Security Level:

30/ May/ 2017

VXLAN
Thinking outside the (DC) Box

christian.kuster@huawei.com

HUAWEI

Agenda

Concept

History

Overlay war

Deployement Scenarios

Network architecture evolution

Huawei Solution overview

What is VXLAN ?

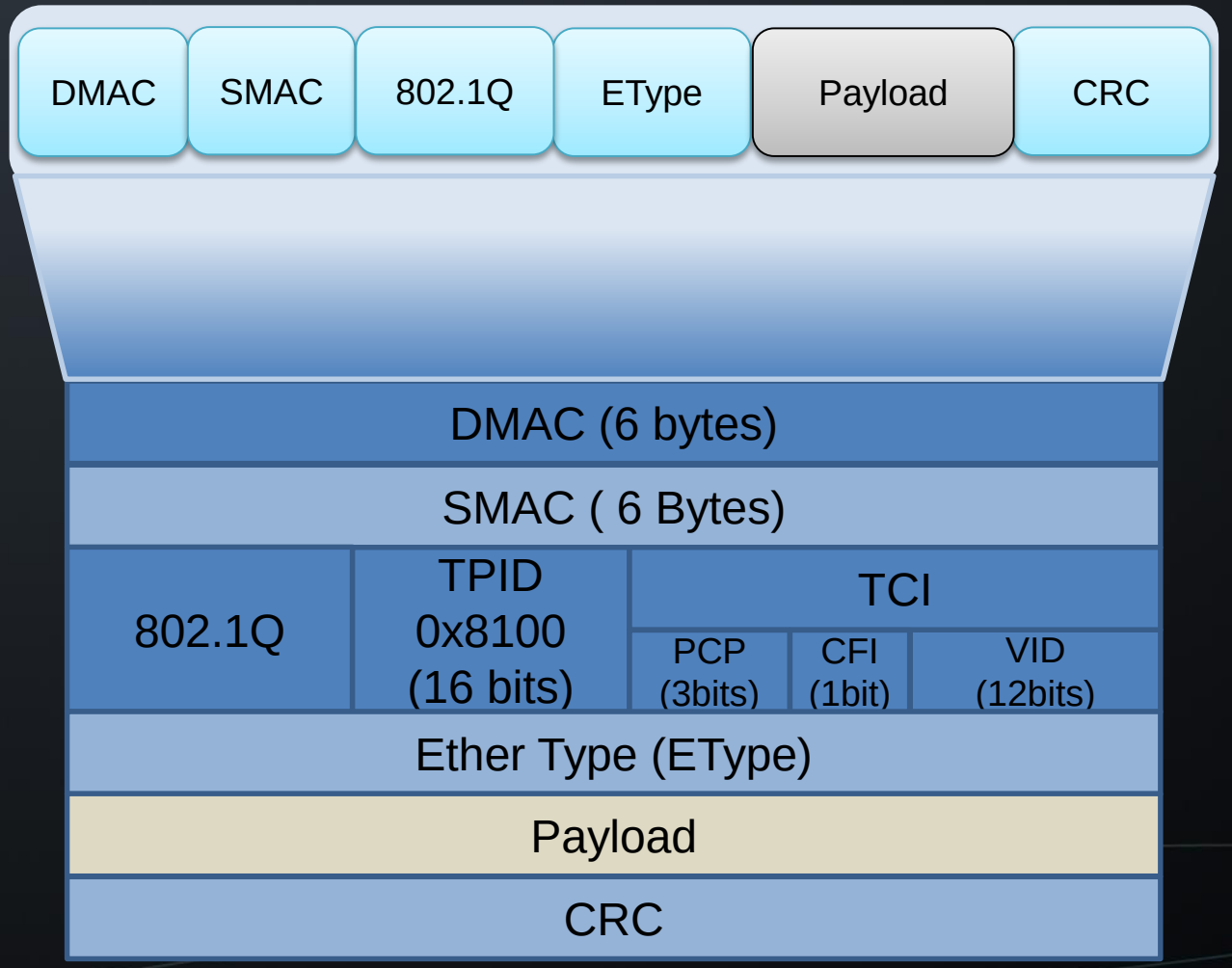
- Virtual Extensible LAN (VXLAN) is a **standard** encapsulation protocol for running an overlay network on **standard** Layer 3 infrastructure.
- An overlay network is a virtual network that is built on top of existing Layer 2 and Layer 3 network technologies to support **elastic** network architectures.
- VXLAN makes it easy for network engineers to scale out a networking environment while logically isolating services and tenants
- VXLAN separates network infrastructure transport from network service transport
- The primary goal of VXLAN is to extend the virtual LAN (VLAN) address space by adding a 24-bit segment ID and increasing the number of available IDs to 16 million

Ethernet – IEEE 802.1Q

Classical Ethernet Frame with IEEE 802.1Q

- VLAN space expressed by 12 bits VID
- Maximum number of segments limited to 4096 VLAN IDs

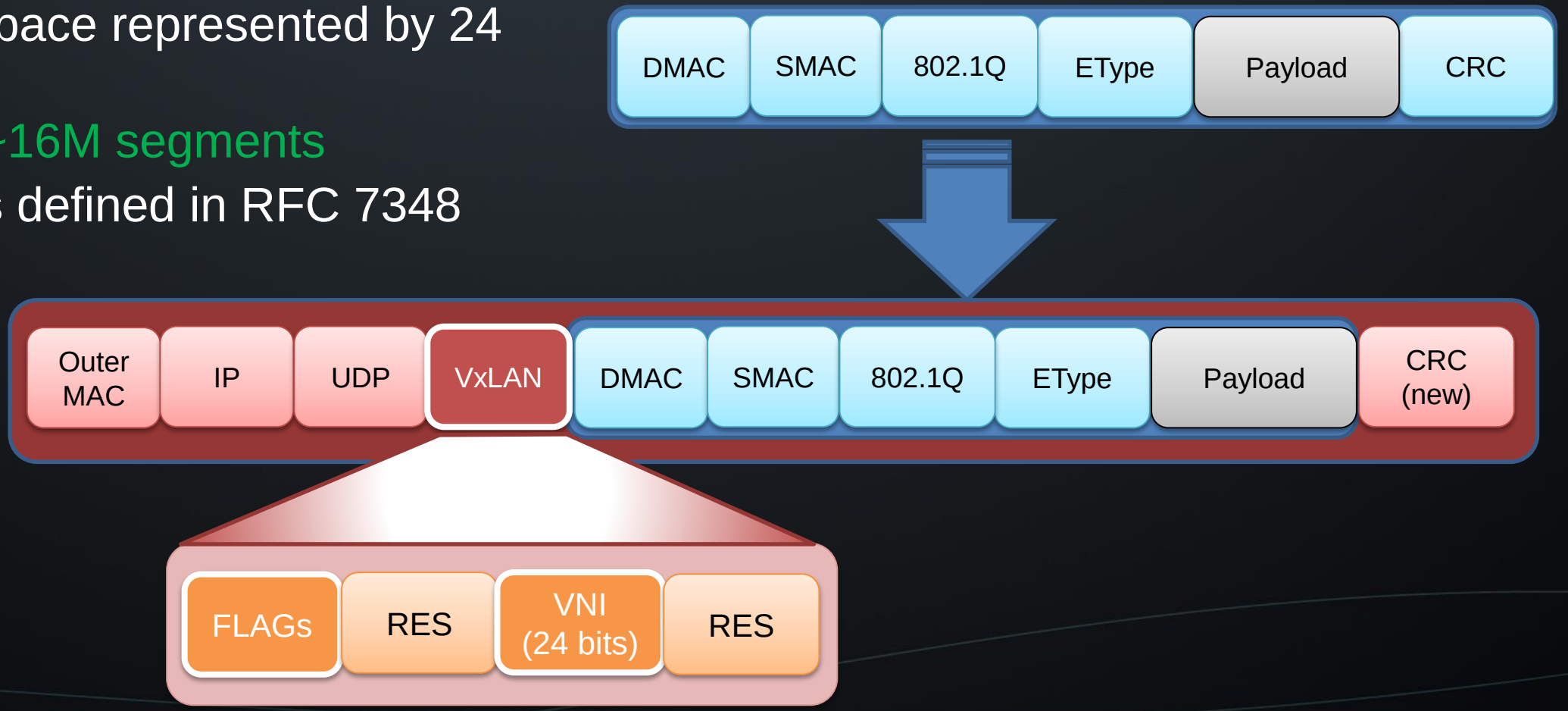
TPID – Tag Protocol Identifier
TCI – Tag Control Information
PCP – Priority Code Point
CFI – Canonical Format Indicator
VID – VLAN Identifier



VXLAN – Virtual Extensible LAN

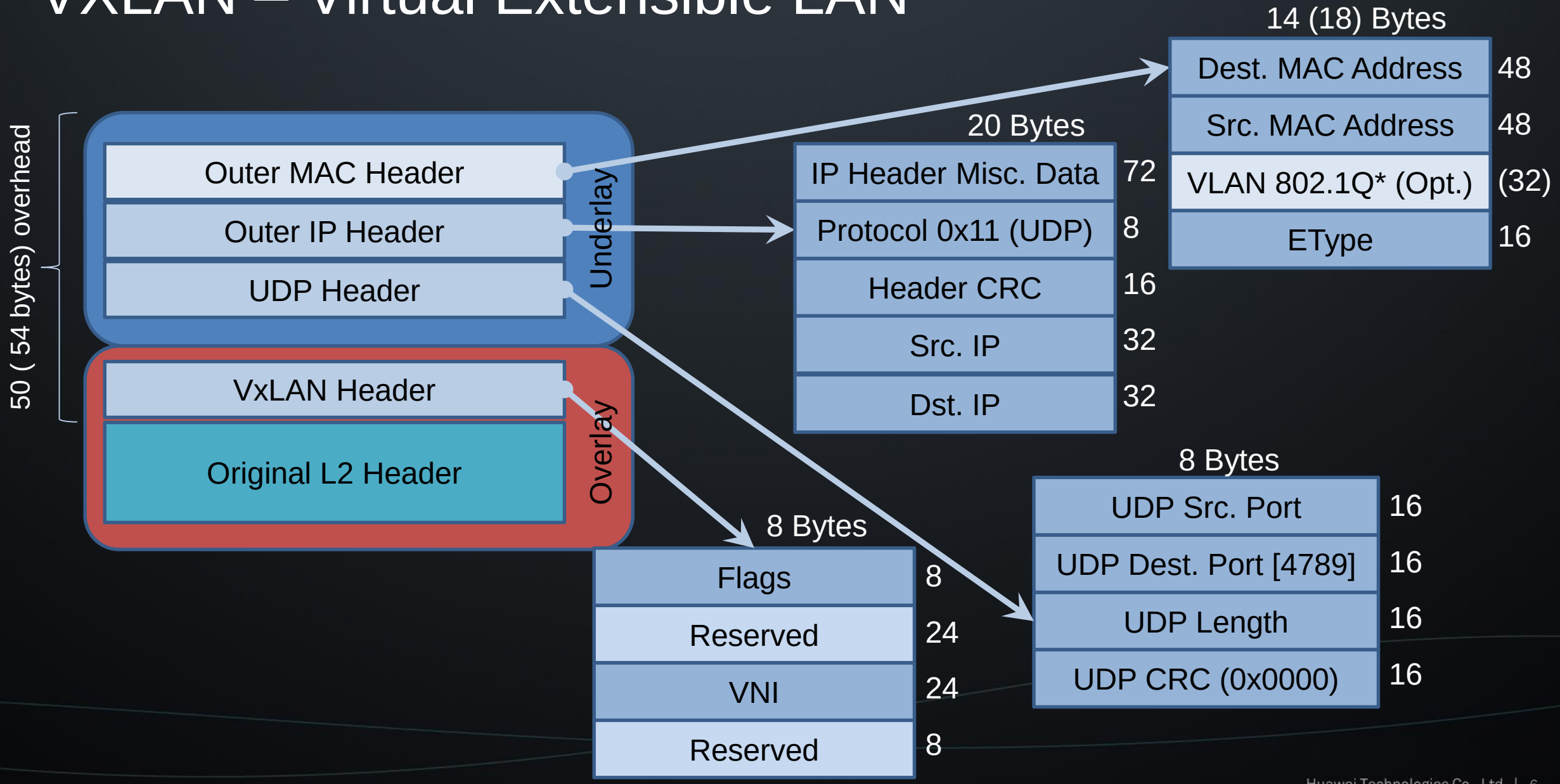
VXLAN – Virtual Extensible LAN

- VXLAN space represented by 24 bits VNI
- Support ~16M segments
- VXLAN is defined in RFC 7348



VNI/VNID – VXLAN Network Identifier

VXLAN – Virtual Extensible LAN



VXLAN – Virtual Extensible LAN

VXLAN header

Flags (8 bits): where the I flag MUST be set to 1 for a valid VXLAN Network ID (VNI). The other 7 bits (designated "R") are reserved fields and MUST be set to zero on transmission and ignored on receipt.

VXLAN Segment ID/VXLAN Network Identifier (VNI): this is a 24-bit value used to designate the individual VXLAN overlay network on which the communicating end systems are situated. End systems in different VXLAN overlay networks cannot communicate with each other.

Reserved fields (24 bits and 8 bits): MUST be set to zero on transmission and ignored on receipt.

VXLAN – Virtual Extensible LAN History

History (source: IETF datatracker):

- draft-mahalingam-dutt-dcops-vxlan-00.txt (August 27 2011)
- draft-mahalingam-dutt-dcops-vxlan-01.txt (February 24 2012)
- draft-mahalingam-dutt-dcops-vxlan-02.txt (August 22 2012)
- draft-mahalingam-dutt-dcops-vxlan-03.txt (February 22 2013)
- draft-mahalingam-dutt-dcops-vxlan-04.txt (May 8 2013)
- draft-mahalingam-dutt-dcops-vxlan-05.txt (October 17 2013)
- draft-mahalingam-dutt-dcops-vxlan-06.txt (November 06 2013)
- draft-mahalingam-dutt-dcops-vxlan-07.txt (January 23 2014)
- draft-mahalingam-dutt-dcops-vxlan-08.txt (February 03 2014)
- draft-mahalingam-dutt-dcops-vxlan-09.txt (April 10 2014)

Current state (May 2017):

- RFC 7348 (August 26 2014): Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks

- draft-ietf-nvo3-vxlan-gpe-04 (April 27, 2017): Generic Protocol Extension for VXLAN (VXLAN GPE)
  - VXLAN GPE is intended to extend the existing VXLAN protocol to provide protocol typing, OAM, and versioning capabilities

VXLAN – Virtual Extensible LAN History

Involved in RFC:

- M.Mahalingam VMware
- D.Dutt Cisco
- K.Duda Arista
- P.Agarwal Broadcom
- L. Kreeger Cisco
- T. Sridhar VMware
- M.Bursell Citrix
- C.Wright Red Hat

Early Backers include:

- Huawei (December 2012)
- Juniper Networks
- Pica8
- Cumulus Networks
- Dell
- Mellanox
- FreeBSD
- OpenBSD
- Joyent

Other Overlay Encapsulations

- NVGRE
  - **N**etwork **V**irtualization using **G**eneric **R**outing **E**ncapsulation (NVGRE). Generic Routing Encapsulation (GRE) is used to tunnel layer 2 packets over layer 3 networks. NVGRE is described in the IETF RFC 7637. Its principal backers are (were) Microsoft and HP.
- STT
  - Stateless Transport Tunneling (STT) is an encapsulation mechanism used by Vmware for communication between server based vSwitches. Based on TCP and a bit more complicated, mainly because it was designed to carry large data packets, up to 64 Kbytes. Described IETF draft-davie-stt-08 (expired). Proposed by Nicira (now acquired by VMware).
- GENEVE
  - **GE**neric **NE**twork **V**irtualization **E**ncapsulation. Based on UDP and promises to address the perceived limitations of the earlier specifications and support all of the capabilities of VXLAN, NVGRE and STT. Described in draft-ietf-nvo3-geneve-04.

Other Overlay Encapsulations and Protocols

- L2TPv3
  - Statefull Tunnel Protocol based on IP directly or UDP.
  - Modification to L2TP(v2) to extend the capability to provide pseudowires not only for PPP but also for other L2 protocols.
  - Defined in IETF RFC 3931, modified by RFC 5641.
  - Emerged from L2F (RFC 2341) via L2TP (RFC 2661)

SW Overlay Vendor Ecosystems

VMWARE:

NSX for Multihypervisors – Openflow based – EOA 2014 – EOGS 2016
Superseded by **NSX-T** (Open vSwitch based)

- **VXLAN**
- STT
- GRE
- GENEVE

NSX for vSphere (former Nicira solution)

- **VXLAN encapsulation**
- all in kernel modules (e.g. dVS, dFW...)
- Propriatery control plane (communication of modules with controller – user world agent)

The two NSX deployements are NOT compatible with each other

SW Overlay Vendor Ecosystems

IBM: DOVE (Distributed Overlay Virtual Ethernet)

Logical components of the DOVE architecture:

- DOVE controllers
- DOVE switches (abbreviated as dSwitch)

DOVE controllers perform management functions, and one part of the control plane functions across DOVE switches.

DOVE switches perform the encapsulation of layer 2 frames into UDP packets using the **Virtual Extensible LAN (VXLAN)** frame format. DOVE switches are running as part of virtual machine hypervisors

HW Overlay Vendor Ecosystems

Cisco: ACI (Application Centric Infrastructure)

To a large extent, ACI is no different than what has been deployed over the past several years in enterprise data centers.

The main difference is the management and policy framework which is based on the APIC (Application Policy Infrastructure Controller).

A leaf-spine ACI fabric is a standard Layer 3 IP fabric which uses **virtual extensible local area network (VXLAN)** as the overlay with HW based VXLAN Gateways

The “DC Network overlay” war

- Three network overlay protocols were aiming for standardization: VXLAN, NVGRE and STT.
- Two protocols to perform edge virtual bridging in hardware: 802.1qbg virtual Ethernet port aggregation and 802.1qbh VN-TAG.
- Two standard protocols for doing multi-path Ethernet --TRILL and Shortest Path Bridging.
- Three proprietary multi-path Ethernet protocols: Brocade VCS, Cisco FabricPath and HP's IRF. Yes, all three really are proprietary.
- There's also OpenFlow, which changes how Ethernet paths are defined in hardware.
- Finally, there are two proprietary protocols for Wide Area Ethernet, Cisco's Overlay Transport Virtualization (OTN) and HP's Ethernet Virtual Interconnect.

That's 13 protocols, all aimed at modernizing Ethernet. With the exception of the last two, 11 are focused on virtual networking and easing operational problems with virtual machines and networking.

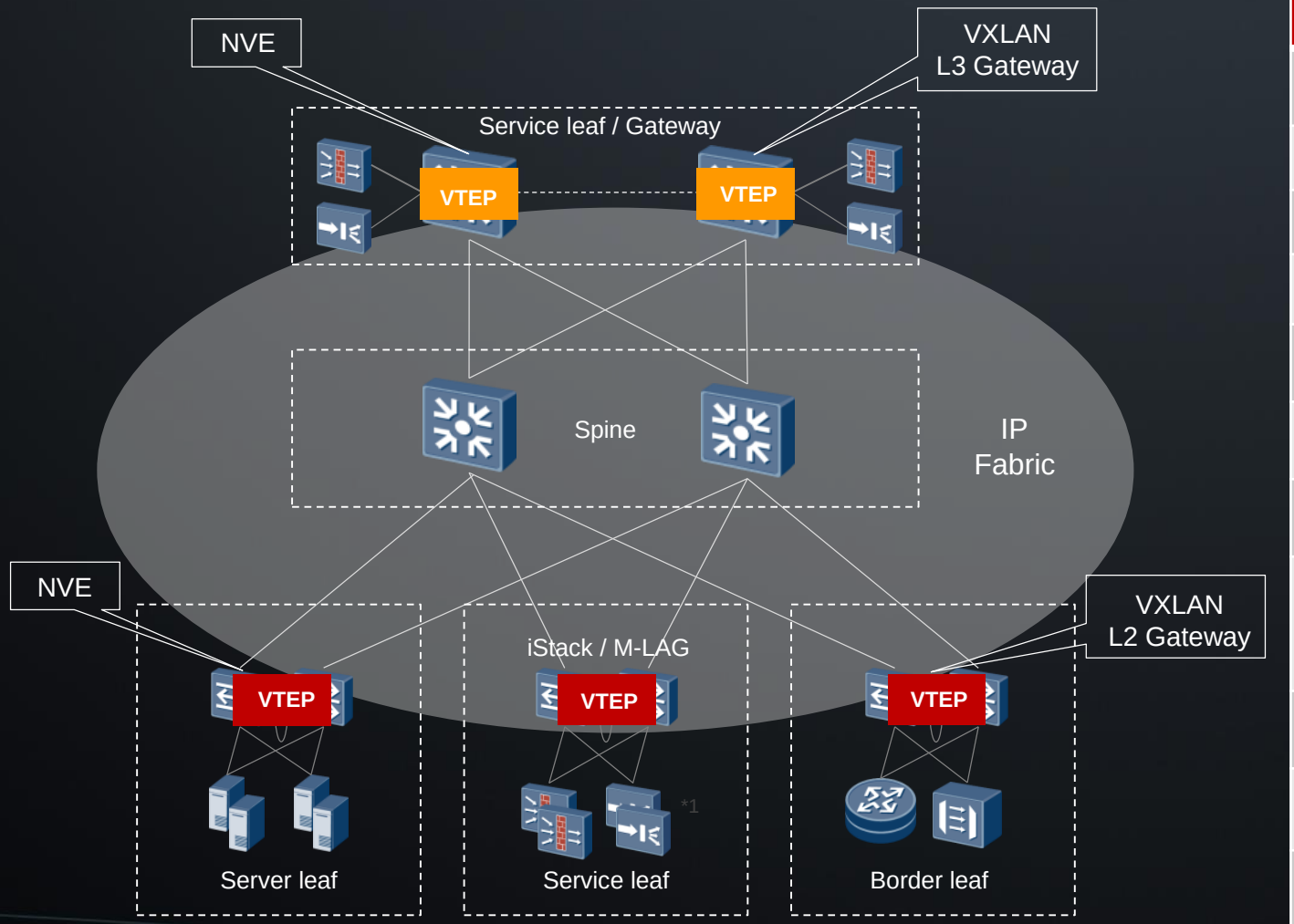
DC network overlay war comes to an end

- What enterprises and service providers really need is a single overlay standard that everyone from hypervisor vendors to hardware vendors can easily implement.
- Once that's in hand, the networking industry as a whole can move onto more interesting problems like e.g. programmability, tenant isolation and L4-L7 service insertion.

And the winner is:

VXLAN

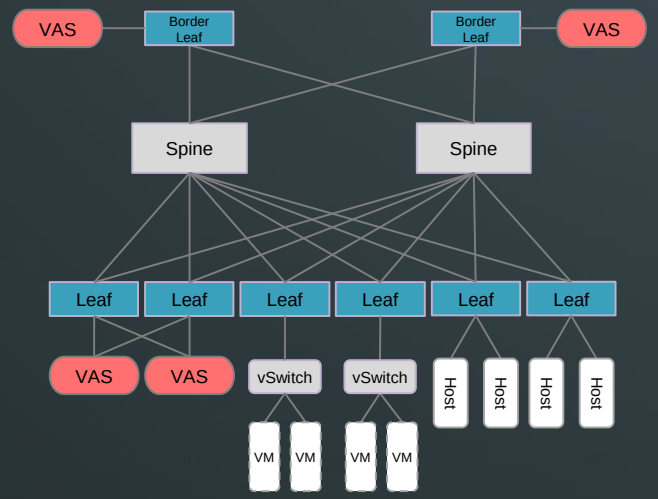
VXLAN Terms



Term	Description
Fabric	A basic physical network for a data center, which is composed of a group spine and leaf nodes.
Spine	Core node of a VXLAN fabric network, which uses high-speed interfaces to connect to functional leaf nodes and provides high-speed IP forwarding.
Leaf	An access node that is deployed on a VXLAN fabric network to connect various network devices to the VXLAN network.
Service leaf	A functional leaf node that connects L4-L7 value-added service devices, such as firewall and LB, to the VXLAN fabric network.
Server leaf	A functional leaf node that connects computing resources (virtual or physical servers) to the VXLAN network.
Border leaf	A functional leaf node that connects to a router or transmission device and forwards traffic sent from external networks to the data center.
NVE	Network virtualization edge, a network entity that implements network virtualization. NVE nodes establish an overlay virtual network on the underlay Layer 3 basic network.
VTEP	VXLAN tunnel endpoints that are deployed on NVE nodes and responsible for VXLAN packet encapsulation and decapsulation. VTEPs are connected to the physical network and assigned IP addresses (VTEP IP) of the physical network. VTEP IP addresses are independent of the virtual network. A local VTEP IP address and a remote VTEP IP address identify a VXLAN tunnel.
VNI	VXLAN network identifier that identifies a VXLAN segment. Traffic sent from one VXLAN segment to another must be forwarded by a VXLAN L3 gateway.
VXLAN L2 GW	Also called a VXLAN bridge, used to transmit non-VXLAN traffic to the VXLAN network and L2 communication within the VXLAN network. It is represented by the symbol VTEP
VXLAN L3 GW	Also called a VXLAN router or VXLAN IP gateway, used for communication between subnets on a VXLAN network. Unless otherwise specified, "gateway" mentioned in this chapter refers to a VXLAN L3 gateway. It is represented by the symbol VTEP

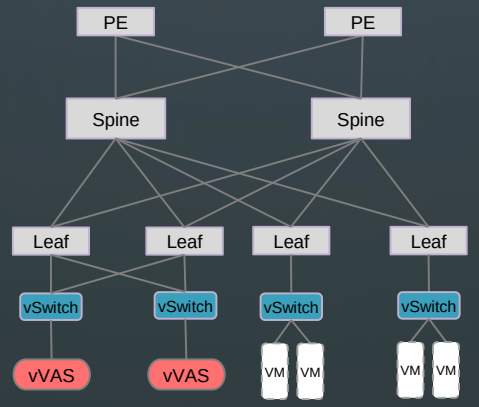
Typical VXLAN deployment scenarios - DCN

Hardware overlay network



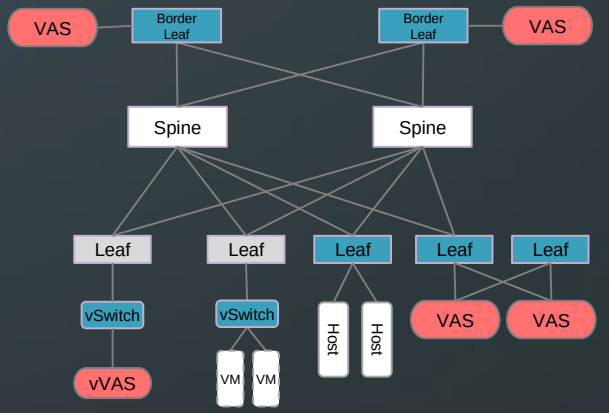
- All NVE nodes are deployed on hardware switches.
- Gateways are deployed on the spine or border leaf devices in **centralized** mode.
- VAS devices are attached to the gateways or service leaf nodes.
- vSwitches connect to the fabric network through VLANs configured on them.

Software overlay network



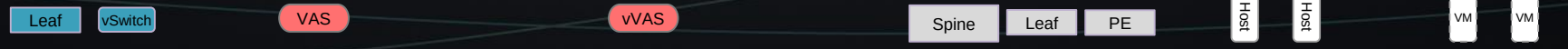
- All NVE nodes are deployed on vSwitches.
- Gateways are deployed on vSwitches in **distributed** mode.
- VAS devices are attached to vSwitches.

Hybrid overlay network



- NVE nodes are deployed on hardware switches and vSwitches.
- Gateways for VMs are deployed on vSwitches in **distributed** mode, and gateways for physical hosts are deployed on the spine nodes in **centralized** mode.
- **VAS devices can be attached to the gateways, service leaf nodes and/or vSwitches**

Symbols:

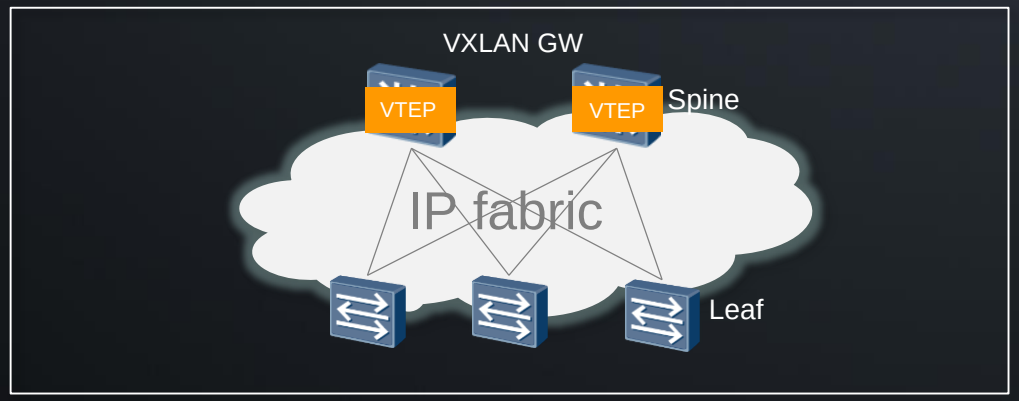


NVE devices (VTEP) | Value-added services (FW,LB) | Virtual Value-added services (vFW,vLB) | Common network devices | Physical servers | Virtual machines

VXLAN Gateway deployment design

Note: When IT/DCN architects and engineers talk about VXLAN gateways, they refer typically to the **VXLAN L3 Gateways**.

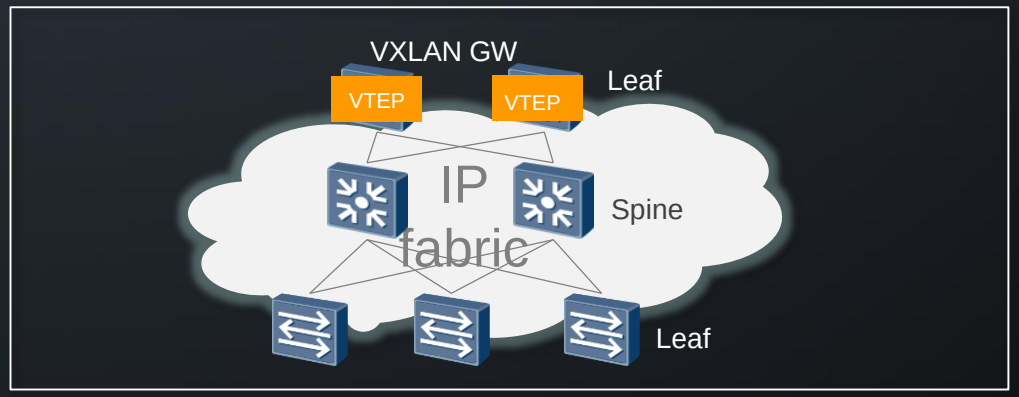
Collapsed Gateway and Spine



Characteristics

- The convergence deployment reduces the number of network devices and lowers the network deployment cost.
- The gateway nodes are closely coupled with the spine nodes, making network expansion difficult. This deployment is applicable to a data center that does not need to be expanded in the near future.
- Gateways cannot be deployed in multi-group mode.

Decoupled Gateway and Spine



Characteristics

- The decoupling deployment facilitates network expansion. Expansion of the spine, leaf, or gateway nodes will not greatly affect the other nodes.
- Multiple groups of gateways can be deployed on a large-sized network.
- Gateways can be deployed in multi-group, multi-active mode.

VXLAN Control plane options

The standard VXLAN defines the encapsulation (forwarding plane).

But what about the control plane? The standard does not really define a «standard» control plane (suggests Multicast though for handling BUM traffic).

Several option exist:

For static tunnels (manual) with «traditional» MAC learning:

- **Unicast with HER (Head end replication)**
- Multicast

«SDN»

- SDN («Contro plane up» aka OpenFlow based separation of control and forwarding plane)

Dynamic Tunnels with BGP based MAC (IP) signaling:

- BGP EVPN (L2 Overlay)
- MP-BGP (L3 Overlay) – **Huawei** L3 VPN solution: VRF – VNI mapping

VXLAN Services

Recap:
We can provide the following overlay services with VXLAN:

- L2 VPN
- L3 VPN

with the following deployement scenarios:

- Software based - Pure OTT
  - Physical network devices just run IP, virtual network devices – aka SW provide VXLAN gateway funtions
- Hardware based - NVE
  - HW network edge devices provide VXLAN gateway funtions, Spines - aka core devices, just run IP
- Hybrid
  - A mix of the two above

Typical Service Provider Services

- L2 VPN
- L3 VPN

- These services are provided today (since ~ 2001-2007) by MPLS based services, e.g. VPLS (2 different standards) for L2VPN, MPLS L3 VPN, now emerging EVPNoMPLS for L2 VPN.
- Pure HW based solutions.
- You absolutely require MPLS capable HW devices end-to-end under your administrative control (and have negotiated specific NNIs with specific designs/agreements for each Service providers.

MPLS in the Datacenter ?

- One might expect that MPLS in the data center would be first choice.
  - ➢ But almost no new Datacenter Network is built with MPLS
- The virtualization capabilities that MPLS brings with IP VPNs and E-VPNs are much needed in modern data centers.
- Virtualization inside a data center needs to be extended outside the data center, either to another data center, or to a VPN customer in a branch office; both of these use MPLS.
- MPLS hasn’t penetrated the DC to any great extent.

The question is, what is the fundamental reason why not?

Why not MPLS in the Datacenter ?

- MPLS is (Supposedly) Complex
  - MPLS is thought to be hard to configure, manage, debug and troubleshoot because of the flexibility of the underlying protocols and the number of knobs and whistles.
  - Isolating issues can be difficult. Mediating between dueling vendors (It’s her broken code! No, it’s his misinterpretation of the standard!) adds a non-technical dimension to the problem.
  - The fact that MPLS control planes appear to have the complexity effect is a sad testament to something having gone south somewhere along the line.

Why not MPLS in the Datacenter ?

- MPLS is Expensive
  - Because of the flexibility that MPLS and the related standards demand
  - Because of the robust and complex control planes that is required with MPLS hardware
  - Because of the (lack of) economy of scale.

Neither MPLS Edge nor MPLS Core functionality is commodity.

Network Architecture Evolution

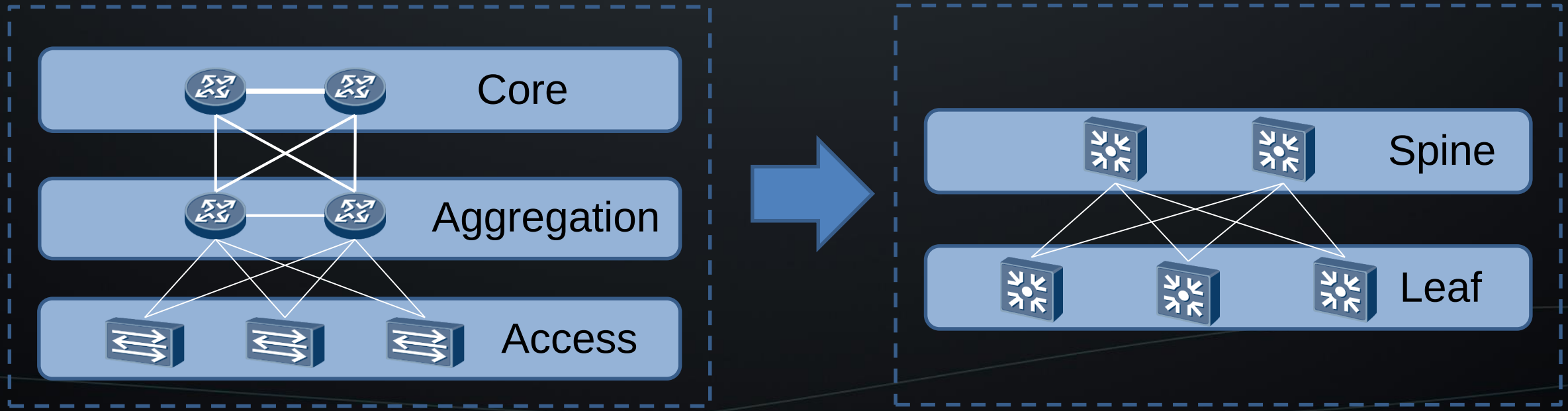
There is a new *(in fact, not so new anymore)* trend for network topology design standards — creating a **fast, predictable, scalable, and efficient** communication architecture.

This architecture originated in the data center... Sorry: CLOUD

Leaf - Spine

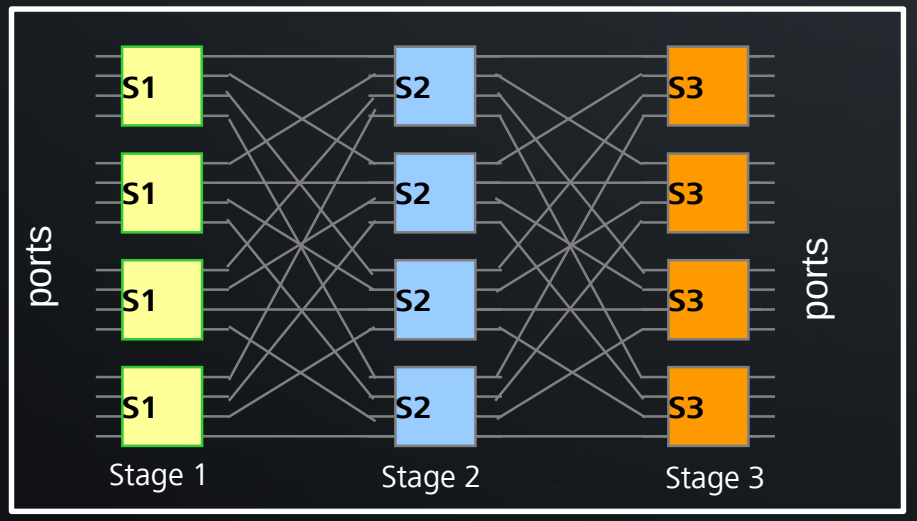
Network Architecture Evolution

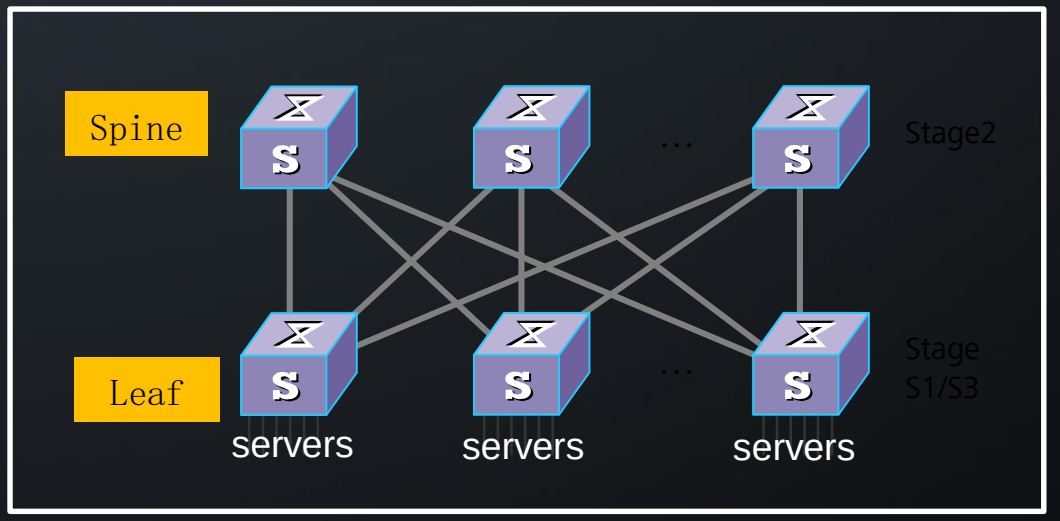
- With the increased focus on **massive data transfers** in the network, the aging three-tier design is being replaced with what is being called the Leaf-Spine design.
- Leaf-Spine architecture is adaptable to the continuously **changing needs** of evolving networks.
- All devices are always exactly the same number of segments away and contain a predictable and consistent amount of delay or latency for transported data



Leaf – Spine origin: Clos

Clos network is a kind of multistage circuit switching network, first formalized by Charles Clos in 1952, which represents a theoretical idealization of practical multi-stage telephone switching systems. Clos networks are required when the physical circuit switching needs to exceed the capacity of the largest feasible single crossbar switch. (source Wikipedia)





CLOS architecture:

- Proven architecture in the design of switching fabric modules
- Non-blocking for any port-port switching

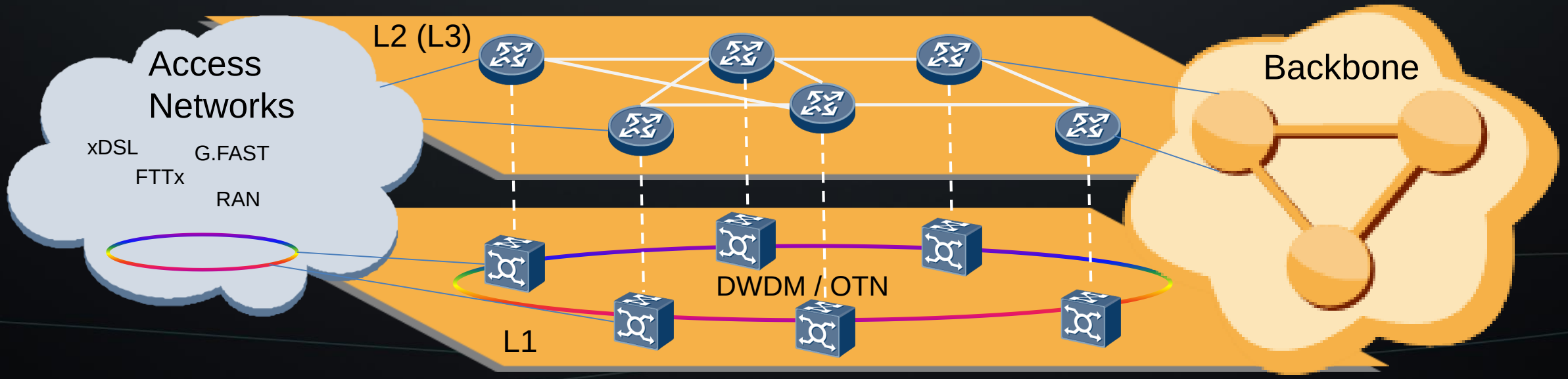
Leaf/Spine architecture characteristics:

- First used in data center networking
- Achieving non-blocking switching for servers

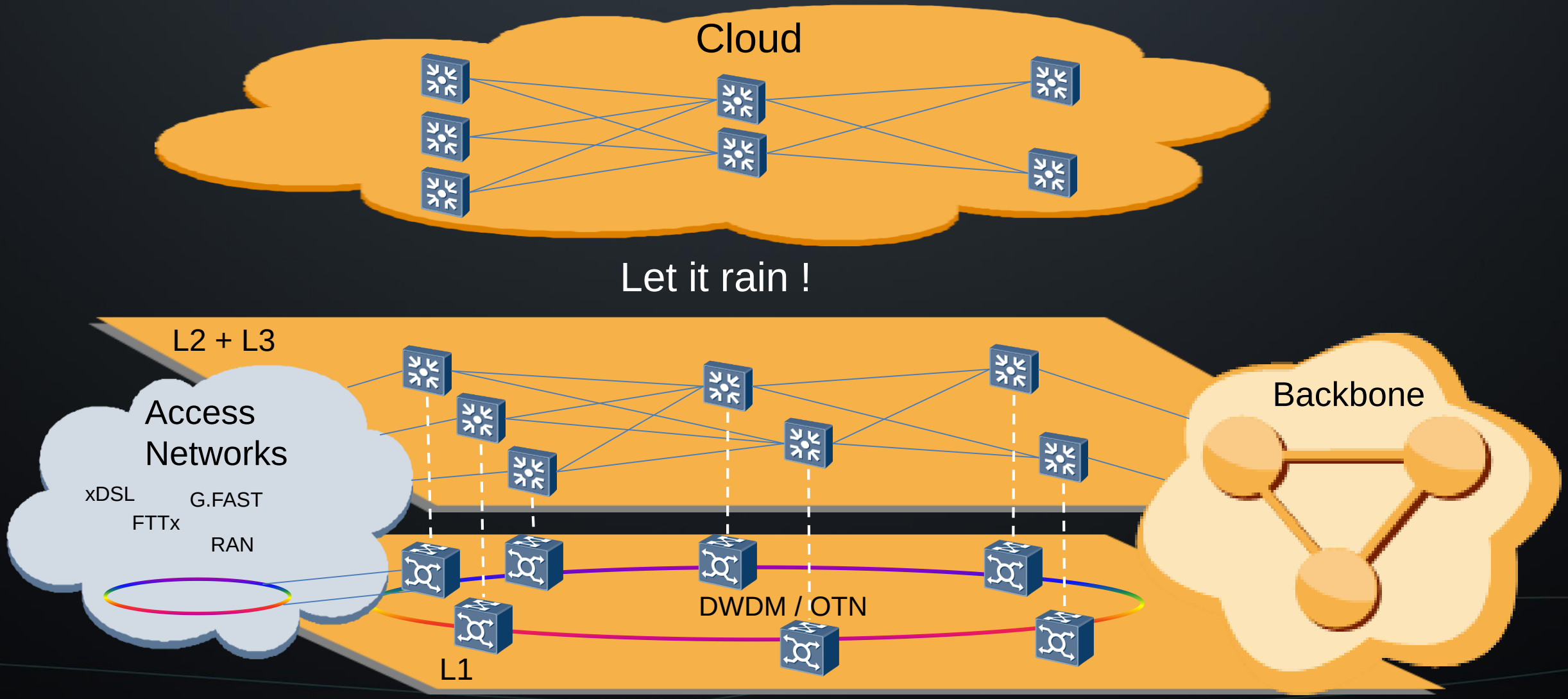
Network Architecture Evolution in the Metro

Traditionally, Metro Network Architectures follow:

- Some sort of the legacy 3 tier model with access, aggregation and core network elements
- the underlying ring structure of the L1 transport infrastructure
- just use L1 transport



Network Architecture Evolution in the Metro



VXLAN in the Metro ?

- One might expect that VXLAN in the Metro would be first choice.
- The virtualization capabilities and simplicity that VXLAN brings with IP VPNs and E-VPNs are much needed in modern Metro networks.
- Virtualization inside a Metro Network needs to be extended outside the Metro Network, either to another Metro Network, to a data center, or to a VPN customer in a branch office.

The question is: What is the fundamental reason why not?

The answer is:

There is no reason !

Why VXLAN in the Metro Network ?

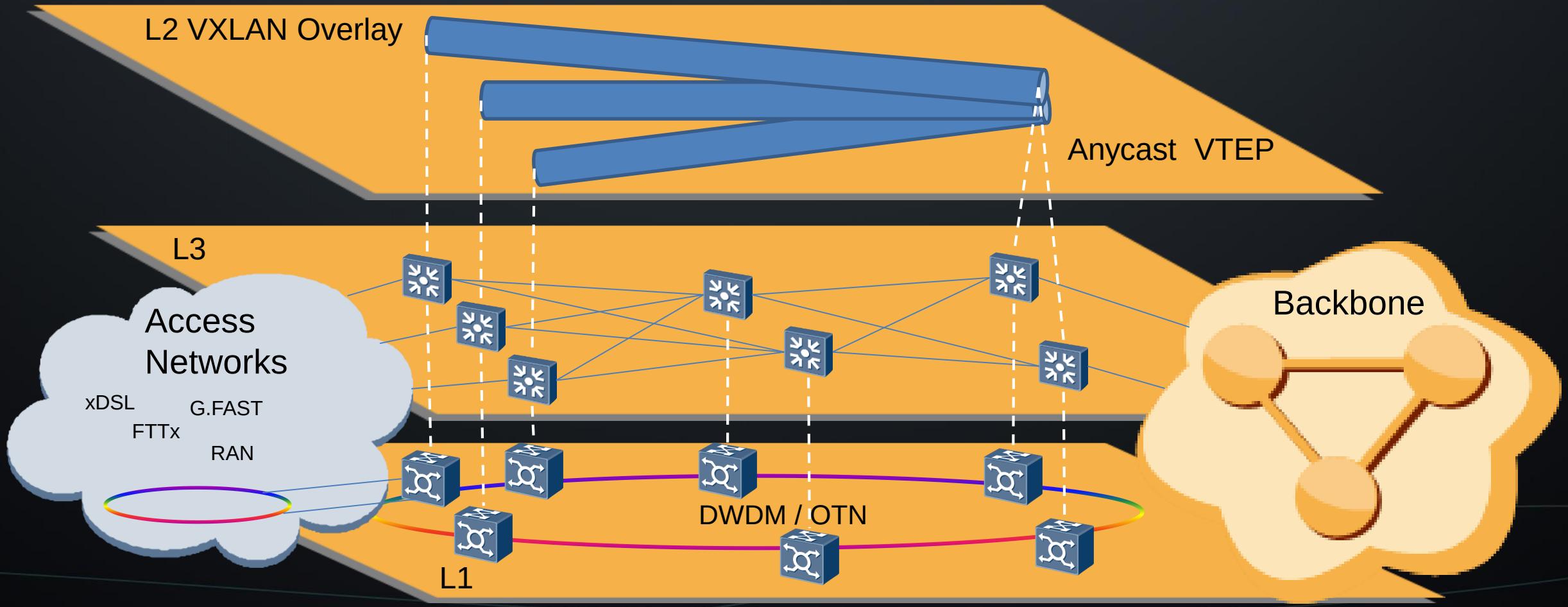
- VXLAN is (Supposedly) simple
  - VXLAN is thought to be easy to configure, manage, debug and troubleshoot because of the simplicity of the underlying protocols (plain IP).
  - The fact that VXLAN itself does not even require a specific control plane in it’s easiest deployment scenario or can “hide” it with SDN is a testament that something has been done right along the line.

Why VXLAN in the Metro Network ?

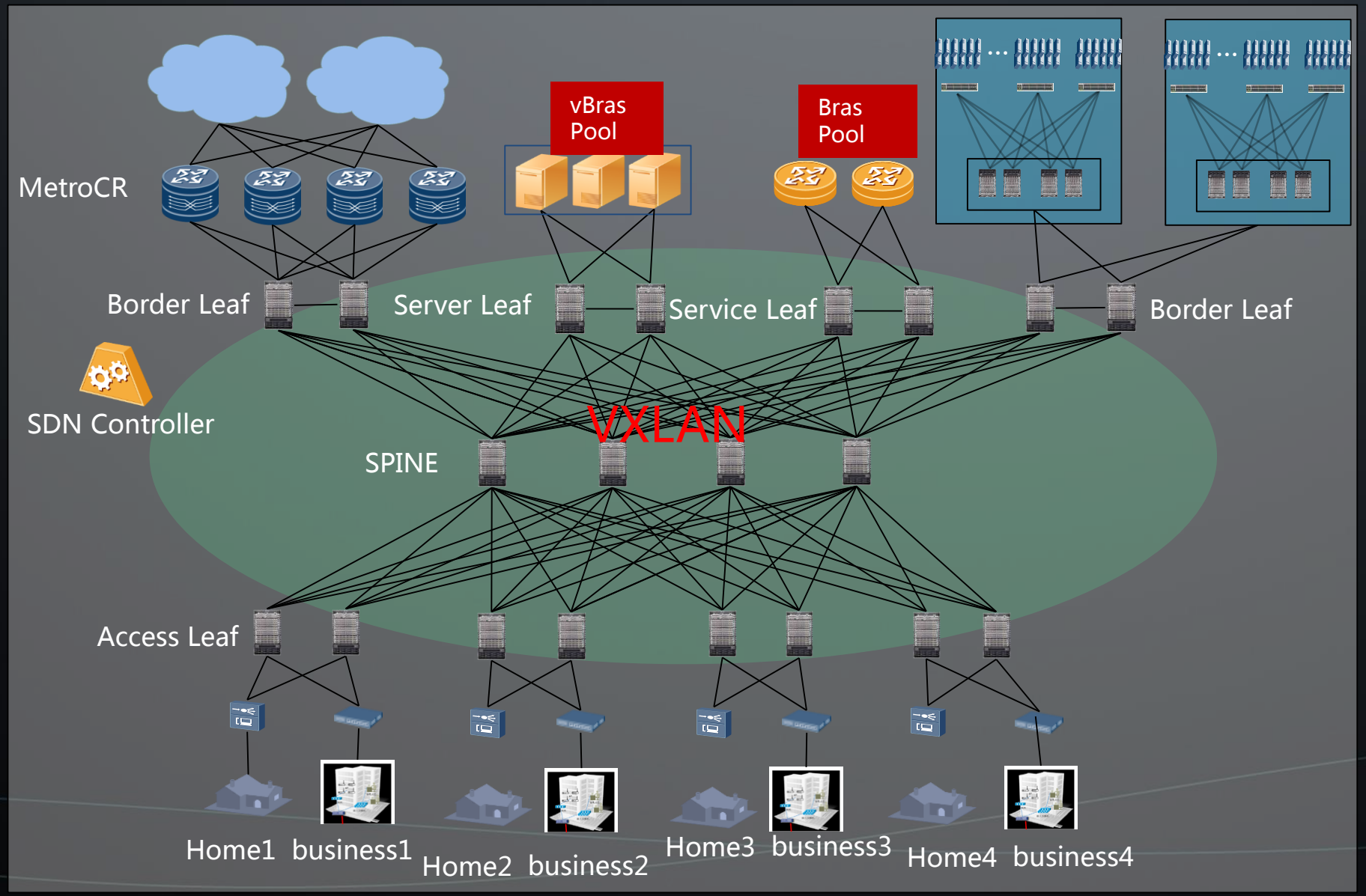
- VXLAN is not Expensive
  - Because of the simplicity that VXLAN demands
  - Because of the robust and simple control plane that is required with VXLAN hardware & software
  - Because of the economy of scale.

IP and VXLAN is commodity in SW & HW

VXLAN overlay in the Metro



Use case: Shanghai Telecom Metro Fabric



FBB and IPTV Subscriber

1. Access Leaf Switch introduces QinQ encapsulated user traffic into VxLAN 。
2. Open up VxLAN tunnel between Access Leaf and Service Leaf through controller
3. Service Leaf access vBras resource pool or hardware based MSE
4. vBras or MSE access metro CR through other VxLAN channels using dynamic routing protocol

Business VPN subscriber

1. Business VPN access Leaf, using "port + vlan" to divide VPC
2. Interconnection between different VPN subscriber access points can be implemented through internal VxLAN tunnel

DC –centric Metro

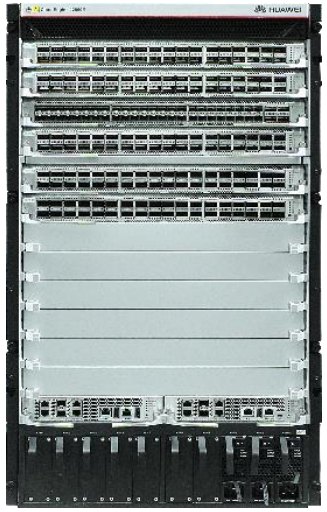
Potential additional requirements compared to a standard DCN solution:

- PTP (Precision Time Protocol) Syncronisation for IP based RAN → mobile backhauling
- Deep buffers for absorbing microbursts & speed mismatches
- VXLAN over IPv6

HUAWEI solution: Cloud Engine series

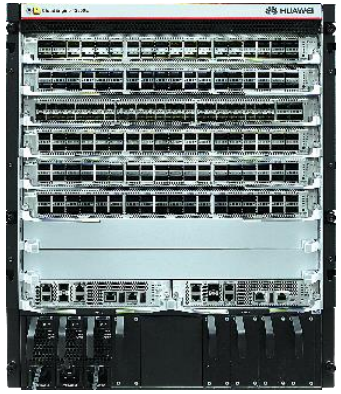
CE12816(E)

23U



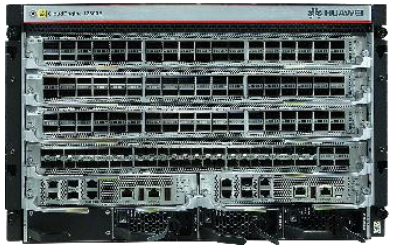
CE12812(E)

18U



CE12808(E)

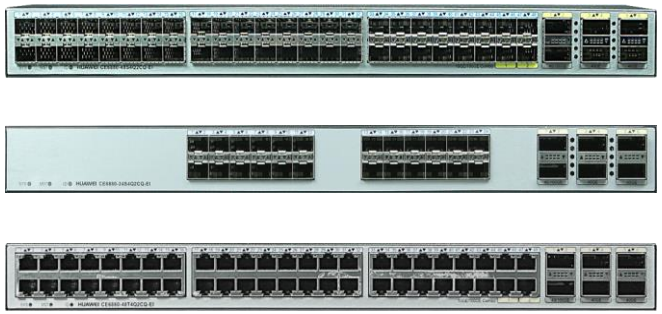
13U



CE12804(E)

7U

Power Consumption per 10GE:
2.32W(Typical), 3.34W(Max)



CE68xx

Next generation Cloud Engine: supports VXLAN over IPv6 in HW, deep buffers (4 GB)
PTP (Precision Time Protocol)

Openess of the solution

Market:

High requirements for openness and flexibility of network devices.

On CloudEngine series switches, the open system runs on the same Linux platform as the VRP (Huawei Network operating system).

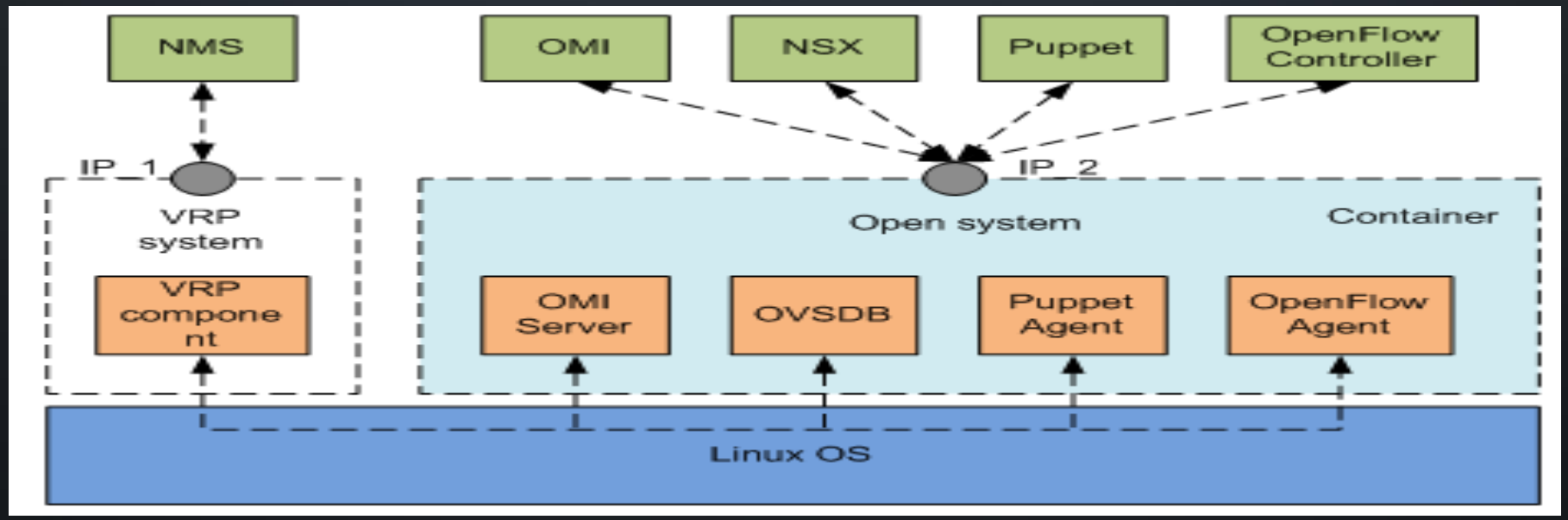
The open system implements device programmability and third-party software integration, meeting requirements for device openness and flexibility.

Benefits

Through the open system, you can :

- Run standard Linux shell script on the device
- Load various third-party software

Open SW Architecture



VRP system: manages all network interfaces of the device and provides secure high-performance network channels through capabilities such as routing, security, AAA and VPN of the VRP protocol stack.

Open system: is the client of third-party software such as OMI, Puppet and OpenFlow. It is connected to the VRP system through the virtual network, and provides standard NetConf and OpenFlow interfaces of the VRP system for third-party software.

Thank You.

Copyright©2016 Huawei Technologies Co., Ltd. All Rights Reserved.
The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.