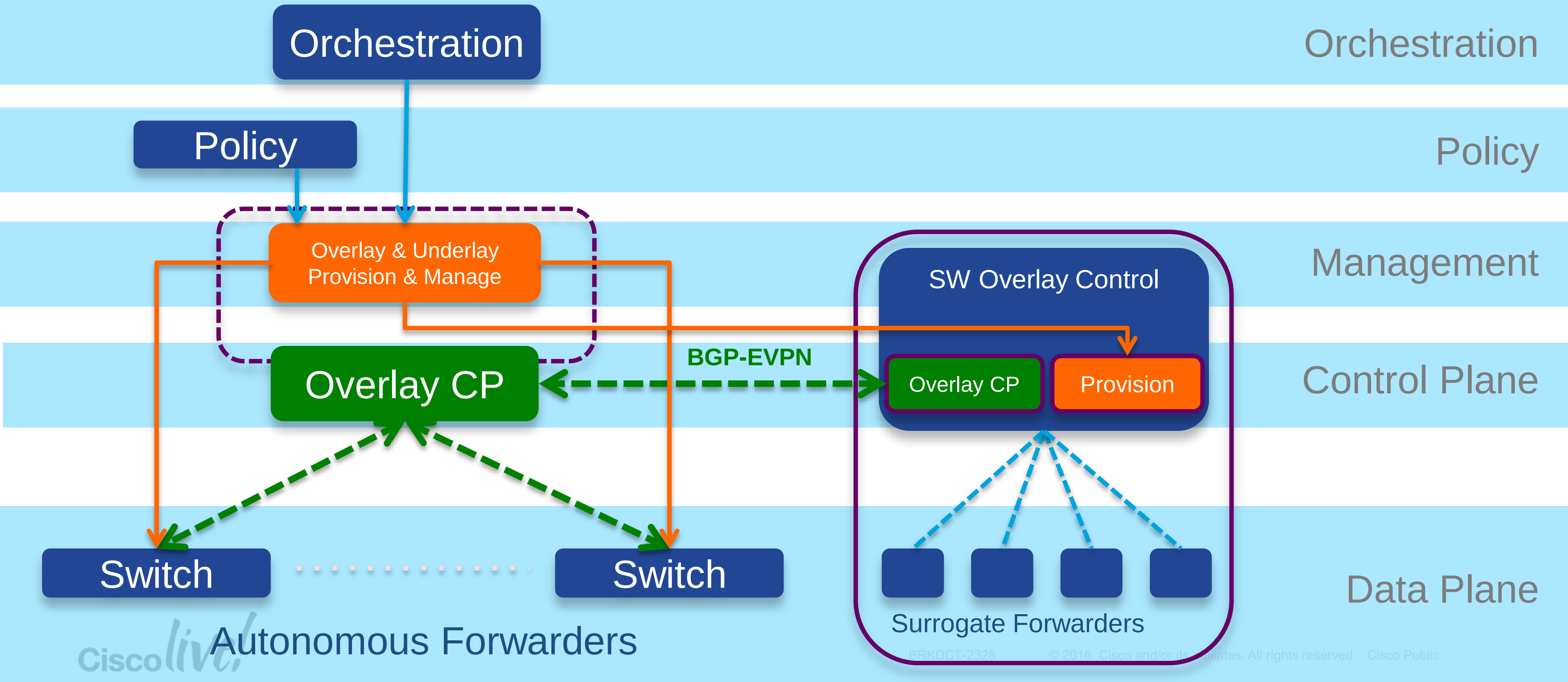
- Tight integration with provisioning/management
- Limited scale



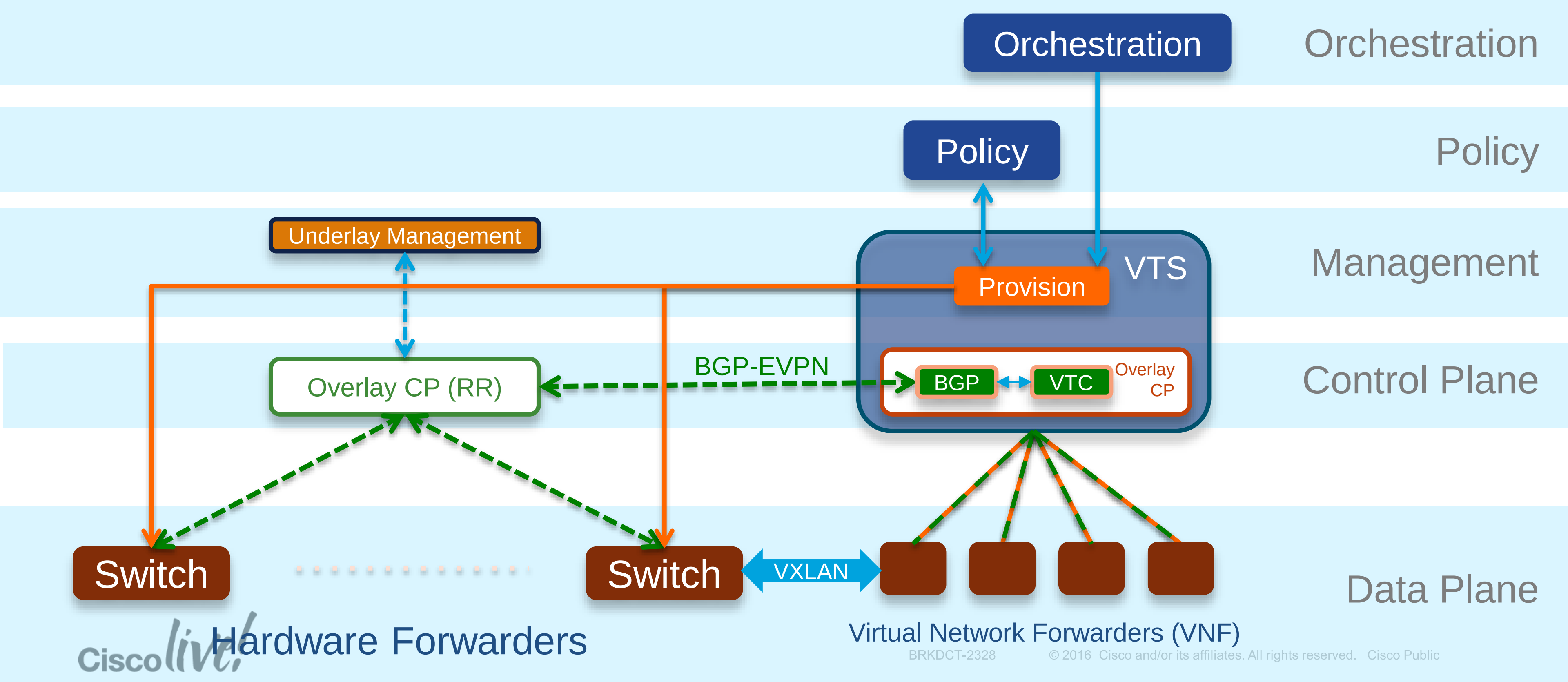
Distributed – Network Protocol

- Loose integration with provisioning/management
- Global Scale

Overlay Reference Architecture



Overlays with Virtual Topology System

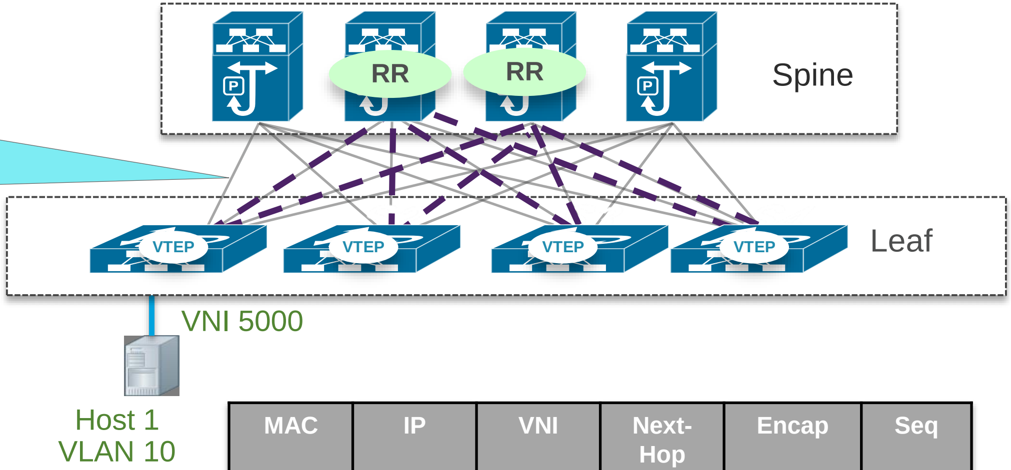


Mobility in BGP EVPN

Host Advertisement

NLRI:
- Host MAC1, IP1
- NVE IP 1
- VNI 5000

Ext. Community:
- Encapsulation: VXLAN, NVGRE
- Cost/Sequence



MAC	IP	VNI	Next-Hop	Encap	Seq
1	1	5000	IP1	VXLAN	0

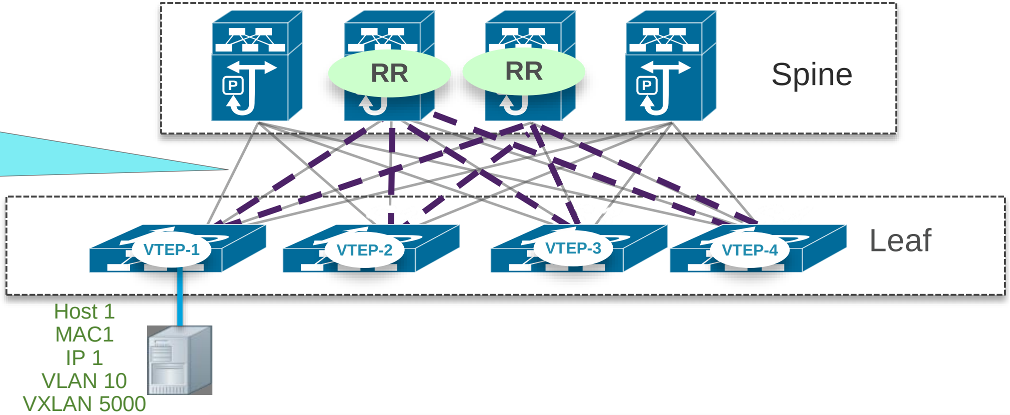
1. Host Attaches
2. Attachment VTEP advertises host's MAC address and IP address to other VTEPs through BGP RR

Mobility in BGP-EVPN

Host Moves

NLRI:
- Host MAC1, IP1
- NVE IP 1
- VNI 5000

Ext. Community:
- Encapsulation: VXLAN, NVGRE
- Cost/Sequence



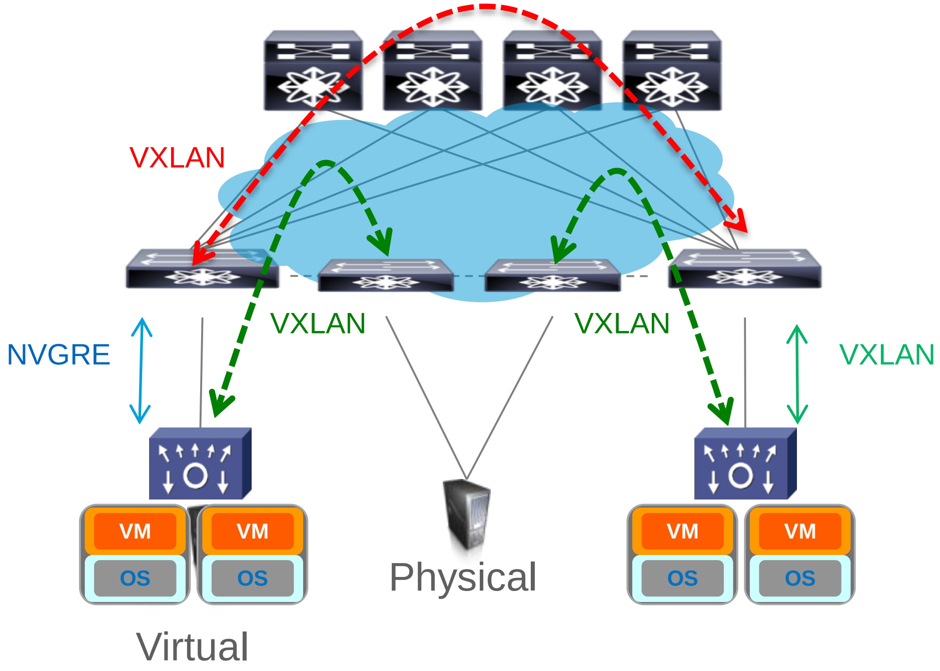
MAC	IP	VNI	Next-Hop	Encap	Seq
1	1	5000	IP3	VXLAN	**1**

1. Host Moves behind switch VTEP-3
2. VTEP-3 detects Host1 and advertises H1 with seq #1
3. VTEP-1 sees more recent route and withdraws its advertisement

Overlays Evolve to Meet Network Challenges

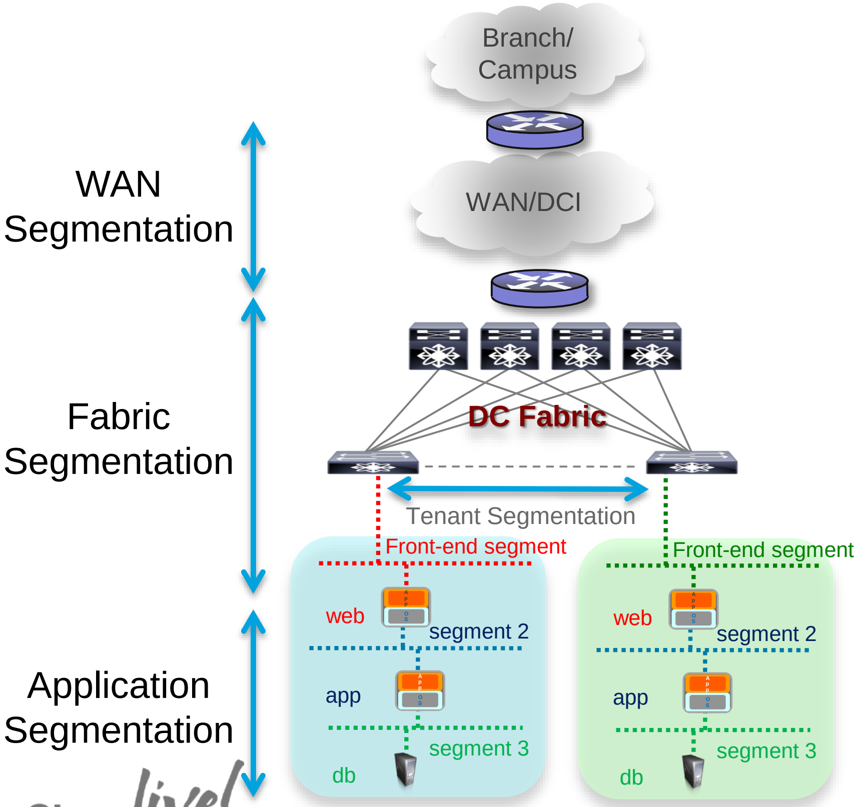
DC-Fabric: Integrated Physical + Virtual overlays

- Physical + Virtual:
  - Hybrid overlay
  - Overlay normalisation
- VXLAN/FP fabrics support a mix of software and HW end-points on a hybrid overlay: No gateways
- ACI Fabrics can normalise host overlay encapsulation:
  - Terminate the encapsulation from the host overlay
  - Translate to a normalised encapsulation in the fabric



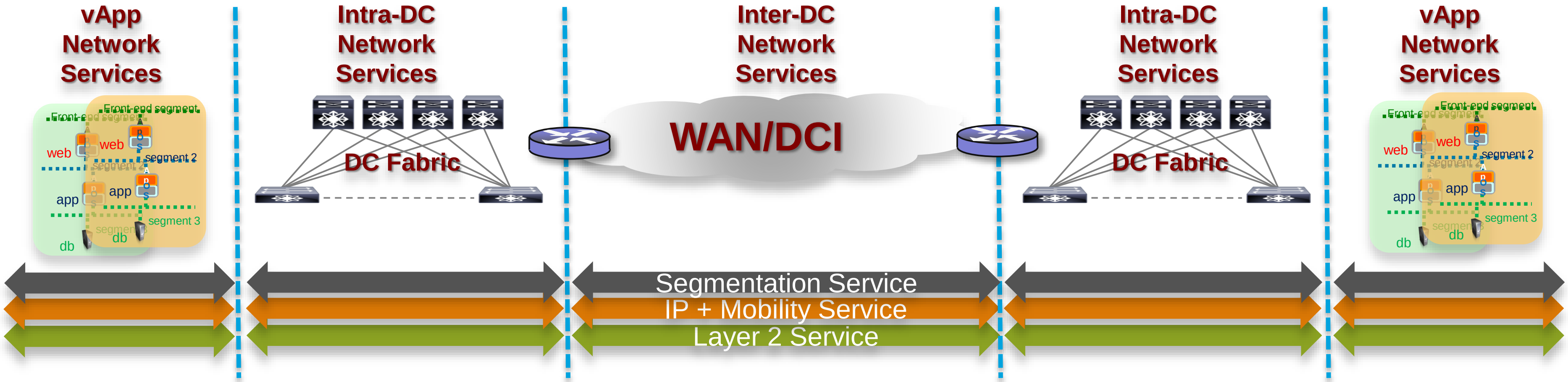
Cisco live!

Segmentation End-to-end



- Segmentation at many levels
- Must be given continuity
  - Across the different network places
  - Across organisations and administrative boundaries
- All relevant technologies include the required segmentation semantics
- The network maps the segments together to provide a scalable and interoperable e2e segmentation solution

Failure Domain Scope



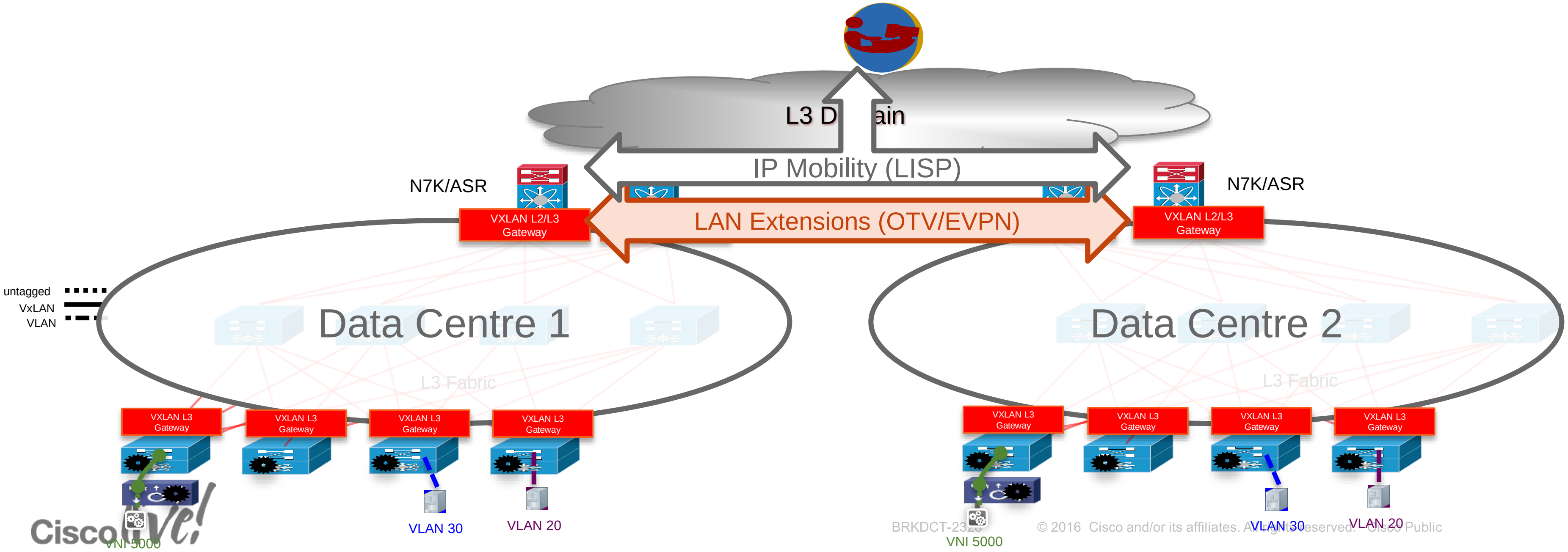
Core Principles of Network Resiliency/Scale applied to Overlay Services

- Clearly delineated Fault Boundaries and service domains
- Control Plane Hierarchy and Federation within and across domains
- Data Plane Boundaries
- Administrative Domain Delineation and Federation

Interconnecting Multiple Data Centres

LAN Extensions and IP mobility

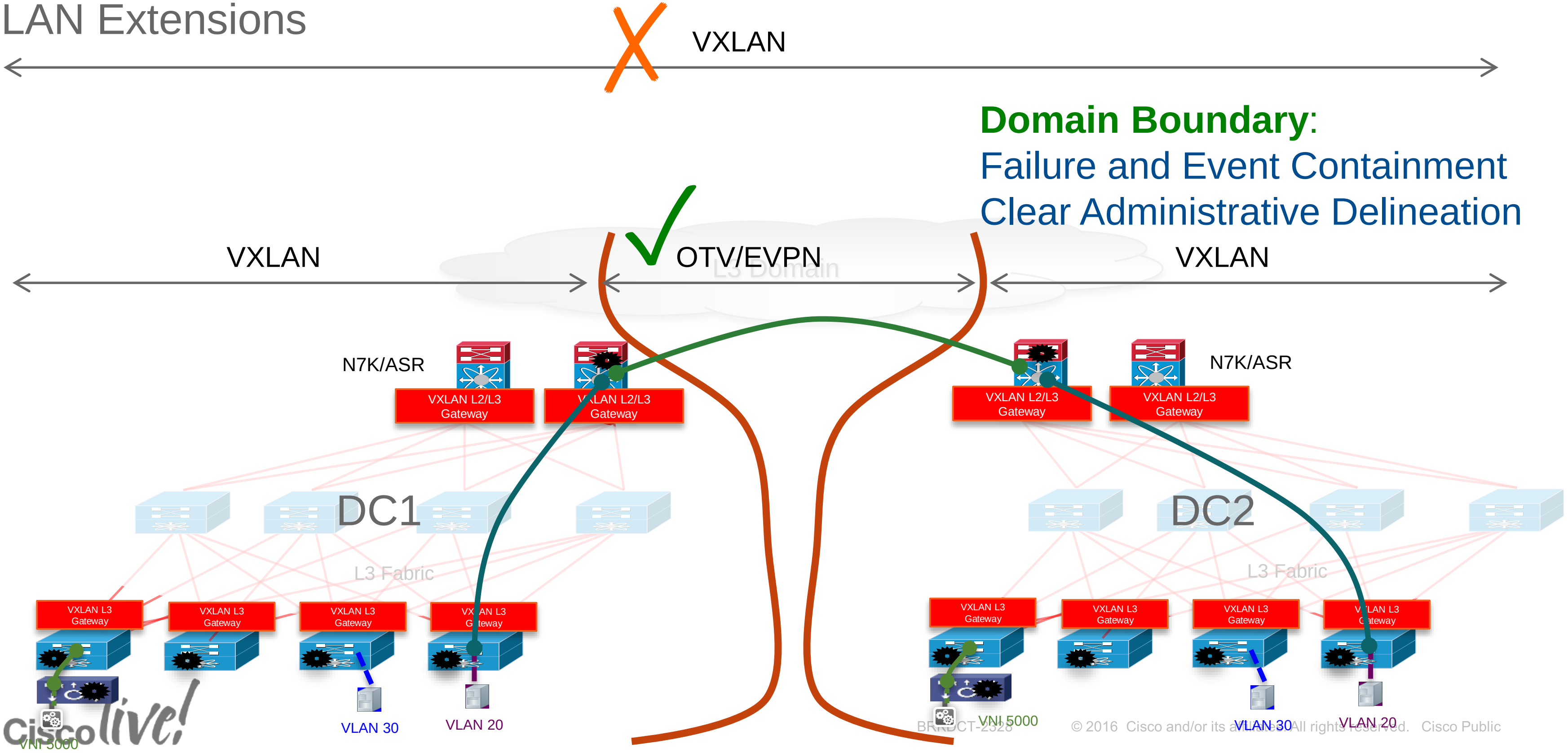
Ethernet extensions between independent fabrics
IP traffic is forwarded via the optimal path (no hair-pinning)



Interconnecting Multiple Data Centres

LAN Extensions

Domain Boundary:
Failure and Event Containment
Clear Administrative Delineation



Interconnecting Multiple Data Centres

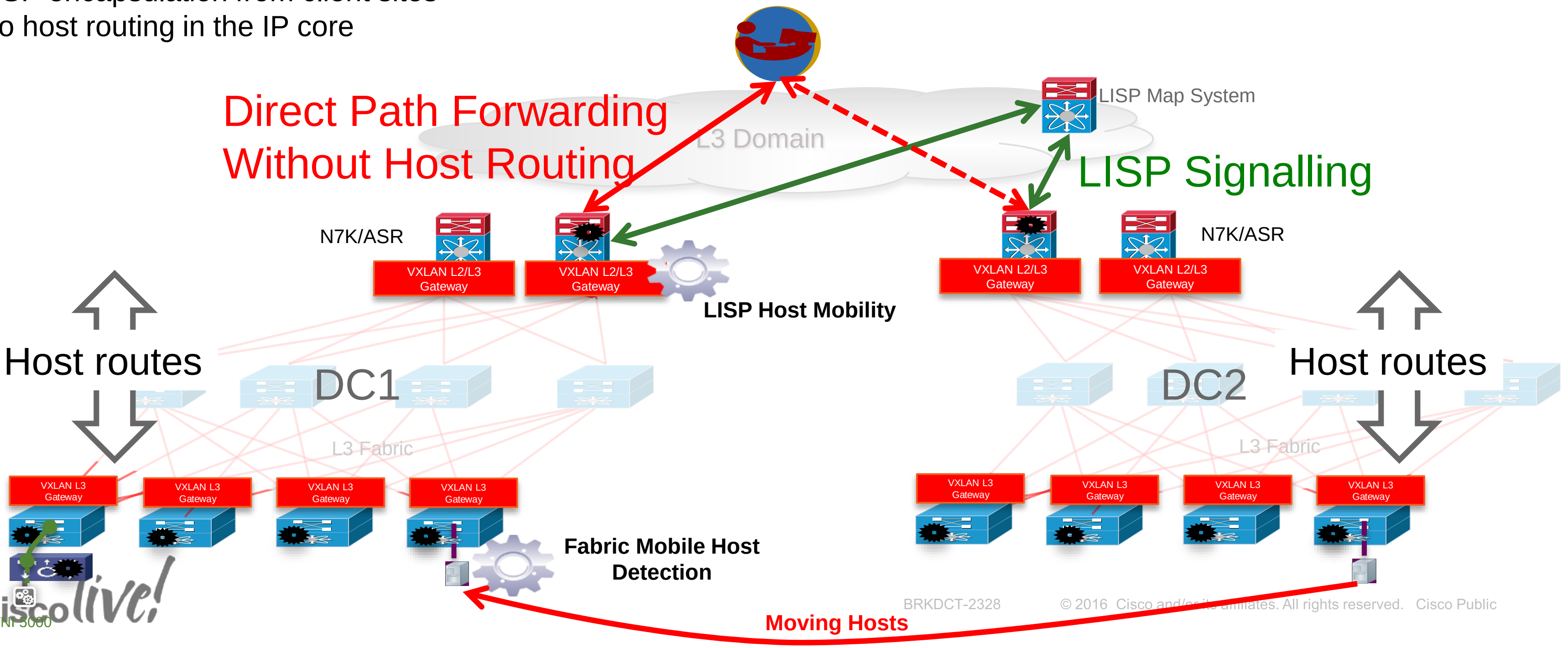
LISP IP Mobility for Optimised Routing

LISP Mobility:

LISP registrations and notifications
LISP encapsulation from client sites
No host routing in the IP core

LISP Signalling:

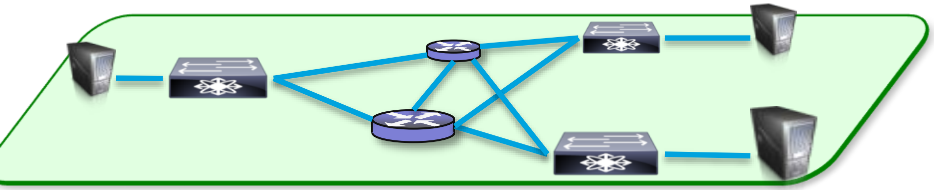
Relay mobility state between sites



Role of the Underlay

Underlying Fabrics

How The Fabric Forwards Traffic



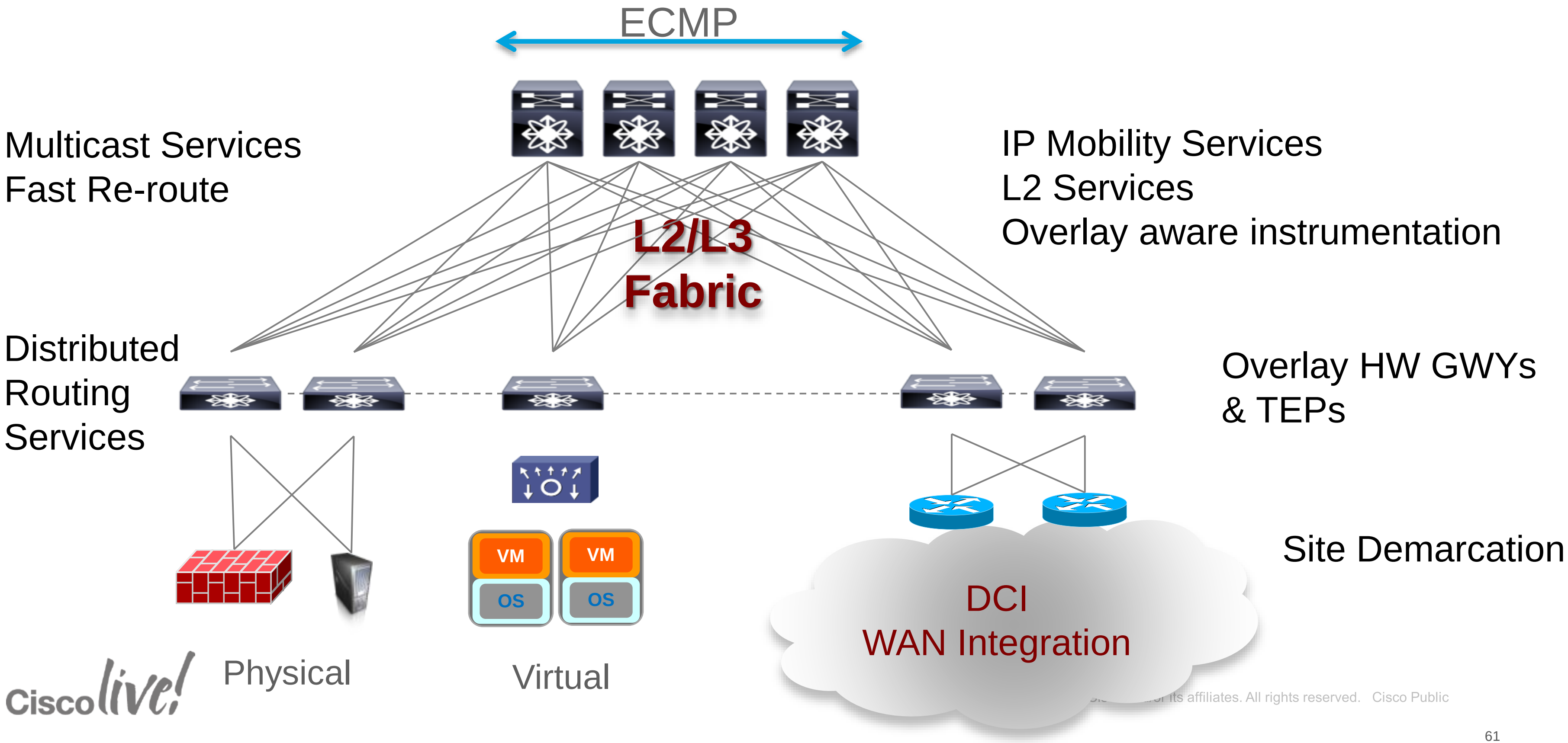
Fabric Characteristics

- High Capacity (10/40/100 GE)
- Line-rate and Low Latency
- Multi-pathed and Resilient (16 way ECMP)
- Simplified/manageable (single touch provision)
- Programmable (1PK, Scripting: Python, POAP)
- Overlay aware (inspect encapsulated traffic)

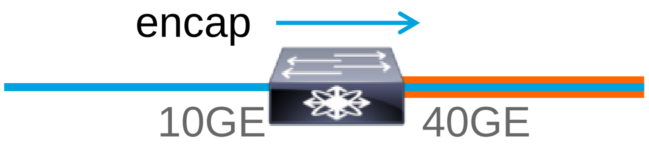
Types of Network Fabric

- IP Network
  - Leverage traditional routing protocols
  - Manage point-to-point links
  - Realise multi-pathed fabric
  - Standards based
- Unified Fabric Network
  - Simplified provisioning and management of multi-pathed fabric
  - Multicast, Load Balancing and multi-topology optimisations
  - Supports multiple types of traffic: IP, Ethernet, FCoE

Fabric Relevance to a Hybrid Overlay



Encapsulation and Effective Throughput

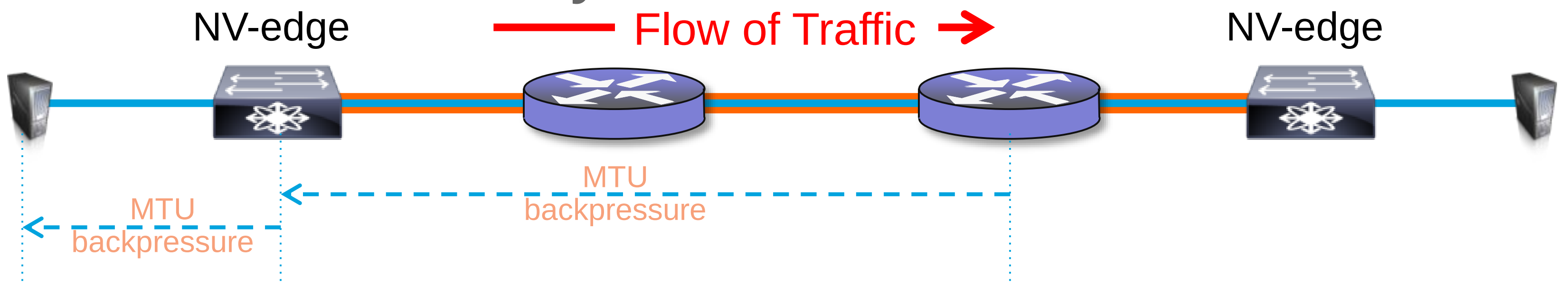


1500bytes/packet (10Gbps) ➔ 1542 bytes/packet (10.1 Gbps)

64bytes/packet (10Gbps) ➔ 106 bytes/packet (10.3 Gbps)

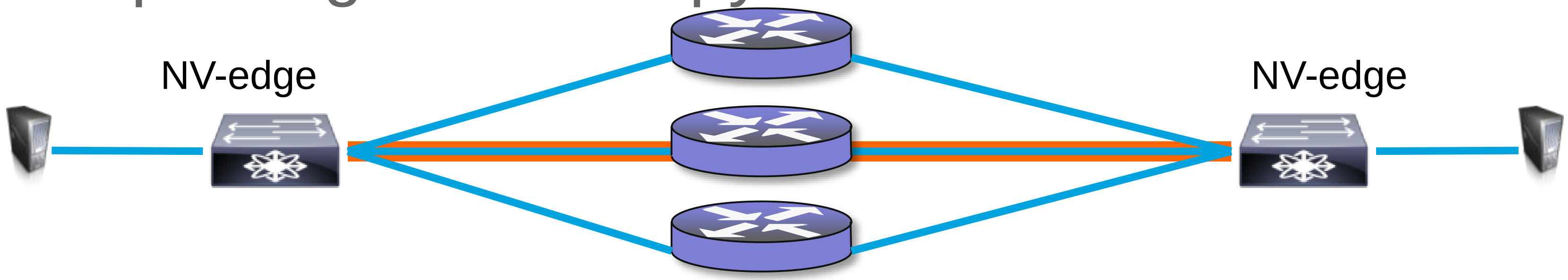
- Encapsulation adds bits to the traffic being sent
- When receiving traffic at full line rate, the encapsulated traffic will exceed the line-rate BW of the egress interface
  - Packet drops
  - Diminished effective throughput
- The uplink BW should be greater than the downlink BW to avoid congestion by encapsulation
  - This is naturally done in the network

MTU Issues: Overlay PMTUD



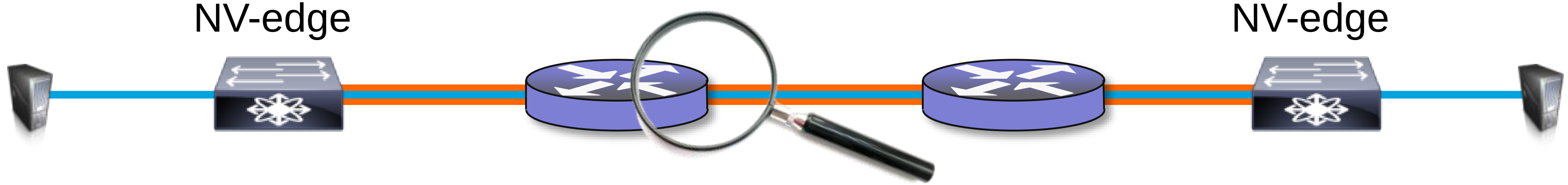
- Encapsulated traffic may exceed max MTU of the path
- When traffic is encapsulated with the Don't Fragment (DF) bit set:
  - If MTU is exceeded: IGMP unreachable message (datagram-too-big) is sent back to the encapsulating NV-edge
  - Encapsulating NV-edge will lower the tunnel MTU accordingly
  - Subsequent packets from the source will trigger an ICMP unreachable message from the NV-edge back to the server (if the traffic from the source has the DF bit set)
- If the DF bit is not set, the device sensing the MTU is exceeded should attempt to fragment the traffic

Multi-pathing and Entropy



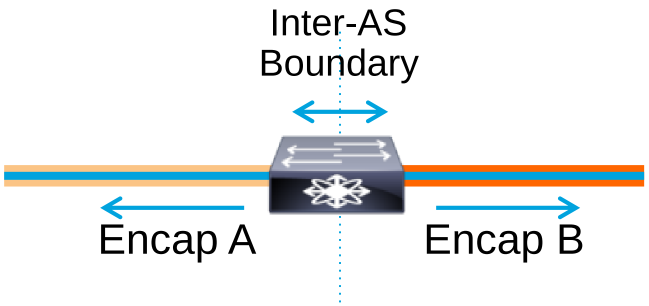
- Tunnel Polarisation: All encapsulated flows tend to look like a single flow between a pair of edge devices
  - Encapsulated traffic always hashes to a single path
- Adding entropy to the encapsulation header can depolarise the tunnels
  - Use all available paths
- UDP headers: Variable UDP source port
- GRE headers: Variable key field
- MPLS headers: Variable LSP label

Instrumentation and Overlay Awareness



- Infrastructure awareness of encapsulated traffic:
  - Outer/Encapsulation header
  - Overlay shim header
  - Internal/Payload header
  - Payload
- Overlay aware Switching & Routing infrastructure:
  - ACLs, QoS, Netflow
- Network Analysis Module (NAM) inspects encapsulated traffic

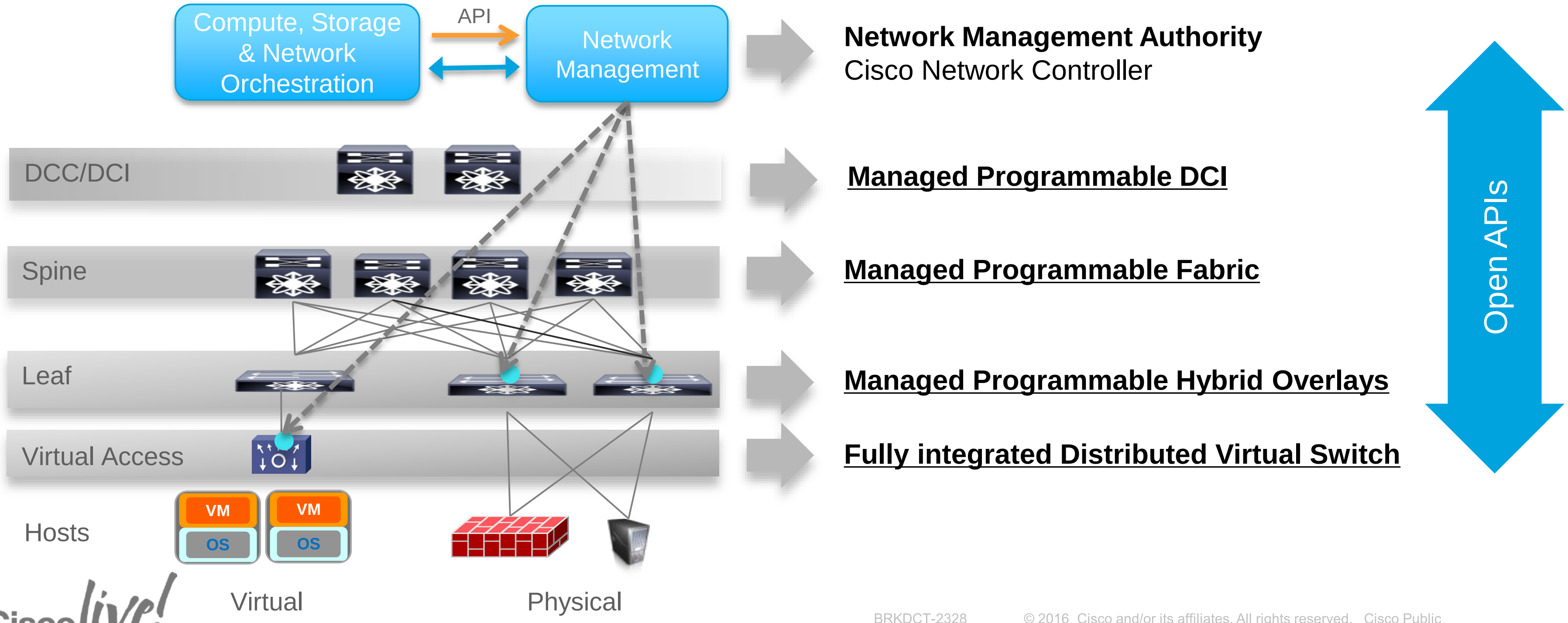
Data Plane and Control Plane Normalisation



- Multi-protocol overlay gateway
- Terminate and map multiple types of encapsulation
  - VXLAN, NVGRE, MPLS, OTV, LISP
- Terminate and re-distribute information between overlay control protocols
  - Controllers, BGP, LISP

Management and Orchestration

Data Centre Fabric Management



Overlay & Underlay Management

Overlay manager

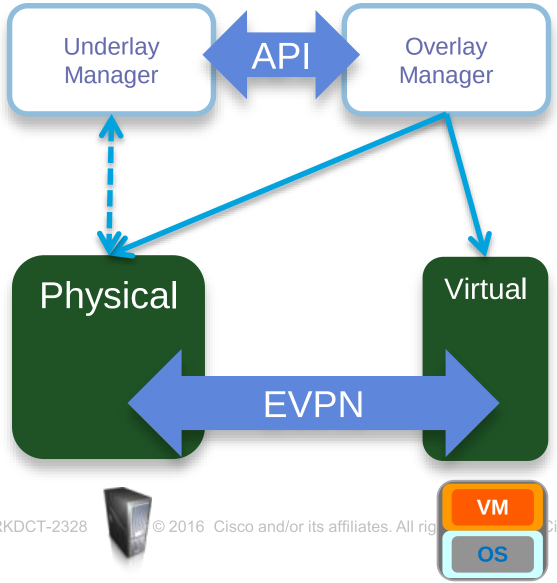
- Provision VXLAN on Virtual and Physical end-points

NMS/EMS for underlay management

- PoAP, Topology Discovery and Inventory, Telemetry, Image Management, etc.
- e.g. DCNM, NFM

Loosely coupled

- API for information exchange
- Combine Underlay/Overlay management under single pane of glass

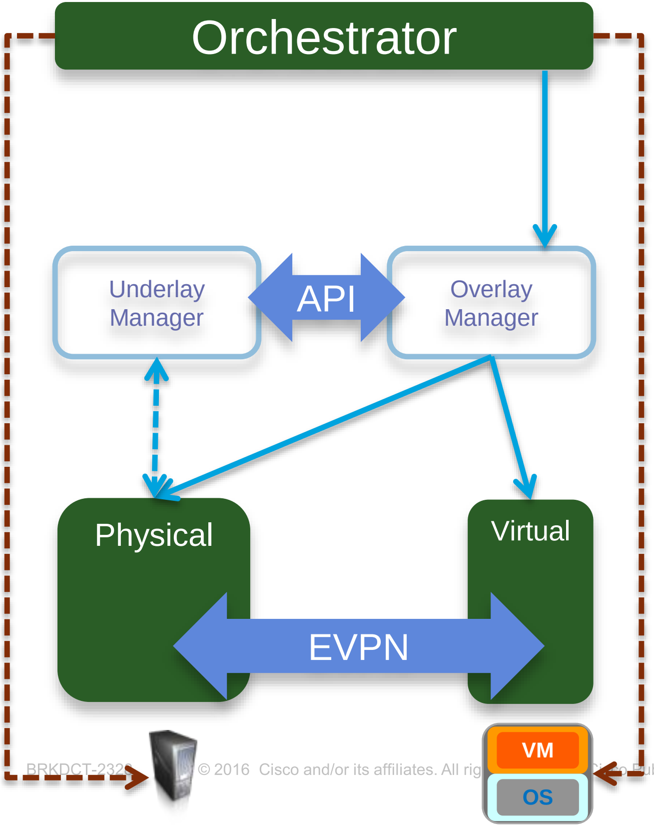


Interface with Orchestrators

Orchestrator events and parameters exchanged with overlay manager through orchestrator API

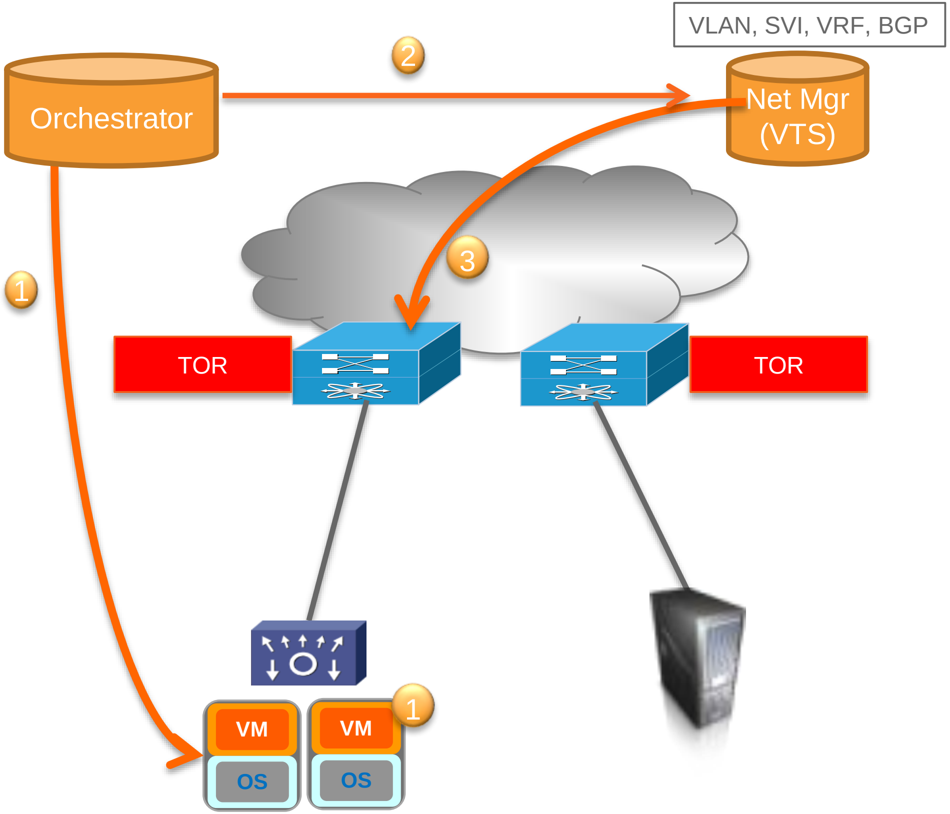
Examples:

- OpenStack,
- UCS director



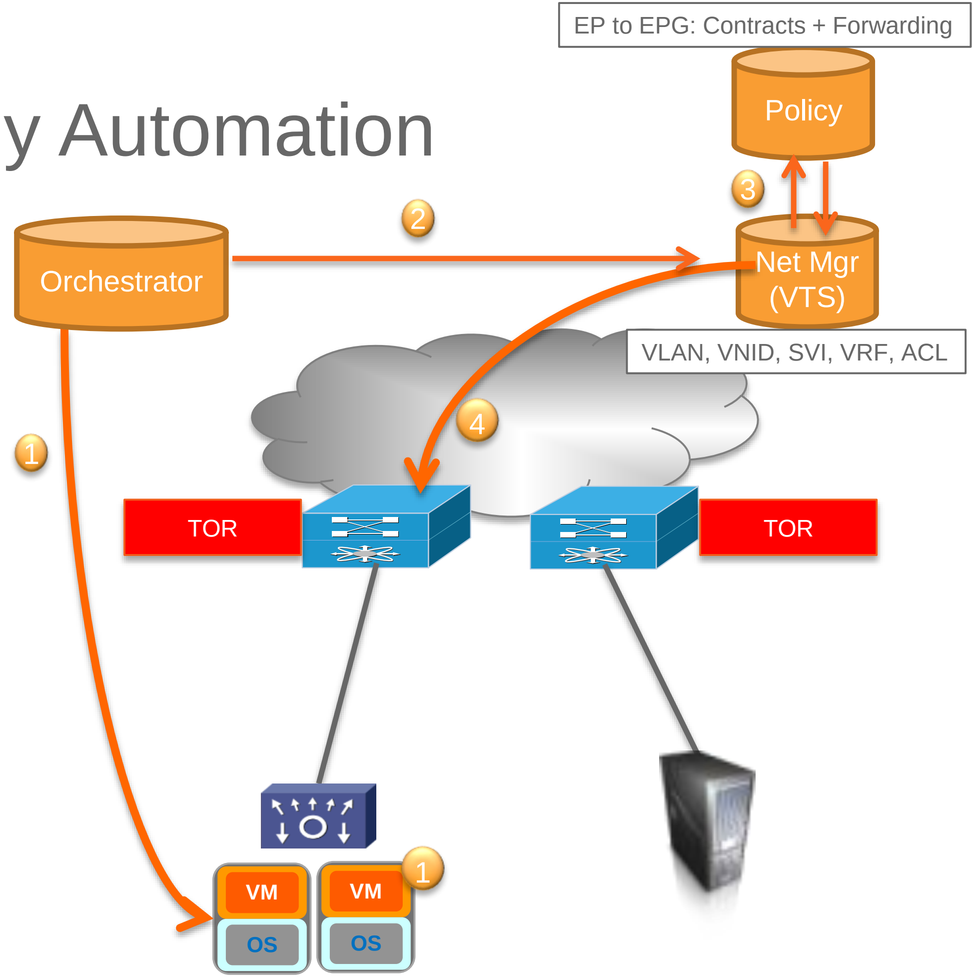
Virtual Topology Automation

- Orchestrator brings up a new or moved host
- The event is passed to the Domain Network Manager
- The Network Manager programs the right VXLAN profile on the appropriate access switches
- Physical and/or virtual switches



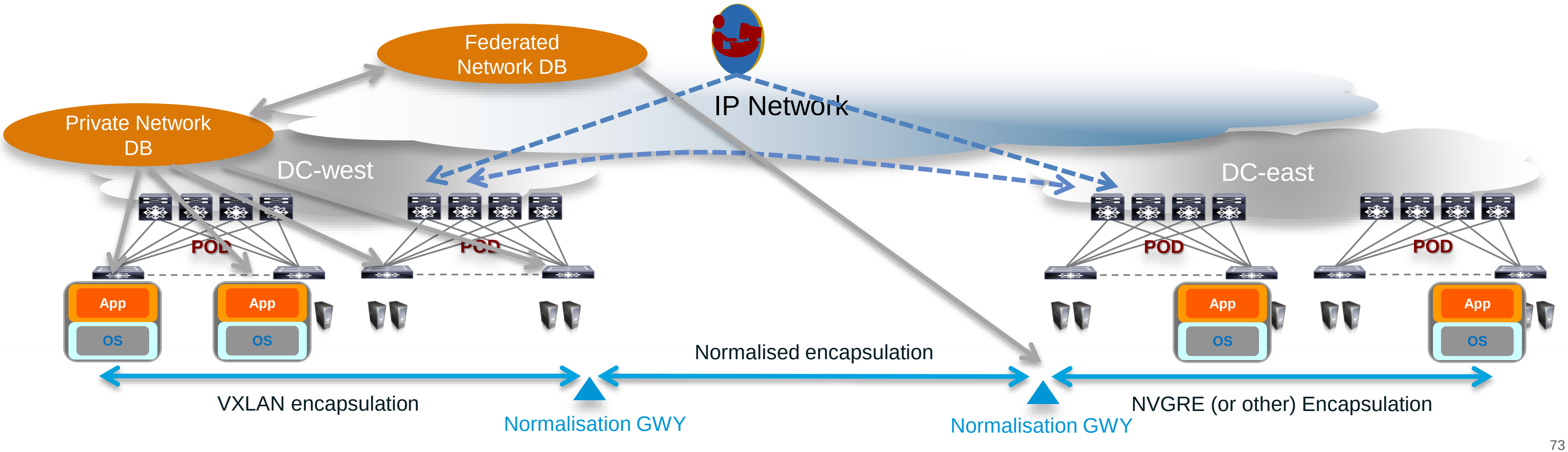
Policy and Virtual Topology Automation

- Orchestrator brings up a new or moved host
- Host “arrival” event is passed to the Network Domain Manager
- Domain Manager queries Policy Repository
- The Domain Manager translates the policy into concrete network constructs & programs the appropriate switches
- Physical and/or virtual switches



Federated/Normalised Overlays Vision

Inter-DC and Intra-DC – LISP/BGP Protocol + Any encapsulation
Virtual and Physical Hosts
Layer 2 and Layer 3
Internet Scale



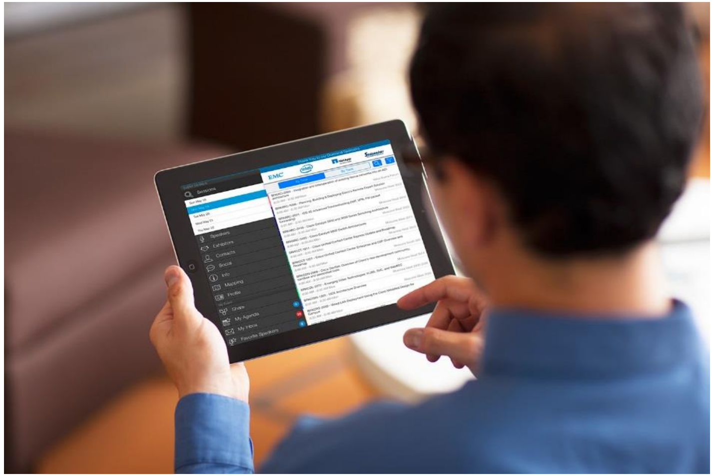
Q & A

Complete Your Online Session Evaluation

Give us your feedback and receive a **Cisco 2016 T-Shirt** by completing the Overall Event Survey and 5 Session Evaluations.

- Directly from your mobile device on the Cisco Live Mobile App
- By visiting the Cisco Live Mobile Site http://showcase.genie-connect.com/ciscolivemelbourne2016/
- Visit any Cisco Live Internet Station located throughout the venue

T-Shirts can be collected Friday 11 March at Registration



Learn online with Cisco Live!
Visit us online after the conference for full access to session videos and presentations.
www.CiscoLiveAPAC.com

Thank you