Move to Modern L3 Fabric While Supporting Legacy L2 Workloads

AOS 2.0 VXLAN Feature Brief

As a network engineer, you need to move to modern L3 fabric for agility and scale, but are held back by many L2 workloads (virtual or physical). You need to easily stretch your L2 segments across the fabric using VXLAN, but are stuck with three options to manage (design, implement, operate) L2 workloads on a physical IP fabric (underlay):

1. Manual, without using a SDN controller
2. OVSDB, using a SDN controller (e.g. Cisco ACI, VMware NSX or Juniper Contrail)
3. Ethernet VPN (EVPN), using vendor-specific control plane technology to allow hosts (bare-metal and VMs) to be placed anywhere in a network, and connected to the same logical L2 overlay network.

Manual:

- Pros: Multi-vendor standardards based technology. No controller lock-in. Ability and flexibility to invest or add a SDN controller later.
- Cons: There are a lot of manual, error-prone, vendor-specific, time-consuming steps involved.

SDN Controller::

- Pros: Ability to connect physical ports and bare metal servers using high speed underlay switch with 10/25/40/100G interfaces. Leverage the same interface used to configure virtual networking infrastructure.
- Cons: You need to invest in a SDN controller that has certified integration with the underlay switch. Assumes the majority of the workload is already virtualized.

Move to Modern L3 Fabric While Supporting Legacy L2 Workloads

AOS 2.0 VXLAN Feature Brief

EVPN+VXLAN:
- Pros: Standards based technology
- Cons: Complex configurations with different vendor-specific deployment types. Interoperability challenges between vendors.

How AOS (Apstra® Operating System) makes the lifecycle operations of VXLAN easy

AOS provides turnkey, vendor-agnostic, autonomous lifecycle automation of VXLAN, spanning design, implementation, and operations — for any virtual, container-based, and bare-metal workloads.

For example, the networking team can automate the configurations and verifications of VXLAN on an IP fabric, without using a SDN controller, or EVPN. Without AOS, these are the manual VXLAN provisioning steps you will have to perform. With AOS, these steps are completely automated:

- ☑ Create VXLAN Network Identifier (VNI) resource pool
- ☑ Configure VTEP VNI interfaces for applying connectivity and isolation policy
- ☑ Configure VTEP source interfaces for VTEP-to-VTEP communication
- ☑ Attach hosts (virtual or bare-metal) that need to be on certain VLANs
- ☑ Bind different VLANs into a contiguous, stretched L2 domain over VXLAN tunnel
- ☑ Configure VXLAN routing between VXLAN and VXLAN or VLAN domains (coming soon)
- ☑ Verify all configurations
- ☑ Figure out what telemetry to gather, in order to validate the operating state of the VXLAN tunnels and underlay fabric meet your desired service expectations
- ☑ Gather the telemetry you need
- ☑ Continuously monitor telemetry you gather, in order to ensure the operating state of the VXLAN tunnels and underlay fabric meet your desired service expectations

The following screenshot shows an example of your VXLAN intent during the design phase. In this case, you want a select group of servers on different VLANs (in different racks) placed onto the same VXLAN segment (same L2 domain).

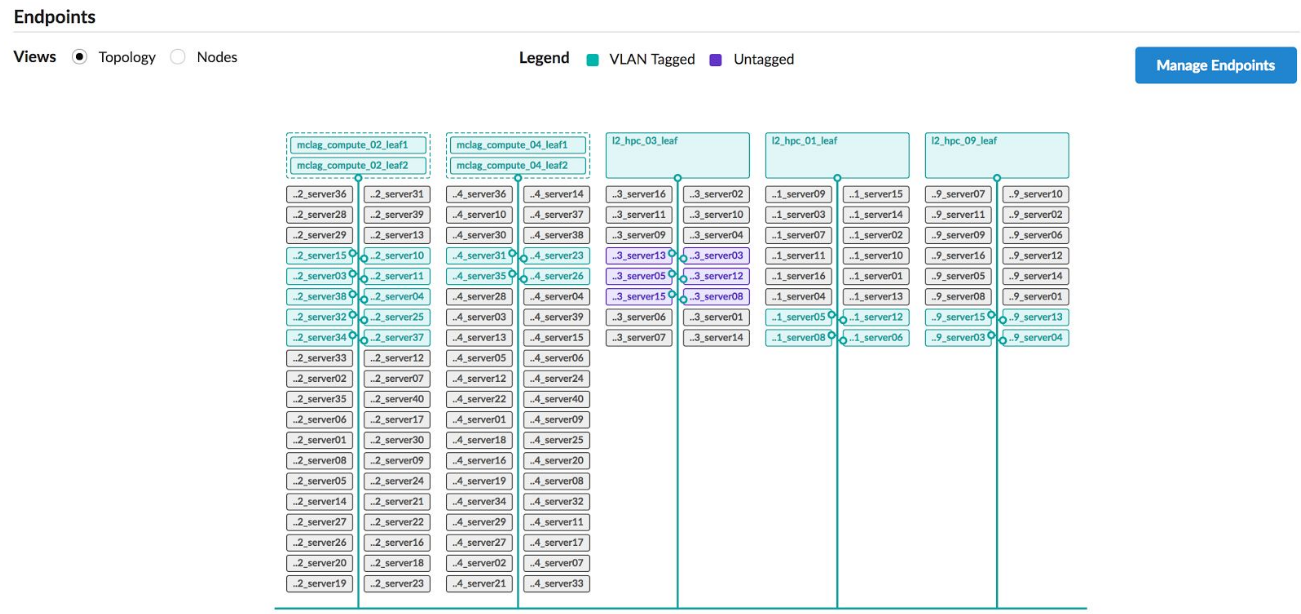
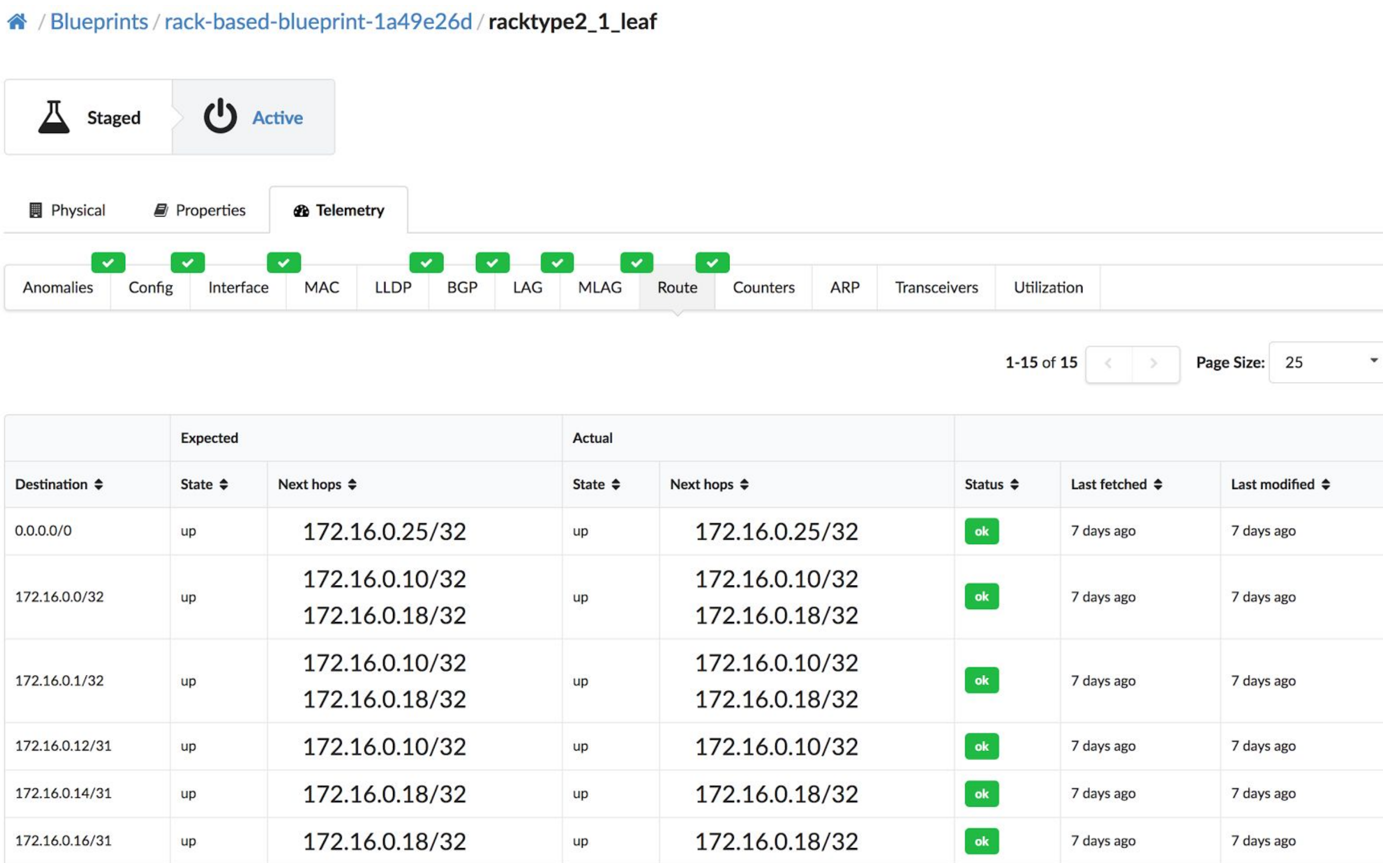


Figure 1: Automated VXLAN Design

Additionally, in the figure above, some of the endpoints are provisioned to support "tagged" interfaces (aqua) while others are provisioned for "untagged" (purple). AOS also manages the automation and intent validation of these endpoints.

The following screenshot shows an example of closed-loop validation. In this case, all route expectations for the VTEPs on "racktype 2_1_leaf" have been validated.



	Expected		Actual				
Destination	State	Next hops	State	Next hops	Status	Last fetched	Last modified
0.0.0.0/0	up	172.16.0.25/32	up	172.16.0.25/32	ok	7 days ago	7 days ago
172.16.0.0/32	up	172.16.0.10/32 172.16.0.18/32	up	172.16.0.10/32 172.16.0.18/32	ok	7 days ago	7 days ago
172.16.0.1/32	up	172.16.0.10/32 172.16.0.18/32	up	172.16.0.10/32 172.16.0.18/32	ok	7 days ago	7 days ago
172.16.0.12/31	up	172.16.0.10/32	up	172.16.0.10/32	ok	7 days ago	7 days ago
172.16.0.14/31	up	172.16.0.18/32	up	172.16.0.18/32	ok	7 days ago	7 days ago
172.16.0.16/31	up	172.16.0.18/32	up	172.16.0.18/32	ok	7 days ago	7 days ago

Figure 2: Example Of Closed-Loop Validations of VXLAN

The following screenshot shows an example of closed-loop validation for the entire integrated underlay and overlay (VXLAN), as a single logical system, across all individual devices.

Move to Modern L3 Fabric While Supporting Legacy L2 Workloads

AOS 2.0 VXLAN Feature Brief

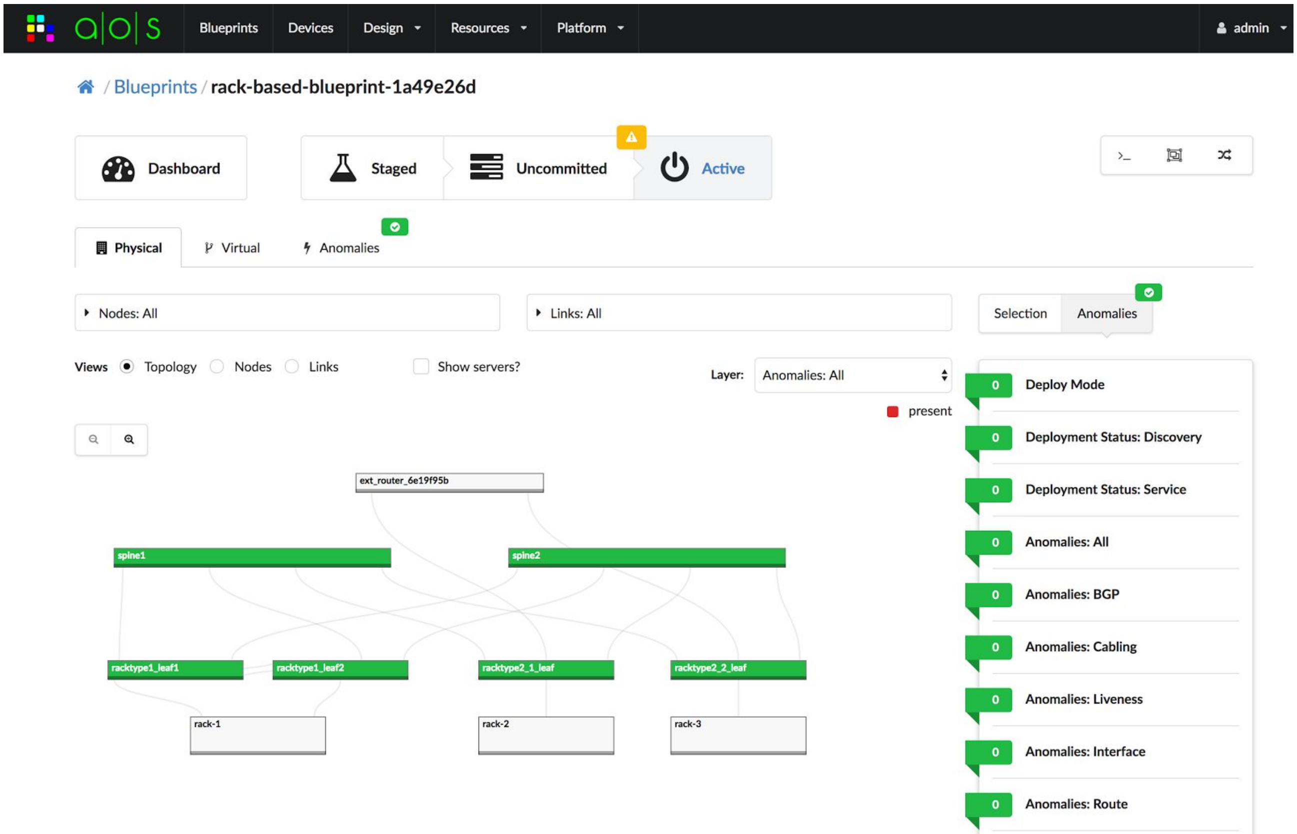


Figure 3: Example Of Closed-Loop Validations of Integrated Underlay and Overlay

Summary

AOS provides turnkey, vendor-agnostic, autonomous lifecycle operations of VXLAN, including design, implementation, and operations for any virtual, container-based and bare-metal workloads.

Soon for operations, AOS provides vendor-agnostic telemetry and correlation between overlay and underlay. For example, easy, instant, fabric wide visibility into VXLAN tunnel utilization for all VXLAN segments configured by AOS on switches. Additionally, AOS provides underlay to overlay correlation when endpoints/VTEPs are instantiated on hypervisors through integration with:

- Leading hypervisor and/or overlay controller, and/or
- Leading Cloud Management Platform (CMP)