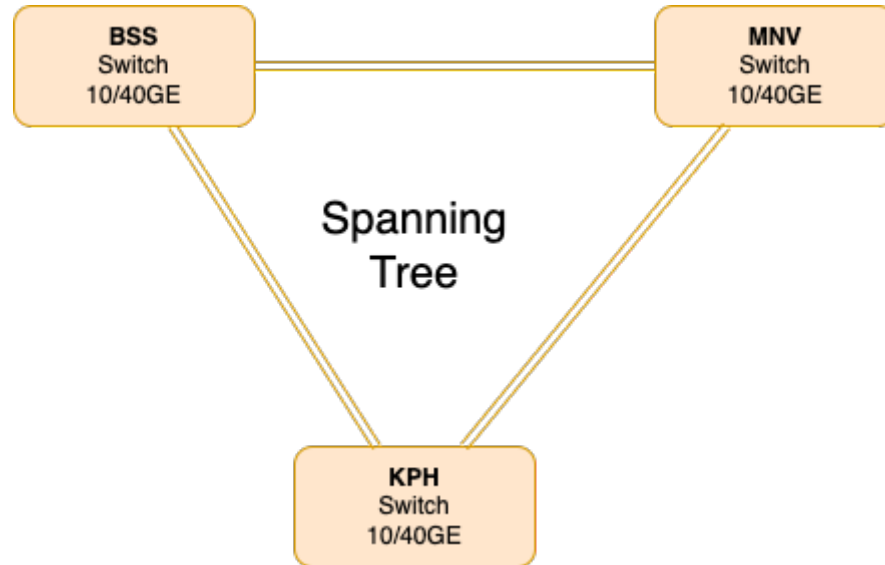


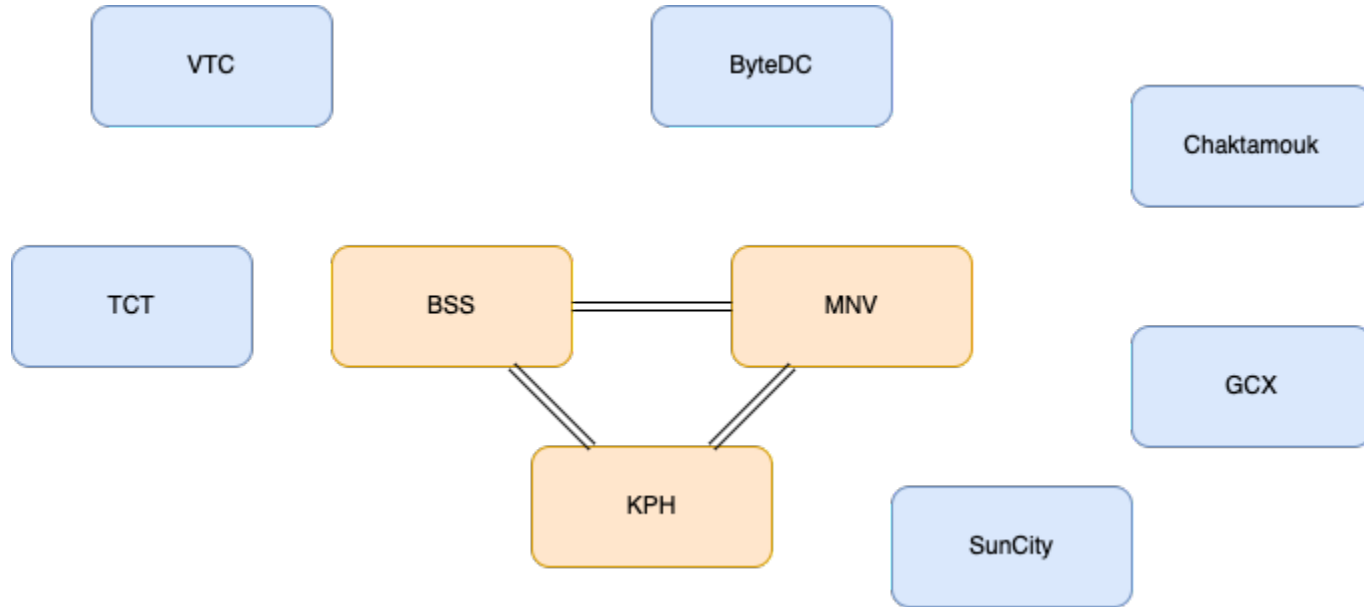
# VXLAN at CNX

VXLAN, Stacks, MLAG, EVPN with Huawei

# CNX 2010 - 2023



# CNX 2023 forward





# Requirements

## Network

- Need to connect 10+ data centers (no STP, need layer 3 routed core)
- Each location needs to have a redundant hardware and uplinks
- No single point of failure within the network (all functions in duplicate)

## Services

- VLAN any point to any point (cross locations)
- Common peering LANs
- 10GE - 100GE



VXLAN



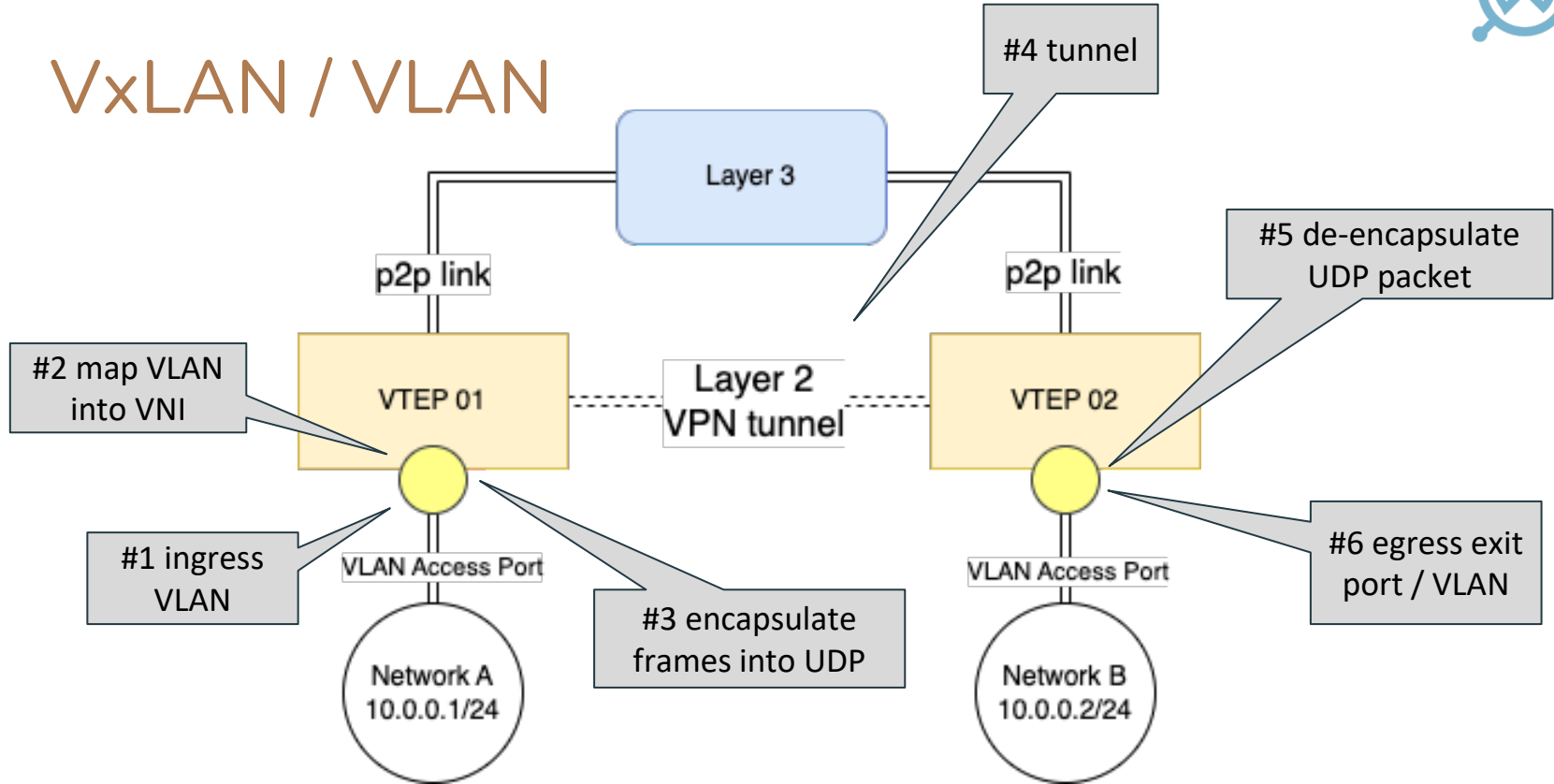
# VXLAN

VXLAN (Virtual Extensible LAN) is a network overlay technology designed to facilitate scalable network virtualization across multiple Layer 2 networks.

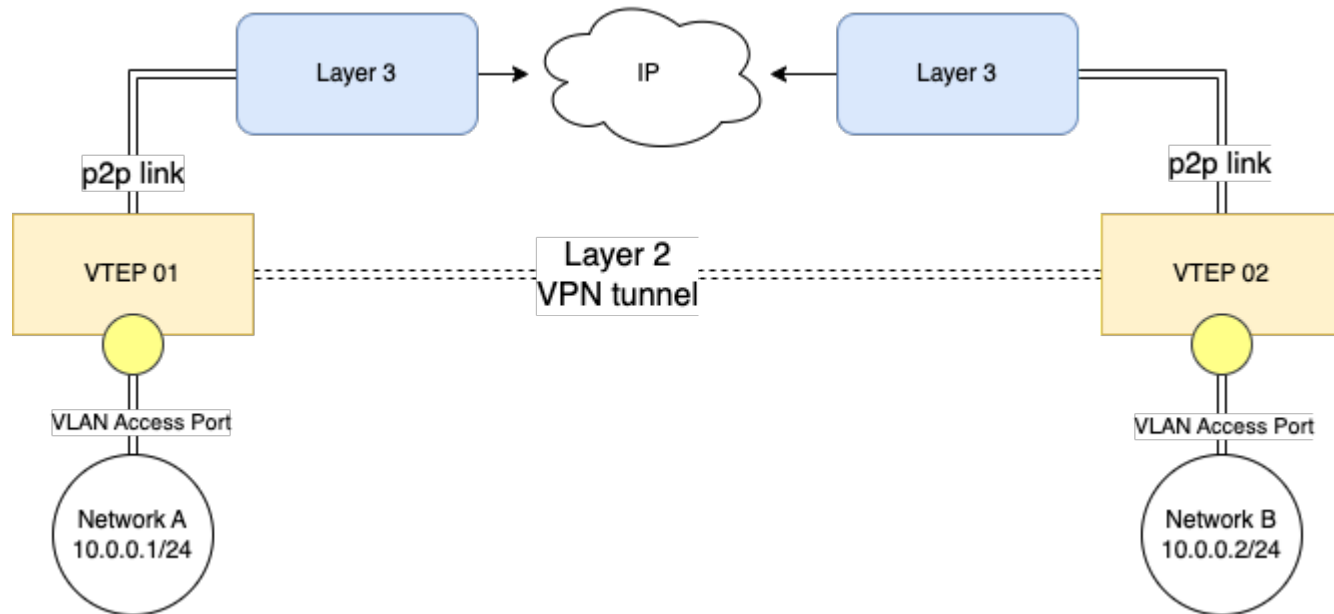
VXLAN encapsulates Ethernet frames within UDP packets, enabling Layer 2 segments to be **extended over an underlying Layer 3 network**.

VXLAN operates by using VXLAN Tunnel Endpoints (VTEPs) that encapsulate and de-encapsulate Ethernet frames. These VTEPs are located within switches, and they handle the VXLAN setup by mapping Layer 2 networks to unique VXLAN Network Identifiers (VNIs).

# VxLAN / VLAN



# VxLAN







# Setup - Bridge Domains (BD)

Role: Functions as a Layer 2 broadcast domain within the VXLAN network. It groups Ethernet interfaces and NVE interfaces to facilitate Layer 2 communication and segmentation.

Linkage: BDs are linked to NVE interfaces (through VNIs) to enable Layer 2 over Layer 3 tunneling.

```
vlan 500
  description "CNX Peering LAN"

interface 25GE1/0/1
  port link-type access
  port default vlan 500

bridge-domain 500
  vxlan vni 500
  l2 binding vlan 500
```



# Network Virtualization Endpoint (NVE)

Role: Acts as the gateway between the VXLAN overlay network and the physical network. It is responsible for encapsulating and de-encapsulating VXLAN packets, allowing for the extension of Layer 2 networks across Layer 3 infrastructures.

Linkage: NVE interfaces are tied to specific VXLAN segments (identified by VNIs) and can be associated with Bridge Domains for Layer 2 network extension. They also interact with VPN Instances when routing VXLAN traffic at Layer 3.

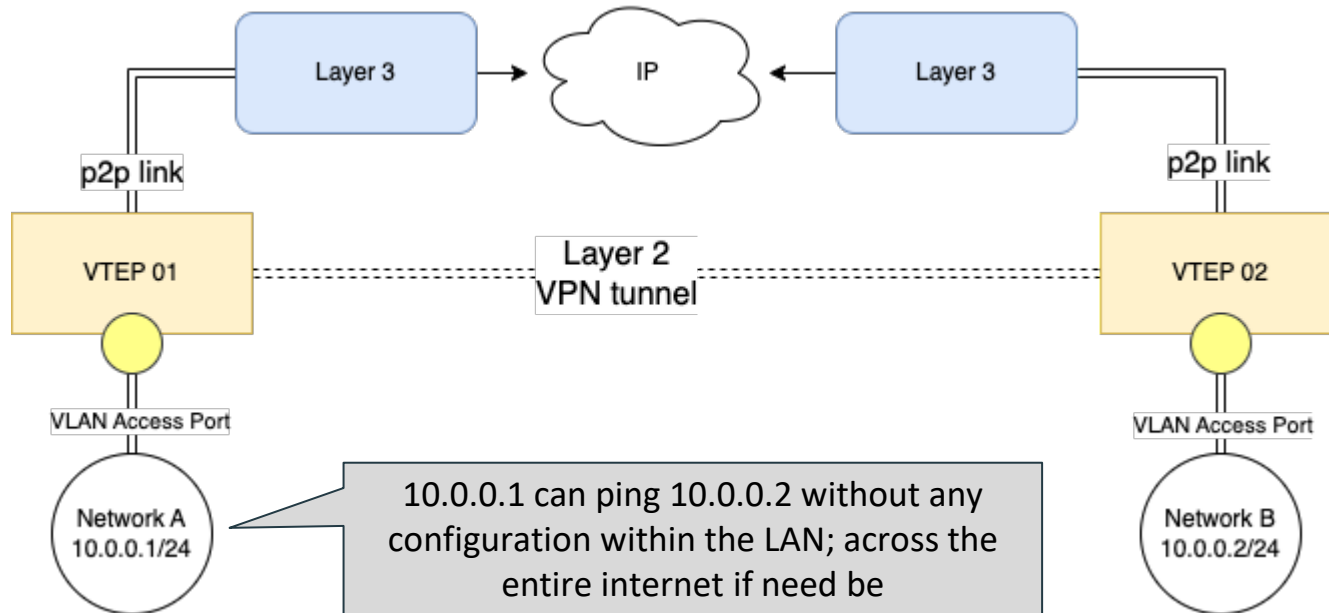
```
# vtep 01
interface LoopBack0
ip address 172.23.127.11 255.255.255.255



interface nve 1
source 172.23.127.11
vni 500 head-end peer-list 172.23.127.21
```

```
# vtep 02
interface LoopBack0
ip address 172.23.127.21 255.255.255.255

interface nve 1
source 172.23.127.21
vni 500 head-end peer-list 172.23.127.11
```

# VxLAN end to end





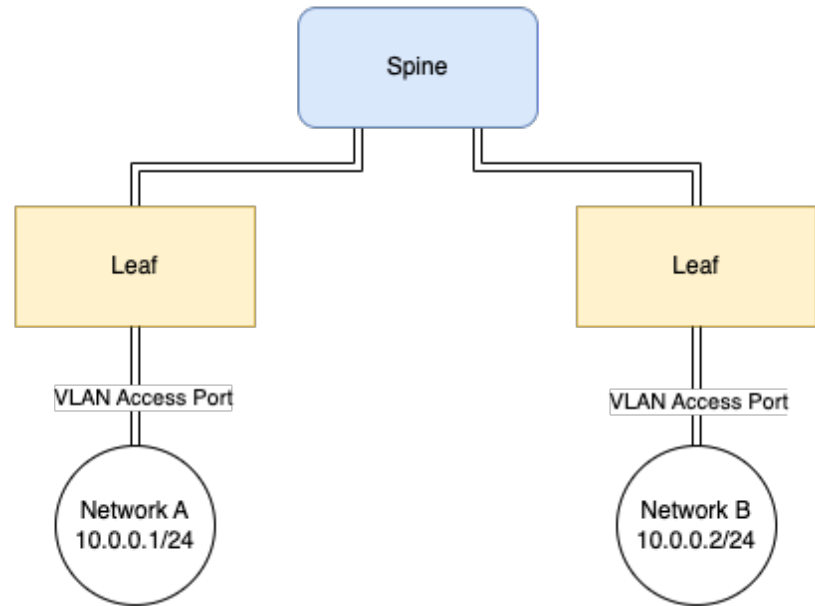
# Spine-leaf, Stacks and MLAG

# Spine-Leaf

Spine-leaf architecture is a popular network topology used in modern data centers, particularly beneficial for environments **requiring high bandwidth** and **lower latency**.

This design is comprised of two primary layers: **the spine** (backbone) and **the leaf** (access layer).

This setup ensures that any device on a leaf switch can communicate with any other device with only one or two network hops, **providing a predictable and uniform latency**.



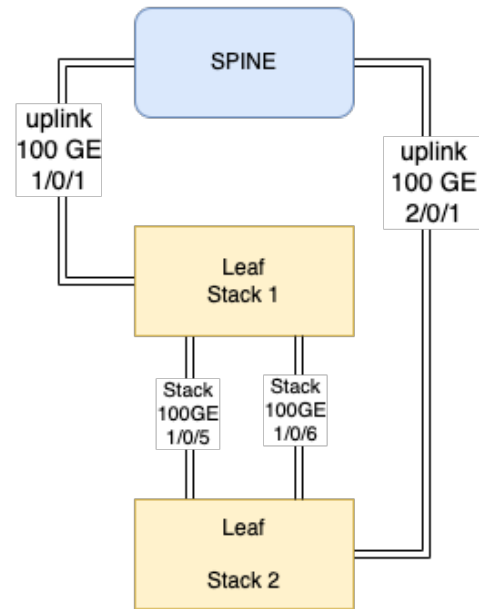
# Stacks - adding resilience and reliability

```
## switch 01
stack
stack member 1 priority 150
stack member 1 domain 10

interface stack-port 1/1
port member-group interface 100ge 1/0/5 to 1/0/6

## switch 02
stack
stack member 1 priority 120
stack member 1 domain 10
stack member 1 renumber 2 inherit-config

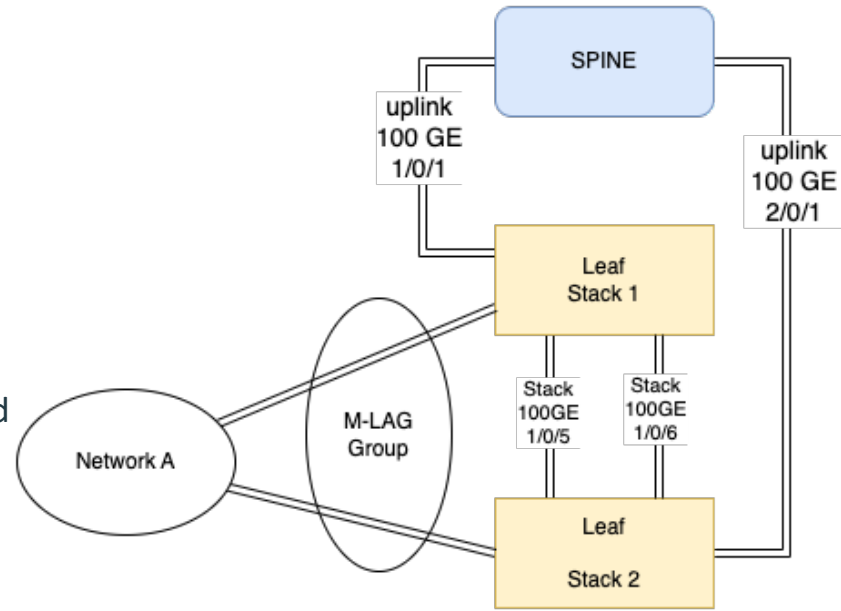
interface stack-port 1/1
port member-group interface 100ge 1/0/5 to 1/0/6
```



# Multi-Chassis Link Aggregation

MLAG (Multi-Chassis Link Aggregation) is a network technology that allows multiple switches to be configured so that they appear as a single device for the purpose of link aggregation.

MLAG enhances network reliability by providing fault tolerance and load balancing capabilities, **minimizing the risk of a single point of failure** and improving overall network performance by distributing traffic across multiple links and devices.

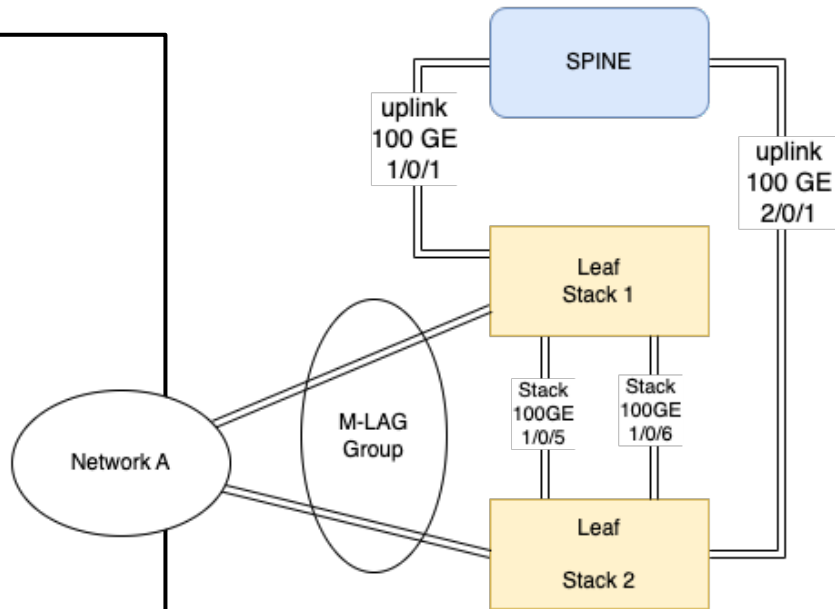


# Multi-Chassis Link Aggregation

```
interface Eth-Trunk1
port link-type trunk
port trunk allow-pass vlan 500 599
mode lacp-static

interface 25GE 1/0/47
eth-trunk 1

interface 25GE 2/0/47
eth-trunk 1
```







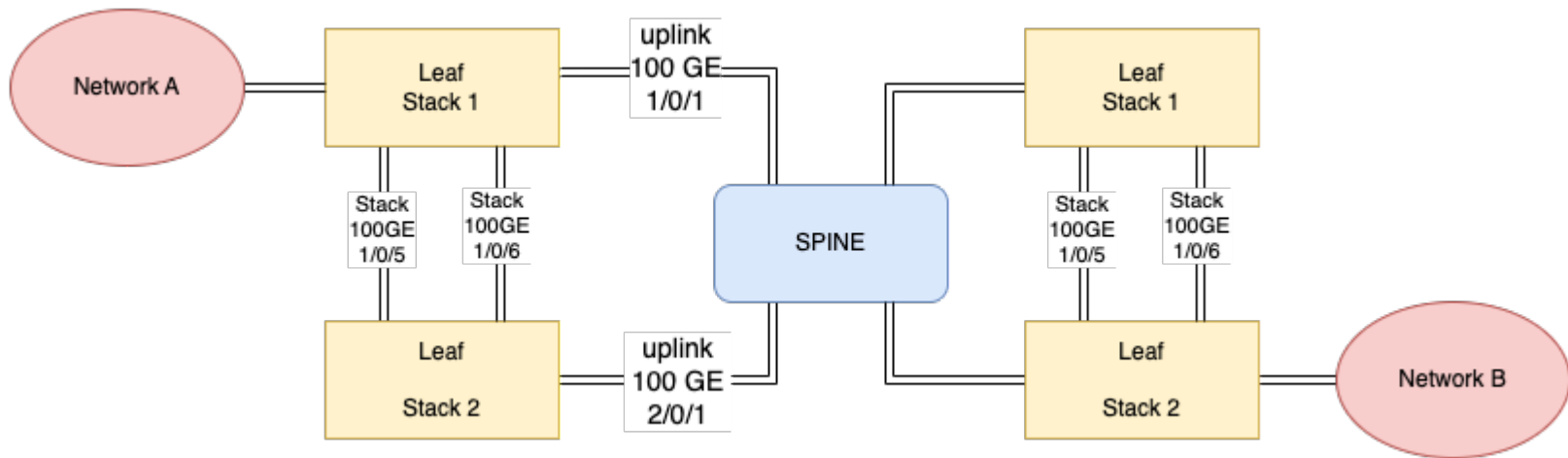
Access



# Access - Single Port

```
interface 25GE1/0/1
port link-type access
port default vlan 500
port mode 10G
```

- Access to P2P service (1:1)
- Access to Peering LAN (1:n)

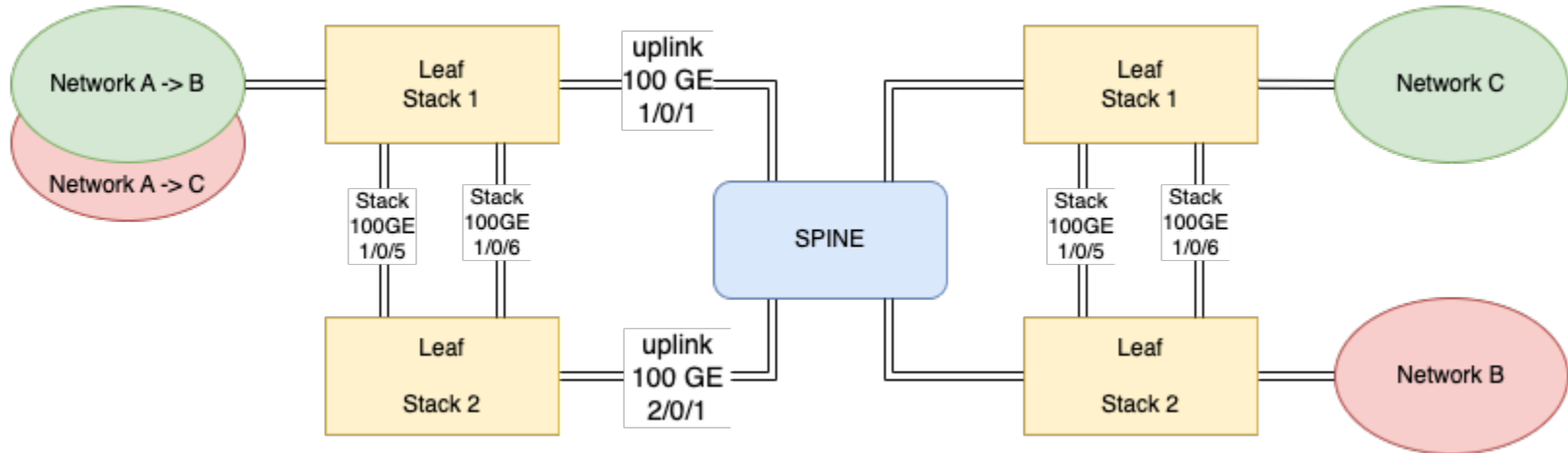




# Access - Multiple Services

```
interface 25GE1/0/1
port link-type trunk
port trunk allow-pass vlan 512 513
```

- Access for multiple services (VLANs) with 1 link



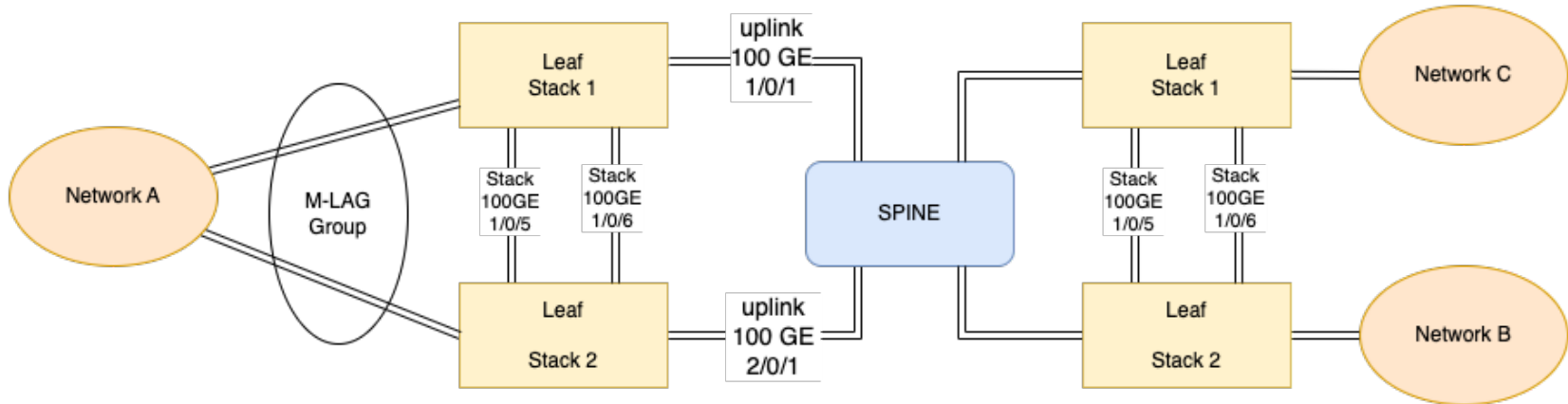


# Access - MLAG - Group

- Access for one or multiple services (VLAN) with MLAG

```
interface Eth-Trunk1
port link-type trunk
port trunk allow-pass vlan 511 512
mode lacp-static
```

```
interface 25GE 1/0/47
eth-trunk 1
interface 25GE 2/0/47
eth-trunk 1
```





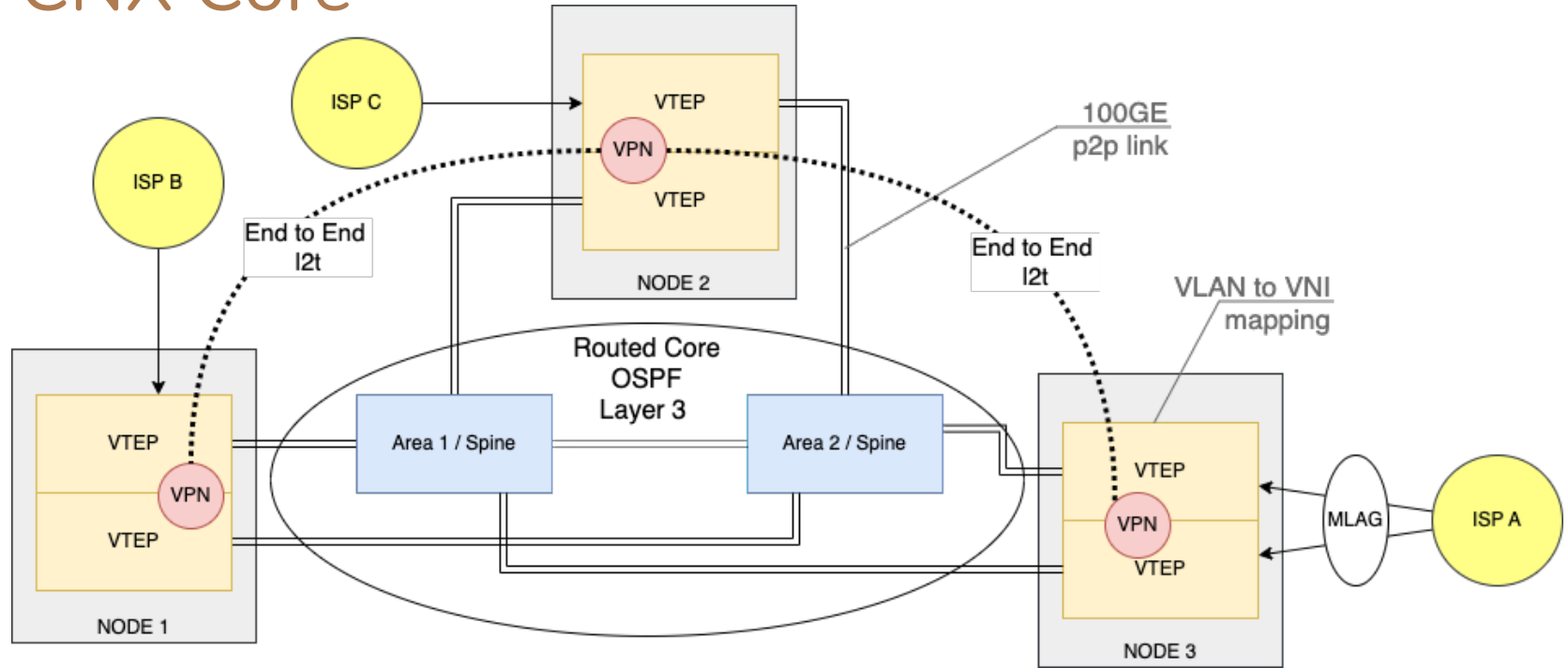
Pulling it together



# Architecture

- 2 switches per location configured as stack
- Each switch has own uplink (point to point, layer 3)
- 2 links from each location to a separate spine location
- 2 spine locations
- OSPF at the core to establish a common L3 routing domain
- Each VTEP (NVE) interface can ping every other VTEP
- Customers can connect using MLAG for additional redundancy

# CNX Core





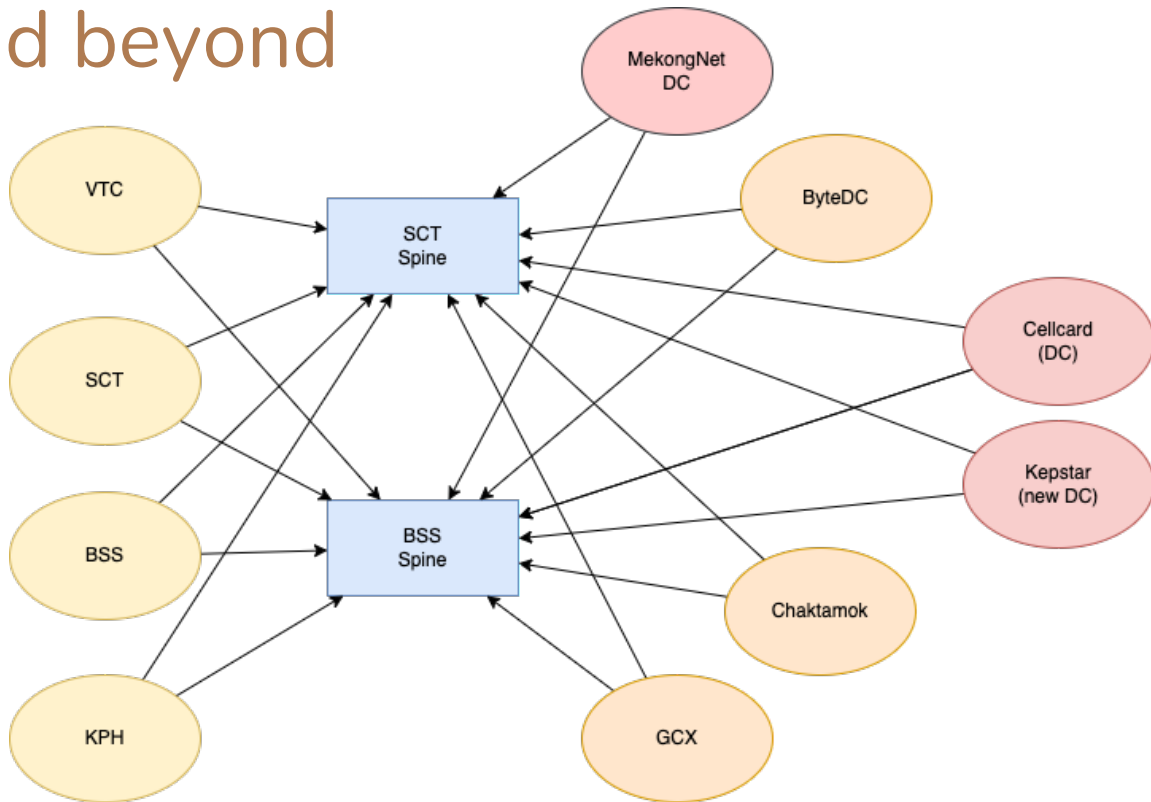
But ...



# CNX 2025 and beyond

Linking all end to end VxLAN  
VPNs over a layer 3 network  
manually ... seems like a  
disaster waiting to happen.

**We need more ...**





EVPN



# Ethernet VPN

EVPN (Ethernet VPN) is an advanced network technology that provides an enhanced control plane mechanism for managing Ethernet traffic within VxLAN tunnels.

It extends VxLAN tunnel services by enabling efficient handling of MAC addresses, which facilitates more scalable and dynamic bridging and routing functionalities across a network.

EVPN utilizes BGP (Border Gateway Protocol) as its primary control plane protocol to advertise and discover network information, including MAC address routes, IP address routes, and ARP entries, which allows for efficient network segmentation and host discovery.



# How VxLAN works

- VxLAN requires each VTEP to be able to form a tunnel with all other VTEP that share a VLAN / BD / VNI
- 10 VLANs that need to be accessible in 10 locations, requires that we configure each of those relationships by hand
- Each VTEP then broadcast to ARP requests to figure out where a certain device is connected
- Every VTEP keeps a list of VxLAN tunnels / VNI-VLAN / mac addresses



# Tunnel Endpoints (VTEP)

Each VTEP needs to know for each connected device in the network:

- VLAN / VNI (BD)
- Device IP address
- Device MAC address / BD / end point (local or another VTEP)
- How to get to this endpoint via our Layer 3 network

**Using EVPN we can move some of this information to a separate control plane**



# EVPN config

- enable BGP EVPN
- setup BGP sessions
- add BGP community attributes to the bridge domain configurations
- configure the Network Virtualisation Endpoint (NVE) to use BGP



# EVPN BGP

## Spine - route reflector

```
evpn-overlay enable

bgp 65000
router-id 172.23.127.2
peer 172.23.127.11 as-number 65000
peer 172.23.127.11 connect-interface LoopBack0

ipv4-family unicast
peer 172.23.127.11 enable
peer 172.23.127.11 reflect-client

l2vpn-family evpn
undo policy vpn-target
peer 172.23.127.11 enable
peer 172.23.127.11 reflect-client
```

## Notes

- We use private ASN
- Router ID is the loopback address
- We configure peers to use the loopback interface as source IP
- Loopback is in OSPF
- Enable peer as route reflector clients
- L2vpn-family evpn bring the VxLAN data into BGP



# EVPN BGP

## Leaf - route reflector client

```
evpn-overlay enable

bgp 65000
router-id 172.23.127.11
peer 172.23.127.2 as-number 65000
peer 172.23.127.2 connect-interface LoopBack0

ipv4-family unicast
peer 172.23.127.2 enable

l2vpn-family evpn
policy vpn-target
peer 172.23.127.2 enable
```

## Notes

- Near same config as RR
- Also use the loopback interface
- Enable the peers
- `policy vpn-target` exports the vpn endpoint information



# Bridge domain (BD) changes

## Previous

# on leaf bridge domain configuration

```
bridge-domain 500
```

```
vxlan vni 500
```

```
I2 binding vlan 500
```

## With EVPN

# on leaf bridge domain configuration

```
bridge-domain 500
```

```
vxlan vni 500
```

```
I2 binding vlan 500
```

```
evpn
```

```
route-distinguisher 65000:500
```

```
vpn-target 65000:500 export-extcommunity
```

```
vpn-target 65000:500 import-extcommunity
```



# NVE changes

## Previous

```
# vtep 01
interface LoopBack0
ip address 172.23.127.11 255.255.255.255
```

```
interface nve 1
source 172.23.127.11
vni 500 head-end peer-list 172.23.127.21
vni 500 head-end peer-list 172.23.127.31
vni 599 head-end peer-list 172.23.127.21
vni 599 head-end peer-list 172.23.127.31
```

## With EVPN

```
# vtep 01
interface LoopBack0
ip address 172.23.127.11 255.255.255.255
```

```
interface nve 1
source 172.23.127.11
vni 500 head-end peer-list protocol bgp
vni 599 head-end peer-list protocol bgp
```



# Summary

- VXLAN as used at CNX allows us to build complex networks with multiple nodes across Phnom Penh
- We have device and link redundancy at every location
- We have a uniform latency across our network
- We are able to map any port in any location to any other port in our network or map any port into a shared VLAN for common access
- We are able to scale the network to multiple data centers and use multiple 100GE links to ensure sufficient capacity per location



Thank you