오픈스택을 위한 아리스타 SDCN 전략
Software Defined Cloud Network

김창민 부장
System Engineer
charles.kim@aristanetworks.com


Arista
Change through Innovation
Morgan Stanley
eBay
Verizon


Data Centers
ARE
Demanding
Complex
Growing
CHANGING

Some keywoards for Datacenter Market

Cloud Computing

Scale-out Design

Open Source

Commodity Hardware

Automation

Network Virtualization

Server Virtualization

DevOps

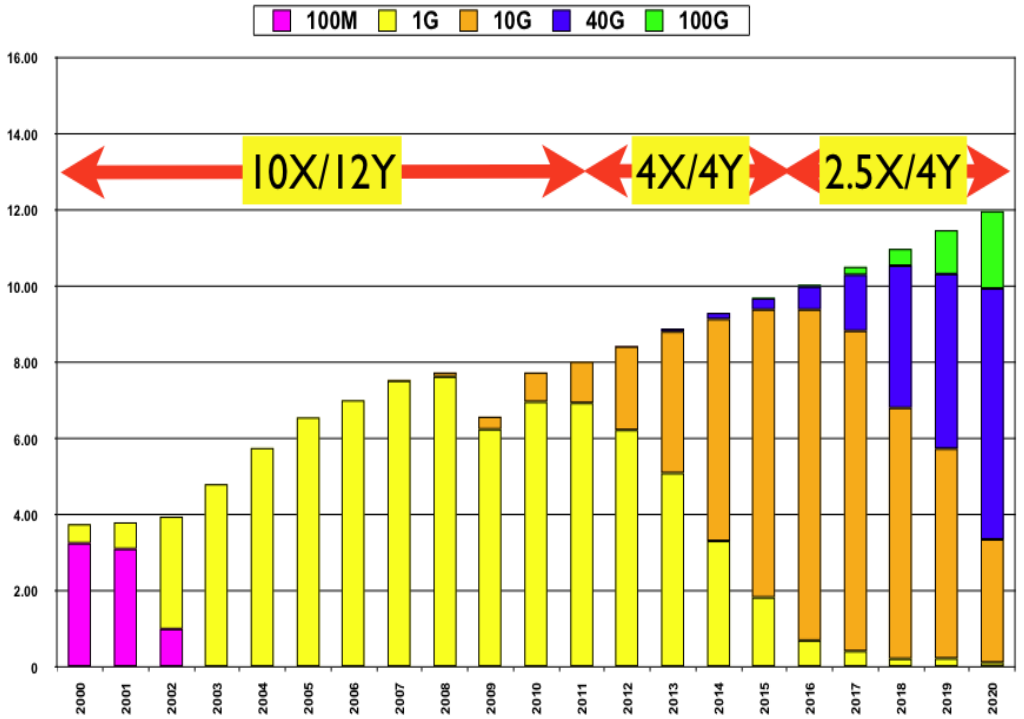
Network Programmability

Software Defined Networking

Widespread Adoption of Ethernet (in servers)

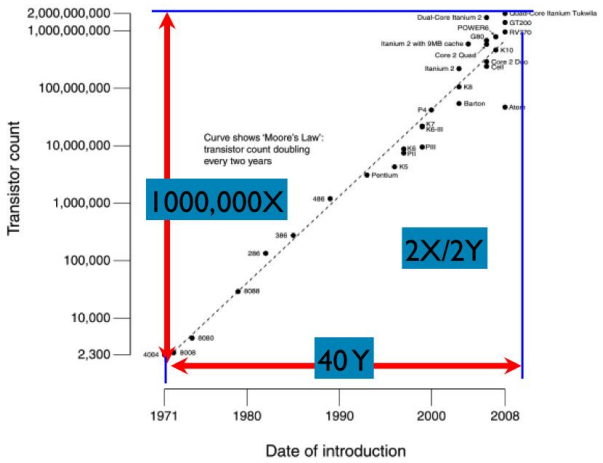
2002:
Fast Ethernet to Gigabit Ethernet

2013/14:
Crossover Gigabit to 10GbE

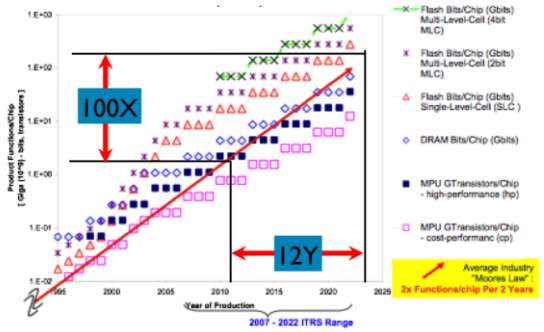


Source: Intel LAN Group

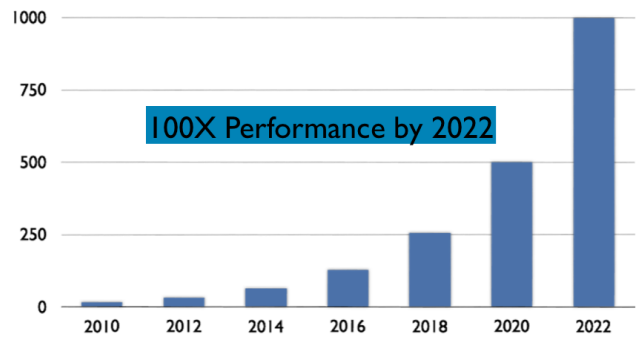
Moore's Law and Networking



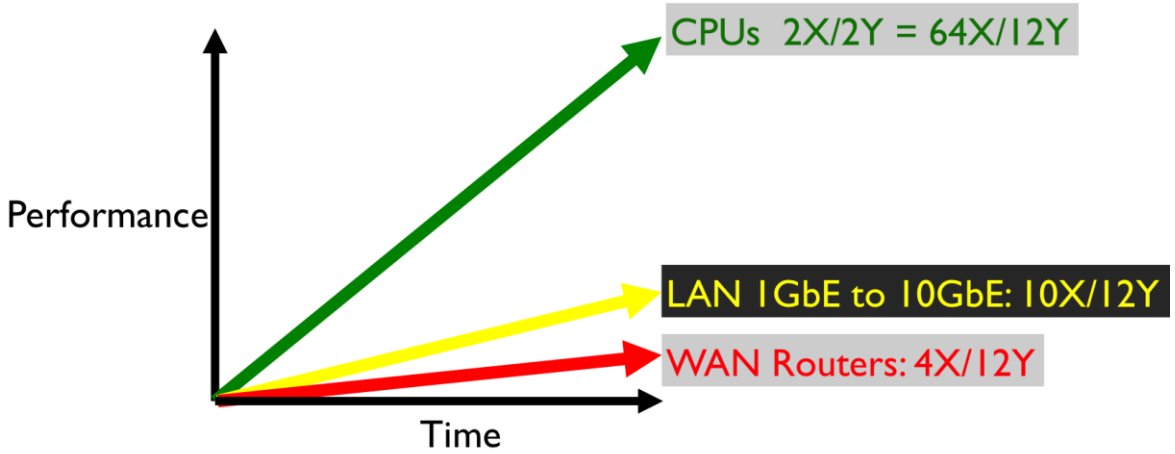
Moore's Law 1971-2011



Semiconductor Technology Roadmap

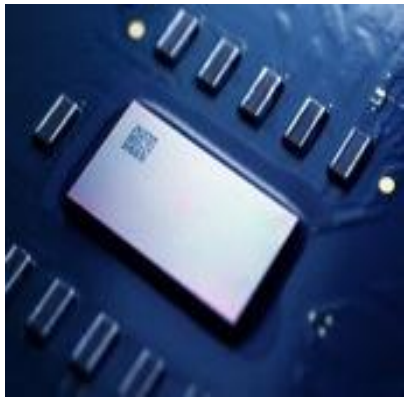


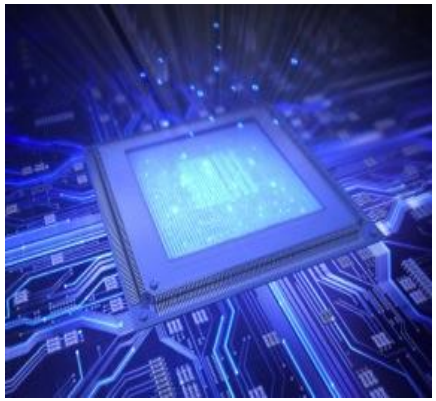
64-bit CPU Cores over Time



Why has Networking not kept up with Moore's Law?

Platforms Optimized for Custom Silicon





1990+ Proprietary ASICs

- Expensive to produce
- Relatively slow (low transistor res)
- Low density
- Power hungry
- Long write-down

2000+ FPGA Processing

- Complex to program
- Low port density/chip
- Result in high component count,
  low feature velocity
- Ultimate flexibility

2010+ Full Custom Silicon

- Standardized features
- High clock rates and density
- Competitive market
- Exceptional innovation rate

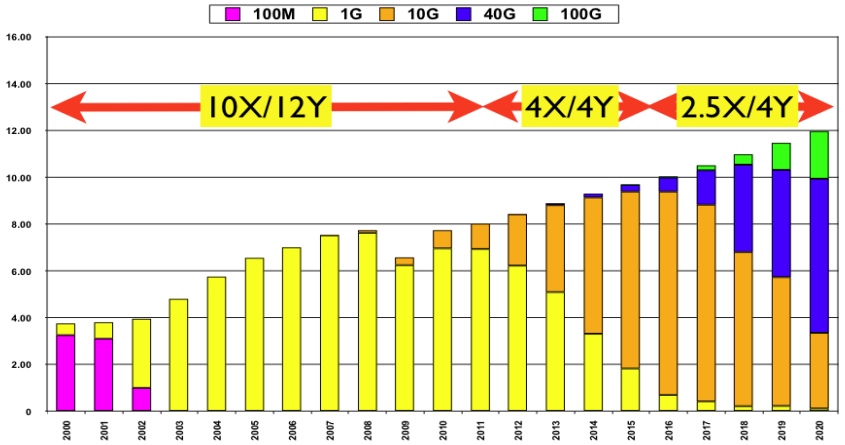
Evolution of Custom Switch Silicon

Technology	130nm	65nm	40nm	28nm
10G ports	24	64	128	256
Throughput	360M PPS	960M PPS	2B PPS	4B PPS
Buffer Size	2 MB	8 MB	16 MB	32 MB
Table Size	16K	64K	128K	256K
Port Speeds	10G	10G/40G	10G/40G/100G	10G/40G/100G
Availability	2007	2011	2013	2015
Improvement	-	3X/4Y	2X/2Y	2X/2Y

Next generation custom switch silicon is on Moore's Law

CPUs driving the Netwok Upgrade

- **Faster CPUs need Faster Networks**
  - Intel Sandybridge driving 10GbE adoption
  - 50% attach rate 2013, 80% by 2015

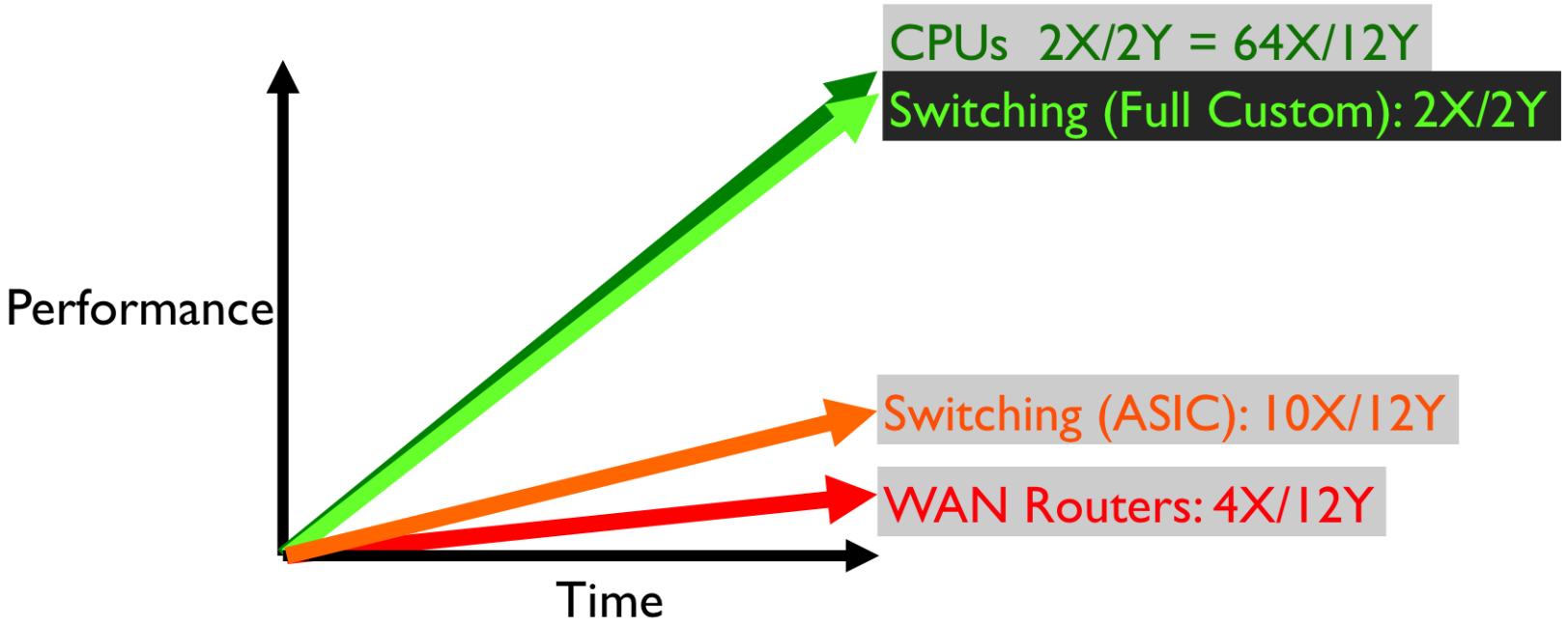


Source: Intel LAN Group

- **10/40/100G Market growing rapidly**
  - $4B in 2010 to $16B in 2016
  - From 5M ports 2010 to 67M ports 2016

- **Faster End nodes need faster Backbones**
  - Many apps drive east/west traffic not north/south
  - Cluster sizes getting larger & larger

Moore's Law and Networking



Next generation custom switch silicon is on Moore's Law

Besides larger tables, what else can 2X/2Y transistors be used for?

SDN

Flexible forwarding requires flexible ways of exposing the underlying functionality

Arista SDCN Architecture

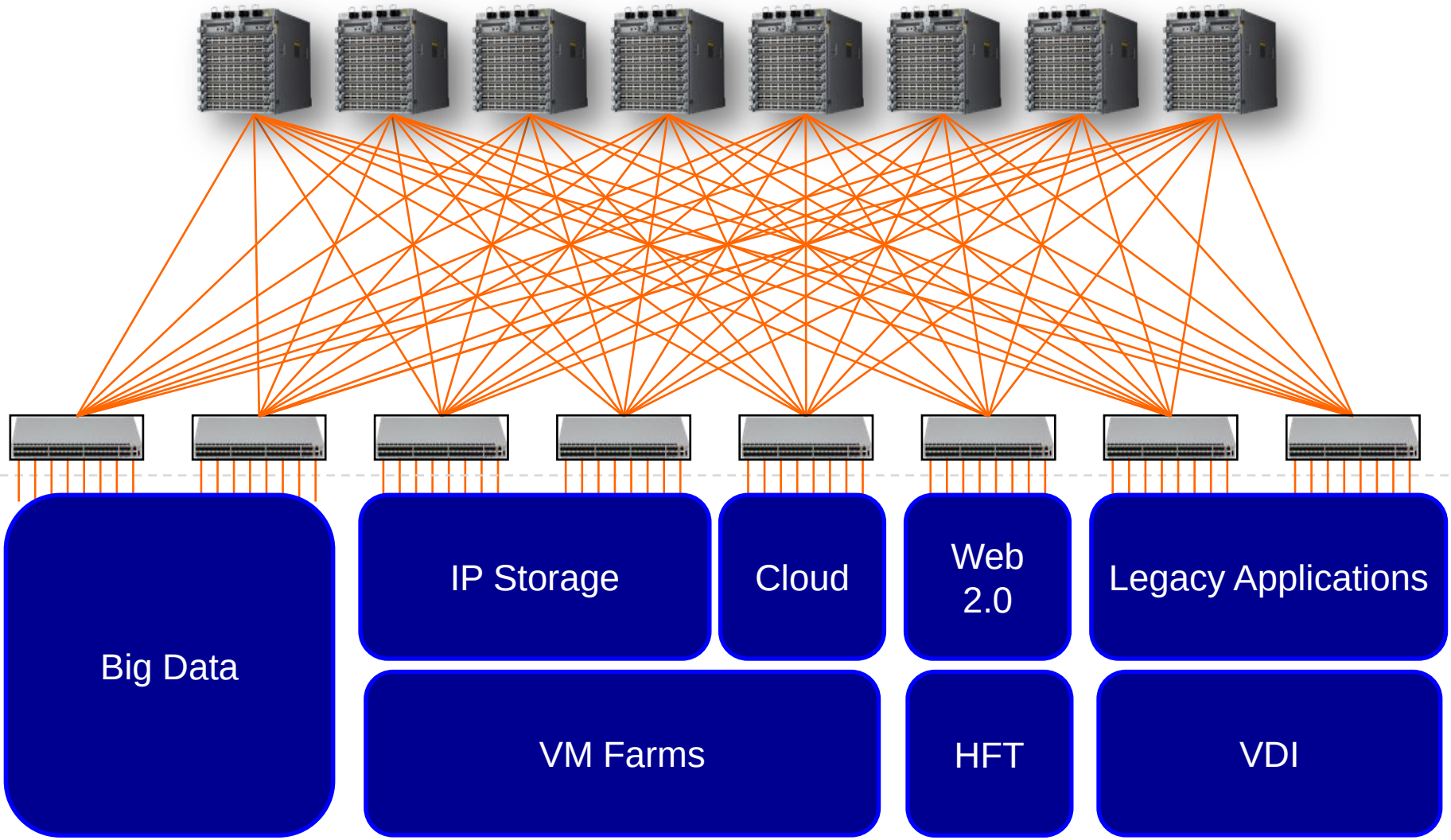
Open Partnering: connecting the network to the best and most powerful infrastructure-centric applications available

Introducing: a series of powerful applications to run on Arista EOS in distributed system

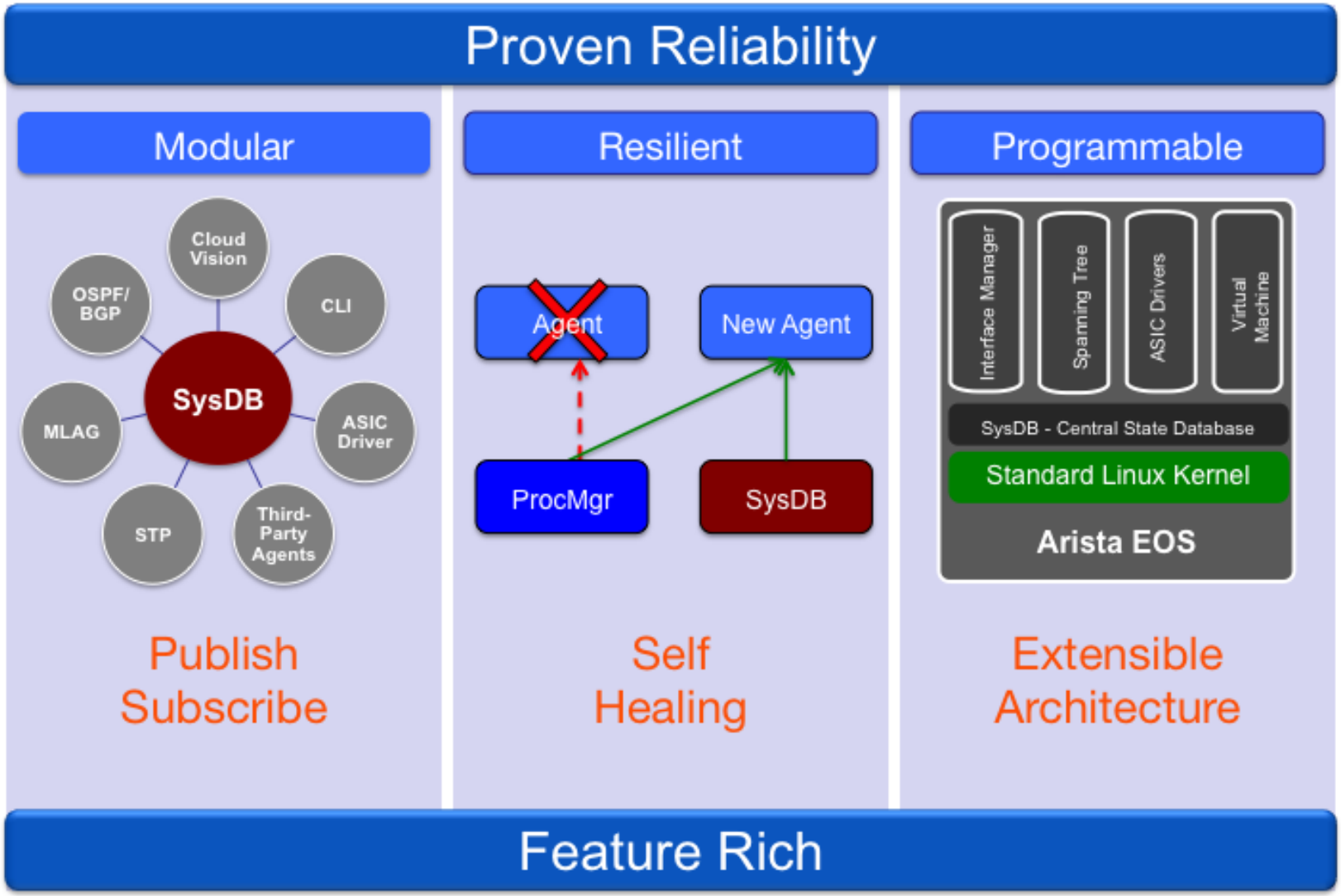
Modular distributed system designed to be customized for customer's IT operations

Universally capable infrastructure – enables any application and workload combination

Universal Cloud Network Design



Arista EOS - Software for the SDCN



Arista EOS - Extensible

EOS itself is basically a bunch of RPMs

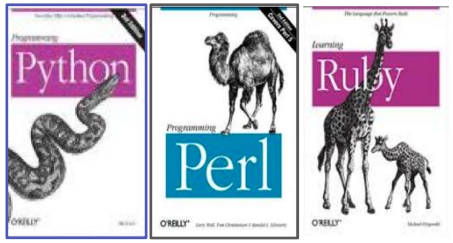
Yes, that is Fedora Linux

Yes, you can install your own RPMs

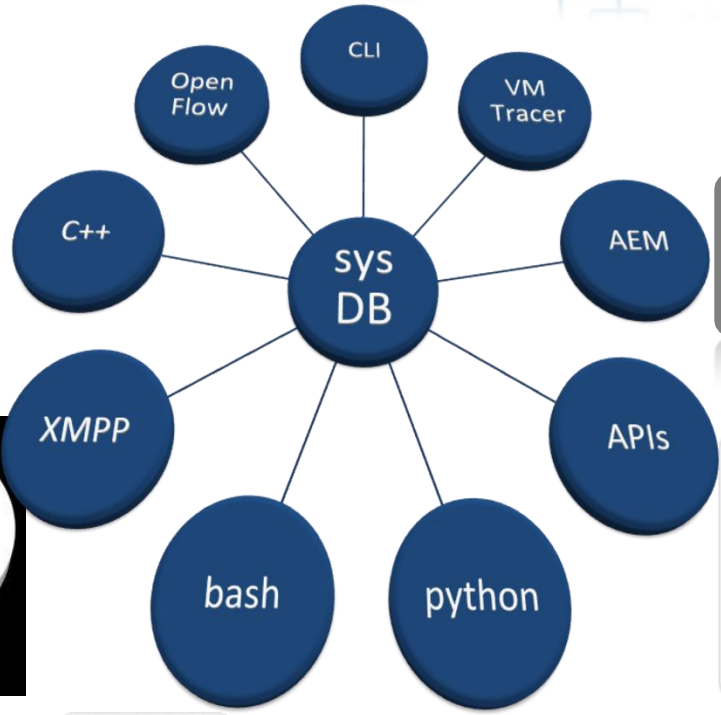
```
[charles@switch ~]$ sudo rpm -qa | head -20
inotify-tools-3.14-1.fc14.i686
libgcc-4.5.1-4.fc14.i686
hdparm-9.27-1.fc13.i686
zlib-1.2.5-2.fc14.i686
lftp-4.0.9-3.fc14.i686
readline-6.1-2.fc14.i386
CliMode-1.0.0-975983.EOS4104devctikku.i686
xz-libs-4.999.9-
0.2.beta.20100401git.fc14.i686
m2crypto-0.20.2-9.fc14.i686
glib2-2.26.0-2.fc14.i686
Cli-1.13.1-978534.EOS4104devctikku.i686
elfutils-libelf-0.152-1.fc14.i686
i2c-tools-3.0.2-4.fc12.i686
libcap-2.22-1.fc14.i686
EosUtils-1.1.0-978534.EOS4104devctikku.i686
cpio-2.11-2.fc14.i686
Cdp-1.0.0-981867.EOS4103OpenFlow.i686
dbus-libs-1.4.0-3.fc14.i686
IpEth-1.0.0-981867.EOS4103OpenFlow.i686
net-snmp-5.7.1-803098.2012eos49Xmerge.i686
[charles@switch ~]$
```

Arista EOS - Self Healing Programmable API's

eAPI simplifies integration with 3rd party tools & products



Install and run Perl, Python, Ruby scripts or other languages



VMware integration into vSphere and vCloud

API integration with OpenStack

Monitor key events on switch and take proactive action

Native API calls developed with key partners

Communicate with 000's of switches using scalable management plane

Customize, install and run BASH scripts

Customize, install and run Python scripts

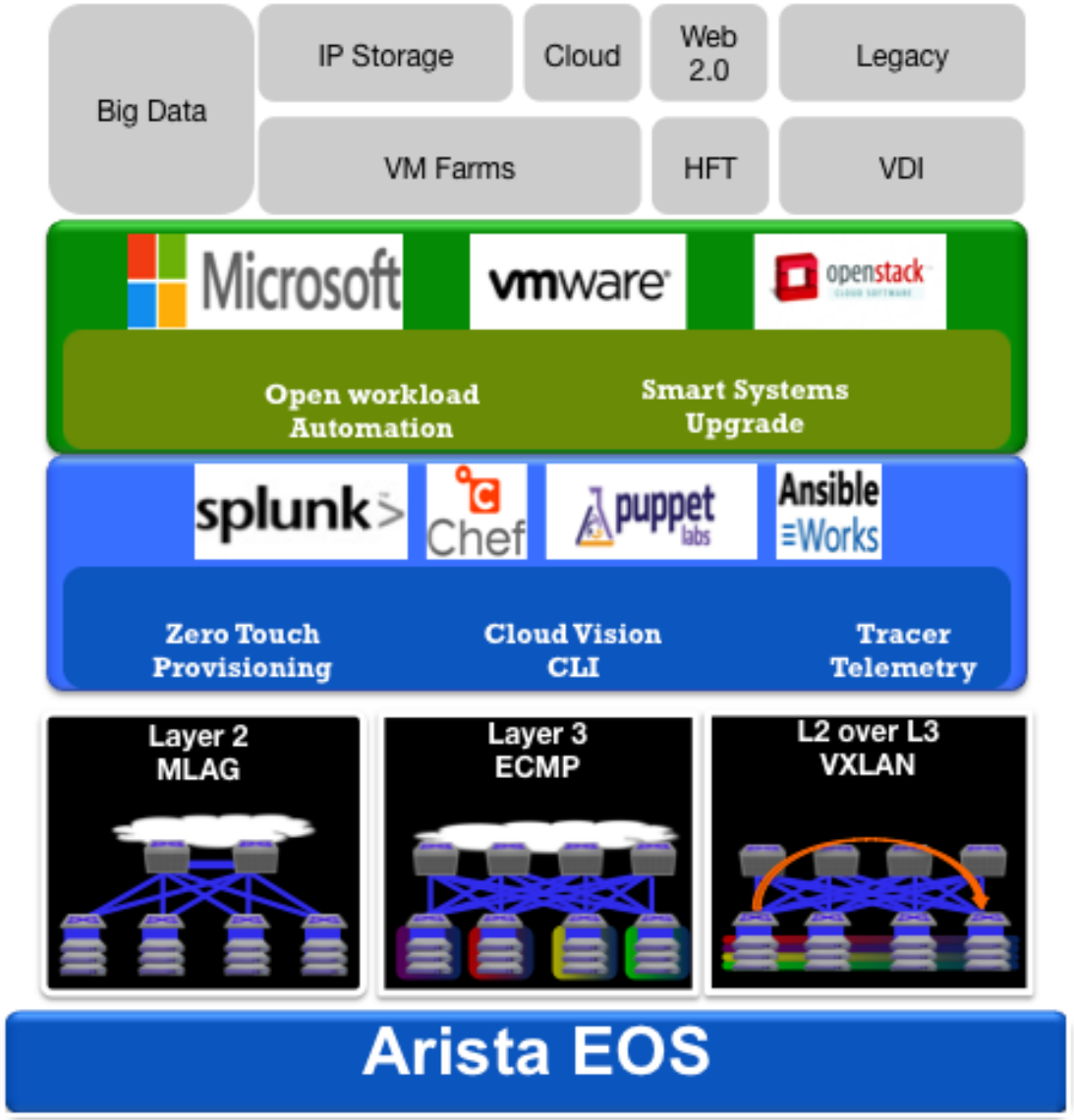
Arista EOS - Software for the SDCN

Applications

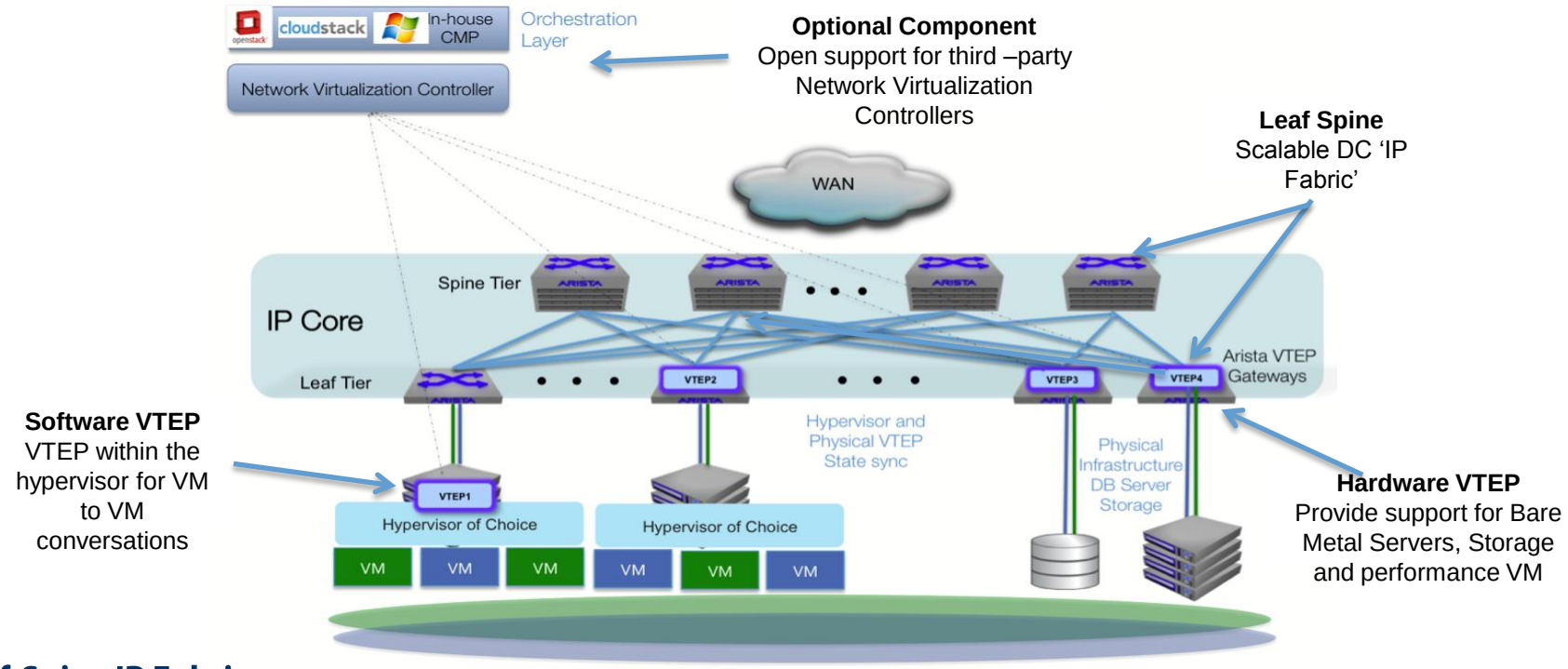
Virtualization

Operations
Deployment

Universal
Cloud Network



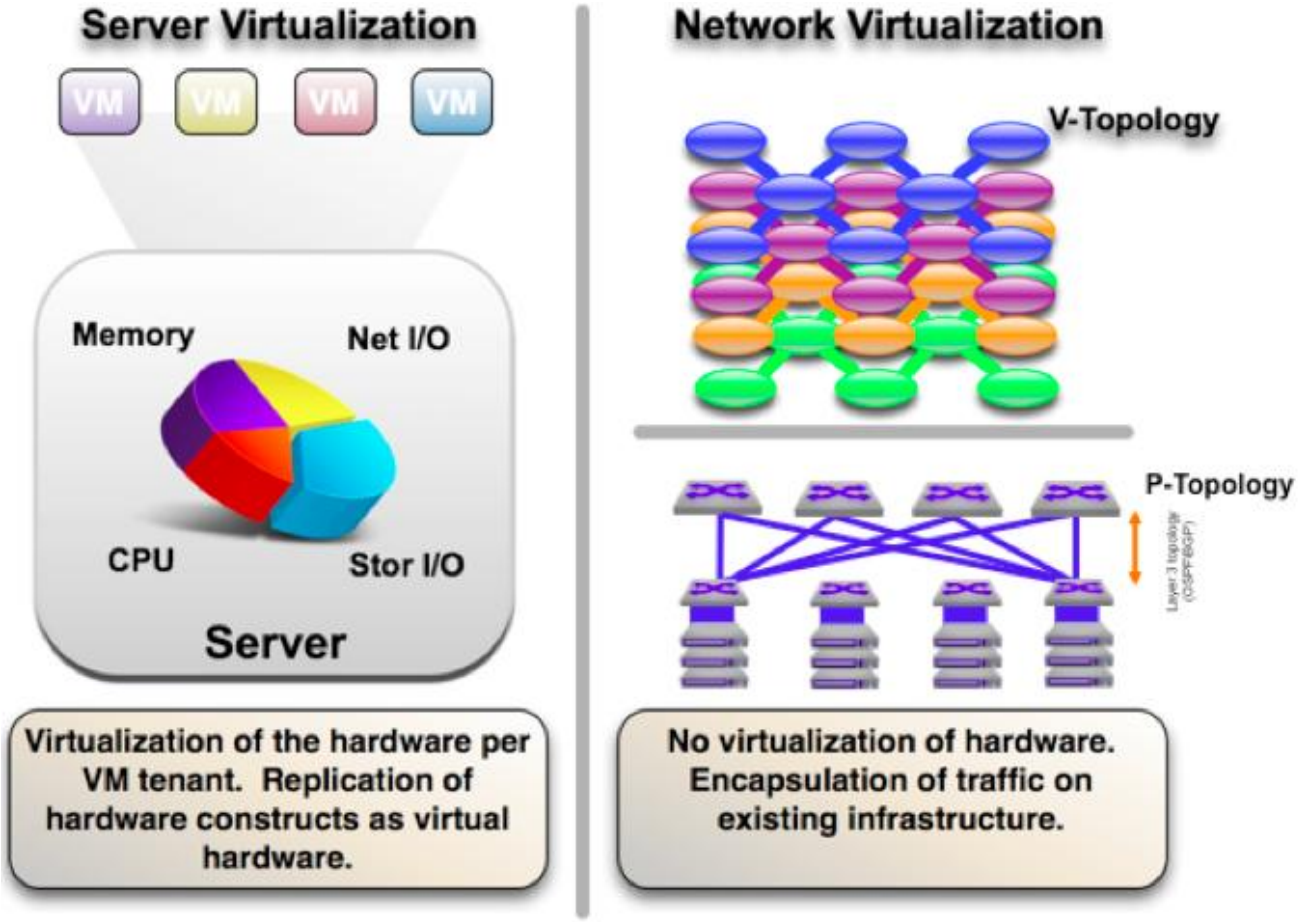
Arista Network Virtualization Architecture



- **Leaf-Spine IP Fabric**
  - Scalable Layer 3 ECMP fabric for horizontal traffic growth of Server to Server communication
- **Software VTEP**
  - VXLAN VTEP component within virtual switch of the hypervisor for VM to VM communication
- **Hardware VTEP**
  - VTEP component within the Leaf/Spine switches for bare metal servers, network services, performance requirements or lack VTEP support in hypervisor
- **Network Virtualization Controllers**
  - Open API, (JSON. OVSDB, Neutron) for open integration into third party NVC controllers for simplified end-to-end orchestration

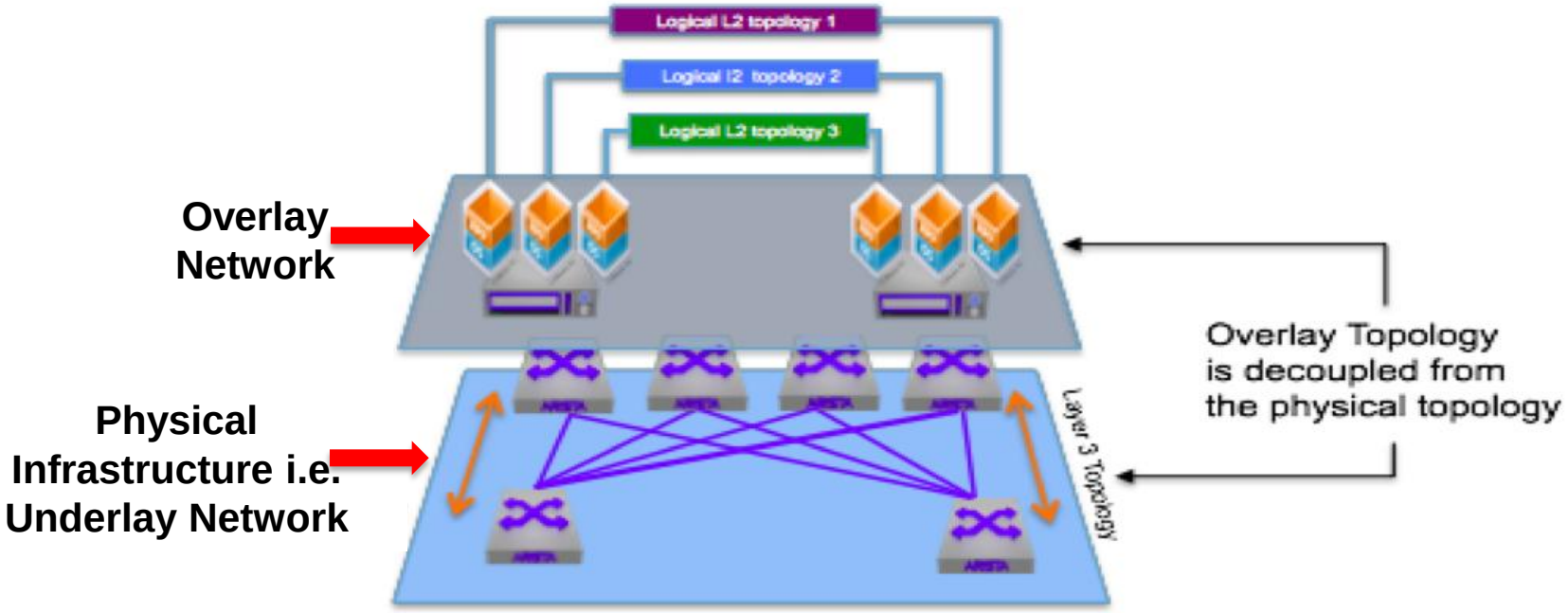
What is Network Virtualization?

Network Virtualization is not the same as Server Virtualization!



Overlays vs Underlays

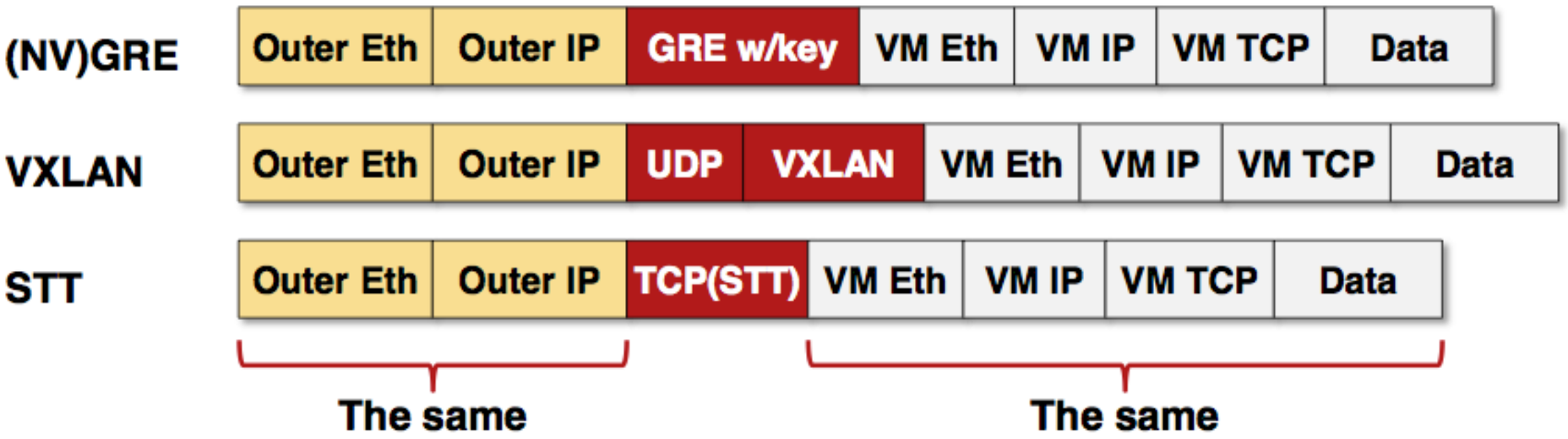
Network virtualization: ability to separate, abstract and decouple the physical topology from a 'logical' or 'virtual' topology by using encapsulated tunneling.



This **logical** network topology is often referred to as an ***'Overlay Network'***.

VXLAN disassociates workloads from physical networks, allowing for possible transition to cloud based providers

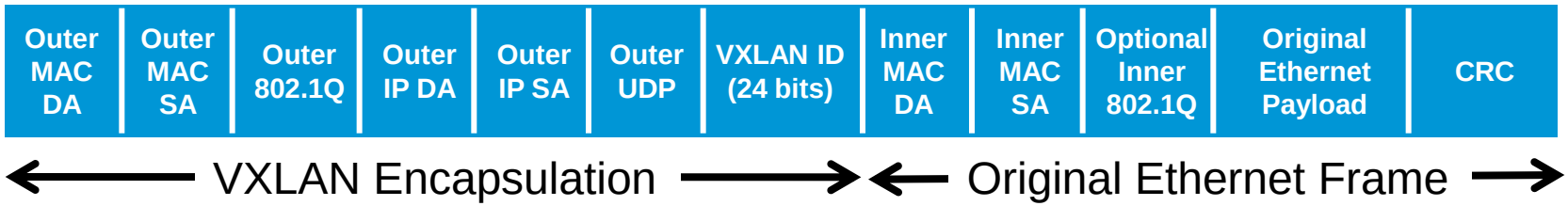
The Encapsulation Wars



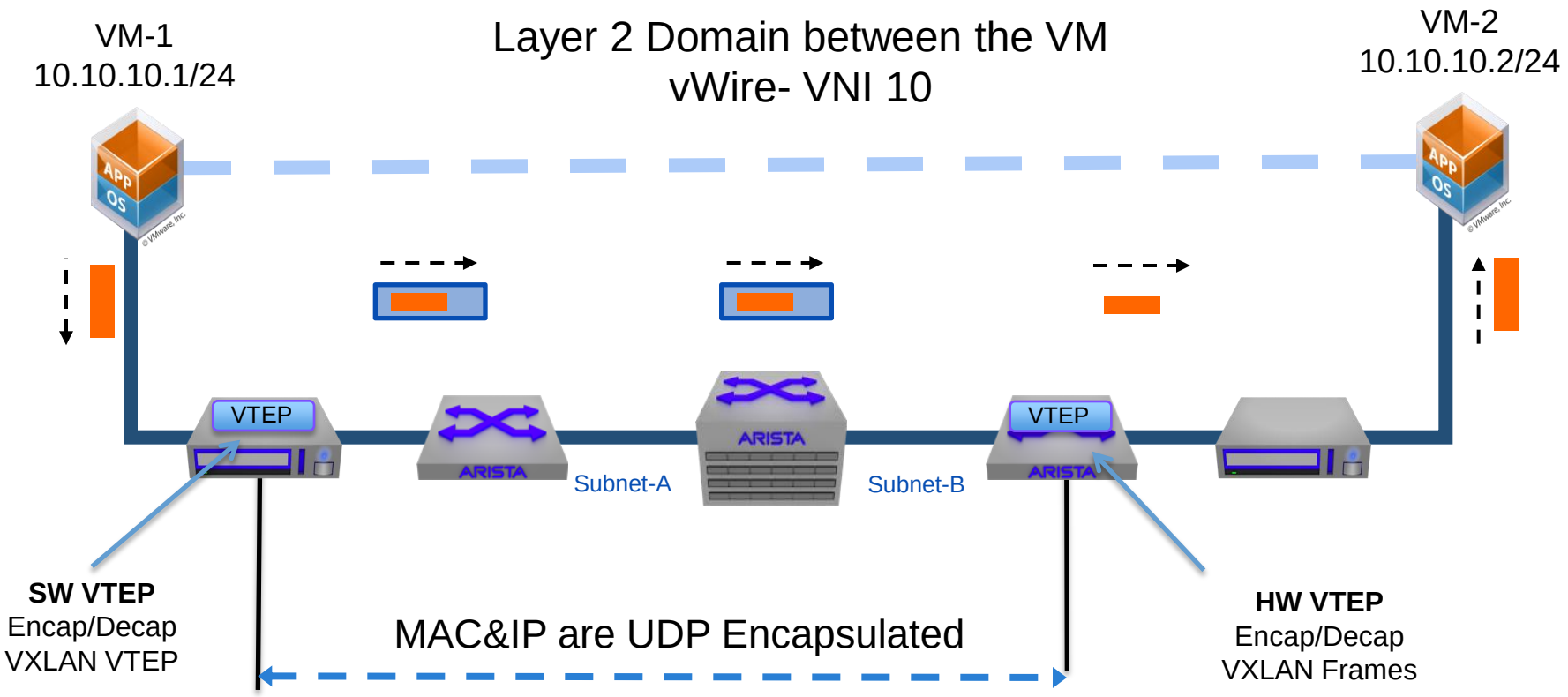
- Three Competing encapsulations
- Minor technological differences (load balancing, TCP offload)
- None supported by legacy networking hardware or IDS/IPS gear
- What really matters is the control plane

Virtual Extensible Local Area Network (VXLAN)

- **Ethernet in IP overlay network**
  - Entire L2 frame encapsulated in UDP
  - 50 bytes of overhead
- **Include 24 bit VXLAN Identifier**
  - 16 M logical networks
- **VXLAN can cross Layer 3**
- **Tunnel between ESX hosts**
  - VMs do NOT see VXLAN ID
- **IP multicast used for L2 broadcast/multicast, unknown unicast**
- **Technology submitted to IETF for standardization**
  - With Arista, Vmware, Red Hat, Citrix, Cisco, and Others



Virtual eXtensible LAN: How does it work?



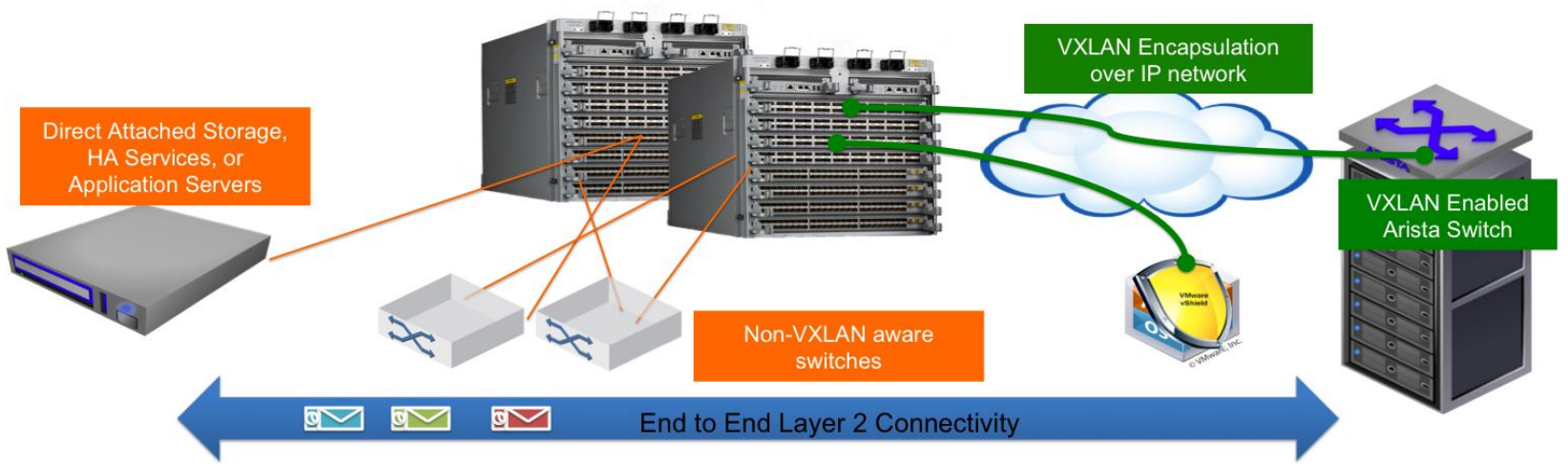
Encapsulation at VTEP node is transparent to IP ECMP fabric

VXLAN Benefits

- **Feature Benefits**
  - Eliminates current networking challenges in the way of on-demand, virtual environment:
    - VLAN Sprawl
    - Single fault domains
    - Scalability beyond 4096 segments
    - Proprietary fabric solutions
    - IP mobility
    - Physical cluster size and locality
  - Enables multi-tenancy at scale
  - Decouples logical networks from physical infrastructure so that applications can be deployed without worrying about physical rack location, IP address or VLAN
  - Based on open and well known standards

VXLAN Use Cases

- Physical to Virtual internetworking
- Multi-hypervisor connectivity and integration
- Multi-tenant Cloud environments
- HA clusters across failure domains
- Dynamic growth
- Dynamic resource management



An Overview of Open Souce Cloud Technologies

SaaS

PaaS

IaaS

Open Source, OpenStack

IaaS

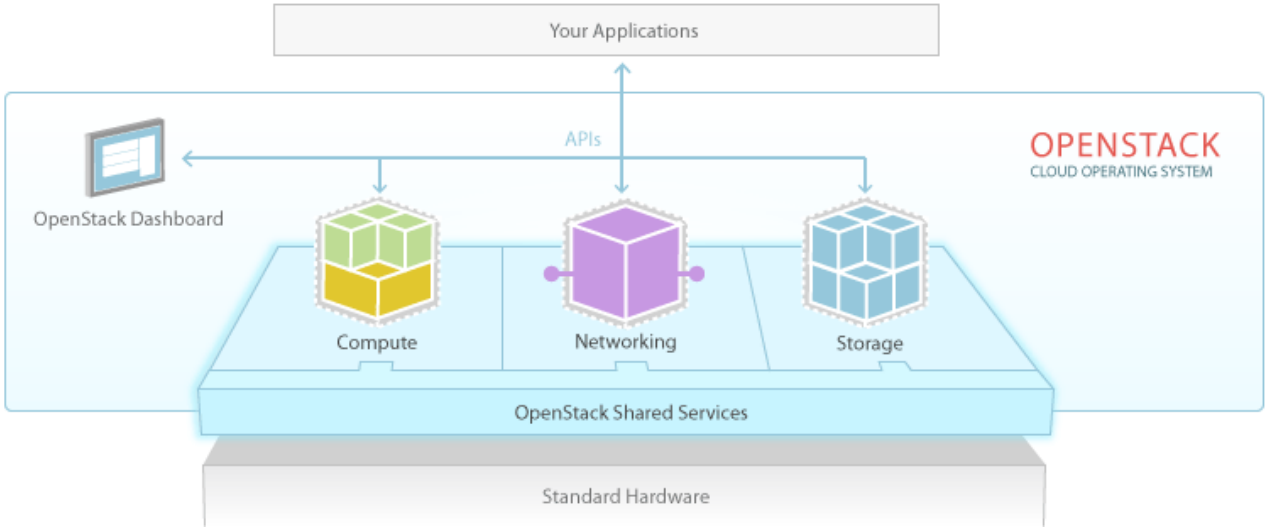
Open Source

June, 2010

+

Apache2.0 License

About OpenStack



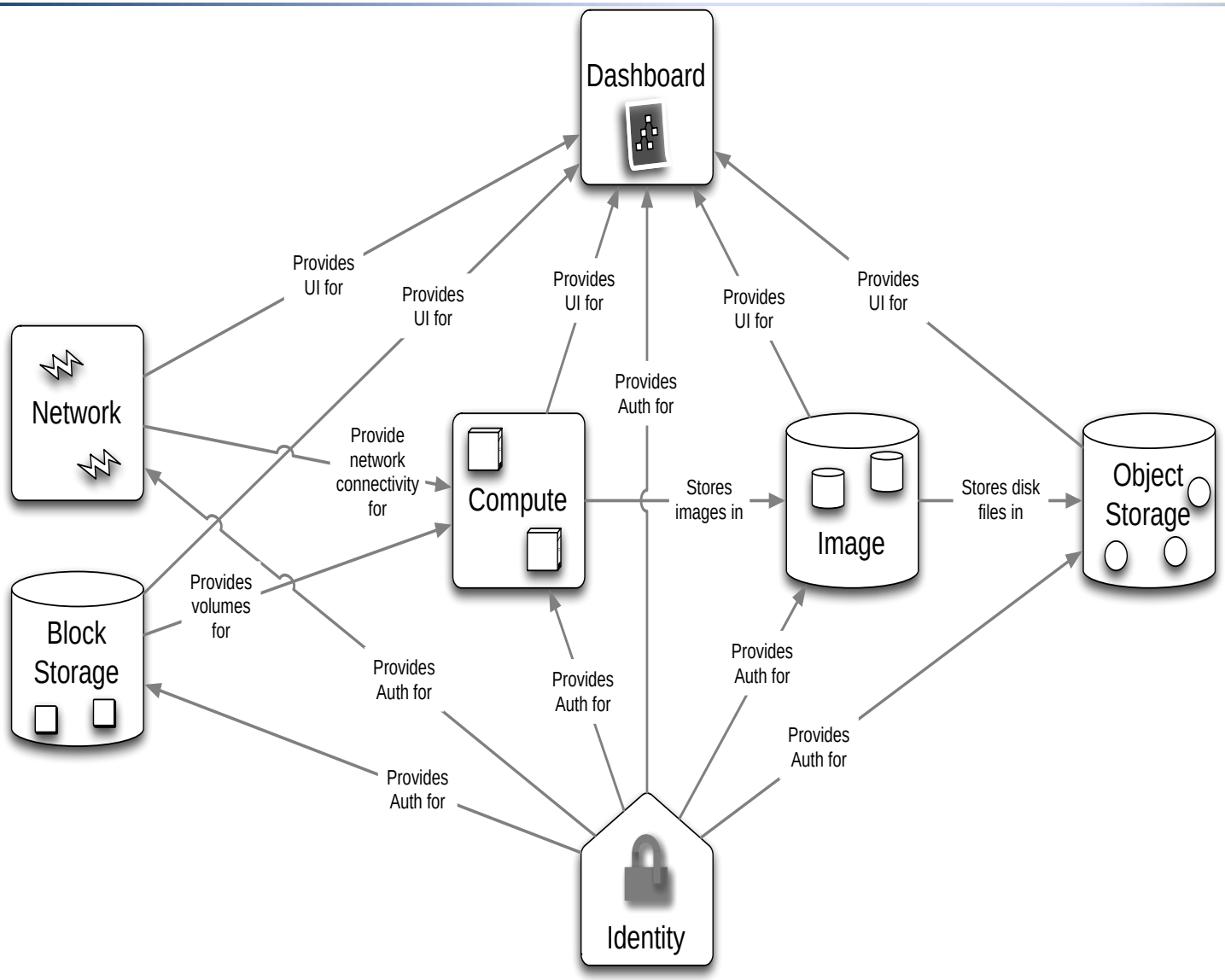
OpenStack is a **cloud operating system** that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed through a dashboard that gives administrators control while empowering their users to provision resources through a web interface.

April 4, 2013: Grizzly Software Release

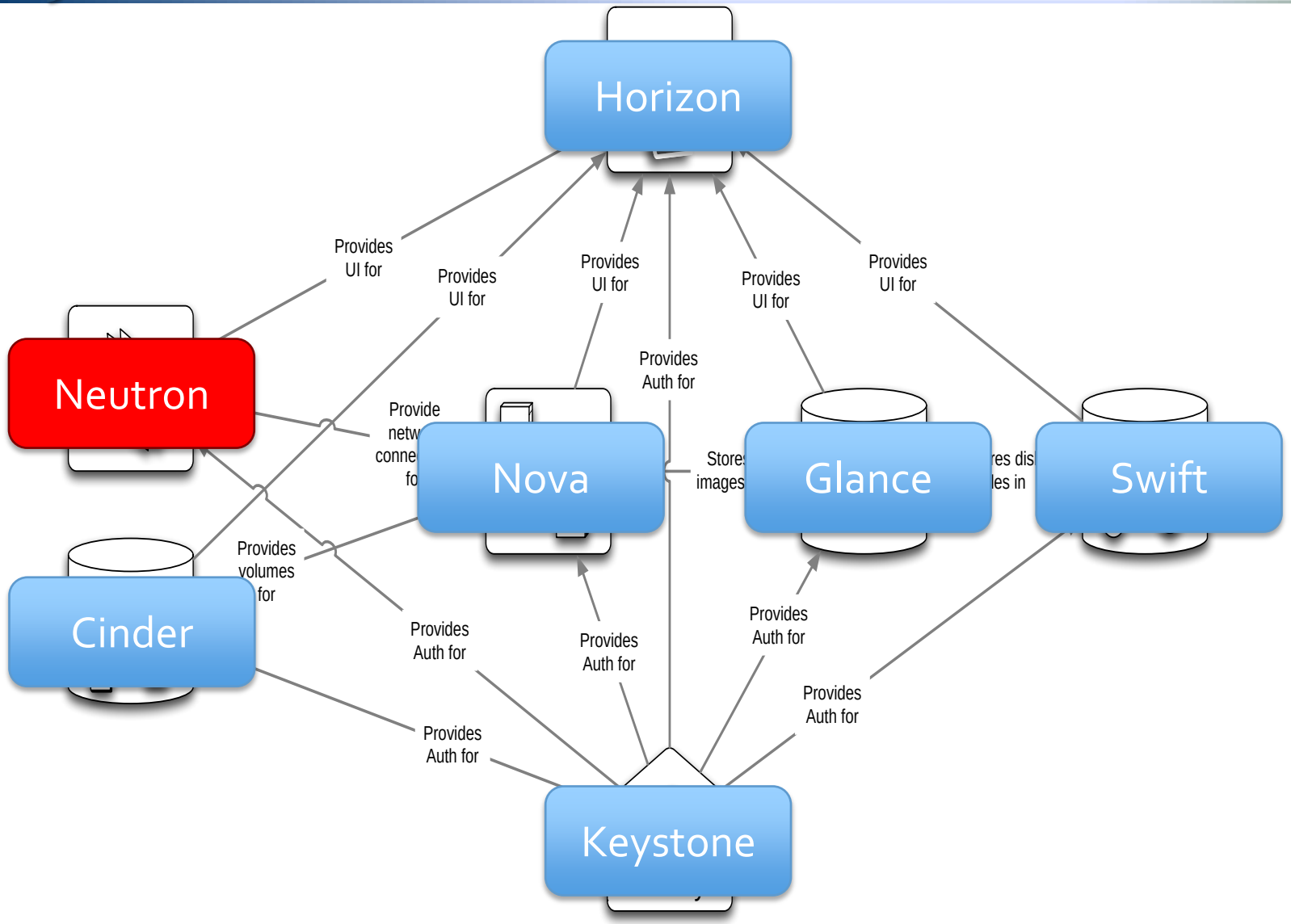
October 17, 2013: Havana Software Release

April 4, 2014 : Icehouse Software Release

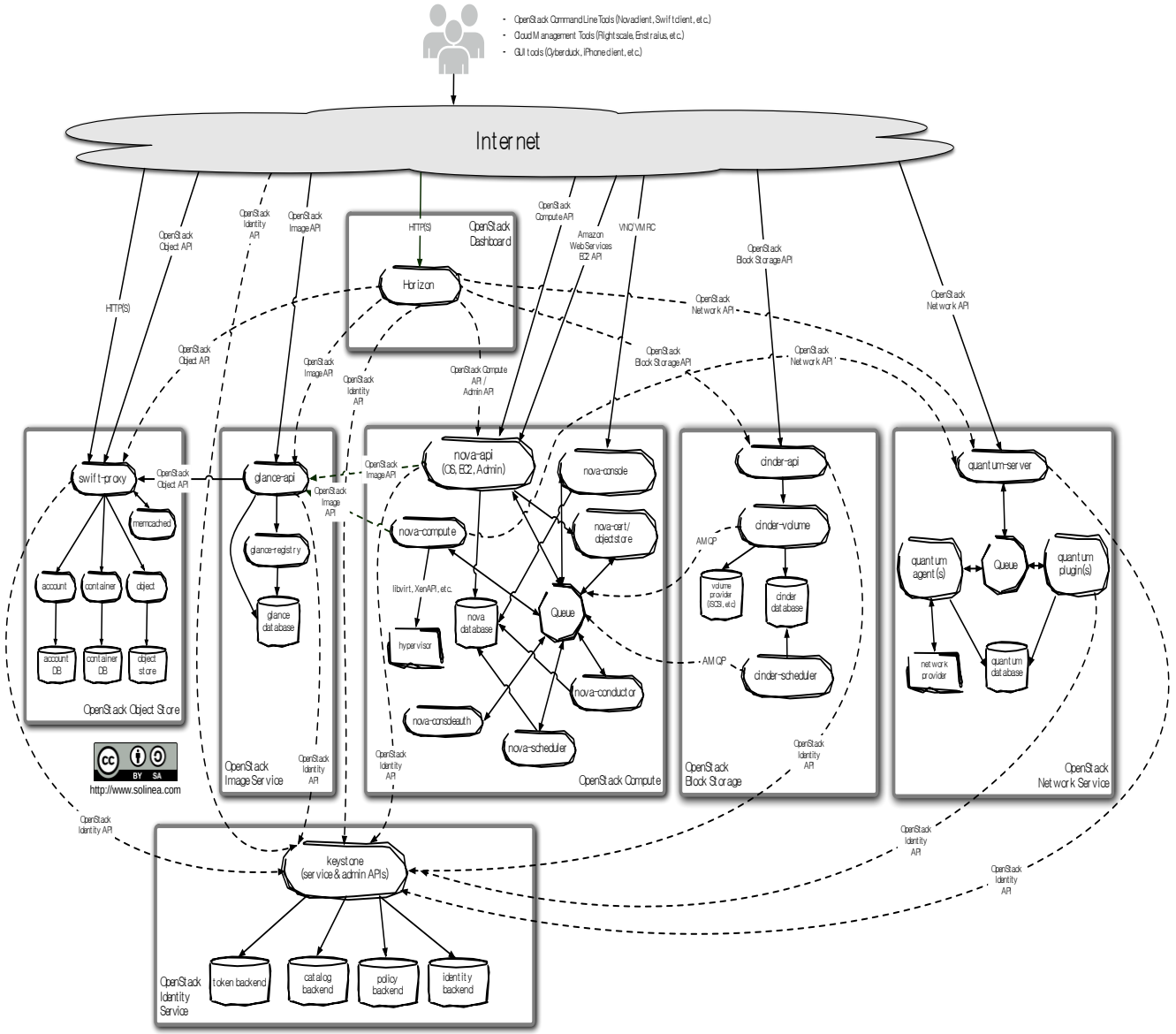
OpenStack Conceptual Architecture



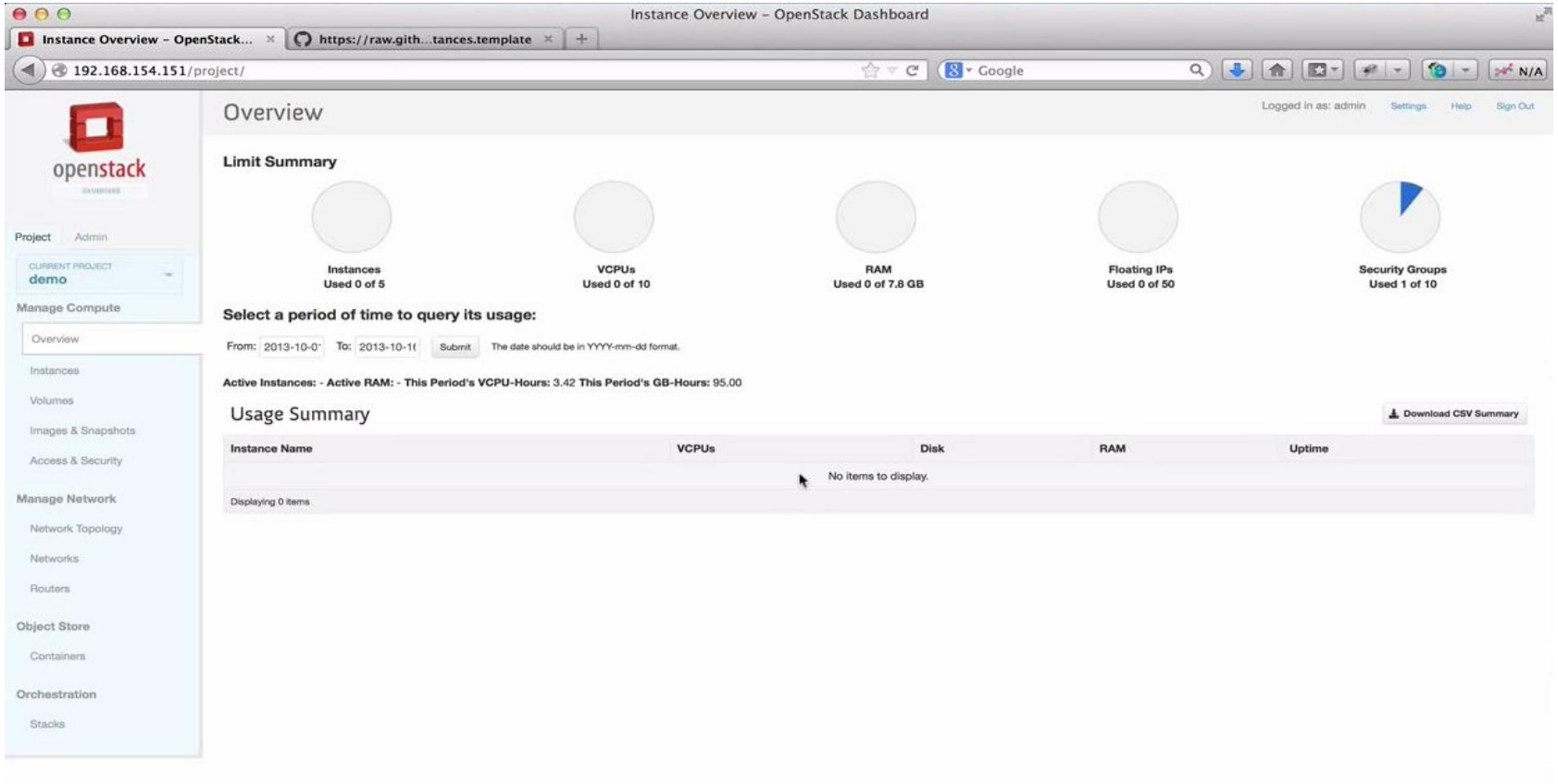
OpenStack Conceptual Architecture - Project Codename



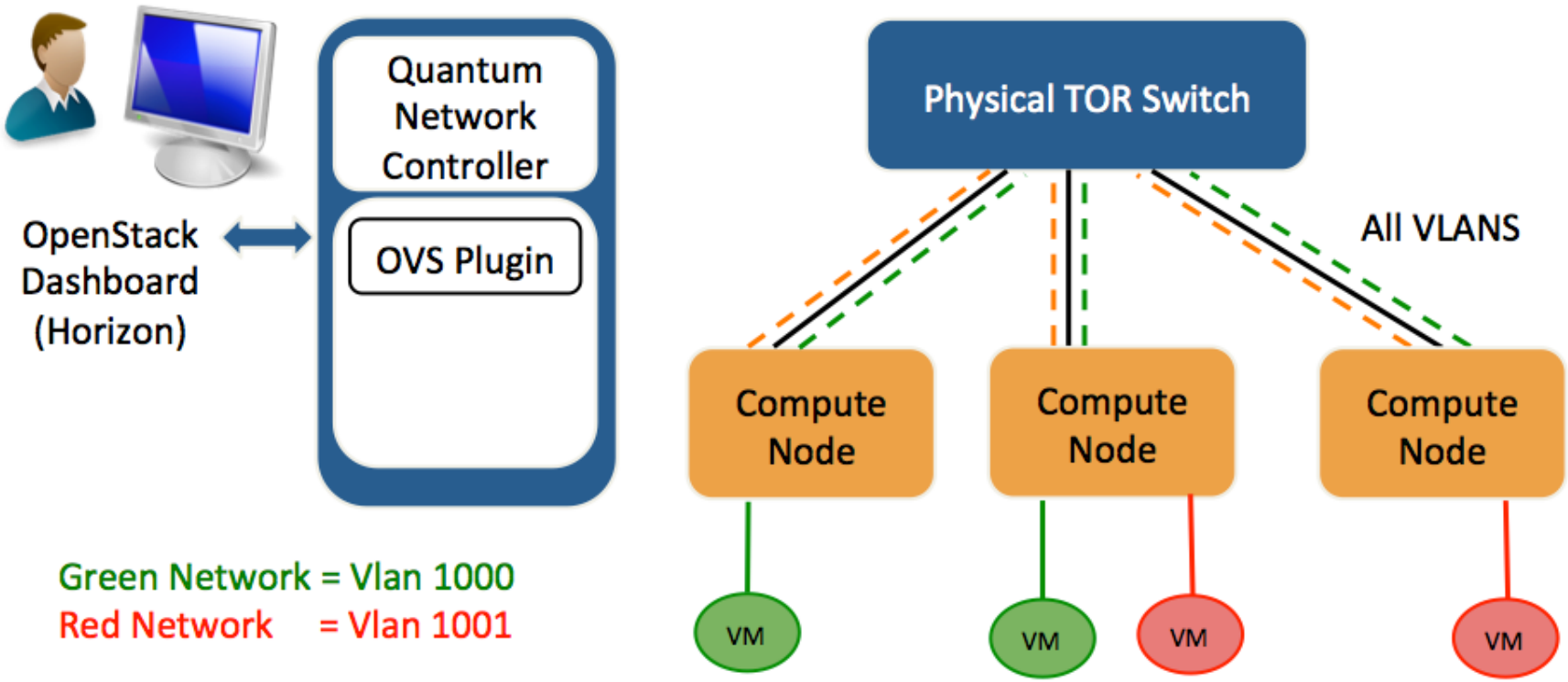
OpenStack Logical Architecture



OpenStack Horizon

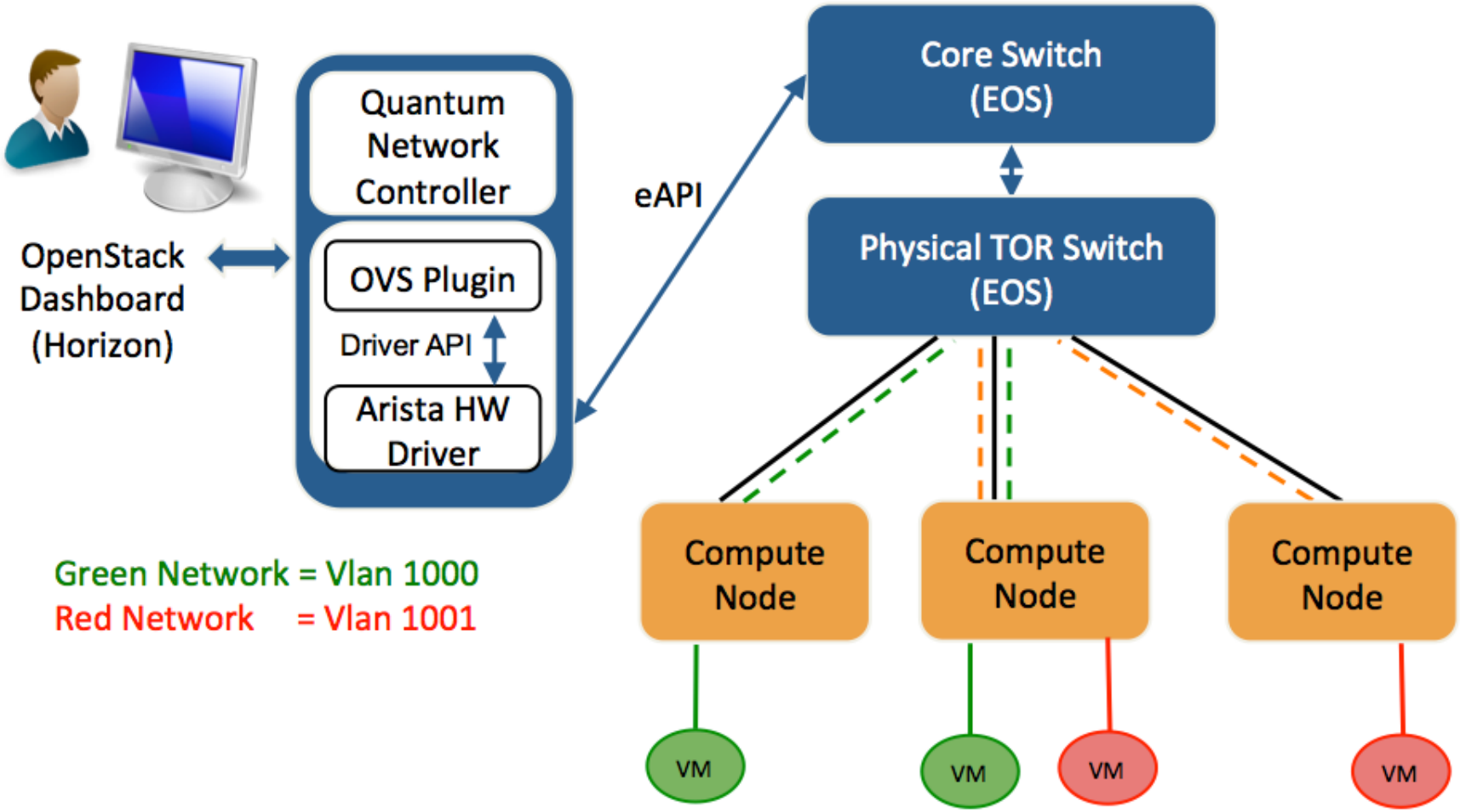


Problem



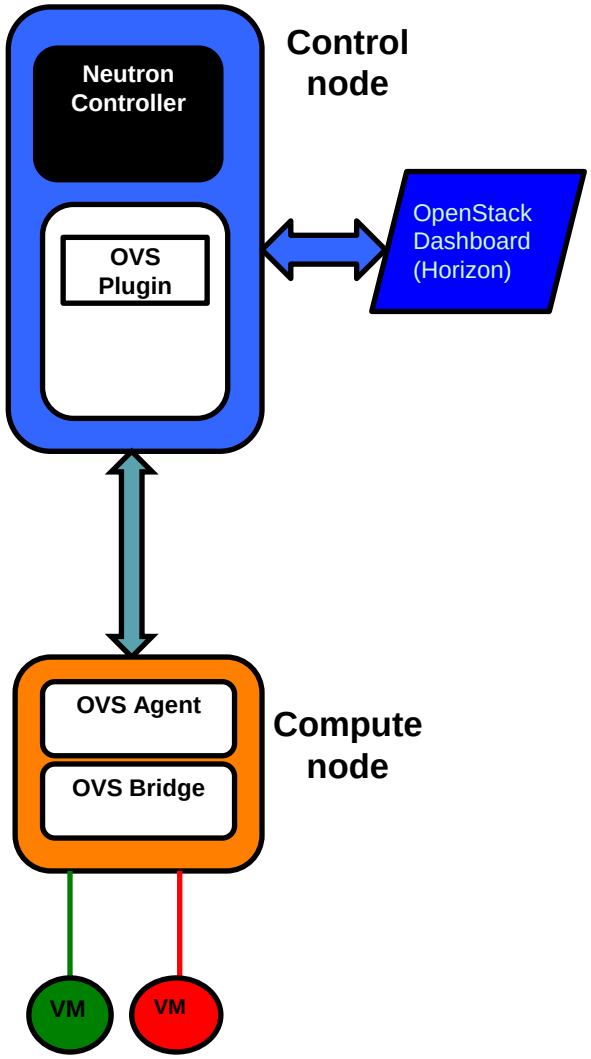
- **Result for the physical network is either :**
  - Suboptimal network design
  - Manual provisioning of tenant networks
  - or need an outside system to automate

Solution


Quantum Network Controller
OVS Plugin
Driver API
Arista HW Driver
OpenStack Dashboard (Horizon)
eAPI
Core Switch (EOS)
Physical TOR Switch (EOS)
Compute Node
Compute Node
Compute Node
VM
VM
VM
VM
Green Network = Vlan 1000
Red Network = Vlan 1001

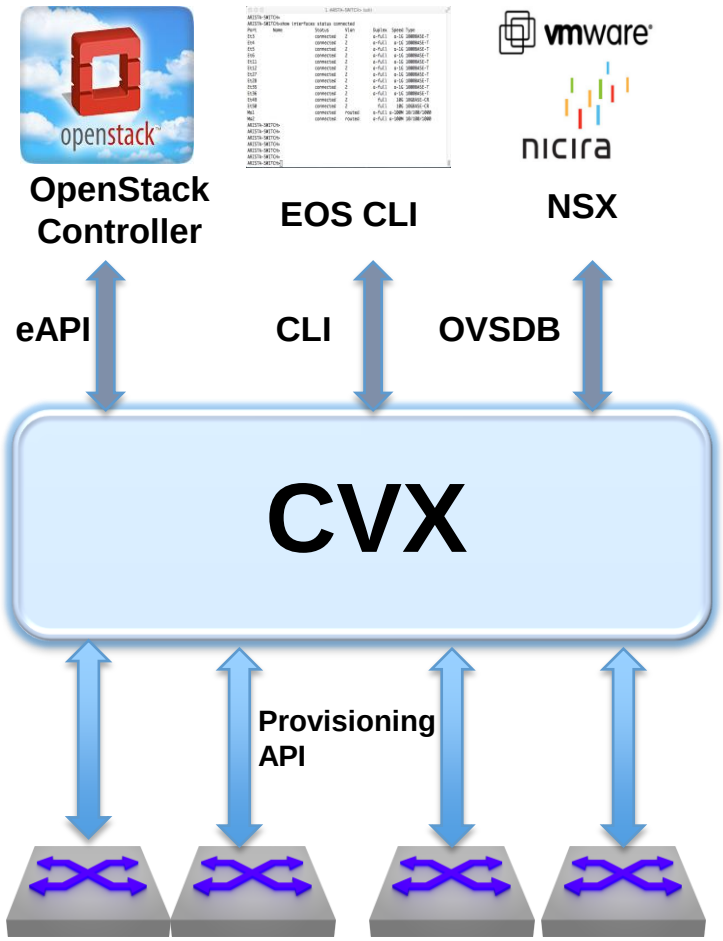
OpenStack Background

- Open vSwitch Plugin configures each vswitch on the compute hypervisor
- Neutron is OpenStack component that provides “networking as a service” between interface devices (e.g., vNICs) managed by other OpenStack services (e.g., nova)
- 2 Neutron releases to date - Folsom and Grizzly



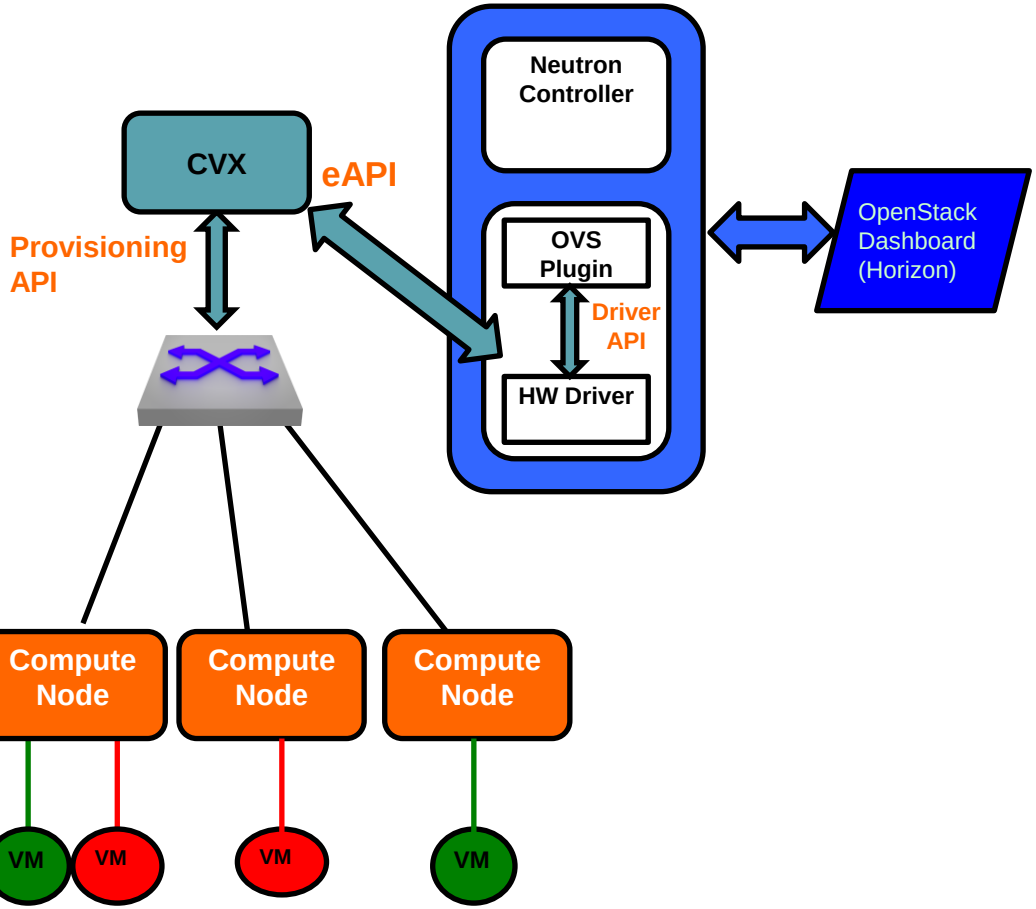
CloudVision eXtenstion (CVX)

- EOS based VM
- Orchestrates group of physical switches running EOS
- Provides single point of visibility and management
- Serves as integration point into other controllers, orchestration systems, NMS
- Can be run as a standalone VM/cluster of VMs or directly on physical switches



Arista OpenStack Integration

- Driver adapter layer sits beneath OVSPlugin – so completely new plugin is not required for each hardware vendor
- OVSPlugin manages VLAN allocation, and HW driver manages vendor-specific hardware provisioning
- Intelligent topology aggregation/discovery
- Dynamic move of both virtual and physical resources



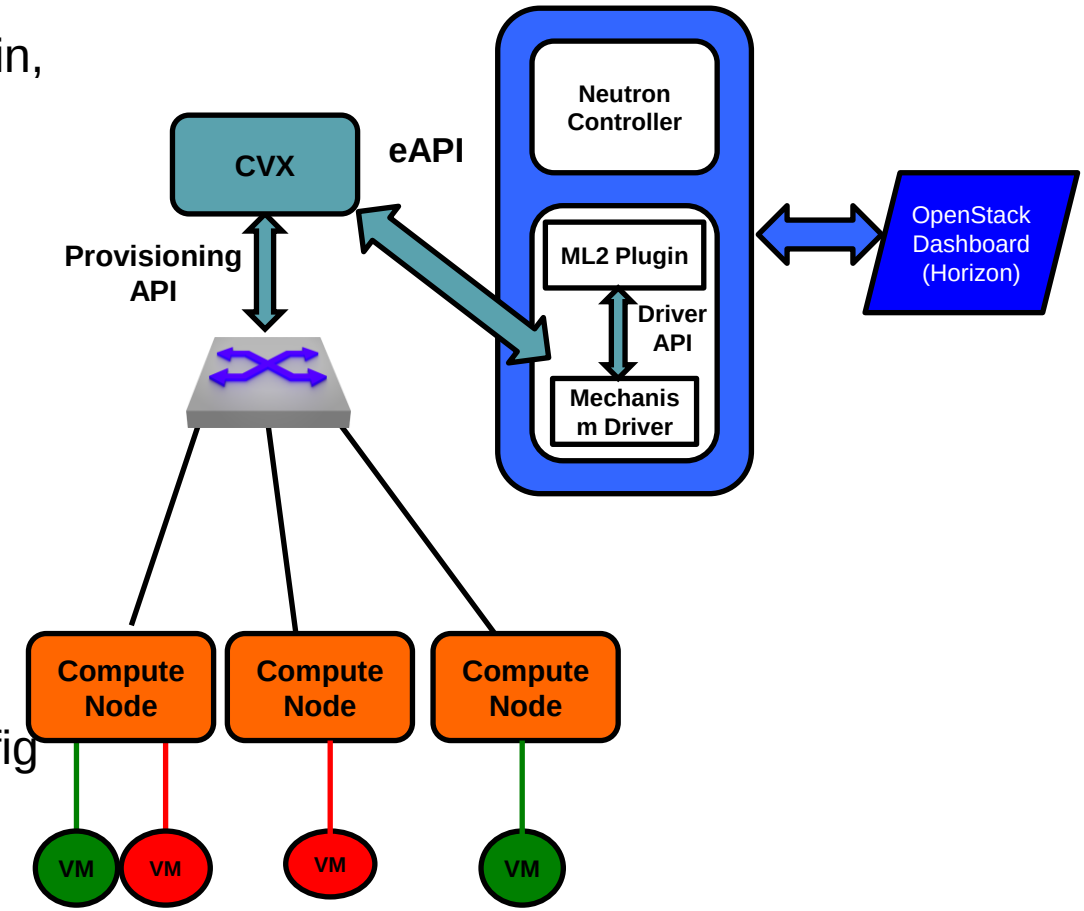
Arista OpenStack Integration

- ML2 Plugin consolidates OVSPlugin, LinuxBridgePlugin, most of vendor specific plugins

- ML2 Plugin with 3 components
  - Core Plugin
  - Type Drivers
  - Mechanism Drivers

- Arista Mechanism Driver for ML2

- Automate provisioning of switches by tapping into virtual network config



OpenStack with VXLAN Deployment

3 viable designs:

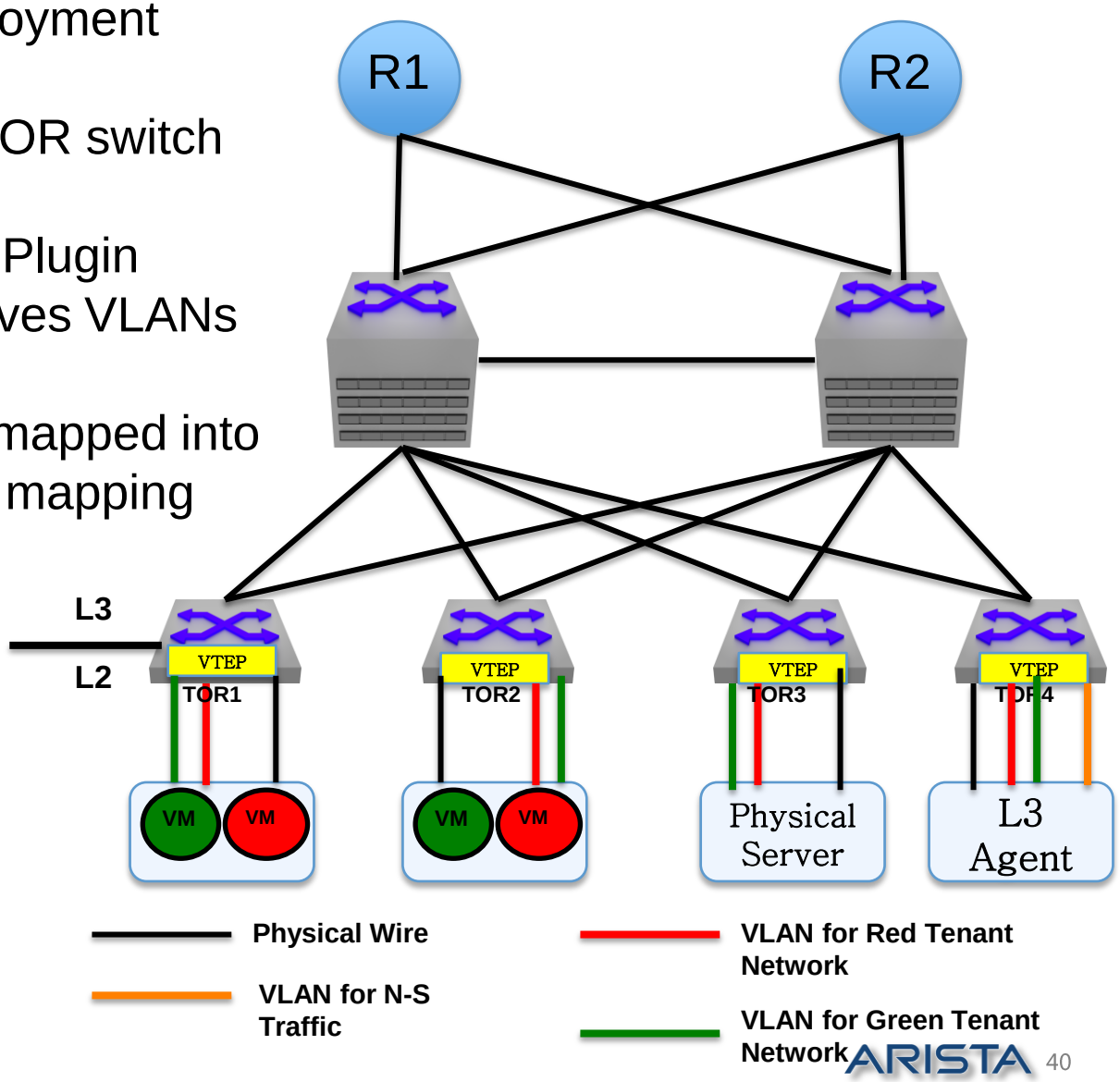
- **All virtual VTEPs**
  - External controller required to distribute VXLAN tables across all VTEPs
  - Downsides to using all virtual VTEPs

- **All hardware VTEPs**
  - VXLAN runs at every TOR switch
  - Quantum plugin operates in same way as with VLANs
  - Limited to up to 4K tenant networks

- **Mix of both virtual and hardware VTEPs**
  - Virtual switches as VTEPs at compute nodes and physica switches as VTEPs when bridging non-virtual resources into logical tenant networks
  - Must use either multicast for BUM traffic or external controller to share VXLAN tables with CVX

All hardware VTEPs

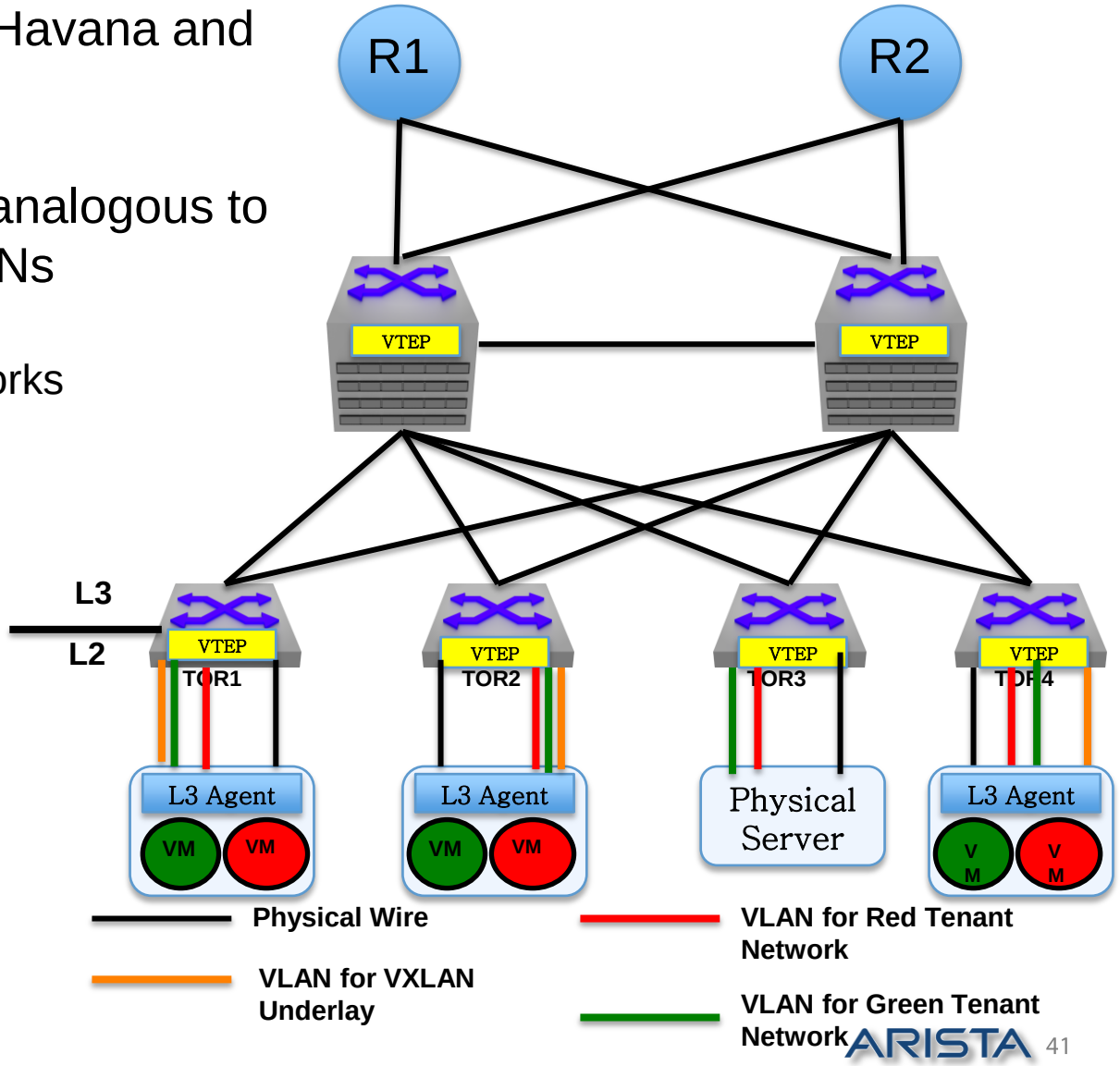
- Folsom and Grizzly Deployment
- VXLAN between every TOR switch
- Arista HW driver in OVS Plugin automatically adds/removes VLANs
- Added/removed VLANs mapped into VXLANs based on static mapping



OpenStack with VXLAN Deployment

- Multi-Host Deployment (Havana and beyond)
- Traffic flow with VXLAN analogous to pure L2 design with VLANs

❖ Traffic between tenant networks
❖ North-South traffic



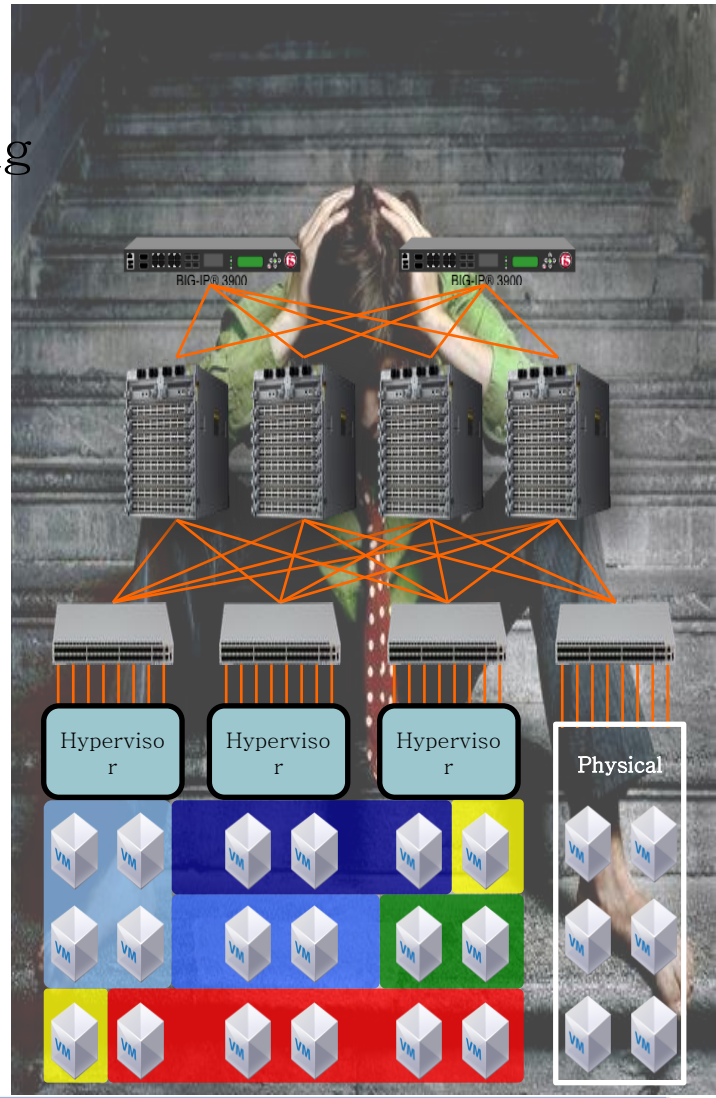
Arista Network – Extensibility Services

How do I easily get my cloud up and running (automated, hands / mistake free, rapid)?

How do I push updates for software or config changes to 10, 100s or 1000s of devices?

What about standardized configuration and corporate compliance?

Did you know server / VM are deployed at scale and their admins have already solved this problem?



How do I deploy & operate my data center?

Zero Touch Provisioning

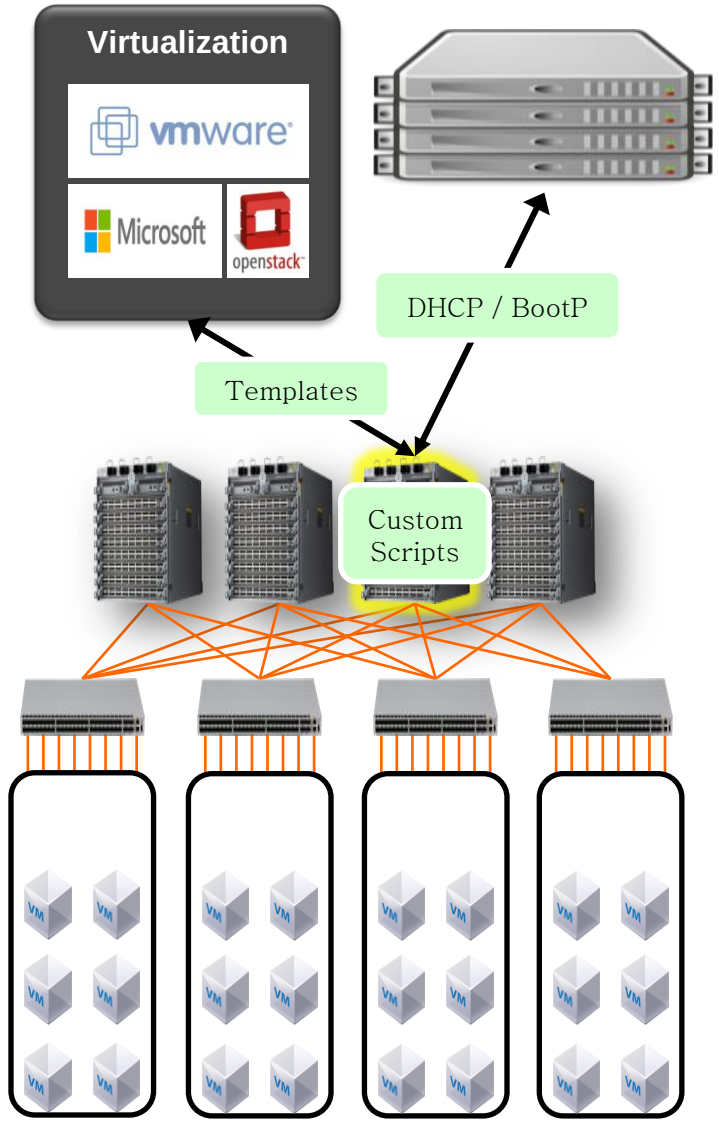
Standards based

Dynamic deployment

Rapid replacement

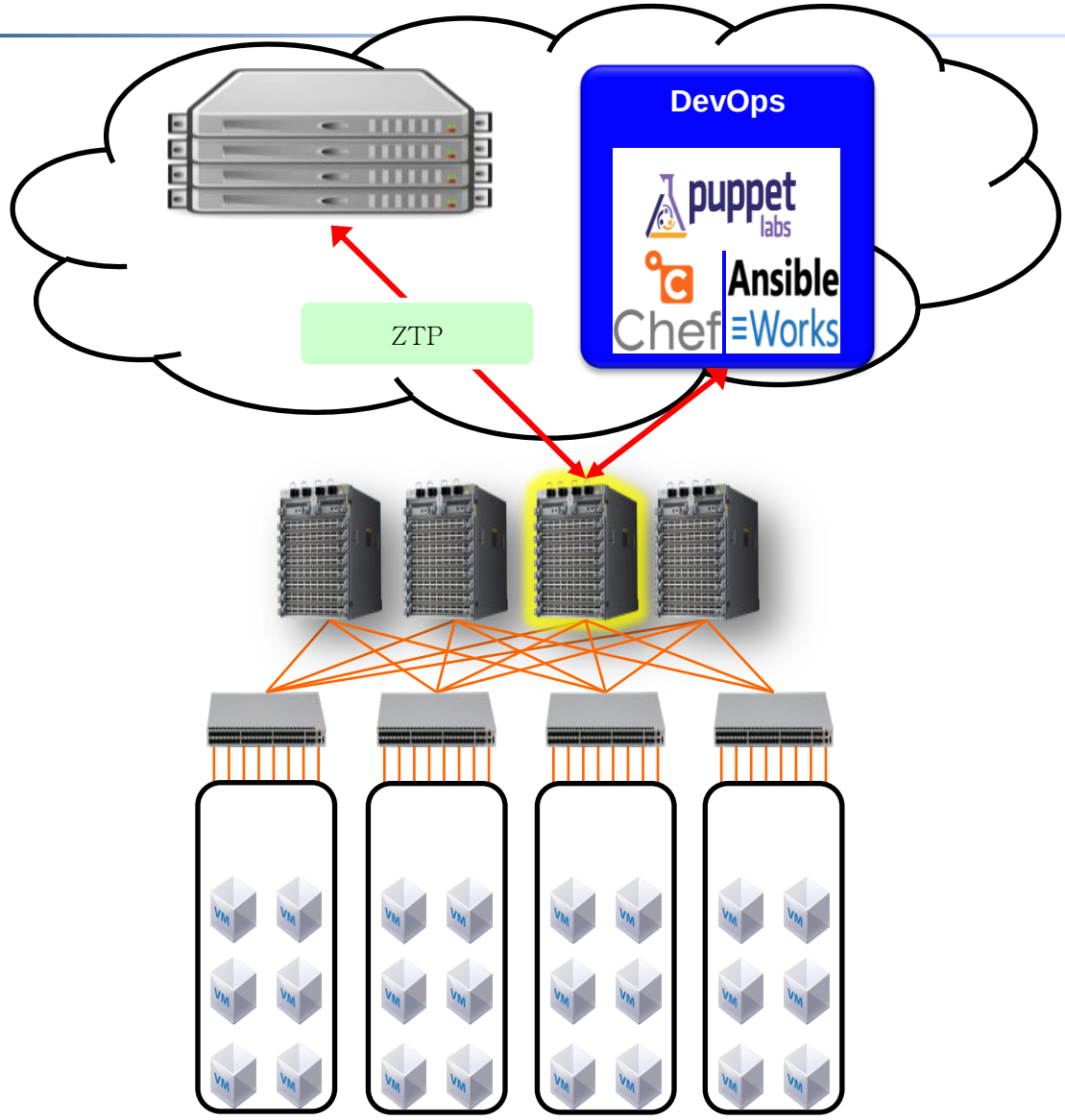
Automate Deployments

Virtual Templates



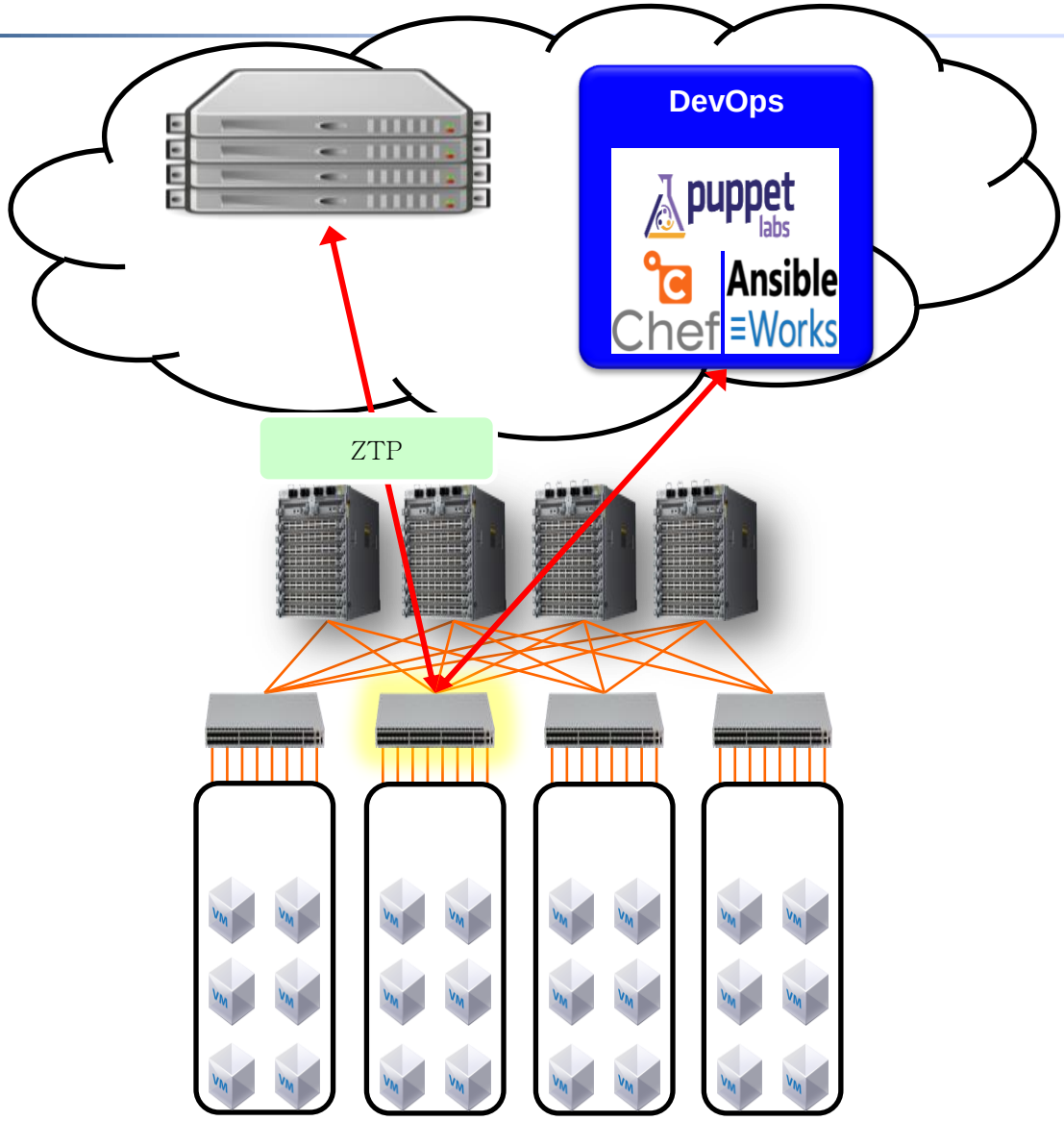
Intelligent self provisioning since 2010

Step 1: Provision the network



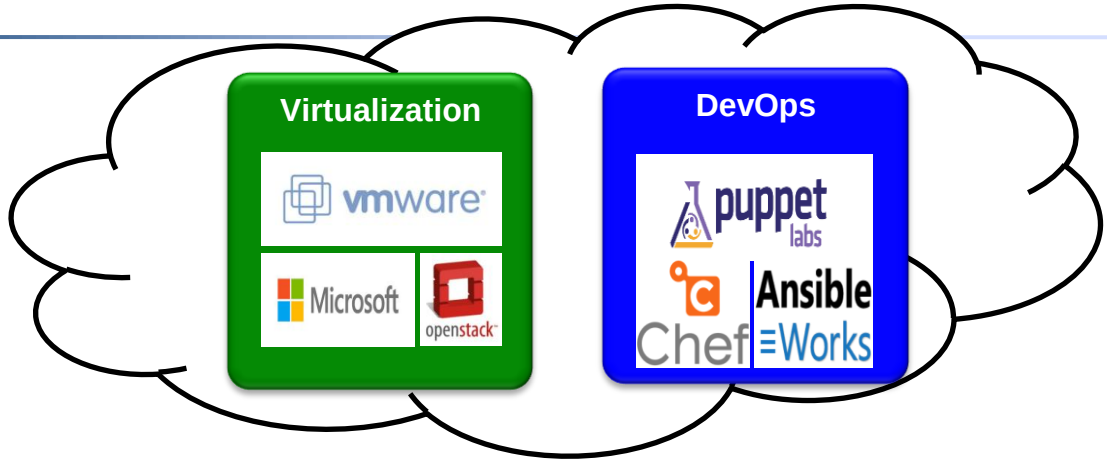
Build the spine

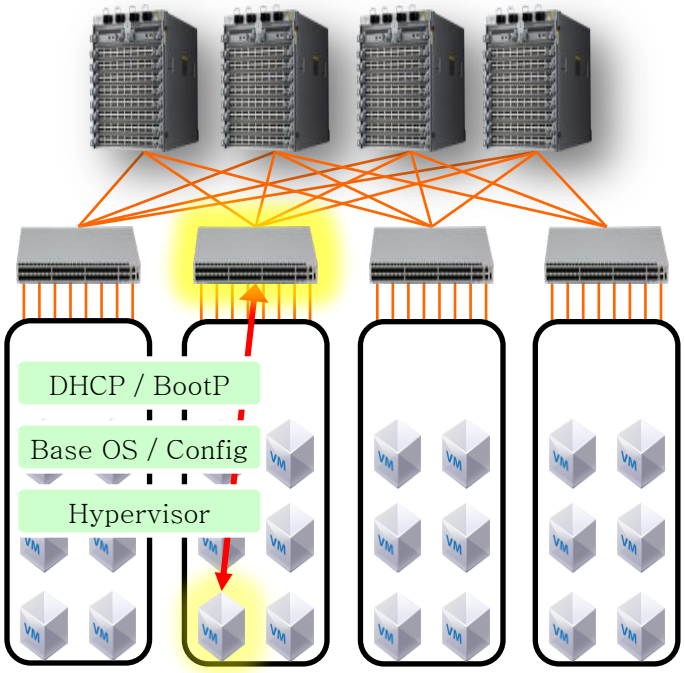
Step 1: Provision the network



Build the leaf

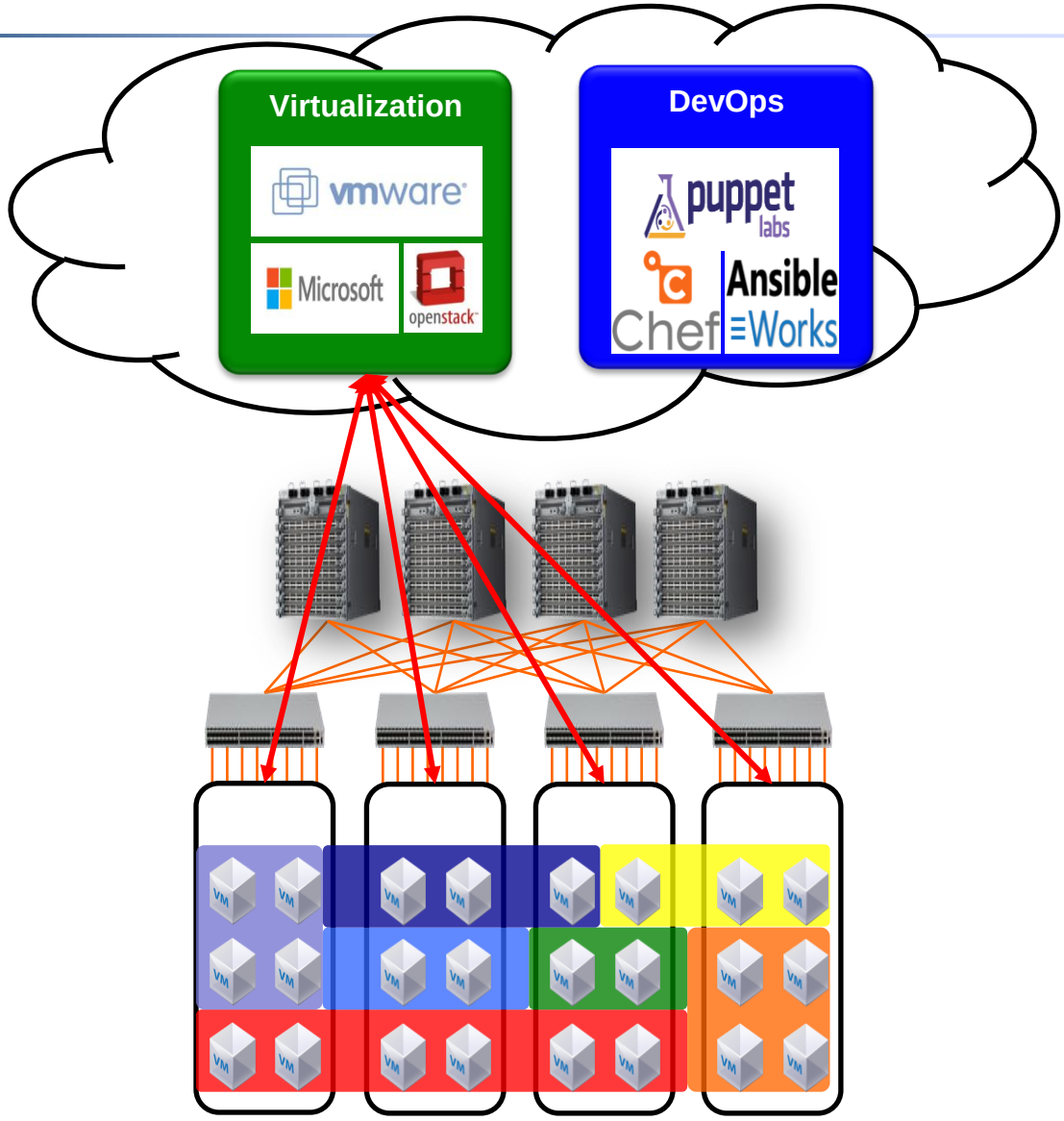
Step 2: Provision the servers





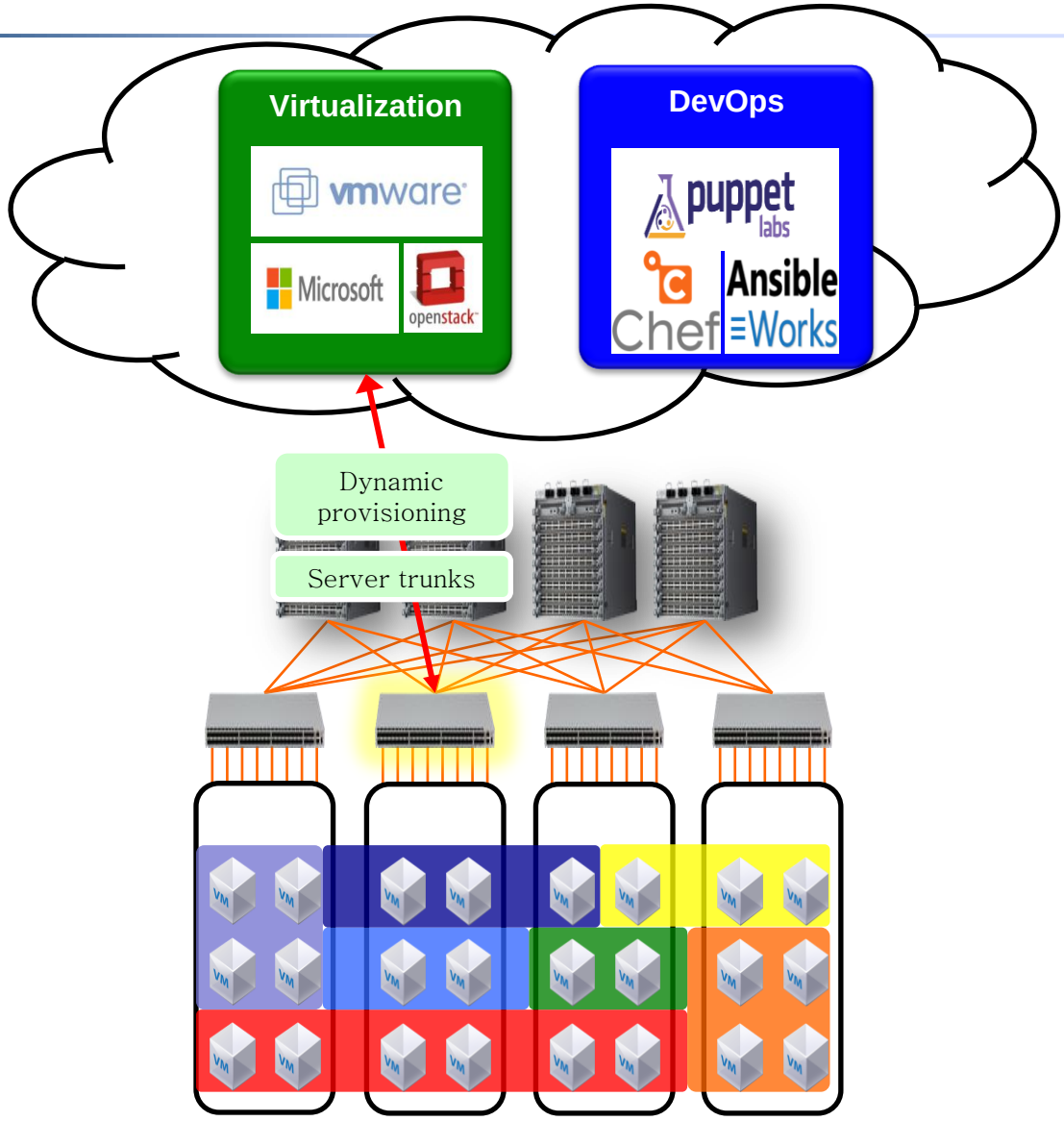
PXE boot the servers

Step 3: Provision the rack



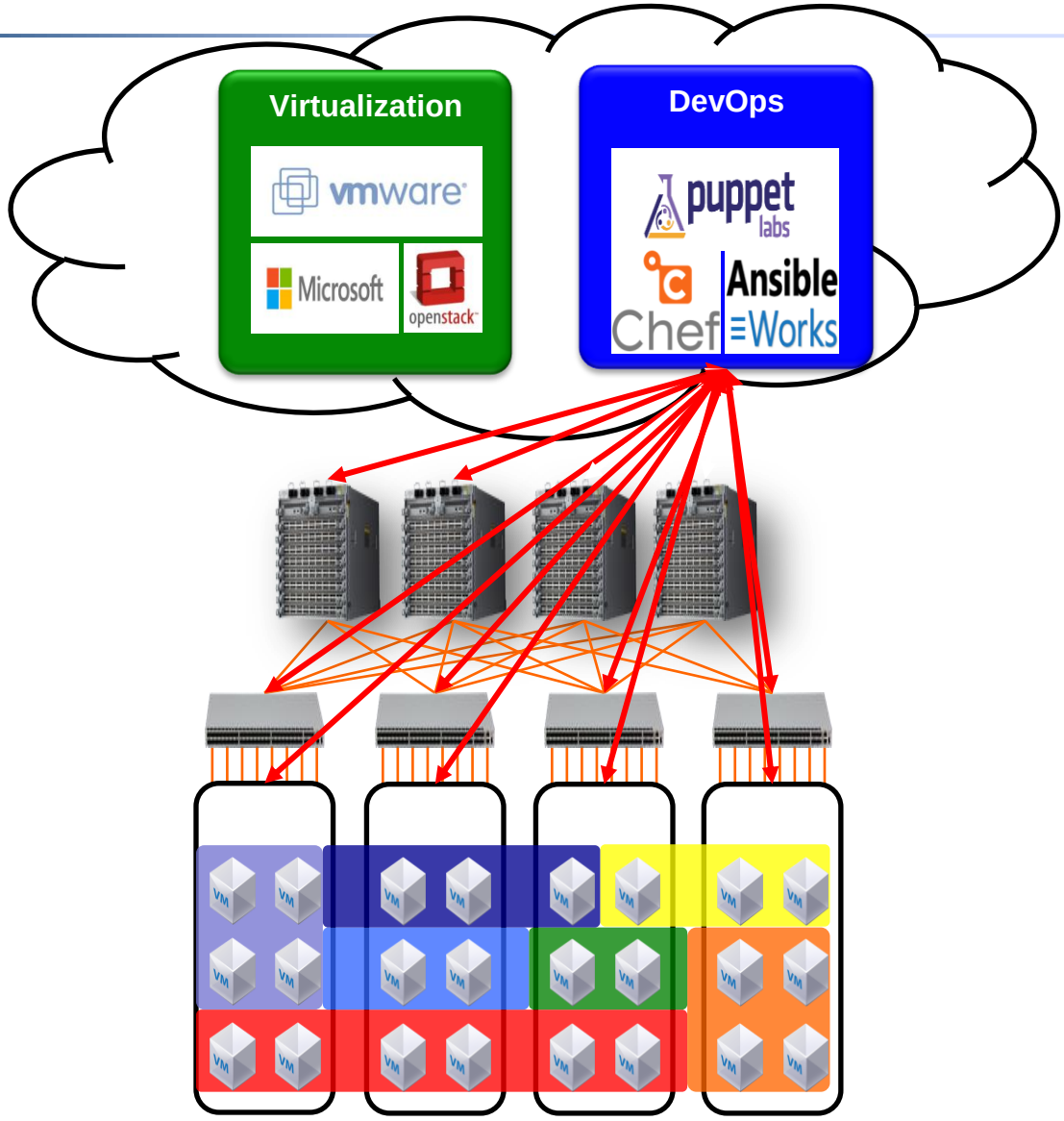
Build the VMs

Step 4: Provision the trunks



vmTracer just in time provisioning

Step 4: Compliance / Config management



Build the rack

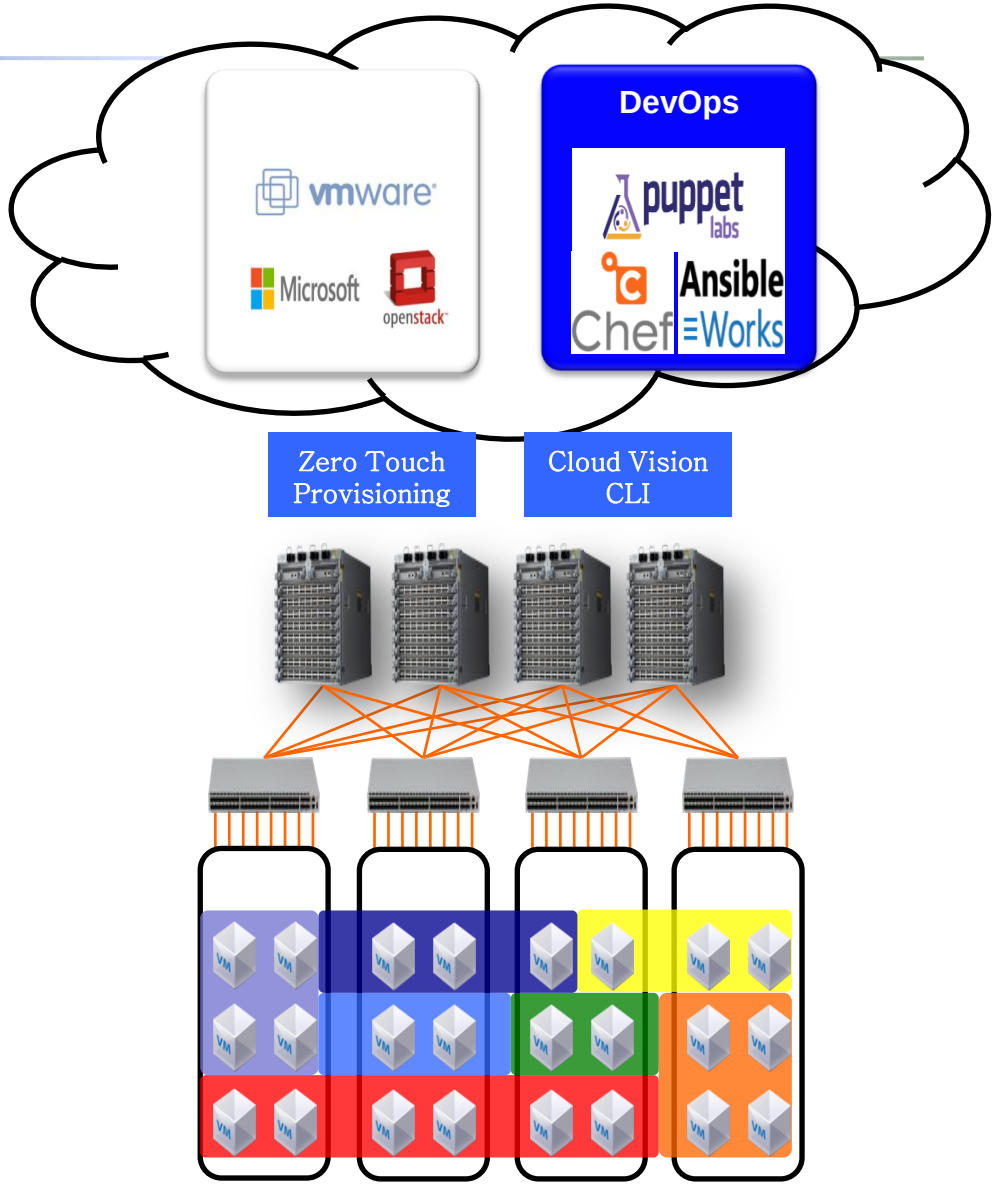
Auto Provisioning - Professional Services

SVC-EE-AUTOPROV-1M-SW

EE-AUTOPROV-SM
(Upto 10 switches)

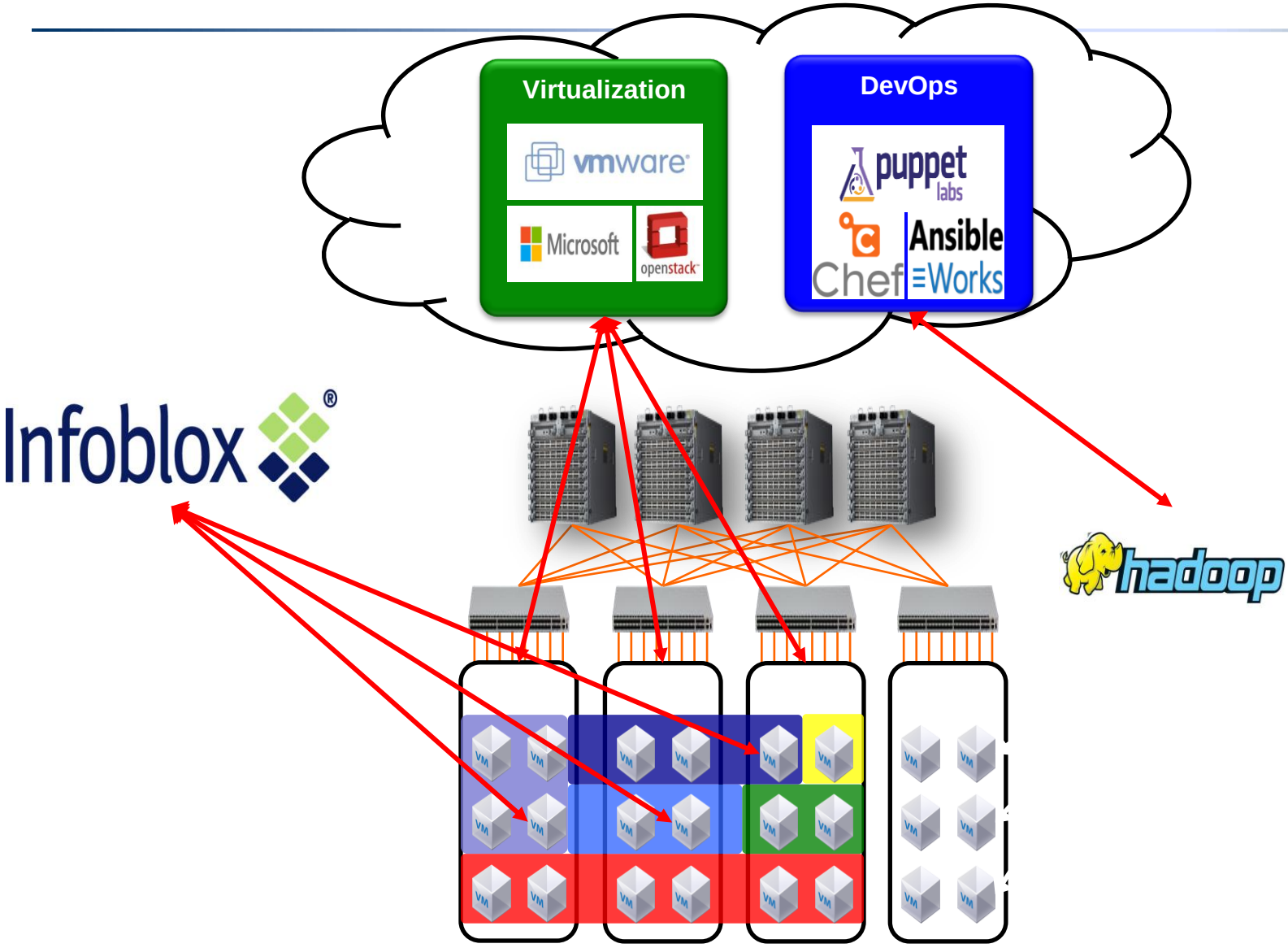
EE-AUTOPROV-MED
(11-50 switches)

EE-AUTOPROV-LG
(51-100 switches)



ZTP, CloudVision and DevOps integration

Step 5: Roll out new applications



Rapid Application Deployment

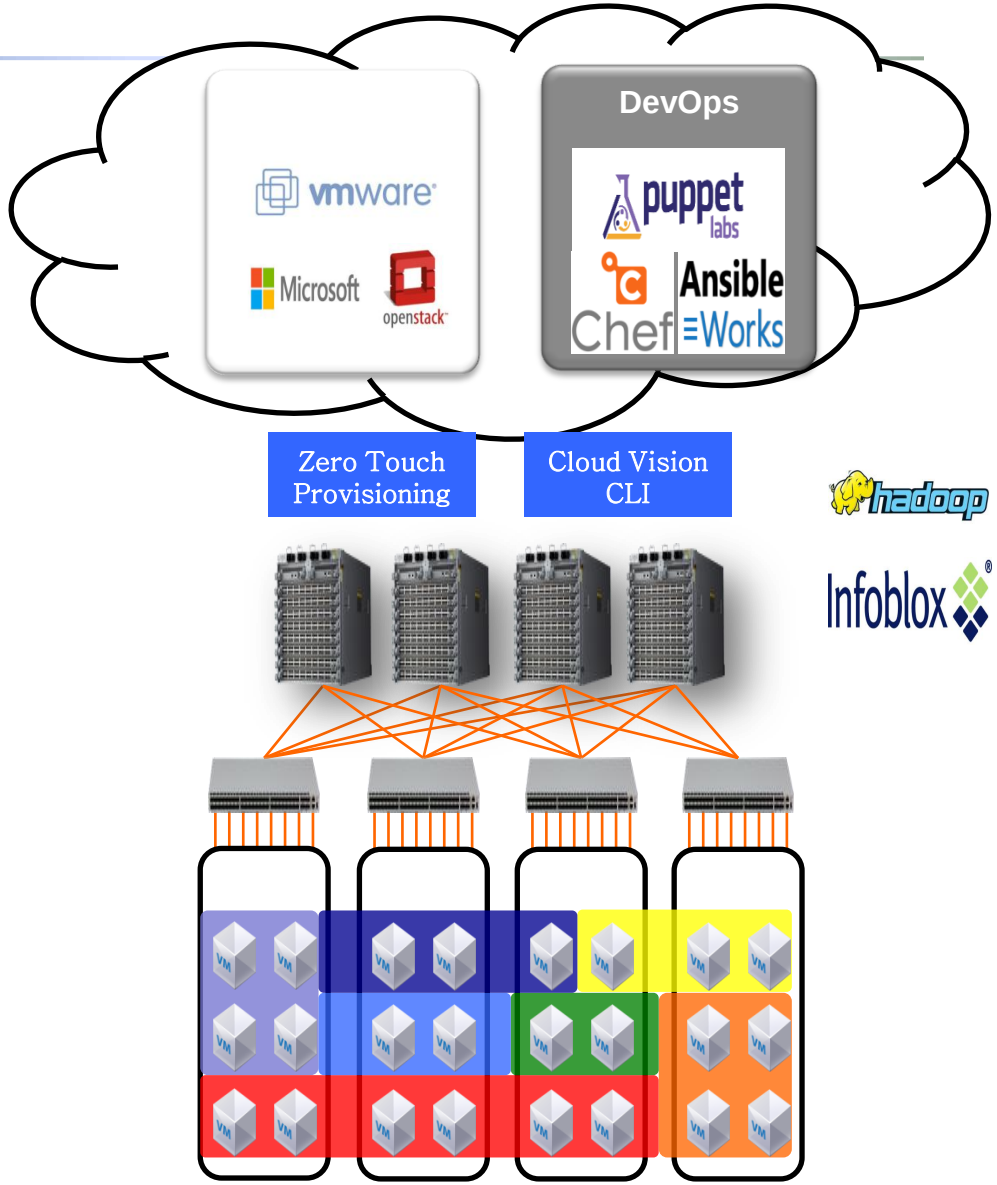
Custom – Extensibility Services

Integrate virtualization platform

Custom application deployment

Home grown back-end EMS

IP Management System



Custom statement of work

Arista is connecting the cloud

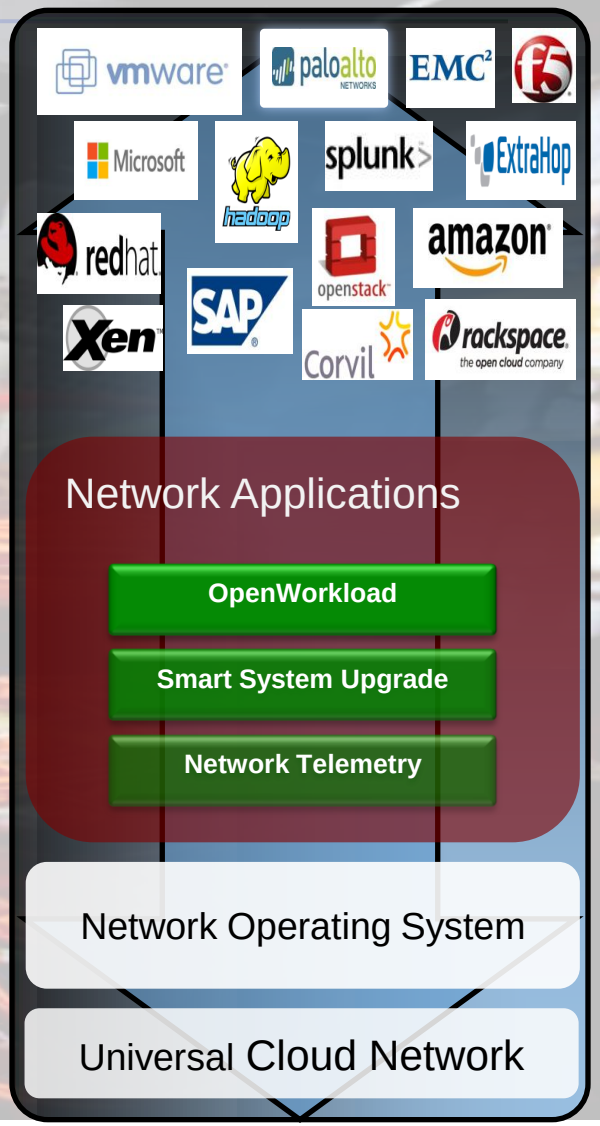
highly reliable operating system

completely programmable

unique focused feature set

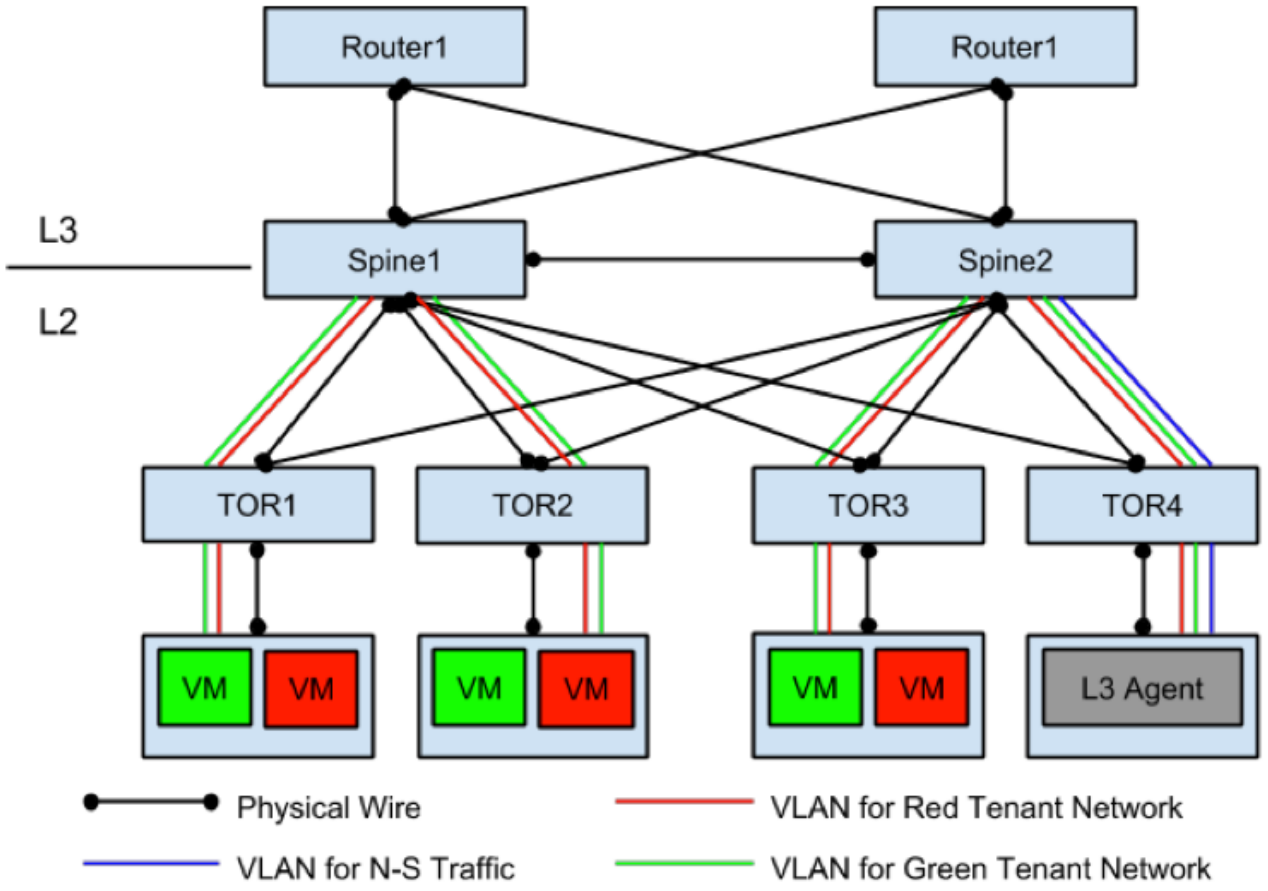
open partner ecosystem

There is a company that is innovating to address your challenges…



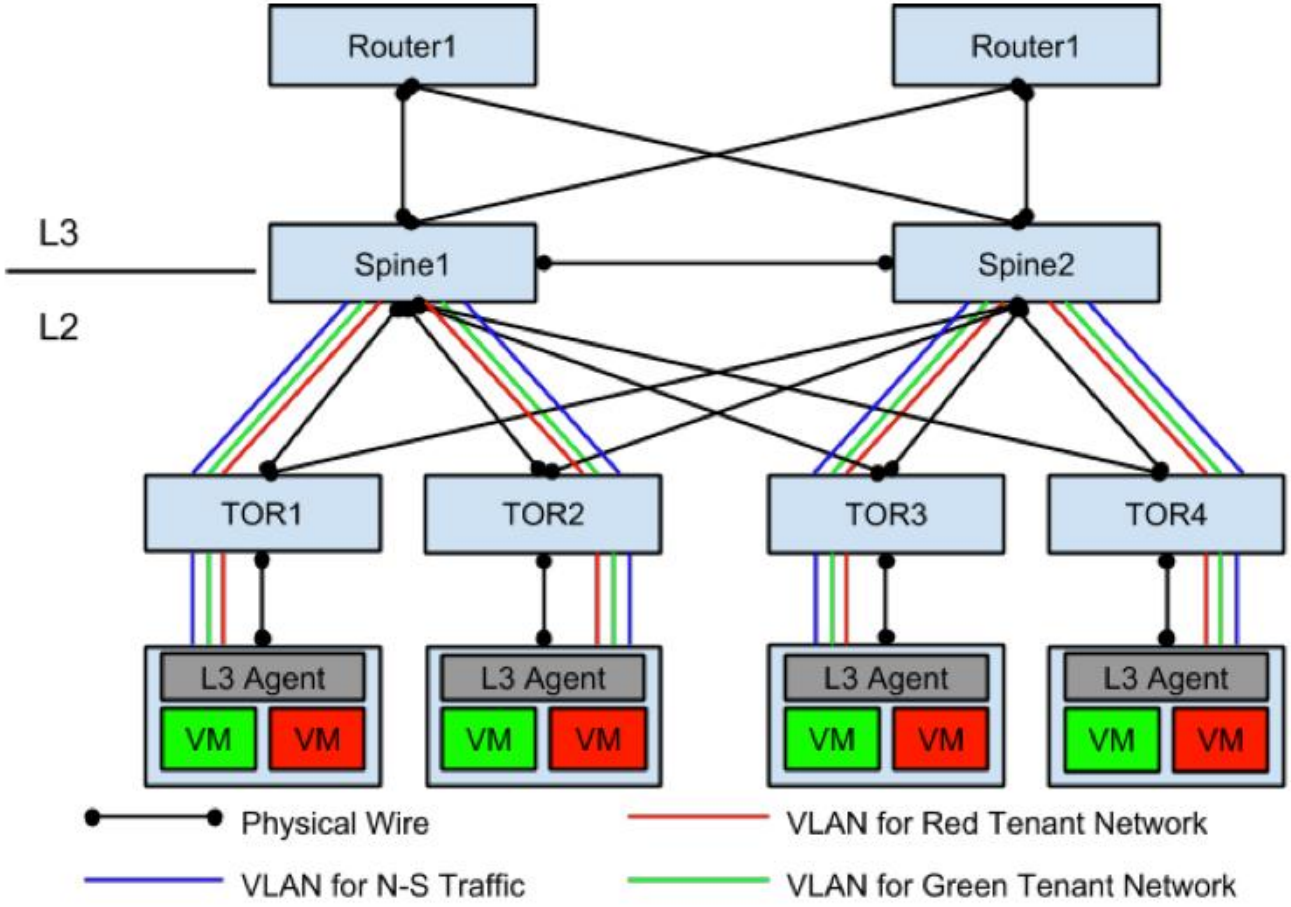

ARISTA

Network Design for an L2 Fabric using VLANs

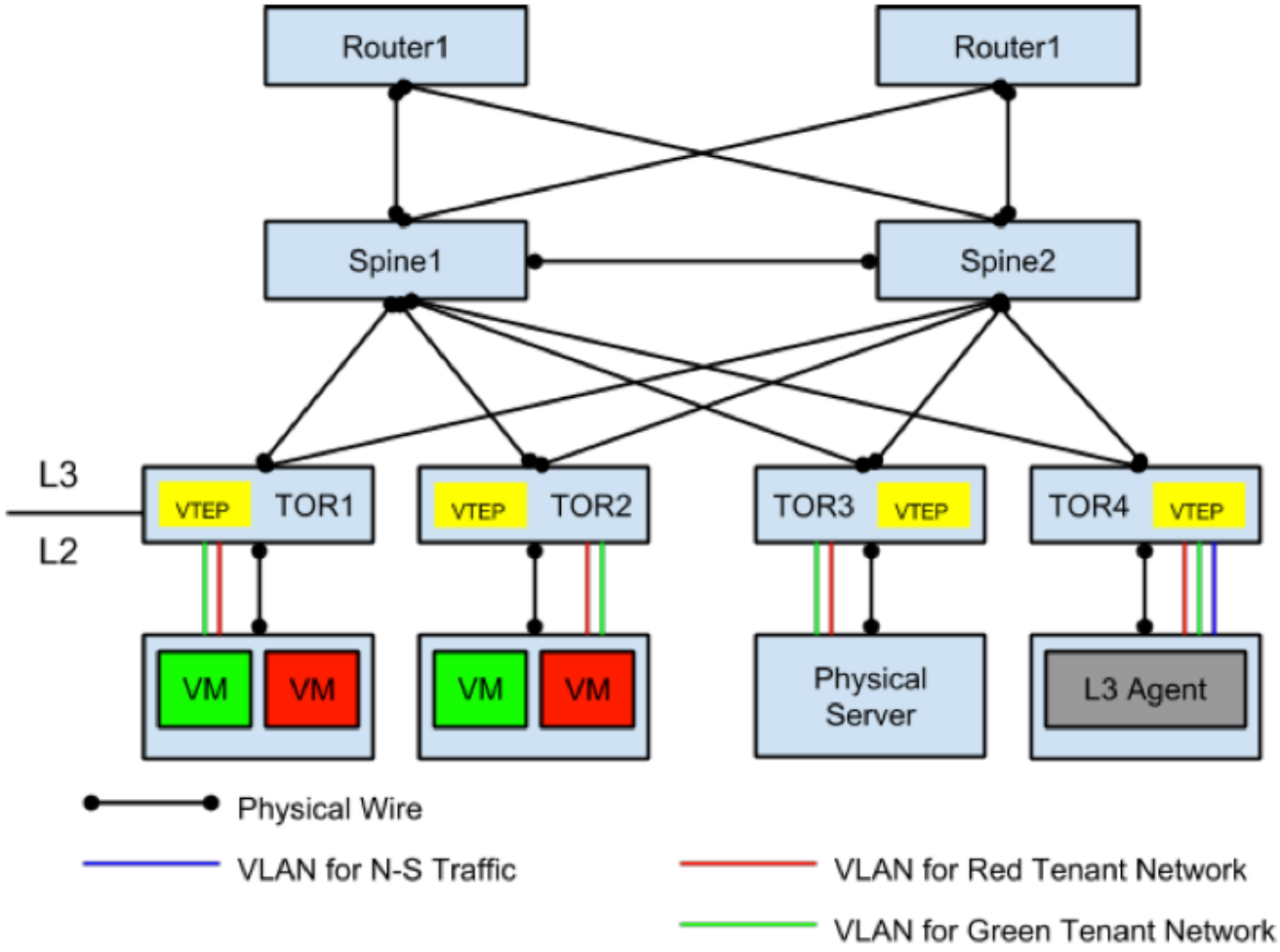


Folsome / Grizzly L2 Deployment

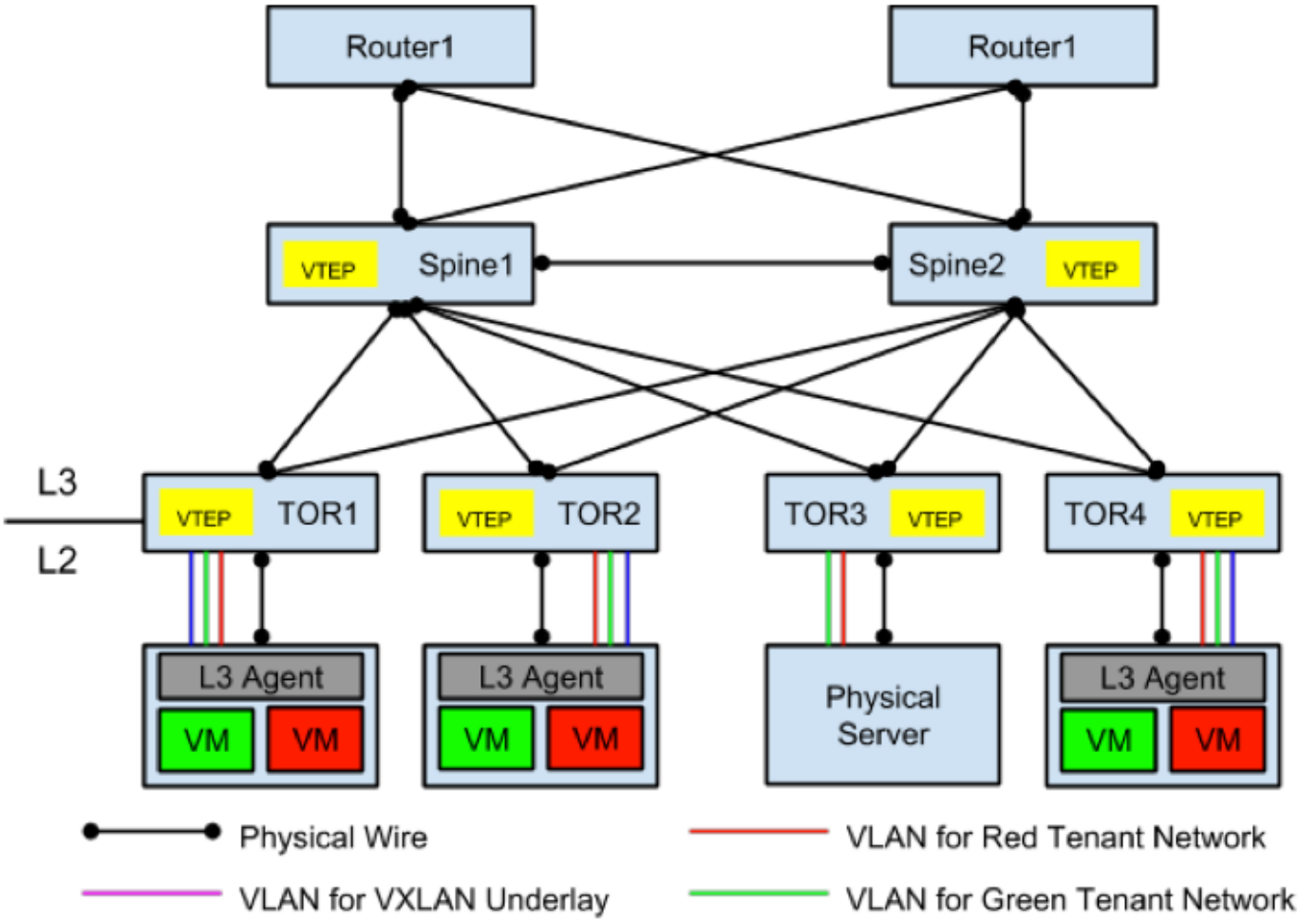
Multi-Host L2 Deployment (Havana and beyond)



Folsome / Grizzly Deployment, no hardware VXLAN routing Support



Multi-Host Deployment (Havana and beyond)



Arista OpenStack Integration Roadmap

- **Arista contributing to ML2 Plugin – Openstack Havana release**
- **Ongoing Certifications – Rackspace, RedHat, Canonical (Ubuntu)**
- **Ongoing partner integration – VMware, Cyan, PlumGrid**

June 2013	Nov 2013	1H2014*
• EFT image available • Supports Folsom and Grizzly • Physical and virtual network visibility • Auto VLAN provisioning • VXLAN fabric by statically configuring VLAN to VNI mappings on each TOR switch. Multicast must be used.	• Supports Havana release • Physical and virtual network visibility • Auto VLAN provisioning • VXLAN fabric by statically configuring VLAN to VNI mappings on each TOR switch. VXLAN fabric can be with or without multicast (by using VXLAN control service on CVX)	• Automate provisioning of VLAN to VNI mappings on each TOR switch • Support VXLAN fabric with all hardware VTEPs (up to 4K tenant networks) • Automate provisioning of routes at spine for public address space • Support for Multi-Host mode