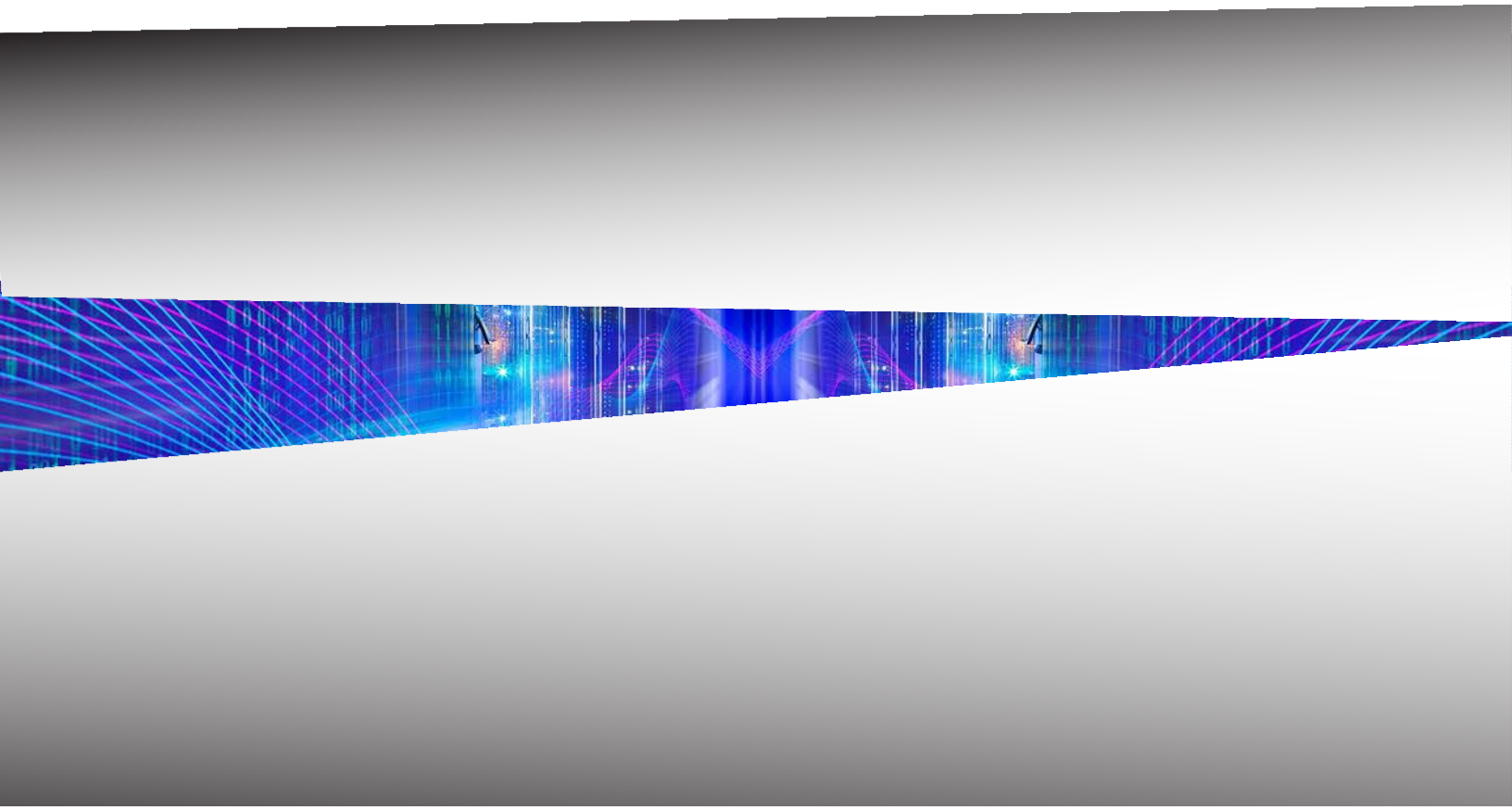
Technical Brief

Pluribus NETWORKS



Achieving a Scale-Out IP Fabric with the Adaptive Cloud Fabric Architecture

Terminology Reference

This is a glossary of acronyms and terms used throughout this document.

AS	Autonomous system is a collection of connected IP routing prefixes under the control of one or more network operators on behalf of a single administrative entity or domain
BFD	Bidirectional forwarding detection is a UDP-based protocol that provides fast detection of Layer 3 next hop failures; it is used in conjunction with a routing protocol, such as BGP
BGP	Border Gateway Protocol is a standardized routing protocol used to exchange routing and reachability data among autonomous systems
Bridge Domain	Configuration construct used to flexibly map single and double VLAN tags to VNIs in order to create scalable fabric-based virtual domains in which bridging is used for traffic forwarding
Cluster	A pair of adjacent Netvisor ONE-enabled switches acting as one logical unit for high availability
Fabric	A set of Netvisor ONE-enabled switches that operate and are managed as a single entity
In-band interface	Internal management interface used as a fabric control port when building a fabric over any IP network
Insight Analytics	Insight Analytics is network performance management (NPM) add-on module to UNUM
L2VPN	Layer 2 VPN; in this design guide context, it is a bridge domain extension across multiple dispersed physical locations over a generic IP transport infrastructure
L3VPN	Layer 3 VPN; in this design guide context, it is an IP domain extension across multiple dispersed physical locations over a generic IP transport infrastructure
Out-of-Band interface	Dedicated out-of-band port on Netvisor ONE-enabled switches, used either as a management-only interface or as a fabric control port to form the fabric and exchange fabric information over the out-of-band management network
Overlay	In the VxLAN context, this refers to all the elements built on top of the generic IP transport infrastructure in order to offer L2VPN and L3VPN functionalities
Pluribus UNUM	Pluribus UNUM unified management, automation and analytics platform software
QinQ	VLAN stacking (sometimes called QinQ) is a technology that double-tags Layer 2 traffic. It was standardized in the IEEE 802.1ad amendment (later incorporated directly into the 802.1Q standard)
Underlay	In the VxLAN context, this refers to the generic IP transport infrastructure used to ensure IP communication among all data centers
VIP	Virtual IP is an IP address that does not correspond to an actual physical device; in this design guide context, it is the IP used by the VRRP instances and the VTEPs
vLAG	Virtual Link Aggregation Group is a Netvisor ONE technology for connecting multiple switches to other devices or to other switches for resiliency and high availability
vLE	Virtual Link Extension is a Netvisor ONE technology that allows the definition of Layer 1 pseudowires that can emulate a direct connection between devices on top of an IP transport network
vNET	A Virtual NETwork is a partition of the Adaptive Cloud Fabric. A vNET is defined by a group of network objects that can operate independently and have dedicated resources, providing multi-tenancy and network segmentation
VRF	VRF is a technology that allows multiple routing spaces to coexist on the same switch; it complements the vRouter construct, offering a highly scalable solution
vRouter	An object used to provide routing between subnets, VLANs and/or vNETs. The vRouter runs in a dedicated operating system container
VRRP	Virtual Router Redundancy Protocol is a networking protocol that provides redundancy of routing paths by creation of virtual routers, which are an abstract representation of multiple routers, i.e., master and backup routers, acting as a group
VTEP	A VxLAN tunnel endpoint is the entity responsible for encapsulating/de-encapsulating VxLAN packets
VTEP HA	VTEP high availability refers to a mechanism designed to ensure redundancy of the VTEP entity
VXLAN	Virtual Extensible LAN is a Layer 2 overlay scheme over a Layer 3 network. It uses MAC-in-UDP encapsulation to provide extension of Layer 2 segments across IP transport networks

Pluribus Networks offers a unique and highly differentiated approach to software-defined networking and is driving a networking revolution toward a more open operating environment. The Adaptive Cloud Fabric™ architecture enables organizations to build scalable private and public clouds that improve service velocity, performance and reliability. The company's innovative Netvisor® ONE software virtualizes open networking hardware to build a holistic, distributed network that is more intelligent, automated and resilient. The company's Insight Analytics™ platform leverages embedded telemetry and other data sources to enable pervasive visibility across the network to reveal network and application performance that speeds troubleshooting and improves operational and security intelligence.

This Technical Brief describes how to design a scale-out IP fabric solution by making the best use of Pluribus' Netvisor ONE and Adaptive Cloud Fabric architecture capabilities combined with open networking switches, such as the Pluribus Freedom™ series switches.

Scale-Out IP Fabric Design

Horizontal scaling, sometimes referred to as "scale-out" or "distributed" network architecture, is a modern approach to building network capacity for the data center, in contrast to traditional vertical scaling, or a centralized, scale-up strategy.

Scale-up network architecture relies on a few centralized, modular, highly capable and expensive network elements. Such modular systems can partially increase their network capacity in both control plane and forwarding plane by adding or replacing supervisory units and line cards respectively. However, capacity expansion is limited by the physical characteristics of the switch chassis, like the number of slots and the switching backplane. As such, significant capacity expansion with scale-up strategy is only possible by replacing the existing modular systems with higher-capacity chassis, which implies a significant cost implication and operational disruptions.

By contrast, a scale-out infrastructure is based on linking together a number of relatively smaller, fixed network form factor switches, which collectively are able to provide the required capacity. Because capacity growth can be achieved by incrementing the number of devices required to meet capacity, the distributed, or scale-out, architecture provides a predictable operational model where both network capacity and cost grow linearly with the number of network elements, making it preferable to the traditional vertical architectural approach.

A key element of the scale-out architecture is guaranteeing that no performance degradation occurs when growing the number of distributed nodes. This means that the addition of a node increases not just forwarding and connectivity capacity, but also the control plane. For example, software-defined networking (SDN) architectures based on a centralized controller impose a limit on the capacity growth of the entire network, which defeats the purpose and value of a horizontal architecture.

The industry best practice for realizing a scale-out architecture is an IP fabric implemented in multi-stage Clos topology, which offers a non-blocking and resilient connectivity model with predictable performance and scaling characteristics, so that traffic workloads are efficiently distributed across the network loads using equal cost multi-path (ECMP).

In comparison with a centralized network approach, one disadvantage of distributed network architectures like IP fabrics is the operational complexity of provisioning and monitoring network operations, services and policies. Pluribus Networks has solved this problem with the Adaptive Cloud Fabric architecture. The Adaptive Cloud Fabric uses an innovative distributed Ethernet switch control plane to cluster IP fabric nodes and create a unified virtual logical chassis with a single management point using either command line interface (CLI) or API.

IP Fabric Design Goals

When building an IP fabric, the main design objectives to be considered are:

- Scale-out architecture with multi-terabit switching capacity employing open networking and merchant silicon-based switches like the Pluribus Freedom series switches
- Non-blocking IP fabric using standard IP protocols
- High availability with fast reconvergence in case of a failure event
- Highly scalable network endpoint database comprising hosts, appliances and virtual machines that attach to the IP fabric
- Highly scalable Layer 2 VPN and Layer 3 VPN services implemented with standard VXLAN protocol
- Forwarding efficiency for traffic hairpinning avoidance, loop avoidance, BUM optimization
- Optimization and simplification of control plane architecture to minimize the need for expensive high-performance nodes, such as switches with very large routing tables
- Operational simplification with single management fabric
- End-to-end visibility of network and application traffic

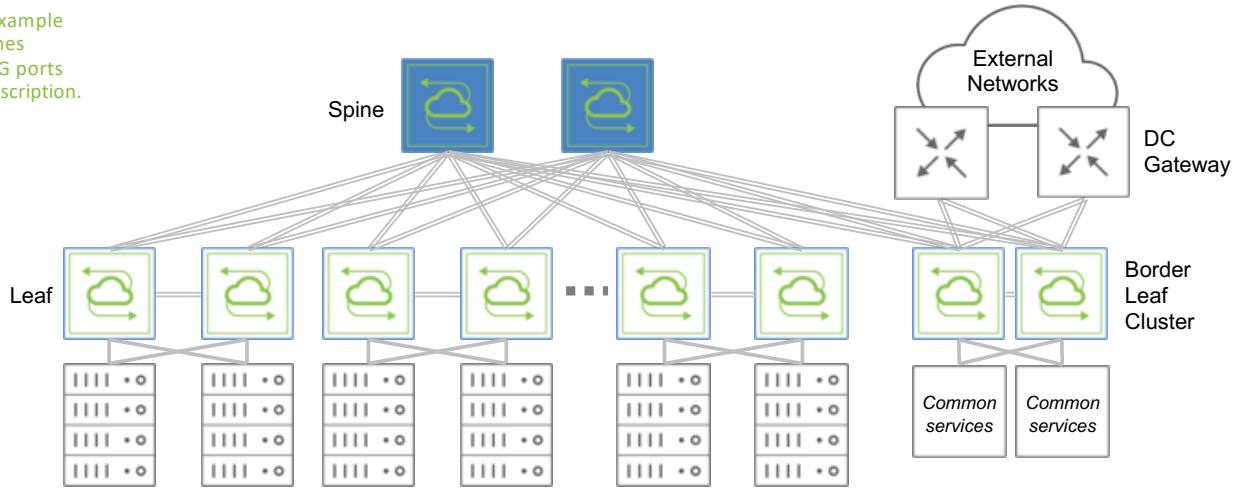
Solution Overview

Let's explore the physical connectivity and scale options of the IP fabric design. Figure 1 below represents a typical data center pod implemented with a three-stage Clos topology. This design is built with Pluribus Freedom 9572-V (48x 10G/25G with 6x 100G uplinks) switches for the leaf layer and two Freedom 9532-C (32x 100G) switches for the spine layer. These switches are running the Pluribus Netvisor ONE operating system. Connectivity to external networks is provided through a pair of leaf switches functioning as redundant border leaves, which connect to a DC gateway function implemented with a pair of any third-party routers.

Leaf switches are paired to form high-availability clusters with 2x 100G inter-switch links and can provide redundant virtual chassis link aggregation group (vLAG) to access ports and highly available VXLAN tunnel end point (VTEP) functions. High availability between leaf and spine layers is based on ECMP with fast failover using BFD.

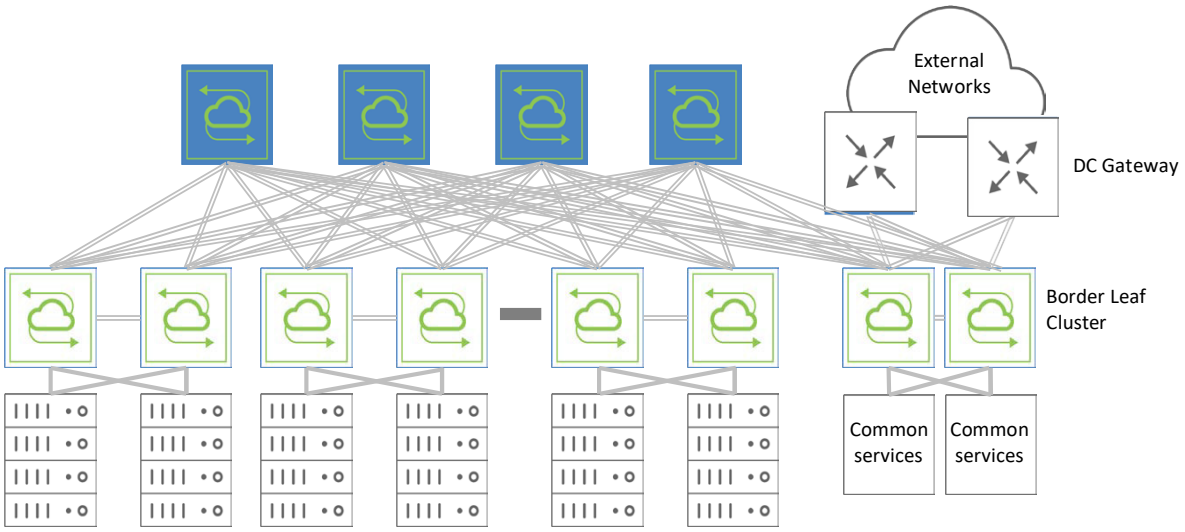
From a capacity perspective, the topology in the figure can grow to 16 leaf switches for a total of 16x 48x 10G = 768x 10G access ports with 480G:400G, equaling a 1.2:1 oversubscription ratio.

Figure 1:
Data center pod example with 16 leaf switches providing 768x 10G ports with 1.2:1 oversubscription.



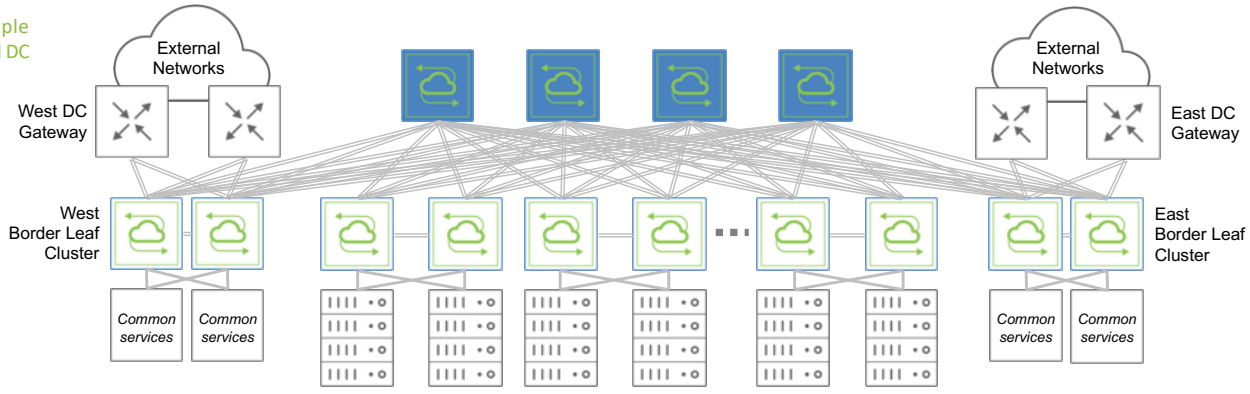
If higher capacity is desired, the topology can be expanded to 32 leaf switches by adding two additional Freedom 9532-C switches. As a result, the Clos fabric can scale up to 1536x 10G access ports with 1.2:1 oversubscription ratio, as represented in Figure 2.

Figure 2:
Data center pod example with 32 leaf switches providing 1536x 10G ports with 1.2:1 oversubscription.



An important aspect of pod design is planning the desired capacity for carrying external traffic, also referred as north-to-south (N-S) traffic, which is forwarded through the DC gateway and border leaf elements. For example, simply interconnecting more physical interfaces between the border leaf and the spine and DC gateway blocks can augment N-S connectivity. Other scaling aspects, like the number of L3VPNs supported on border leaf and DC gateway for both control and forwarding function, can lead instead to optionally incrementing the capacity of the entire N-S functional block by inserting additional border leaf and/or DC gateway pairs, as represented in the example of Figure 3. Further discussion on scaling N-S interconnection function is addressed later in this document.

Figure 3:
Data center pod example with 4 border leaf and DC gateway switches.



IP Underlay

The underlay defines the Layer 2 and Layer 3 connectivity of the physical infrastructure. The Adaptive Cloud Fabric architecture does not impose a specific underlay protocol for the IP fabric and provides different standard options based on eBGP, iBGP and OSPF, following industry best practices to implement ECMP in a Clos topology.

One of the advantages of the control plane of the Adaptive Cloud Fabric architecture is clear underlay-overlay separation. The fabric does not rely on the underlay routing protocols for exchanging overlay information, like the location of endpoints and L2VPN or L3VPN constructs. This approach makes the underlay function lean and reduces the amount of forwarding state on the spine layer to the routing information strictly necessary to provide reachability between the leaf nodes with a single VRF instance. The immediate effect of this simple approach is fast convergence and reconvergence in case of link or node failures. Moreover, this simple design defines the spine as a pure transport element that provides high port density without the need for very large Layer 2 and Layer 3 hardware tables, nor very expensive deep buffers, thanks also to the proposed oversubscription ratio, making the Freedom 9532-C switches an ideal non-blocking high-density choice for the spine function.

Adaptive Cloud Fabric

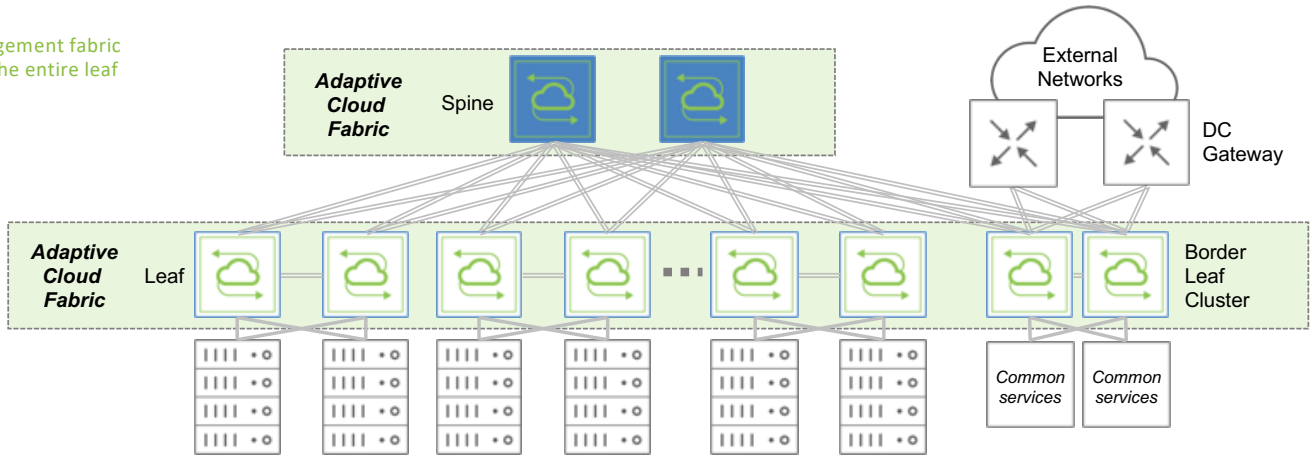
The Pluribus Adaptive Cloud Fabric architecture uses an innovative distributed control plane that can manage all fabric nodes, whether colocated or geographically distributed, from a single CLI- or API-based management point. Any fabric node can act as a single point of management for the entire fabric, thus significantly reducing fabric configuration complexity and the possibility of human error. By providing complete parity between the CLI commands and the API-based configuration, the fabric is interoperable with a centralized management station, such as Ansible or the Pluribus UNUM™ fabric and device manager.

The Adaptive Cloud Fabric's advanced transactional model guarantees that device configuration is maintained consistently across network nodes and supports configuration rollback capabilities. Therefore, a single point of provisioning provides consistent network-wide configuration with powerful commands that can operate on a list of dispersed fabric devices rather than on individual ones.

The Adaptive Cloud Fabric's control plane provides intuitive and less error-prone mechanisms to automatically configure functional super-entities from a number of physical or logical components. Examples of automatic, time-saving mechanisms include switch clustering, auto-LAG, automatic tunnel creation, bridge domains, distributed VRFs and many more.

For the example data center pod design in this paper, since the leaf and spine layers are two distinct functional network blocks, it is recommended to create two separate fabric instances, in order to provide two individual management domains for each functional block, thus preserving homogeneity, provisioning and monitoring simplicity. For example, as represented in Figure 4, all Pluribus Freedom switches in the leaf layer are part of the same Adaptive Cloud Fabric instance, thus simplifying and centralizing the management and monitoring of all data center endpoints.

Figure 4:
Single management fabric comprising the entire leaf layer.

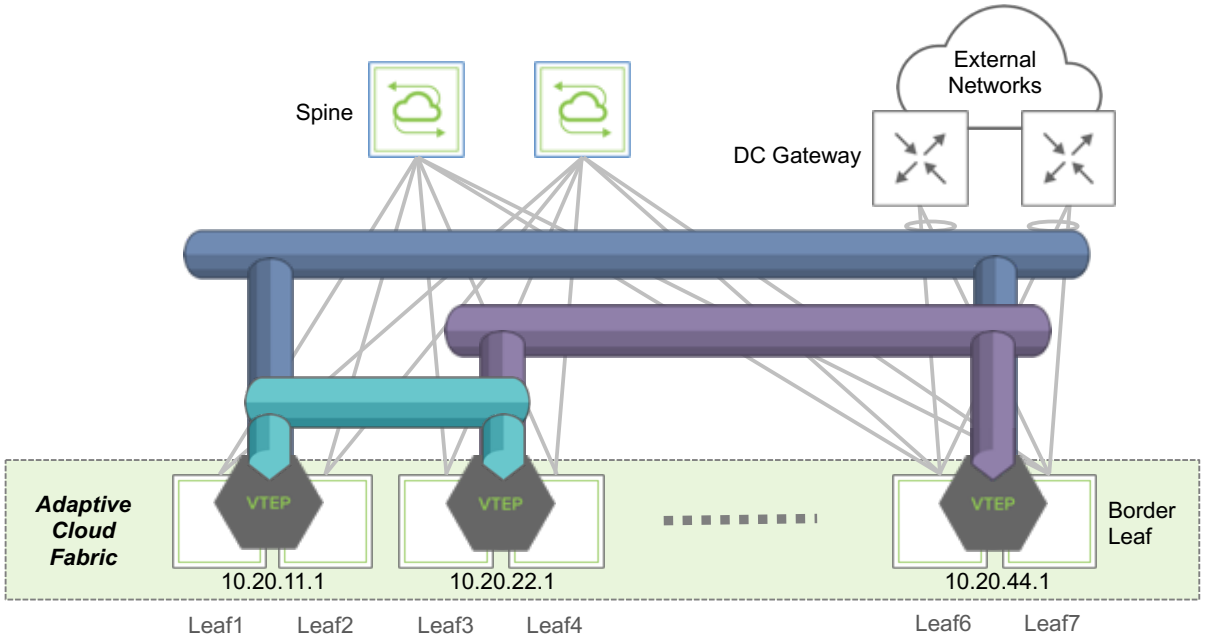


Automatic Provisioning of Tunnels

The Adaptive Cloud Fabric overlay approach is based entirely on a standard Virtual Extensible LAN (VXLAN) implementation. The VXLAN tunnel end point function (VTEP) is implemented following a critical design requirement for DC networks: to deploy a VXLAN transport in conjunction with a high-availability (HA) configuration at the edge of the VXLAN-based fabric. This guarantees path redundancy and service continuity in case of link or device failure. In VXLAN, edge redundancy support is also known as VXLAN tunnel endpoint high availability (VTEP HA).

A VTEP HA object is provisioned on each redundant switch cluster part of the leaf fabric. The VTEP HA pair shares a common virtual IP, in addition to having two individual physical IP addresses used for point-to-point services. With the cluster synchronization function, a VTEP pair can therefore act as a single logical VXLAN endpoint using a single shared VIP as source address. Similarly, a destination VXLAN endpoint can be reached by using its HA pair's common VIP as destination address. This enables the creation of an overlay network of VXLAN interconnections based on virtual, not physical, addresses, which offers embedded physical as well as logical redundancy.

Figure 5:
VTEP HA objects with automatic tunnel provisioning.



With the Adaptive Cloud Fabric control-plane, the instantiation of VTEP objects triggers the automatic creation of all the required VXLAN tunnel connections in both directions between switch clusters, resulting in a significantly lower amount of time for configuration efforts.

The example in Figure 5 includes three switch cluster pairs, which leads to the automatic provisioning of 12 unidirectional tunnel objects:

```
CLI (network-admin@leaf-1) > tunnel-show

switch scope   name                                  type  vrouter-name peer-vrouter-name local-ip   remote-ip  active state error route-info     ports auto-tunnel
------ ------- ------------------------------------- ----- ------------ ----------------- ---------- ---------- ------ ----- ----- -------------- ----- ------------
leaf-1 cluster auto-tunnel-10.20.11.1_10.20.44.1 vxlan Leaf-1       Leaf-2            10.20.11.1 10.20.44.1 yes    ok          10.20.44.0/29 49    auto
leaf-2 cluster auto-tunnel-10.20.11.1_10.20.44.1 vxlan Leaf-2       Leaf-1            10.20.11.1 10.20.44.1 yes    ok          10.20.44.0/29 49    auto
leaf-1 cluster auto-tunnel-10.20.11.1_10.20.22.1 vxlan Leaf-1       Leaf-2            10.20.11.1 10.20.22.1 yes    ok          10.20.22.0/29 49    auto
leaf-2 cluster auto-tunnel-10.20.11.1_10.20.22.1 vxlan Leaf-2       Leaf-1            10.20.11.1 10.20.22.1 yes    ok          10.20.22.0/29 49    auto
leaf-3 cluster auto-tunnel-10.20.22.1_10.20.44.1 vxlan Leaf-3       Leaf-4            10.20.22.1 10.20.44.1 yes    ok          10.20.44.0/29 49    auto
leaf-4 cluster auto-tunnel-10.20.22.1_10.20.44.1 vxlan Leaf-4       Leaf-3            10.20.22.1 10.20.44.1 yes    ok          10.20.44.0/29 49    auto
leaf-3 cluster auto-tunnel-10.20.22.1_10.20.11.1 vxlan Leaf-3       Leaf-4            10.20.22.1 10.20.11.1 yes    ok          10.20.11.0/29 49    auto
leaf-4 cluster auto-tunnel-10.20.22.1_10.20.11.1 vxlan Leaf-4       Leaf-3            10.20.22.1 10.20.11.1 yes    ok          10.20.11.0/29 49    auto
leaf-7 cluster auto-tunnel-10.20.44.1_10.20.22.1 vxlan Leaf-7       Leaf-8            10.20.44.1 10.20.22.1 yes    ok          10.20.22.0/29 49    auto
leaf-8 cluster auto-tunnel-10.20.44.1_10.20.22.1 vxlan Leaf-8       Leaf-7            10.20.44.1 10.20.22.1 yes    ok          10.20.22.0/29 49    auto
leaf-7 cluster auto-tunnel-10.20.44.1_10.20.11.1 vxlan Leaf-7       Leaf-8            10.20.44.1 10.20.11.1 yes    ok          10.20.11.0/29 49    auto
leaf-8 cluster auto-tunnel-10.20.44.1_10.20.11.1 vxlan Leaf-8       Leaf-7            10.20.44.1 10.20.11.1 yes    ok          10.20.11.0/29 49    auto
```

Overlay Design

This chapter discusses the logical elements that enable L3VPN services in a Pluribus fabric, or in other words, the IP domain extension constructs across multiple dispersed physical top-of-rack switches over a generic IP transport infrastructure.

The Adaptive Cloud Fabric supports Anycast Gateway, which enables endpoints to use the same virtual MAC+IP gateway addresses on all leaf switches to support seamless endpoint mobility and increase routing efficiency. Therefore, it is possible to perform the Layer 3 gateway function for data center endpoints directly on the first-hop switch. This enables much more efficient and scalable routing, without unnecessarily increasing the amount of control plane on the switch CPU. In fact, no routing protocols like VRRP are used in this design to provide active-active redundancy for Anycast Gateway configuration pairs.

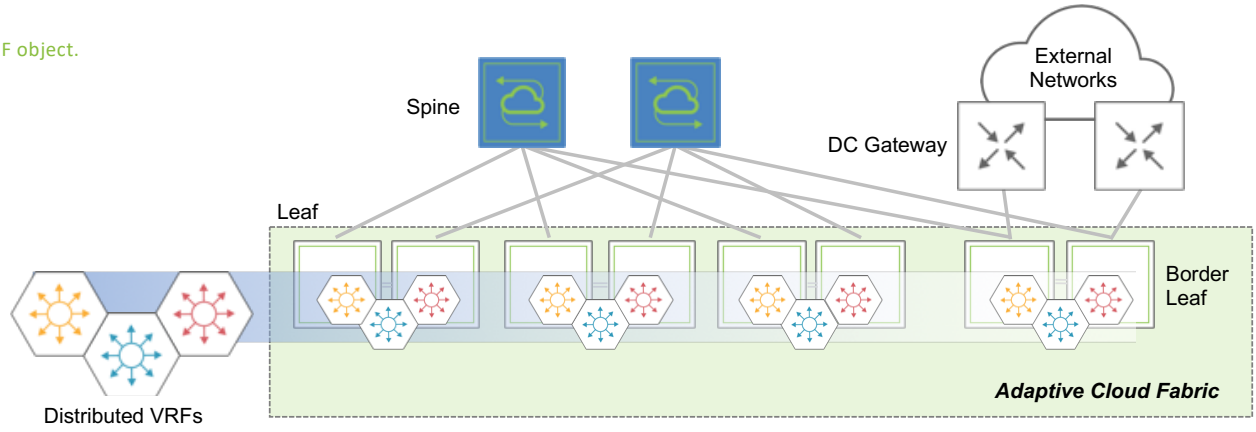
In addition, with multiple tenants, assigning different virtual routing and forwarding (VRF) instances to them is now supported in conjunction with Anycast Gateways, which means that the VRF Layer 3 segregation function can now be performed on each first-hop leaf switch or switch pair in a distributed fashion in hardware. This guarantees the maximum VRF scalability possible, limited only by the specific forwarding ASIC capabilities. Many open networking switches can support more than 1,000 VRFs per switch. The CLI image below shows the number of VRF instances created on a physical leaf switch.

```
CLI (network-admin@ebc-leaf-1*) > vrf-show count-output
name    vnet scope  anycast-mac       vrf-gw vrf-gw2 active hw-router-mac     hw-vrid
------- ---- ------ ----------------- ------ ------- ------ ----------------- -------
VRF-1   0:0 fabric 64:0e:94:40:00:02 ::      ::      no     00:00:00:00:00:00 -1
VRF_2   0:0 fabric 64:0e:94:40:00:02 ::      ::      yes    66:0e:94:65:65:90 1
VRF_3   0:0 fabric 64:0e:94:40:00:02 ::      ::      yes    66:0e:94:65:65:90 2
VRF_4   0:0 fabric 64:0e:94:40:00:02 ::      ::      yes    66:0e:94:65:65:90 3
VRF_5   0:0 fabric 64:0e:94:40:00:02 ::      ::      yes    66:0e:94:65:65:90 4
VRF_6   0:0 fabric 64:0e:94:40:00:02 ::      ::      yes    66:0e:94:65:65:90 5
[snip]
VRF_999 0:0 fabric 64:0e:94:40:00:02 ::      ::      yes    66:0e:94:65:65:90 997
Count: 999
```

The above truncated view shows that there are 999 overlay VRF instances defined on this particular switch, of which 997 are active in hardware. This example demonstrates how the fabric control plane is aimed at maximizing hardware scale; it shows that, while VRFs can be provisioned in the entire fabric, they are actually dynamically installed in hardware only when there are actual subnets and bridge domains assigned to local physical ports. A further benefit of the fabric underlay/overlay decoupling is that the spine layer does not have to implement any multi-VRF IP transport, and as such, does not require any significant VRF scale.

As represented in Figure 6, the Anycast Gateway and distributed VRF functions can be managed by relying on fabric-wide objects like the subnet, thus avoiding repeating the anycast gateway IP address and subnet prefix configuration on each physical leaf switch. For example, in a typical implementation in the industry of an Ethernet VPN (EVPN) fabric, each EVPN node requires the provisioning of an Anycast Gateway interface for each overlay subnet. Assuming a fabric of 32 leaf switches, provisioning 1000 VRFs with three subnets each would require 3,000 commands 32 times. This would equal 96,000 commands, which is a huge provisioning state, even when using centralized configuration management tools. In the case of the Pluribus Adaptive Cloud Fabric, the equivalent provisioning state is 97% fewer commands with only 3000 subnet objects.

Figure 6:
The distributed VRF object.



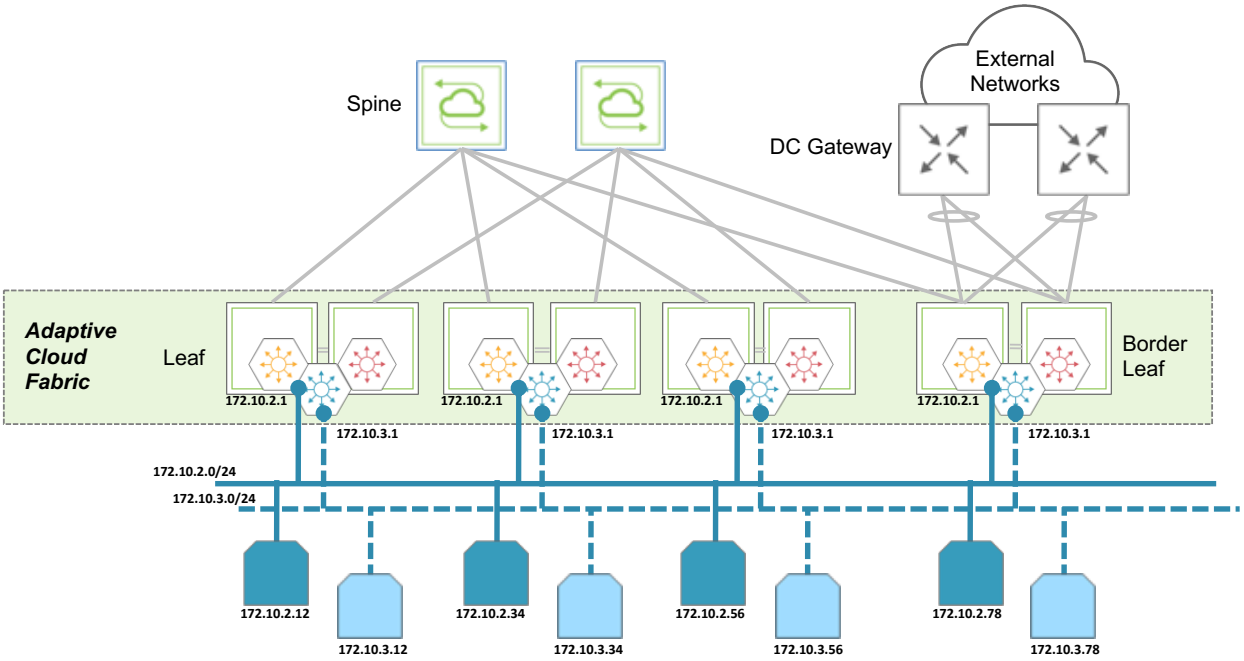
East-to-West Traffic Design Example

One use case example of a L3VPN service implemented using distributed VRFs is for optimizing east-to-west (E-W) routed traffic between subnets whose endpoints, such as virtual machines, are horizontally dispersed and are free to migrate across the entire leaf layer. Using Figure 7 as a reference, an example of E-W subnet configuration for the entire leaf fabric is provided below, where an Anycast Gateway for the prefixes 172.10.2.0/24 and 172.10.3.0/24, part of VRF-1, is assigned to the bridge domain corresponding to VXLAN VNIs 500012 and 500013:

```
CLI (network-admin@ebc-leaf-1) > subnet-create name subnet-vxlan-500012 scope
fabric vxlan 500012 network 172.10.2.0/24 anycast-gw-ip 10.10.2.1 vrf VRF-1

CLI (network-admin@ebc-leaf-1) > subnet-create name subnet-vxlan-500013 scope
fabric vxlan 500013 network 172.10.3.0/24 anycast-gw-ip 10.10.3.1 vrf VRF-1
```

Figure 7:
Distributed VRF for east-to- west subnets (blue VRF).



Similar to VRF objects, subnet objects consume hardware resources only when corresponding bridge domains are physically active on the switch; for example, when provisioned on local physical interfaces or VTEPs. The following CLI output shows an example where the same subnet object, defined once globally for the whole fabric, is not activated on the first two fabric switches:

```
CLI (network-admin@ebc-leaf-2) > subnet-show vrf VRF-1
switch     name                 scope  vnet vlan vxlan  network        vrf    anycast-gw-ip state  hw-state
---------- -------------------- ------ ---- ---- ------ -------------- ------ ------------- ------ --------
ebc-leaf-1 subnet-vxlan-500012 fabric       0    500012 172.10.2.0/24 VRF-1 172.10.2.1     ok     inactive
ebc-leaf-2 subnet-vxlan-500012 fabric       0    500012 172.10.2.0/24 VRF-1 172.10.2.1     ok     inactive
ebc-leaf-3 subnet-vxlan-500012 fabric       12   500012 172.10.2.0/24 VRF-1 172.10.2.1     ok     active
ebc-leaf-4 subnet-vxlan-500012 fabric       12   500012 172.10.2.0/24 VRF-1 172.10.2.1     ok     active
ebc-leaf-6 subnet-vxlan-500012 fabric       12   500012 172.10.2.0/24 VRF-1 172.10.2.1     ok     active
ebc-leaf-7 subnet-vxlan-500012 fabric       12   500012 172.10.2.0/24 VRF-1 172.10.2.1     ok     active
ebc-leaf-1 subnet-vxlan-500013 fabric       13   500013 172.10.3.0/24 VRF-1 172.10.3.1     ok     active
ebc-leaf-2 subnet-vxlan-500013 fabric       13   500013 172.10.3.0/24 VRF-1 172.10.3.1     ok     active
ebc-leaf-3 subnet-vxlan-500013 fabric       13   500013 172.10.3.0/24 VRF-1 172.10.3.1     ok     active
ebc-leaf-4 subnet-vxlan-500013 fabric       13   500013 172.10.3.0/24 VRF-1 172.10.3.1     ok     active
ebc-leaf-6 subnet-vxlan-500013 fabric       13   500013 172.10.3.0/24 VRF-1 172.10.3.1     ok     active
ebc-leaf-7 subnet-vxlan-500013 fabric       13   500013 172.10.3.0/24 VRF-1 172.10.3.1     ok     active
```

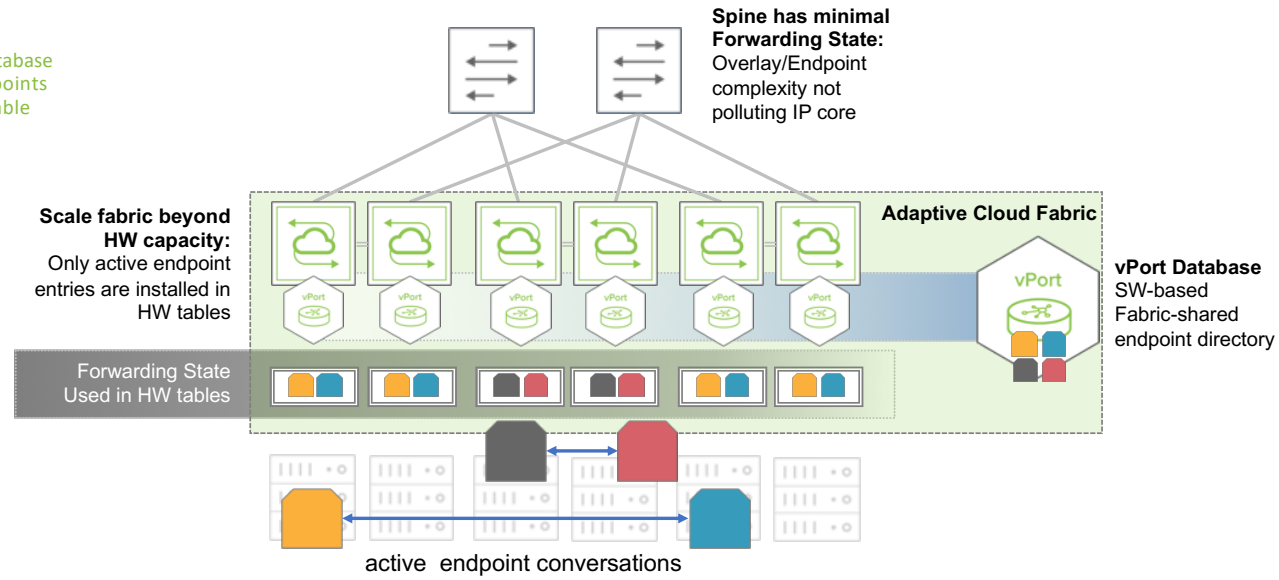
Scaling Endpoints with Conversational Forwarding

Virtual ports (vPorts) are software Layer 2 entries associated with any ports that a switch performs MAC address learning on. While a simple hardware Layer 2 table is limited in its capacity by a switch's dedicated ASIC memory size, Netvisor ONE software runs in the control plane processor's much larger DRAM memory space, and is capable of tracking a large list of Layer 2 entries, much larger than what could fit into the limited space of a typical hardware table. This logical software extension of the Layer 2 table is called the vPort database and is represented in Figure 8.

vPort database entries are persistent and are synchronized across the Fabric architecture. This allows every Fabric member to be aware of every other Layer 2 table entry across the Fabric. The history of each vPort entry is tracked and replicated; therefore, on any fabric member, for example, by using the "vport-history-show" command, it is possible to display vPort changes across different switches and across different ports over time. Instances where this capability is particularly useful are when tracking mobile endpoints/virtual machines and troubleshooting purposes.

Since the Layer 2 table is the foundation of all Layer 2 switching functions on an individual device, it also means that the overarching vPort database's information aggregates all the history of the Layer 2 activity of all the fabric members. Consequently, the vPort database can be considered the authoritative distributed endpoint directory and switching activity history book for the entire Adaptive Cloud Fabric deployment.

Figure 8:
Distributed vPort database can scale fabric endpoints beyond hardware table capacity.



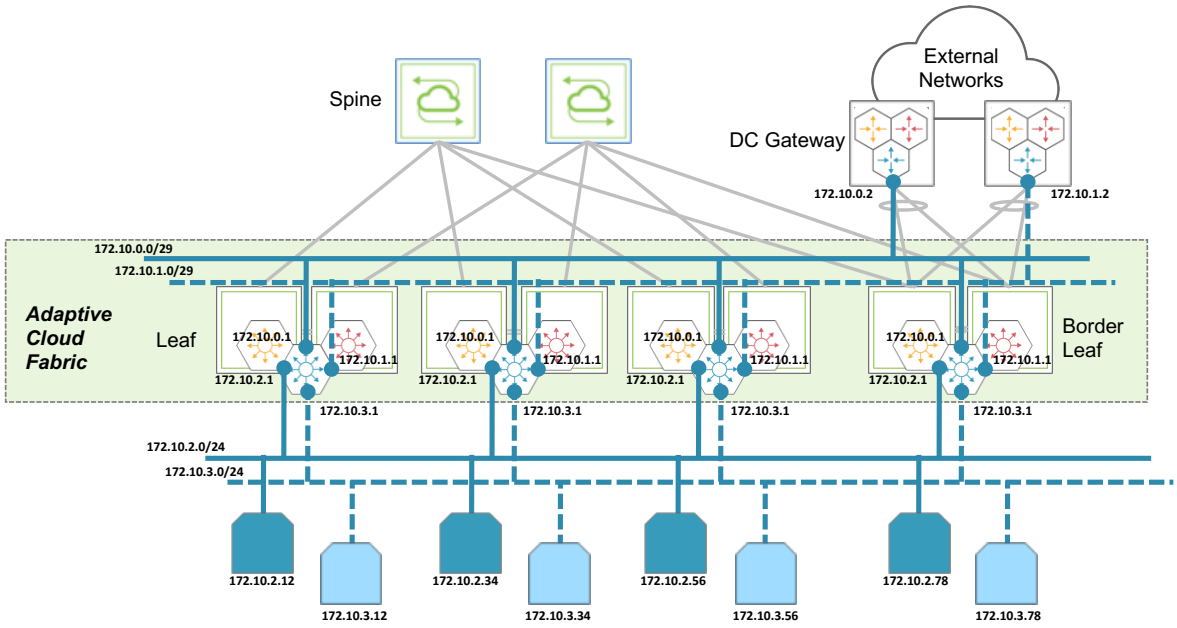
The Adaptive Cloud Fabric implements very sophisticated fabric control plane logic that significantly reduces the dependency of hardware tables for synchronizing the endpoint database across fabric members as it occurs with standard EVPN, instead using a more optimized mechanism based on the versatile vPort database.

Pluribus refers to this as vPort forwarding, but it is also known as conversational forwarding. As depicted in Figure 8, its logic consists of installing only active destination entries in hardware tables, in contrast with the EVPN approach of installing all fabric endpoint entries, regardless of what actual endpoint conversations are in progress and require hardware forwarding. When a new conversation occurs, the destination address of a generic packet can be successfully looked up in the distributed vPort database and subsequently installed in hardware, so as to avoid having to flood the packet to all its possible destinations. Consequently, Adaptive Cloud Fabric member endpoints can scale in a software-based vPort database to half a million, or 10 times higher than the hardware scale of a typical leaf switch.

North-to-South Overlay Interconnection

The previous chapter addressed the traffic paths within the same, or between different, extended subnets within the data center pod or E-W. The last component of the distributed VRF configuration enables north-to-south interconnection of L3VPN services, allowing hosts connected to subnets behind the VRFs in the fabric to access services that are outside of the data centers, including the public internet. This is achieved by pointing each VRF to a next-hop gateway, or two for equal cost redundancy. The VRF attribute defining its northbound gateways has local switch scope relevance, allowing this way to have diversity of northbound connectivity for a certain VRF based on physical location, which is very useful for L3VPN services dispersed over multiple data center locations.

Figure 9:
Distributed VRF for north-to- south subnets (blue VRF).



As represented in Figure 9, the overlay VRF discussed is now provisioned with two additional northbound subnets with subnet mask 29, which provide reachability in and out of the VRF from/to the DC gateway:

```
CLI (network-admin@ebc-leaf-1) > subnet-create name subnet-vxlan-500010 scope
fabric vxlan 500010 network 172.10.0.0/29 anycast-gw-ip 10.10.0.1 vrf VRF-1

CLI (network-admin@ebc-leaf-1) > subnet-create name subnet-vxlan-500011 scope
fabric vxlan 500011 network 172.10.1.0/29 anycast-gw-ip 10.10.1.1 vrf VRF-1
```

The VRF is also configured with the IP address of each DC gateway as default next-hop:

```
CLI (network-admin@ebc-leaf-1) > switch * vrf-modify name VRF-1  vrf-gw
172.10.0.2 vrf-gw2 172.10.1.2
```

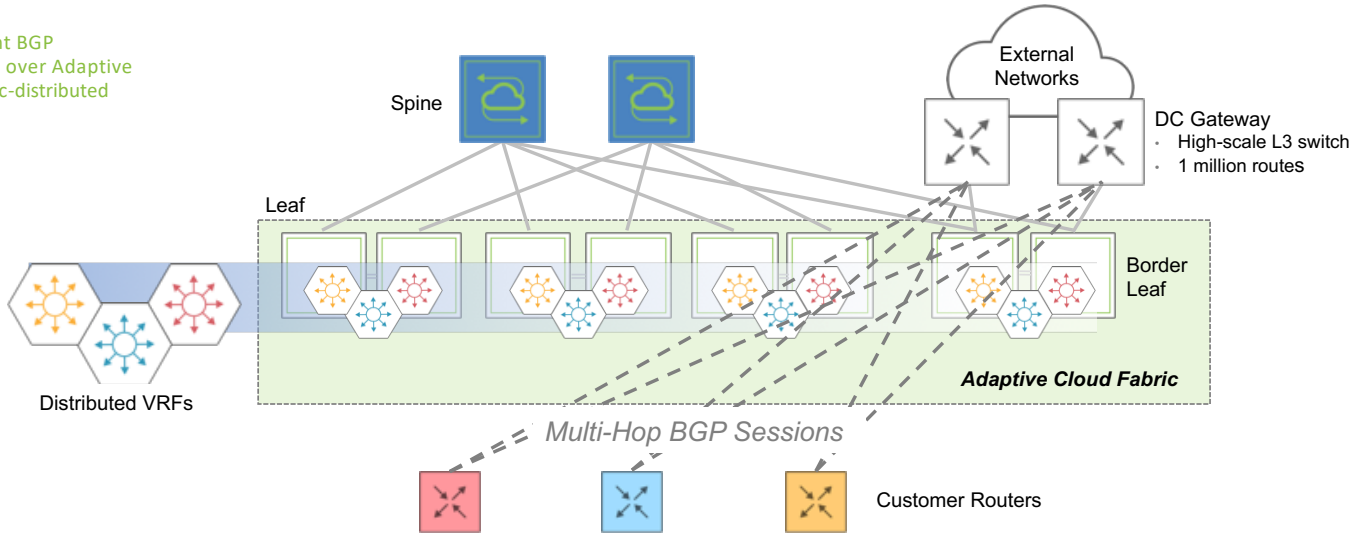
To provide inbound reachability for the VRF inside the pod, the DC gateways will need to be provisioned with a static route for the VRF subnets that need to receive traffic from external networks, using the adjacent Anycast Gateway addresses as next hop:

```
DC-Gateway-1# ip route vrf VRF_1 172.10.2.0/23 172.10.0.1
DC-Gateway-1# ip route vrf VRF_1 172.10.2.0/23 172.10.1.1
DC-Gateway-2# ip route vrf VRF_1 172.10.2.0/23 172.10.0.1
DC-Gateway-2# ip route vrf VRF_1 172.10.2.0/23 172.10.1.1
```

The number of routes that are required in order to implement L3VPNs on leaf switches and border leaf nodes is strictly limited to the number of connected subnets in each active VRF, plus two default routes per VRF. As the typical hardware routing scale in modern leaf switches is of the same order as the Layer 2 domain's capacity, this design scales well as it adapts to the hardware architectural model available in most open networking top-of-rack switches and can thus provide thousands of L3VPN services with full horizontal scale with merchant silicon economics.

One last important aspect of L3VPN services is the capability to announce external network routes to tenant speakers that connect to each distributed VRF. As discussed earlier, the Adaptive Cloud Fabric's distributed VRF is a lean object that is designed to achieve maximum L3VPN scale on open networking leaf switches like the Freedom 9572-V. As such, it minimizes control plane resource requirements by not relying on dynamic routing adjacencies. As illustrated in Figure 10, external network routes can be announced to customer routers distributed horizontally in the data center pod by establishing a two-hop BGP session with the corresponding VRF in the DC gateway. In this case, a distributed VRF provides a high-capacity, high-scale multi-tenant transit network for north-to-south bidirectional traffic, which is completely agnostic on the quantity of external and internal network routes exchanged by customer routers.

Figure 10:
Multi-tenant BGP adjacencies over Adaptive Cloud Fabric-distributed VRF transit.



The DC Gateway function, aggregating all external and internal network routes for all tenants, must be capable of scaling to the order of one million routes, one thousand VRFs and one thousand BGP sessions.

If higher routing capacity is needed globally for the entire pod, the DC gateway function and border leaf function can be scaled out, as described previously in Figure 3 of this document.

Scalable Hierarchical Layer 2 DC Interconnection

The previous chapters discussed common DC designs providing scalable east-to-west and north-to-south overlay interconnections based on distributed routing capabilities.

Certain other DC designs instead require bare-bones Layer 2 transport for both intra-DC (intra-private cloud) and inter-DC (external cloud handoff) connectivity.

The main Layer 2 design limit in such cases is that with the IEEE 802.1Q standard, a network design is constrained to *4094 possible VLAN numbers* – too few to be able to provide sufficient scalability and flexibility in most scenarios.

However, when more Layer 2 identifiers are needed, and a hierarchy of network entities can be established, the *VLAN stacking technology* (informally also called *QinQ*, after the name of Cisco's original implementation, also known as 802.1Q Tunneling) can be utilized to scale up to 16 million (4094 x 4094) identifiers using *two (concatenated, i.e., "stacked") VLAN tags instead of one*.

The IEEE organization has standardized the VLAN stacking technology with the *Provider Bridges* standard, also known as *IEEE 802.1ad*, which was later incorporated into the IEEE 802.1Q-2011 revision of the LAN/MAN standard.

IEEE 802.1ad refers to providers of services, such as transparent LAN Services in metro networks. Additionally, there are a number of data center designs that can tap the scalability of the VLAN stacking technology: for example, data center networks in which one tag (the so-called outer tag) is used to identify a data center customer, while the second tag (the so-called inner tag) is used to identify a customer service.

With this technology it is possible to identify up to 16M services for up to 4K customers. The double-tagged traffic is oftentimes handed off to a provider (for example, a cloud exchange provider), which can terminate it for end-to-end connectivity. Hence, this scheme can be used to deploy *hybrid cloud designs* in which private clouds integrate with external clouds.

In practice in such designs, QinQ (i.e., VLAN stacking) augments the basic 802.1Q capabilities by massively scaling the number of usable network identifiers organized in a hierarchy.

Pluribus combines this technology with the VXLAN-based *Adaptive Cloud Fabric* to yield an even higher degree of scalability and flexibility. In fact, VXLAN IDs (aka VNIs) are 24 bits long (vs. 12 bits of VLAN IDs), and therefore are a perfect match for *double-tagging* Layer 2 network transports.

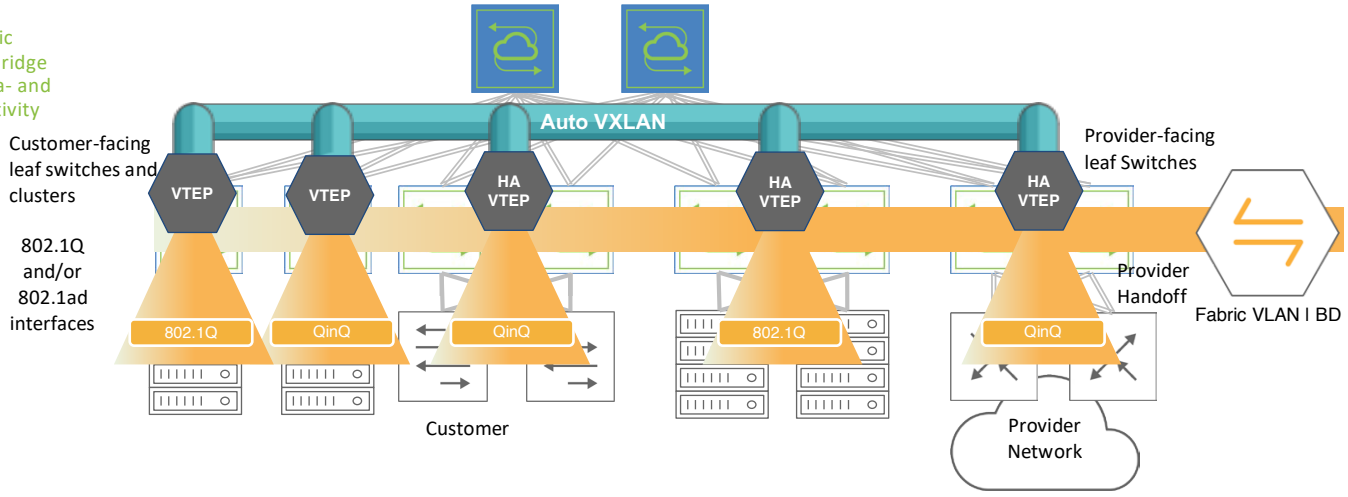
In addition, VLAN ID to VXLAN ID mappings enable a great degree of flexibility for network designers that Pluribus substantiates with a new type of configuration object called *VXLAN-based bridge domain* (BD).

VXLAN-based bridge domains are supported starting from Netvisor ONE release 5.1.2: they can be mapped to single or multiple 802.1Q VLANs as well as to dual IEEE 802.1ad VLAN tags, covering all possible design requirements.

The different available configuration models for VXLAN-based bridge domains are:

- *Single tag mapping*, in which an outer VLAN tag (for example, a customer ID) is mapped to a VNI on an IEEE 802.1ad port and the inner VLAN tags are preserved inside the VXLAN encapsulation. This type of mapping can be used on customer-facing "QinQ access" interfaces.
- *Double tag mapping*, in which an outer VLAN and inner VLAN tag pair is mapped to a VNI on an IEEE 802.1ad port (traffic is received double-tagged in ingress and is marked with two tags in egress after VXLAN decapsulation). This type of mapping can be used on multi-VLAN ports (sometimes called "QinQ trunks") facing, for example, an external cloud provider.
- *Single 802.1Q tag mapping*, in which a single 802.1Q VLAN (or multiple 802.1Q VLANs) are mapped to a common VNI (for example, for inter-DC communication within the same customer's private cloud network).

Figure 11:
Hierarchical fabric structure using bridge domains for intra- and inter-DC connectivity



The various configuration models of bridge domains yield an unprecedented level of flexibility in terms of advanced Layer 2 transport services, with the ability to reuse VLANs across tenants, support QinQ hierarchies and handoff links and aggregate multiple VLANs in the same overlay construct to support high-scale L2 tenant services.

Conclusion

The Pluribus Adaptive Cloud Fabric architecture built on open networking hardware offers a scalable, cost-effective and robust solution to build a reliable, efficient and high-scale IP cloud fabric for the data center. The fabric addresses all the most common requirements that such a solution usually entails, including multi-tenant scale, redundancy, predictable growth capability, fast convergence in case of a failure event and multi-tenancy. Additionally, it provides a set of differentiators that are unique in the industry, such as:

- Single management fabric for each pod functional block
- Intelligent and scalable L3VPN and L2VPN services configurable as single fabric-wide objects
- Built-in flow analytics for L2VPN and L3VPN services with the Insight Analytics platform
- Optimization and simplification of control plane architecture to minimize the need for expensive high-performance nodes, such as switches with very large routing tables

The Pluribus Adaptive Cloud Fabric is powered by the deployment-proven Netvisor ONE OS, which is an open, secure and programmable next-generation network OS that is purpose-built to optimize the power and performance of bare metal open networking hardware. Deployment-proven in production in mission-critical enterprise and carrier networks, Netvisor ONE meets the most stringent performance requirements and delivers the maximum levels of reliability and flexibility at scale without compromise.

Netvisor ONE runs on many Open Compute Project (OCP) and Open Network Install Environment (ONIE) hardware-compliant switches, including devices from D-Link Systems, Dell EMC and Edgecore Networks, as well as the Pluribus Freedom Series of network switches. This flexibility allows organizations the choice of open networking hardware to build scale-out networks with 10, 25, 40 or 100 Gigabit Ethernet interfaces. This allows an entire data center to be built with only a few physical switch models to improve operational consistency, lower costs and simplify sparing strategies.

Pluribus Networks, Inc.
www.pluribusnetworks.com