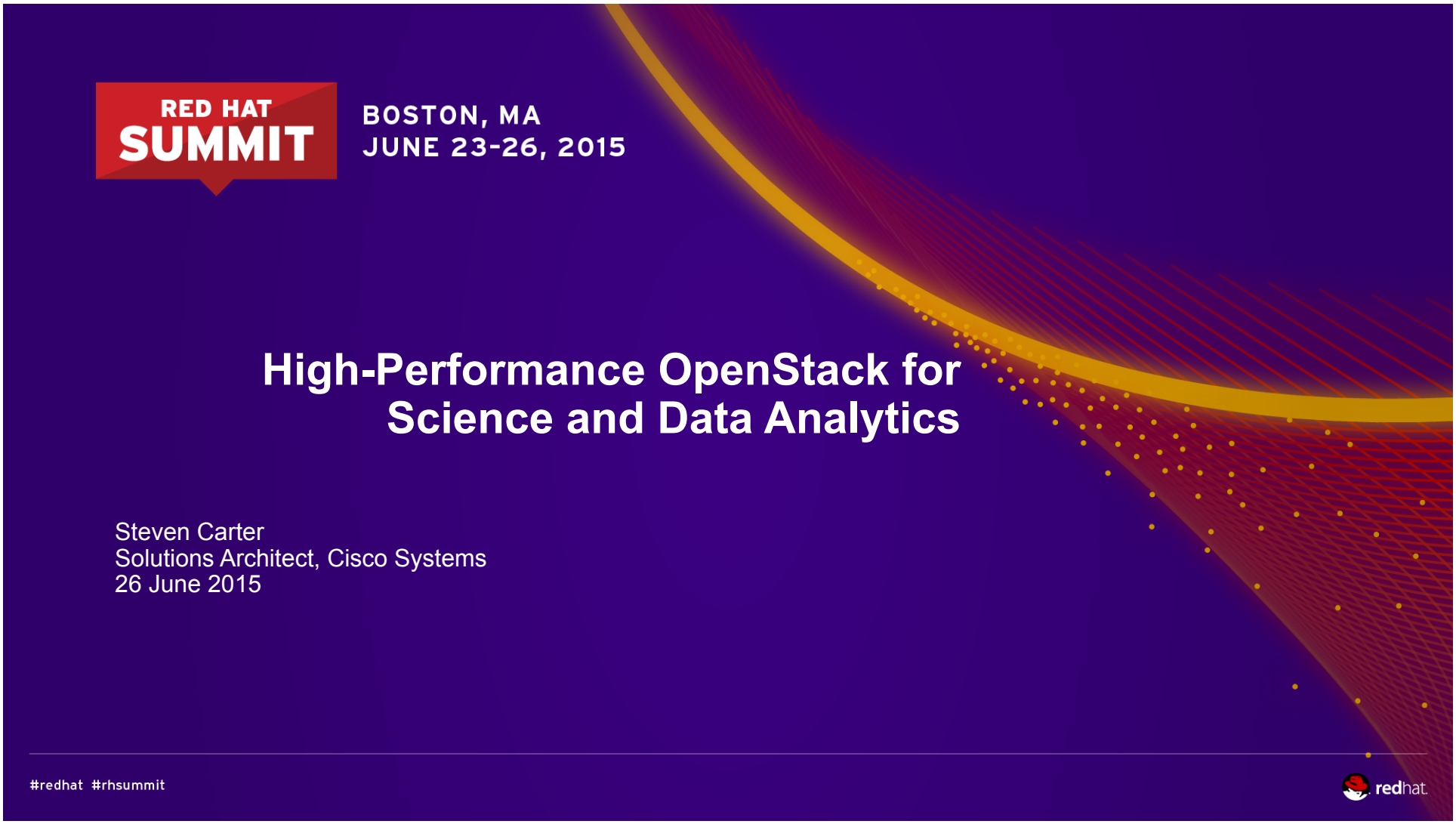

RED HAT
SUMMIT
BOSTON, MA
JUNE 23-26, 2015
High-Performance OpenStack for Science and Data Analytics
Steven Carter
Solutions Architect, Cisco Systems
26 June 2015
#redhat #rhsummit
redhat.

Agenda

- Overview
- Networking
- Storage
- Hardware
- Schedulers
- Use Cases

HPC on OpenStack?

Similarities:

- Compatible Workloads:
  - HPC workloads among the first stateless workloads
- Achieve Economies of Scope
  - Major vendors working on innovation in the space

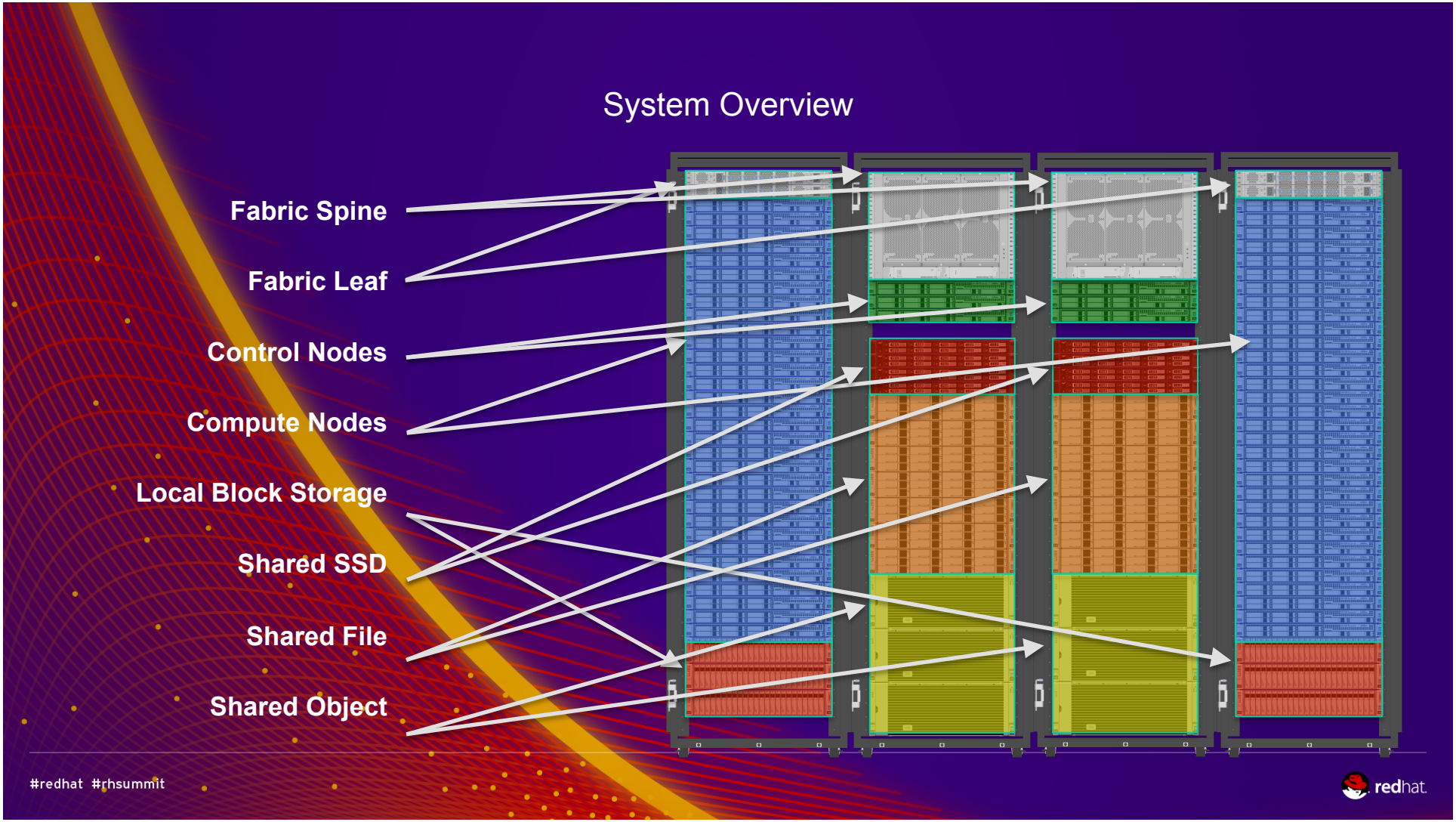
Differences:

- Science requires data!
  - To/From the WAN
  - To/From Central Storage
  - Between Instances
- Science requires more Compute/Memory
  - Less/No Oversubscription
  - GPUs

System Overview

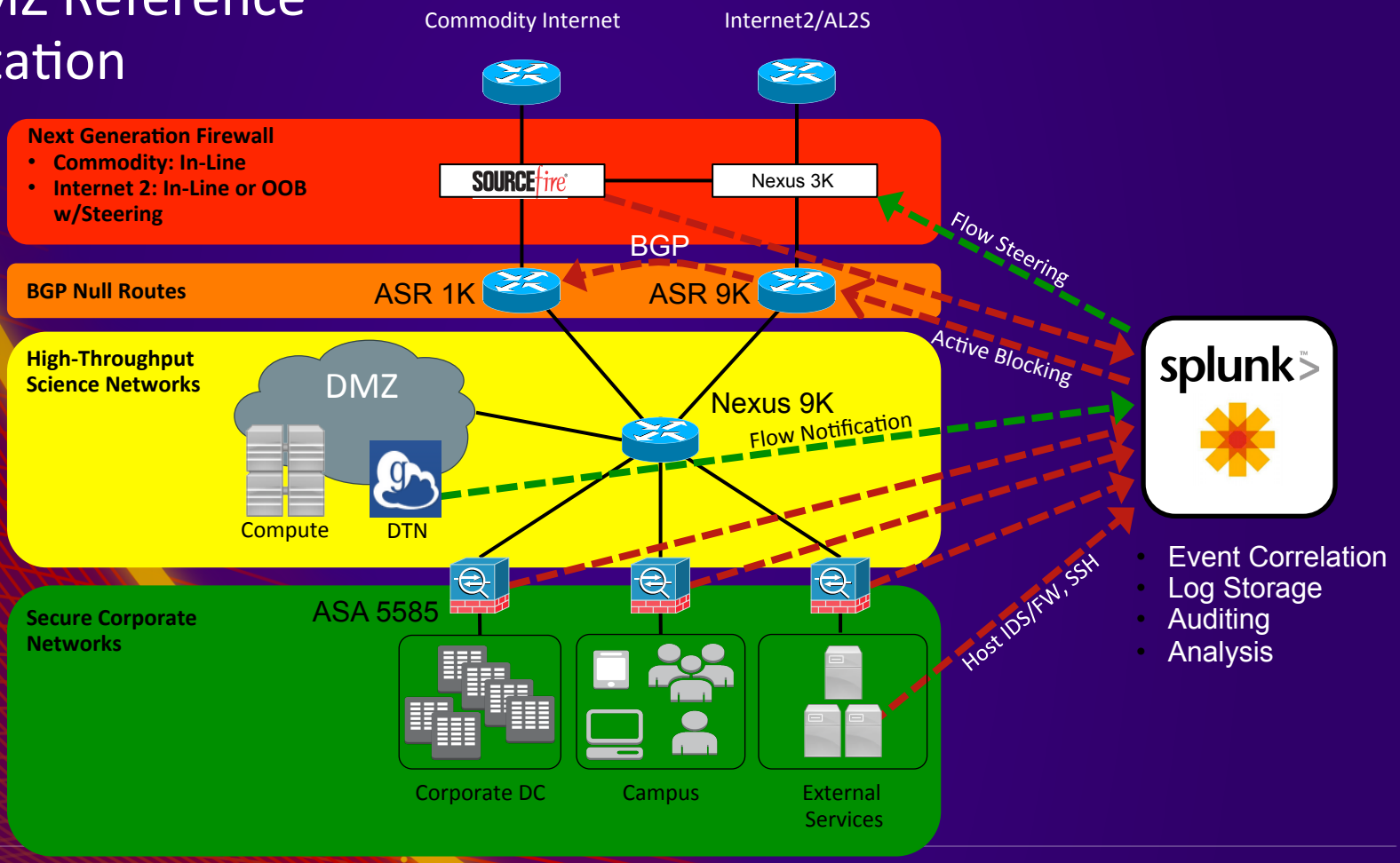
- Purpose Built
- Scalable Architecture
- Easily extensible by adding Scalable Units
- Network architected for Data Movement, Low Latency
- Hardware optimized for Cycles/Memory

System Overview



Networking: Science DMZ

Science DMZ Reference Implementation



Science DMZ Components

- Most informed view of the status of the network, servers, and apps.
- Provides event correlation by severity, rate, and between events
- Provides for auditing and reporting capabilities
- Leverage existing skill by writing logic in Splunk search language

- Provides single northbound REST API
- Application specific logic in between
- Hardware specific southbound APIs

- Software-as-a-Service (SaaS) solution to manage transfers
- GridFTP extensions provides parallelism (i.e., the use of multiple socket connections between pairs of data movers), restart markers, and data channel security.
- GridFTP control plane provides the source and destination information for the flows it sets up
- Effectively authenticates flows before they bypass security

OpenFlow Data Flow Steering

Base setup depending on mode:

Out-Of-Band IDS:

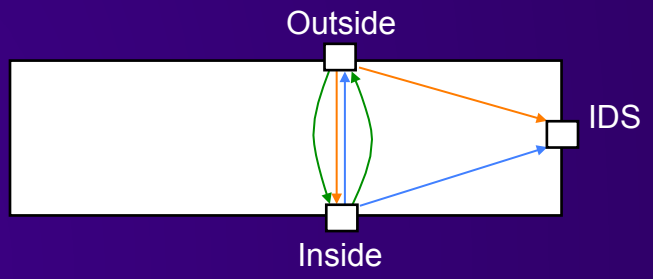
<priority>100</priority>
<in-port>54</in-port>
<output-node-connector>52</output-node-connector>
<output-node-connector>25</output-node-connector>

In-Band Firewall/IPS:

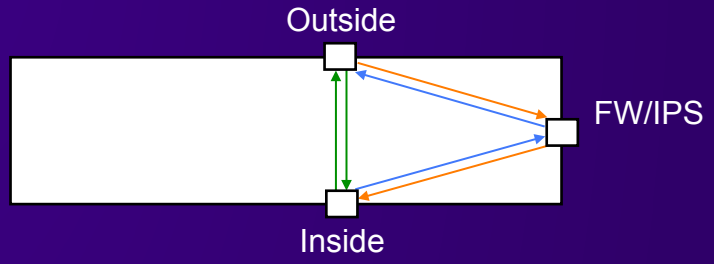
<priority>100</priority>
<in-port>54</in-port>
<output-node-connector>25</output-node-connector>
<in-port>25</in-port>
<output-node-connector>52</output-node-connector>

Bypass operation the same for both modes

<priority>200</priority>
<in-port>54</in-port>
<output-node-connector>52</output-node-connector>

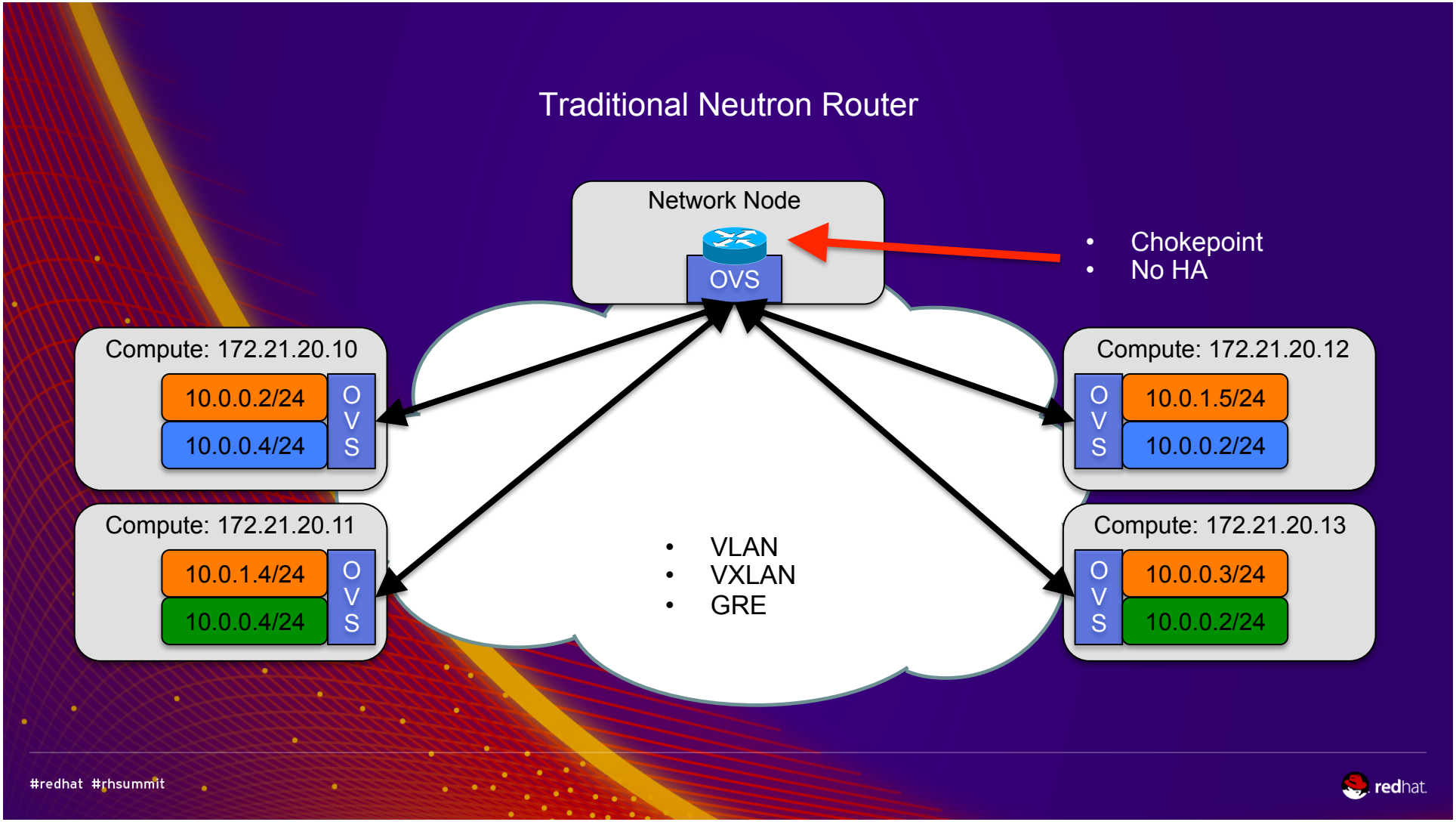


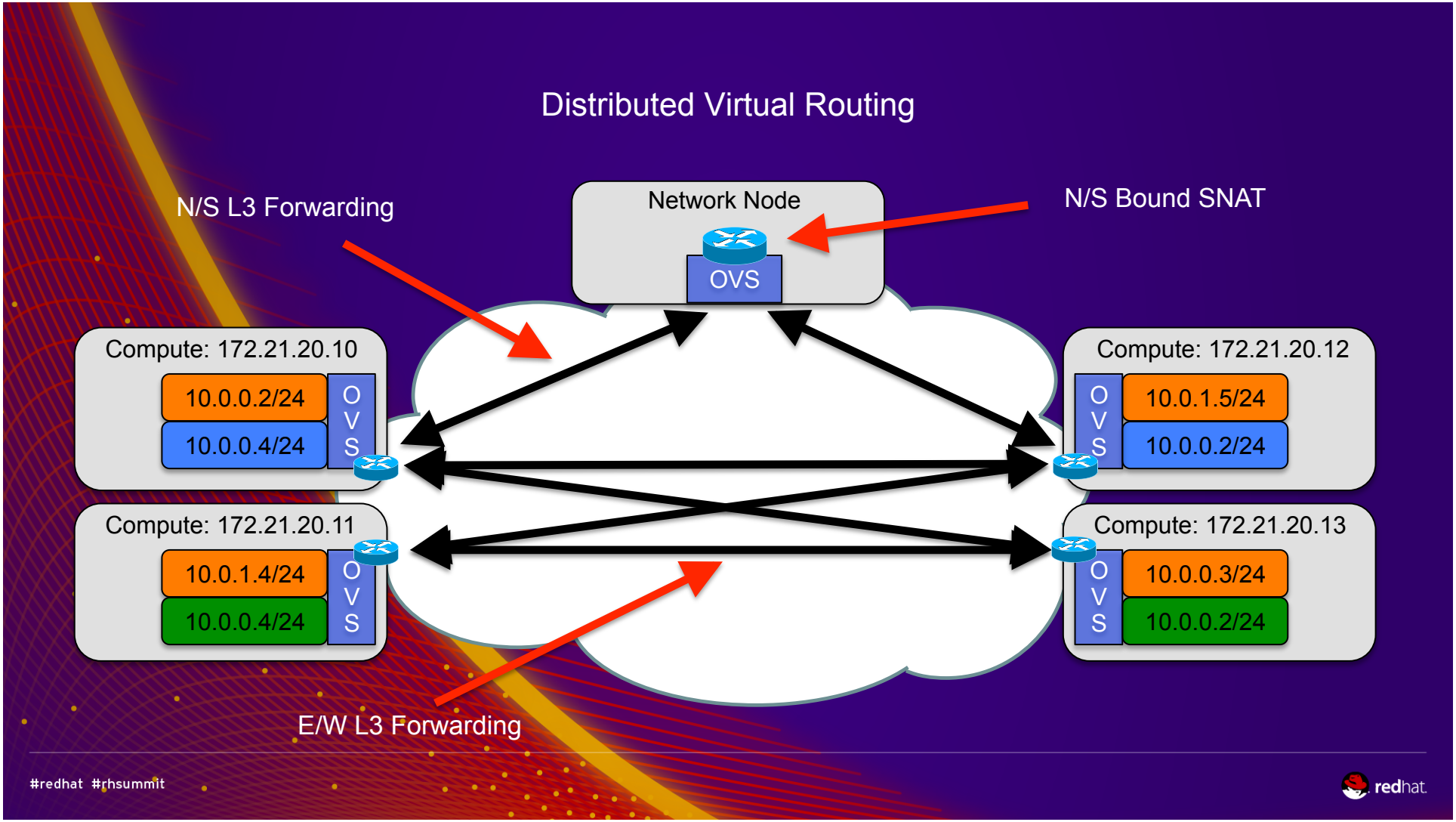
Out-Of-Band IDS



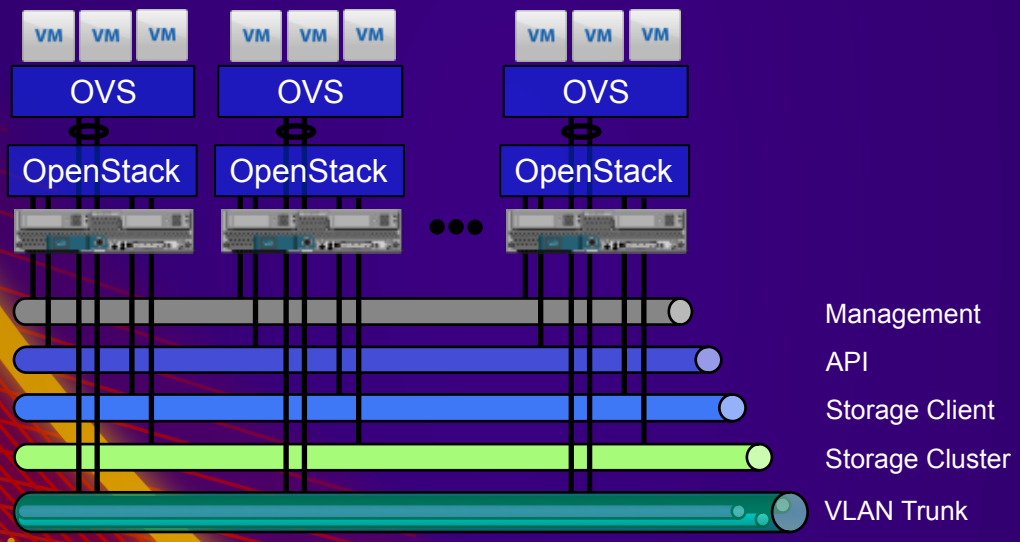
In-Band FW/IPS

Networking: Fabric


Traditional Neutron Router
Network Node
OVS
Chokepoint
No HA
Compute: 172.21.20.10
10.0.0.2/24
10.0.0.4/24
OVS
Compute: 172.21.20.12
10.0.1.5/24
10.0.0.2/24
OVS
Compute: 172.21.20.11
10.0.1.4/24
10.0.0.4/24
OVS
VLAN
VXLAN
GRE
Compute: 172.21.20.13
10.0.0.3/24
10.0.0.2/24
OVS


Distributed Virtual Routing
N/S L3 Forwarding
Network Node
OVS
N/S Bound SNAT
Compute: 172.21.20.10
10.0.0.2/24
10.0.0.4/24
OVS
Compute: 172.21.20.12
OVS
10.0.1.5/24
10.0.0.2/24
Compute: 172.21.20.11
10.0.1.4/24
10.0.0.4/24
OVS
Compute: 172.21.20.13
OVS
10.0.0.3/24
10.0.0.2/24
E/W L3 Forwarding

Neutron Provider Networks

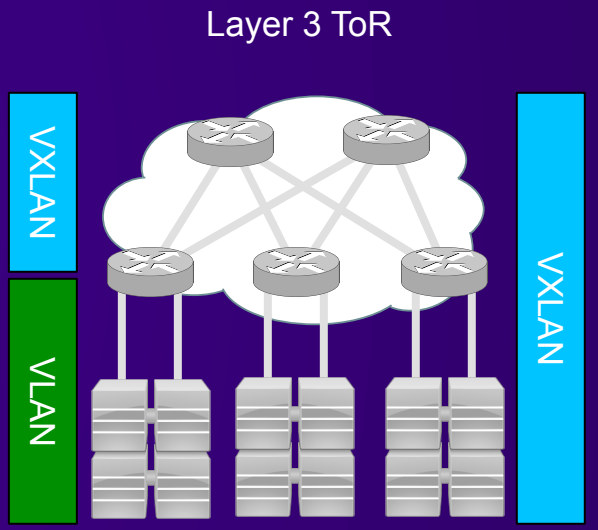


Overlays

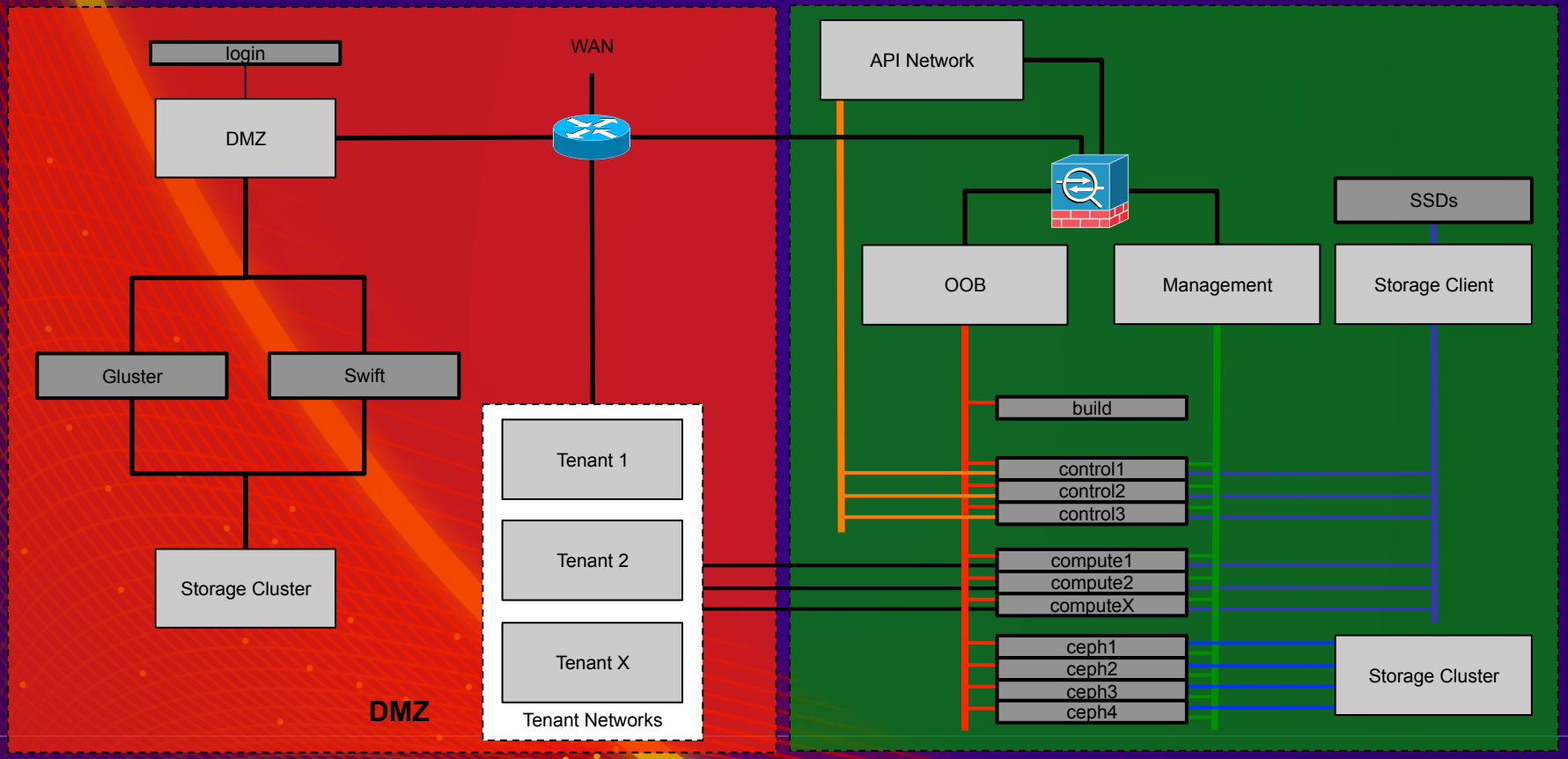
VLAN
- Performing
- Pervasive
- 12 bits of tenant networks of tenants
- Layer 2 ToR

VXLAN
- Less Performing
- Less Pervasive
- Control plane coalescing around standard EVPN
- 24 bits of tenant networks (twice a much, right!)
- Rapidly evolving
- Layer 3 ToR

GRE
- Blech



Network Overview



Storage

Storage Access from Instance

Local to Node:

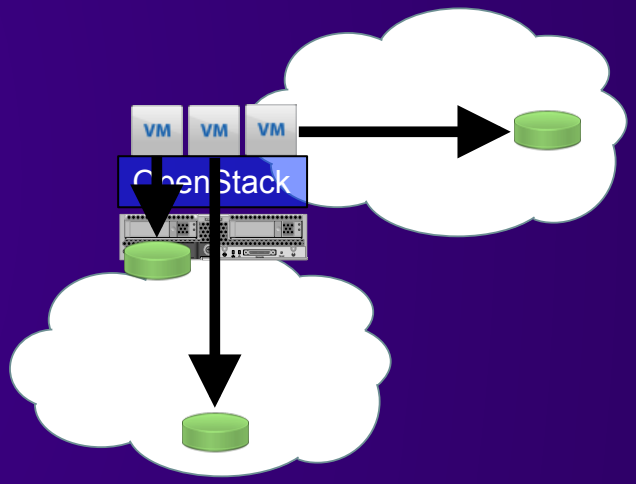
- Local Disks

Remote through OpenStack:

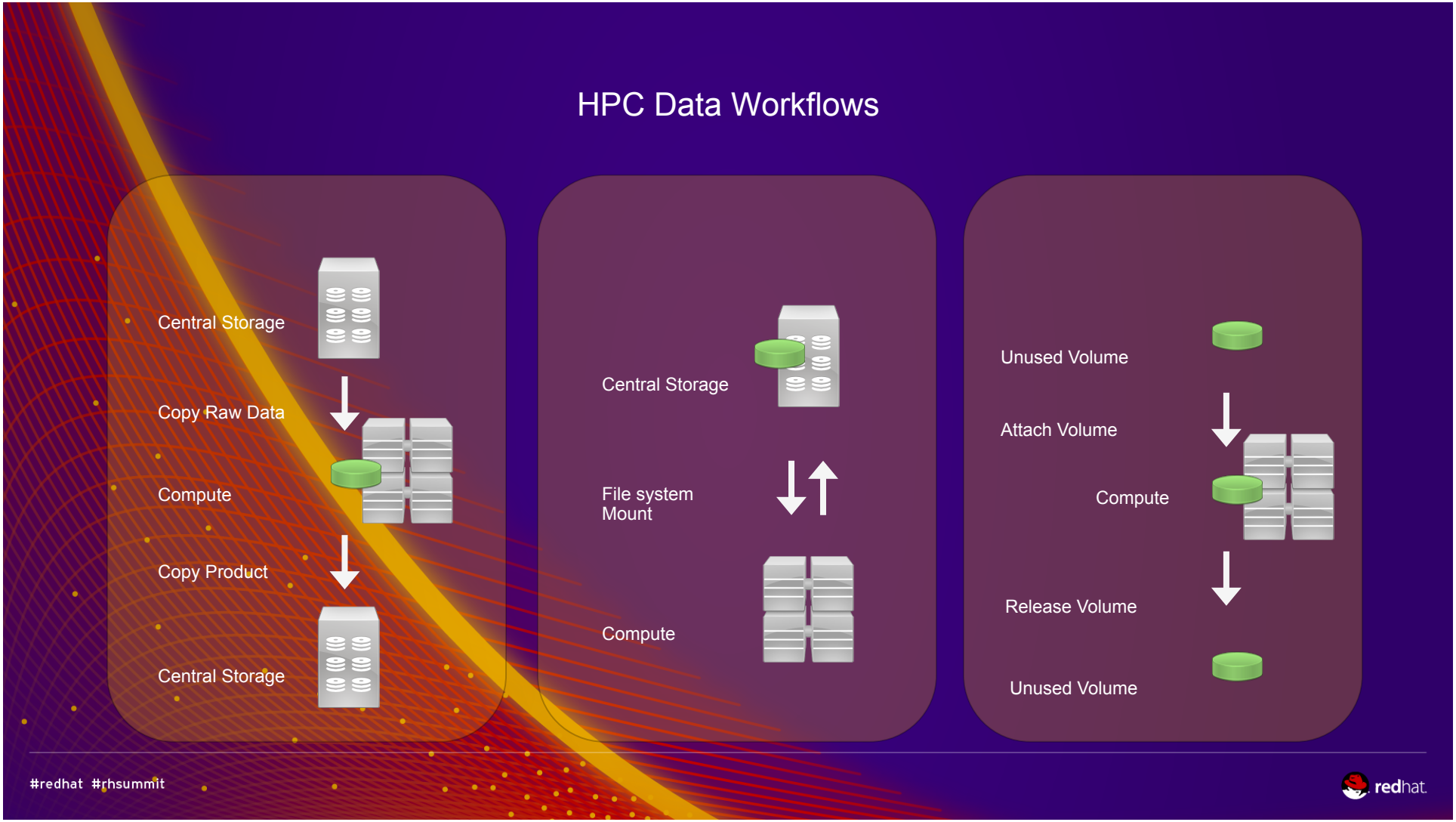
- Cinder
- Manilla

Remote, direct from Instance:

- Swift
- NFS
- Gluster
- Lustre



HPC Data Workflows



Local Temporal Storage

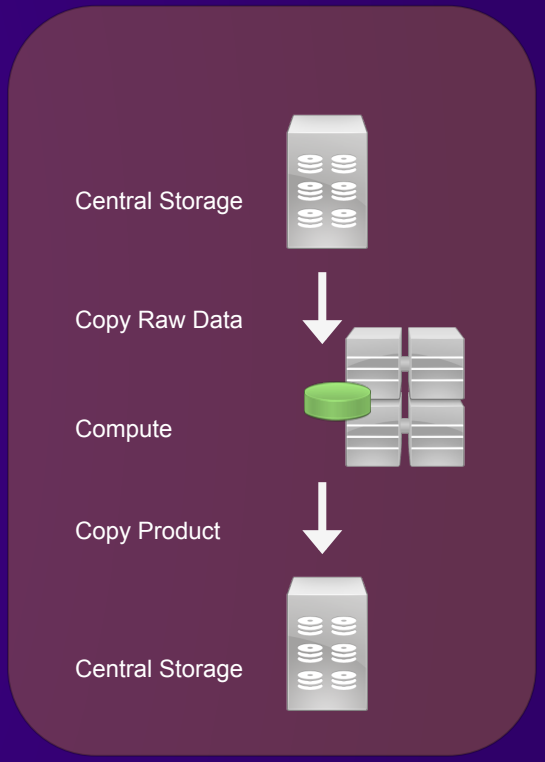
Central Storage: Swift, Possibly File
Instance Storage: Local, Cinder

Pros:

- Favors Object Storage – No mounting needed in instance
- Most Portable
- Most Conducive to SSD Usage
- Easiest on Shared Storage

Cons:

- Requires movement on/off compute nodes

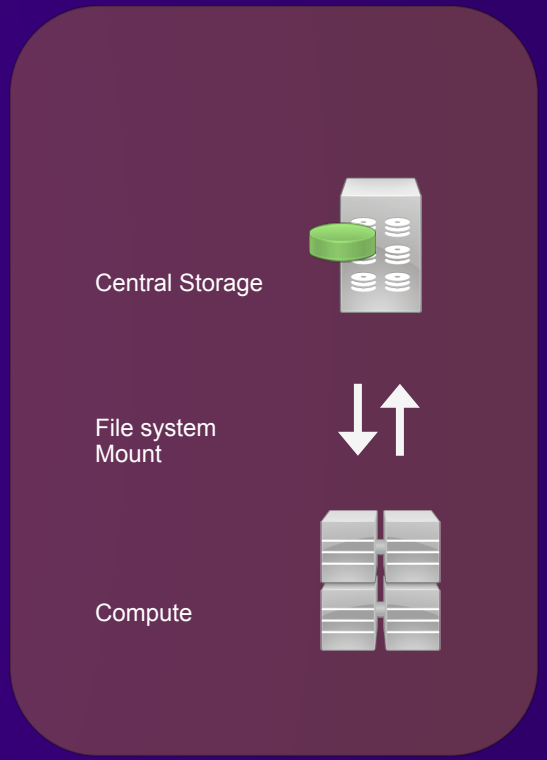


Central Persistent Storage

Central Storage: File (NFS, Lustre, Gluster, GPFS, etc)
Instance Storage: Minimal

Pros:
- No copies

Cons:
- Hardest on Shared Storage
- Requires good network access for tenants
- Need mount on Instance



Persistent Volume

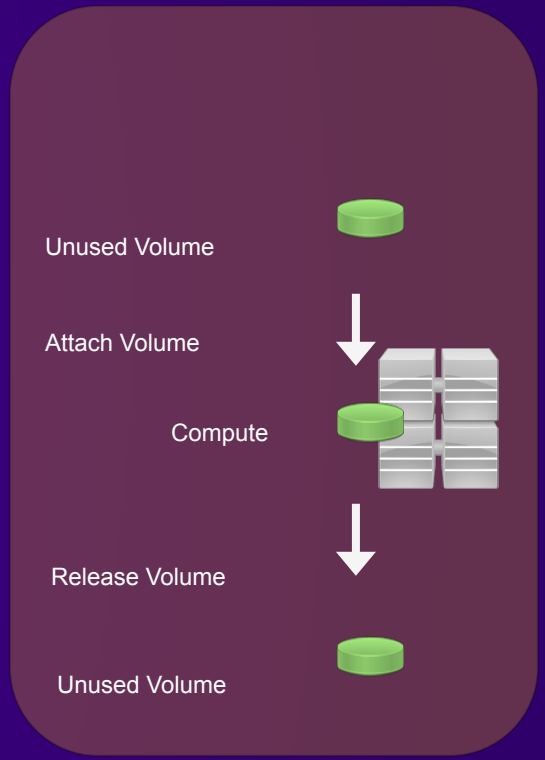
Central Storage: Cinder (Ceph, NFS, NetApp, etc.)
Instance Storage: Minimal

Pros:

- Generally Filesystem
- No copies

Cons:

- Limited ccessability to data (single node access to volumes)



Hardware

Platforms

B200
- 2 sockets, mid-memory, limited storage, dense footprint

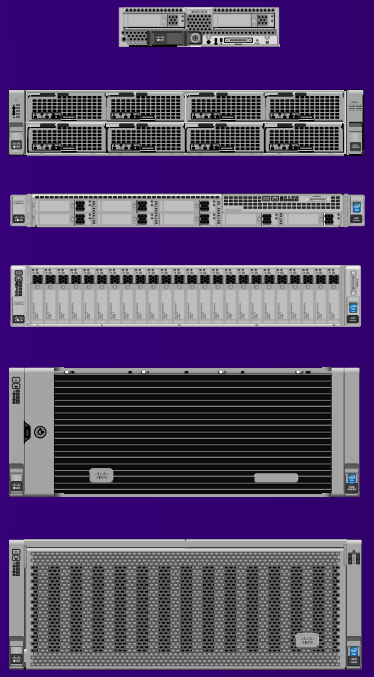
M4308 Cartridges
- 1 socket, mid-memory, shared storage, dense footprint

C220
- 2 sockets, mid-memory, limited storage

C240
- 2 sockets, mid- to large-memory, storage, expansion

C460
- 4 sockets, large memory, storage, large expansion

C3160
- 2 sockets, mid-memory, large storage

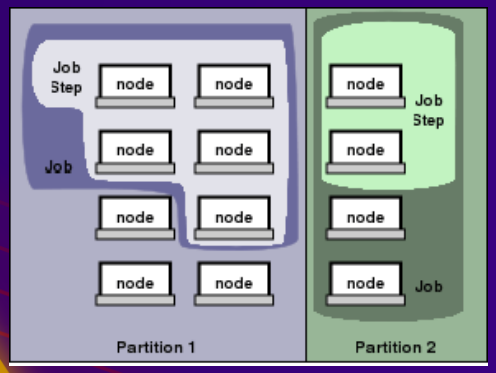


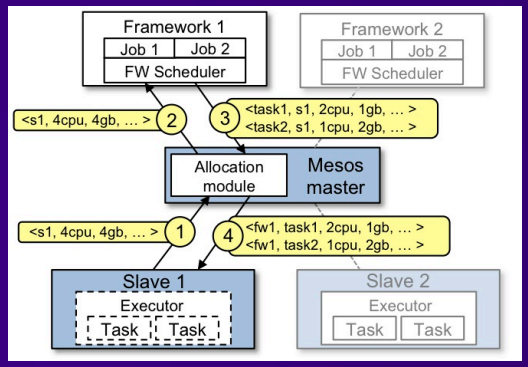
Containers

- More about packaging then security
- Great for portability
- Compelling when running directly on hardware (e.g. Iron)
- Gives better access to hardware for things like GPU and low-latency interconnect
- Need to consider noisy neighbor problems
- Works will with cartridges because it is a 1:1 mapping from container to physical

Scheduling

Traditional HPC





- Traditional batch scheduler
- Curse grained scheduling
- Allocates entire node to job
- Virtual nodes added to queue
- Optimized for job completion time

- Cloud oriented “Distributed Systems Kernel”
- Fine grained scheduling
- Abstracts CPU, memory, storage, and other resources
- Optimized for throughput and utilization

Extension to Public Clouds

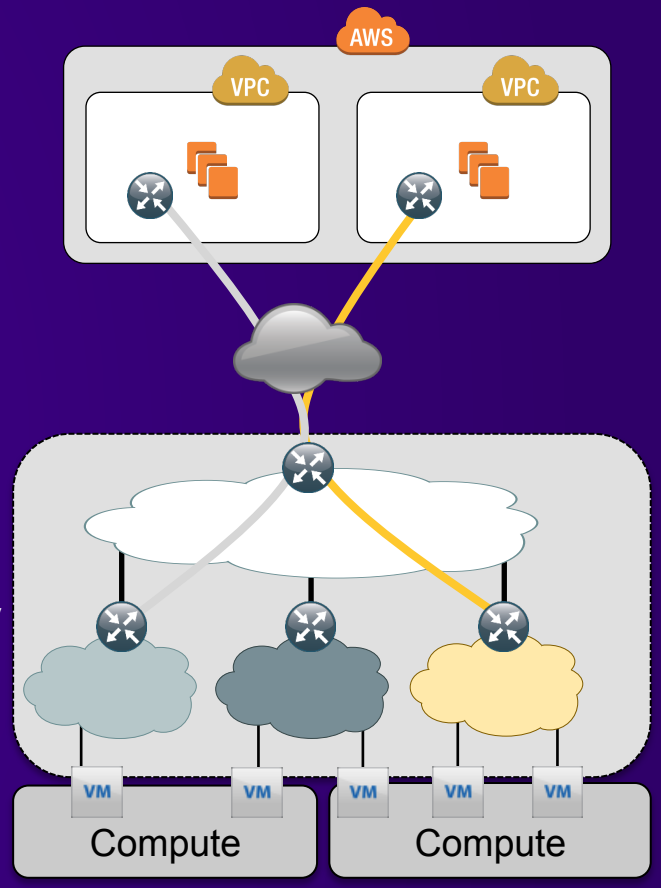
Extension to Public Clouds

Scheduler Based

- Workers can be distributed across clouds
- Need access to scheduler
- Need access to storage

Network Extension

- VPN
- Direct Connect



Use Cases

Use Cases

Application: Seismic Data Analysis

Infrastructure:

- Network: Provider VLANs
- Compute: Rack Servers
- Storage: Object w/local scratch

Scheduling:

- Celery/Mesos

“Burst” to AWS

Application: Genomics

Infrastructure:

- Network: Provider VLANs
- Compute: Blades
- Storage: NFS

Scheduling:

- Grid Engine

“Burst” to AWS


RED HAT
SUMMIT
LEARN. NETWORK.
EXPERIENCE OPEN SOURCE.
#redhat #rhsummit
redhat.