What You Make Possible
NX-OS Routing and Layer 3 Switching

Arkadiy Shapiro
Technical Marketing Engineer
NX-OS and Nexus 7000
arshapir@cisco.com
Session Goals

At the end of the session, the participants should:

- Understand NX-OS Layer 3 **software architecture** and features
- Understand **configuration highlights and best practices** associated with NX-OS Layer 3 IP routing features
- Understand how NX-OS and Nexus switches can be used to build **highly available** Layer 3 networks
Session Anti-goals

This session does not include:

- Deep dive on any Nexus switch hardware or NX-OS software architectures
- Technical discussion on Layer 3 protocols, technologies and associated merits
- Detailed roadmap discussion for Layer 3 features on NX-OS
- Detailed configuration and troubleshooting information for NX-OS Layer 3 features
For purposes of this presentation:

- All features are mentioned with assumption of latest NX-OS software release and all licenses installed
- NX-OS – IOS comparisons refer to most commonly deployed configuration on Catalyst 6500 which may not be the latest IOS version for that platform
- “Graceful Restart” (“GR”) and “Non-Stop Forwarding” (“NSF”) are complementary technologies (not the same) to achieve protocol high availability in NX-OS and Nexus 3000/5000/7000
- “NSR”, “Non-stop routing” and “stateful restart” will refer to the same technology and behavior in NX-OS
Agenda

- Layer 3 Requirements
- NX-OS Layer 3 Software Architecture
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
Agenda

- **Layer 3 Requirements**
  - Network Design Requirements
  - System Requirements
- **NX-OS Layer 3 Software Architecture**
- **Unicast Routing Protocols**
- **Multicast Protocols**
- **First Hop Redundancy Protocols**
- **Routing Policy and Policy-based Routing (PBR)**
- **Layer 3 High Availability and Fast Convergence**
- **Summary**
Typical Enterprise Network Deployment

Layer 3 Requirements
Layer 3 Features by Place in the Network

### Enterprise Core
- **L3:** Routing Table Scale or Lean Core, BGP, IGP, BFD
- MPLS P/PE

### DC Core
- **L3:** Routing Table, BGP, IGP, BFD, Dual Stack, IPv6/v4 translation, ECMP
- MPLS P/PE

### DC Aggregation
- **L2 / L3 Boundary:** FHRP, ARP / ND / IGMP / MLD Handling, DHCP relay
- **L3:** OSPF, IS-IS, EIGRP, BGP, VRF-lite, MPLS PE, PIM, Dual stack, summarization, BFD
- DCI: OTV, LISP

### DC Access
- **Routed Access Layer:**
  - Basic Layer 3, OSPF Stub, EIGRP Stub, Static / Default Routes, Dual Stack
Layer 3 Access Considerations

- Advantages:
  - Lower L2 table utilization
  - Smaller L2 domain → eliminate STP

- Drawbacks:
  - More L3 configuration points
  - VM mobility constrained to a smaller L2 domain

- Use OSPF stub area, EIGRP stub or static default routes

- Fabric Extender technology enables larger L2 domains without adding STP hops
Layer 3 Aggregation Considerations

What’s really happening behind the curtain

Layer 3 Requirements

I’m the active default gateway (FHRP active)

I’m still here... (vPC state sync., FHRP hello, Routing Protocol hello, BFD...)

I’m still here too... (vPC state sync., FHRP hello, Routing Protocol hello, BFD...)

What’s my IP address? (DHCP request)

Who is my default gateway? (ARP request)

Remote MAC Updates (DCI)

Process multicast state (Multicast DR)

Aggregation

Access

Relay DHCP query
Listen to DHCP reply (DHCP request)

L3 links northbound

L2 links southbound

Network Stress

11
Control Plane at Layer 3 Aggregation

Frequent bursty requests

<table>
<thead>
<tr>
<th>Feature</th>
<th>Function</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spanning Tree BPDU Processing</td>
<td>STP BA, Network Port</td>
<td>Every 2 seconds/VLAN or MST</td>
</tr>
<tr>
<td>First Hop Redundancy</td>
<td>HSRP, VRRP, GLBP</td>
<td>Every 3 seconds/group</td>
</tr>
<tr>
<td>Address Resolution</td>
<td>ARP / ND, FIB Glean</td>
<td>Up to 1 request per sec per host</td>
</tr>
<tr>
<td>DHCP</td>
<td>DHCP Request, Relay</td>
<td>Up to 1 request per sec per host</td>
</tr>
</tbody>
</table>

- Aggregation L2 boundary device must maintain significant state
- Control Plane Policing (CoPP) and HW rate limiters essential to protect CPU
Aggregation Table Requirements
Layer 2 and Layer 3

- **MAC Table** - L2 MAC Addresses:
  - Every L2 switch needs to learn about every MAC address in all configured VLANs
  - Aggregation layer typically has visibility to all VLANs thereby requiring MAC address scalability

- **ARP / ND (Host Route) table** - Aggregation switch needs to have an entry for every host (physical or virtual)

- **Route table (RIB / FIB)** – Layer 3 Routes (v4, v6, Multicast), direct and learned

---

### Route table (RIB/FIB)

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPA</td>
<td>IP E</td>
</tr>
<tr>
<td>IP B</td>
<td>IP F</td>
</tr>
<tr>
<td>IP C</td>
<td>IP G</td>
</tr>
<tr>
<td>IP D</td>
<td>IP H</td>
</tr>
</tbody>
</table>

### ARP / ND (Host Route) Table

<table>
<thead>
<tr>
<th>IP Address</th>
<th>MAC Address</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPA</td>
<td>MAC A</td>
<td>VLAN100</td>
</tr>
<tr>
<td>IPB</td>
<td>MAC B</td>
<td>VLAN101</td>
</tr>
<tr>
<td>IPC</td>
<td>MAC C</td>
<td>VLAN102</td>
</tr>
<tr>
<td>IPD</td>
<td>MAC D</td>
<td>VLAN 100</td>
</tr>
</tbody>
</table>
Layer 3 Core

- Network backbone
- Allows growth in aggregation PODs
- 100% Layer 3
- Use routed links instead of SVIs
- MPLS P / PE functionality
- Fast Convergence
- ISP multi-homing
- Routing table scale
Data Center Scalable Fabric with Layer 3

What if traditional multi-tier network does not work for me?

- Options for massively scalable data center fabric:
  - Fabric Path
  - Pure Layer 3

- Why Layer 3 fabric?
  - Unified Control Plane (BGP)
  - Limit broadcast domains
  - BigData, High Performance and Grid Computing
  - Inverse virtualization / DAS

- BGP is becoming protocol of choice
  - Better multi-pathing (Add-Path, AS-Path Multipath Relax)
  - Relax loop avoidance mechanisms (Local AS, AS Override)
  - In some cases, no IGP at all!
Agenda

- **Layer 3 Requirements**
  - Network Design Requirements
  - System Requirements
- **NX-OS Layer 3 Software Architecture**
- **Unicast Routing Protocols**
- **Multicast Protocols**
- **First Hop Redundancy Protocols**
- **Routing Policy and Policy-based Routing (PBR)**
- **Layer 3 High Availability and Fast Convergence**
- **Summary**
Nexus 7000 Layer 3 / MPLS Support

Hardware and Software

- Layer 3 support on all F2 and M-series I/O modules
- MPLS support on all M-series I/O modules
- F1 modules support Layer 3 and MPLS via proxy-mode
- Layer 3 support: NX-OS 4.0(1) onwards
- MPLS support: NX-OS 5.2(1) onwards
Nexus 7000 Layer 3 / MPLS Support

Fabric Extenders

- Layer 3 port mode on FEX (default) from NX-OS 5.2(1)
- FEX port options:
  - L2 interface / port-channel terminated by SVI
  - L3 interface / sub-interface / port-channel
- Connected routes only in both cases (no routing adjacency over FEX)
Nexus 7000 Layer 3 / MPLS Licensing

- **Enterprise Services License** - all routing protocols except RIP
- **MPLS License** – MPLS features
- **XL License** – higher FIB table sizes (optional)
- **Base** - all other Layer 3 features (SVIs, Layer 3 ports, FHRP, IGMP etc)
Nexus 5500 Layer 3 Support

Hardware and Software

- Layer 3 support: NX-OS 5.0(3) N1 onwards
- Routed mode for all ports (exc. FEX ports)
- SVIs for all switchports (inc. FEX ports)
- 160 Gbps throughput per module / daughter-card

**N55-D160L3 and N55-D160L3-V2**
- Single daughter-card
- Nexus 5548P / 5548UP only
- In-rack upgradeable for deployed units

**N55-M160L3 and N55-M160L3-V2**
- Expansion module
- Nexus 5596UP only
- No front-panel ports
Nexus 5500 Layer 3 support

Ver 1: N55-D160L3 and N55-M160L3

<table>
<thead>
<tr>
<th>Capability</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4 Longest Prefix Match Routes</td>
<td>8k (16K with uRPF disabled)</td>
</tr>
<tr>
<td>IPv4 Host Table</td>
<td>8,000</td>
</tr>
<tr>
<td>IP Multicast Routes</td>
<td>4,000</td>
</tr>
<tr>
<td>L3 Interfaces</td>
<td>1K</td>
</tr>
<tr>
<td>VRF</td>
<td>1K</td>
</tr>
</tbody>
</table>

Ver 2: N55-D160L3-V2 and N55-M160L3-V2

<table>
<thead>
<tr>
<th>Capability</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4 Longest Prefix Match Routes</td>
<td>8k (16K with uRPF disabled)</td>
</tr>
<tr>
<td>IPv4 Host Table</td>
<td>16,000</td>
</tr>
<tr>
<td>IP Multicast Routes</td>
<td>8,000</td>
</tr>
<tr>
<td>L3 Interfaces</td>
<td>1K</td>
</tr>
<tr>
<td>VRF</td>
<td>1K</td>
</tr>
</tbody>
</table>

Hardware capabilities only, check documentation for software support status.
Nexus 5500 Layer 3 Licensing

- **Layer 3 Base license**
  - $0 with daughter card / expansion module purchase
  - Installation required
  - SVIs, L3 routed ports on non-FEX interfaces, routed ACLs and uRPF
  - Static Routing, RIPv2, OSPFv2, EIGRP-stub
  - HSRP, VRRP
  - IGMP v2/v3, PIMv2
  - Maximum 256 non-connected OSPF routes

- **Layer 3 Enterprise license**
  - Layer 3 Base license must be present
  - Full EIGRP, BGP
  - VRF-Lite
  - Scalable to 8K routes
Nexus 3000 Layer 3 Support

Hardware and Software

- Layer 3 support from day 1 (NX-OS 5.0(3)U1)
- Enable line-rate Layer 3 on all ports:
  - N3064
  - N3016
  - N3048
- Similar hardware capabilities as Nexus 5500 Gen 2
  Layer 3 expansion module / daughter-card
- Layer 3 software capabilities target ToR deployment in scale-out fabrics
# Nexus 3000 Layer 3 Licensing

## Layer 3 Requirements

### Nexus 3000 Layer 3 Licensing

#### LAN Enterprise

- **Advanced IP Routing**: Higher scale for IGP; BGP and VRF-Lite

<table>
<thead>
<tr>
<th>Advanced IP Routing</th>
<th>Higher scale for IGP; BGP and VRF-Lite</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Base</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic L3 features</td>
<td>Inter-VLAN routing, Static routes, RIPv2, ACLs, OSPFv2 (limited to 256 routes), EIGRP stub, HSRP, VRRP and uRPF</td>
</tr>
<tr>
<td>IP Multicast</td>
<td>PIM SM, SSM, MSDP</td>
</tr>
</tbody>
</table>

- Licensing scheme identical to Nexus 5500’s
  - **Difference**: cost associated with Base license
- Customer must buy and install both licenses for full Layer 3 support
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
  - Unicast Routing and Forwarding
  - Multicast Routing and Forwarding
  - Layer 3 Virtualization
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
IP Unicast Routing and Forwarding

- NX-OS and Nexus switches strongly decouple control plane and data plane
- Routing and forwarding tables built on control plane using routing protocols or static configuration
  - OSPF, EIGRP, IS-IS, RIP, BGP for dynamic routing
- Tables downloaded to forwarding engine hardware for data plane forwarding
NX-OS and Nexus 7000 Unicast Routing Architecture

- Neighbor management
- Protocol database
- Add/Delete prefixes

Reverse Path Forwarding (RPF) updates

Route accept from uRIB
Push routes to hardware

Translate routes to hardware format
Program hardware forwarding engine

Common API to clients
Digest routing info
Select best nexthop(s) per prefix
Apply routing policy
Build final routing table

Common API to clients
Digest routing info
Select best nexthop(s) per prefix
Apply routing policy
Build final routing table
Unicast Routing Information Base (URIB)

- IPv4 (U4RIB) and IPv6 (U6RIB)
- Client-driven and designed for easy insertion of new clients
  - Adds and stores:
    - Best paths and next hops
    - Backup paths and next hops
    - Remote MPLS labels with routes
  - Resolves Recursive Next Hops
  - Downloads routes and remote MPLS labels to FIB
  - No direct Interface Manager or IP event handling
  - Table-based (multi-topology “capable”)
NX-OS and Nexus 7000 Unicast Routing Architecture

Unicast Routing Information Base (uRIB)

u4RIB

u6RIB

FIB Manager

Forwarding Hardware

Supervisor

I/O Module

RIP  IS-IS  EIGRP  Static  OSPF v2  OSPF v3  BGP  ARP

AM

mRIB

nx-os-layer-3-software-architecture
Forwarding Information Base (FIB)

- Receives route updates from URIB with outgoing MPLS labels:
  - BGP (VPNv4 and VPNv6) / static routes + VPN labels
  - IGP / static routes + LDP labels
- Programs routes in the hardware and passes label information to LFIB for programming
- Prefixes chained to table and identified by (VRF, topology)
- Each table is tied to a VDC descriptor block
- FIBs on I/O modules belonging to same VDC are programmed identically
NX-OS and Nexus 7000 Unicast Routing Architecture

Unicast Routing Information Base (uRIB)

- RIP
- IS-IS
- EIGRP
- Static
- OSPF v2
- OSPF v3
- BGP
- ARP

mRIB

u4RIB

u6RIB

uFDM

FIB Manager

Forwarding Hardware

Supervisor

show ip bgp

show forwarding [vrf <vrf>] [ipv4|ipv6] route module <mod>
show forwarding [vrf <vrf>] adjacency module <mod>

show system internal forwarding ipv4 route
show system internal forwarding adjacency

I/O Module

show routing [ipv4|ipv6] [<prefix>] [vrf <vrf>] (show ip route)
Management and troubleshooting

```plaintext
switch# show routing ipv4 direct
31.3.3.3/32, ubest/mbest: 2/0, attached
  *via 31.3.3.3, Lo0, [0/0], 1w5d, direct

switch# show routing ipv4 10.13.34.4 detail
10.13.34.4/30, ubest/mbest: 1/0
  *via 10.13.32.6, Eth8/2, [110/6], 01:19:22, ospf-1, intra
    via 34.34.34.34, [200/0], 01:19:06, bgp-100, internal, tag 100
      recursive next hop: 34.34.34.34/32

switch# show routing ipv6 2142:142:34:109::1:1 detail
2142:142:34:109::1:0/112, ubest/mbest: 1/0
  cand ubest/mbest: 1/0, ufdm in/update: 1/0
  *via ::ffff:34.34.34.34%default:IPv4, [200/0], 01:20:18, bgp-100, internal, tag 300 (mpls-vpn)
    recursive next hop: ::ffff:34.34.34.34/128

!Recently learned routes
switch# show routing updated since 03:35
!Routes pointing to specific neighbor
switch# show routing next-hop 10.1.1.1
```
FIB

Management and troubleshooting

switch# show forwarding vrf default ipv4 route 10.13.34.4 module 8
Prefix       Next-hop       Interface
10.13.34.4/30 10.13.32.6   Ethernet8/2

switch# show forwarding vrf default ipv6 route 2142:142:34:109::1:1 module 8
Prefix       Next-hop       Interface
*2142:142:34:109::1:0/112 10.13.32.6   Ethernet8/2

switch# show forwarding vrf default ipv4 adjacency module 8
next-hop     rewrite info   interface
10.13.32.2   0026.980a.2943  Ethernet8/1
10.13.32.6   001a.a223.a034  Ethernet8/2

switch# show forwarding vrf default ipv6 adjacency module 8
next-hop     rewrite info   interface
2015:32:48:109::2 001c.b0cb.3580  Eth4/5.109
fe80::21c:b0ff:fecb:3580 001c.b0cb.3580  Eth4/5.109

Adjacency Table
Agenda

- Layer 3 requirements
- **NX-OS Layer 3 Software Architecture**
  - Unicast Routing and Forwarding
  - **Multicast Routing and Forwarding**
  - Layer 3 Virtualization
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
IP Multicast Routing and Forwarding

- Forwarding tables built on control plane using multicast protocols
  - PIM-SM, PIM-SSM, PIM-Bidir, IGMP, MLD, MSDP
- Tables downloaded to:
  - Forwarding engine hardware for data plane forwarding
  - Replication engines for data plane packet replication
NX-OS and Nexus 7000 Multicast Routing Architecture

- **IGMP**: Add (*,G) & (S,G) from reports
- **MSDP**: Add (S,G) from SAs
- **PIM**: Add (*,G) & (S,G) from Join/Prune & Register/Assert
- **PIM6**: Add (*,G) & (S,G) from reports
- **ICMPv6 / MLD**: Add (*,G) & (S,G) from reports

**m4RIB**
- Multicast Routing Information Base (mRIB)
- *Add routes, OIFs
  - Update when RPF changes*

**m6RIB**
- *Route accept from mRIB
  - Push routes to hardware*

**mFDM**
- *RPF updates*

**FIB Manager**
- *Translate routes to hardware format
  - Program hardware forwarding and replication engines*

**Forwarding Hardware**

**uRIB**

---

**NX-OS Layer 3 Software Architecture**
Multicast Routing Information Base (MRIB)

Highlights

- IPv4 (M4RIB) and IPv6 (M6RIB)
- Interface with various client processes to provide/learn multicast routes
- Aggregate route-information provided by multiple protocols into single multicast routing table
- RPF services via U4RIB / U6RIB
- Client-driven and designed for easy insertion of new clients
- Detailed traffic statistics
NX-OS and Nexus 7000 Multicast Routing Architecture

Multicast Routing Information Base (mRIB)

- m4RIB
- m6RIB

mFDM

FIB Manager

Forwarding Hardware

- IGMP
  - show ip igmp route (show ip igmp groups)
- MSDP
  - show ip msdp route
  - show ip msdp sa-cache
- PIM
  - show ip pim route
- PIM6
  - show ipv6 pim route
- ICMPv6 / MLD
  - show routing [ipv4|ipv6] multicast [vrf <vrf>] [source-ip] [group-ip] [summary] (show [ipv4|ipv6] mroute)

show forwarding distribution ip multicast route
show forwarding distribution ip igmp snooping

show forwarding [ipv4|ipv6] multicast route [source <ip>] [group <ip>] [vrf <vrf>] module <mod>
show system internal forwarding ip multicast route
show system internal ip igmp snooping
MRIB and FIB
Management and troubleshooting

switch# `show routing multicast 239.1.1.1 summary`
Source packets bytes aps pps bit-rate oifs
(*,G) 767 84370 110 0 0 bps 2
10.1.1.2 9917158 1269395810 127 4227 4 mbps 2

switch# `show ip mroute detail`
(150.1.1.1/32, 226.1.2.1/32), uptime: 1d01h, mrib pim ip
Stats: 10369/1910536 [Packets/Bytes], 171.200 bps
Incoming interface: port-channel 17, RPF nbr: 20.1.17.1
Outgoing interface list: (count: 4)
  port-channel 57, uptime: 1d01h, pim

switch# `show forwarding multicast route group 226.1.2.1/32 source`
  150.1.1.1/32 mod 1
(150.1.1.1/32, 226.1.2.1/32), RPF Interface: port-channel 17, flags:
Received Packets: 0 Bytes: 0
Number of Outgoing Interfaces: 4
Outgoing Interface List Index: 43
  port-channel 27 Outgoing Packets:41505884 Bytes:4087857190
  port-channel 37 Outgoing Packets:0 Bytes:0
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
  - Unicast Routing and Forwarding
  - Multicast Routing and Forwarding
  - Layer 3 Virtualization
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
Virtual Device Contexts (VDC)

Highlights

✓ Carve a single Nexus 7000 switch into multiple logical network entities
✓ Flexible separation of hardware and software resources
✓ Isolate software faults and reduce fate sharing
✓ Securely delineate administrative domains
✓ All MPLS and Layer 3 features are VDC aware
Layer 3 Virtualization (VRF)

Highlights

- Virtual Routing and Forwarding (VRF) provides logical network segmentation
- Virtualizes IP routing control and data plane functions
- For each VRF:
  - Independent routing and forwarding decisions
  - IPv4 and IPv6 unicast/mcast tables are created automatically
- VRF membership of each interface dictates which forwarding table to use
Layer 3 Virtualization (VRF)

Pre-defined VRFs

- **Management VRF**
  - For management purposes only
  - Only mgmt 0 interface can be a member
  - mgmt 0 can not be assigned to another VRF
  - No routing protocols (static only)

- **Default VRF**
  - All Layer-3 interfaces are in the default VRF
  - Default routing context for all show and exec commands
  - All IPv4 and IPv6 routing protocols run in a default VRF context unless other VRF context is specified
  - Similar to IOS global routing table
VRF Awareness

- In NX-OS, everything is VRF aware!
  (IPv4 and IPv6)

<table>
<thead>
<tr>
<th>Service</th>
<th>NX-OS Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>VRF Specific Static ARP</td>
<td>✓</td>
</tr>
<tr>
<td>PBR-set VRF</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware AAA (RADIUS)</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware AAA (TACACS+)</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware BGP</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware DHCP</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware DNS</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware FTP / TFTP</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware NTP</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware OSPF</td>
<td>✓</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Service</th>
<th>NX-OS Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>VRF Aware Ping</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware SCP</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware SNMP agent</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware SSH</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware Syslog</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware Telnet</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware Traceroute</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware uRPF</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware FHRP protocols</td>
<td>✓</td>
</tr>
<tr>
<td>VRF Aware WCCP</td>
<td>✓</td>
</tr>
</tbody>
</table>
MPLS Layer 3 VPN
Nexus 7000

- Unicast - IPv4 and IPv6 (6VPE)
- Multicast - IPv4 (MVPN)
- Most widely deployed MPLS application to provide scalable segmentation
- Supports network consolidation and access to shared services
- Provides IPv6 connectivity over the same IPv4/MPLS L3VPN infrastructure
- Easier IPv4 to IPv6 migration

More details: BRKMPL-2102 - Deploying MPLS-based IP VPNs
### MPLS Layer 3 Unicast VPN and VRF-lite

**Nexus 7000**

- **VPN feature set for** [VRF-lite](https://www.cisco.com/c/en/us/solutions/collateral/service-provisioning/vrf-lite/white_paper-c11-459928.html) and **MPLS VPNs**:

<table>
<thead>
<tr>
<th>Sub-feature area</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route-leaking (Import/Export)</td>
<td>Yes – MPLS VPN or VRF-lite</td>
</tr>
<tr>
<td>Route filtering and limits</td>
<td>• Per VRF max route limit in RIB</td>
</tr>
<tr>
<td></td>
<td>• Import / export map</td>
</tr>
<tr>
<td>PE-CE routing protocols</td>
<td><strong>IPv4:</strong> OSPF v2, EIGRP, BGP, Static</td>
</tr>
<tr>
<td></td>
<td><strong>IPv6:</strong> BGP, Static</td>
</tr>
<tr>
<td>VPN site loop prevention</td>
<td>• Site of Origin for EIGRP and BGP</td>
</tr>
<tr>
<td></td>
<td>• Down Bit for OSPF</td>
</tr>
</tbody>
</table>
Route Leaking Options in NX-OS
Nexus 7000

- VRF-lite scenario:
  - Static routes
    - `vrf context test2
      ip route 11.0.0.0/24 2.2.2.2 vrf test`
    - `vrf context test
      ip route 10.0.0.0/24 1.1.1.1 vrf test2`
  - Import vrf default + static route
    - `ip route 10.0.0.0/24 1.1.1.1 vrf test`
    - `vrf context test
      address-family ipv4 unicast
      import vrf default map test`
  - PBR VRF select

- VRF-lite or MPLS VPN scenario:
  - Route target import / export
Route Leaking in NX-OS
PBR VRF select

- PBR configuration:

```plaintext
feature pbr
vrf context backup
vrf context prod-db
vrf context prod-app
!
route-map return-traffic-from-backup-vrf permit 10
  match ip address backup-to-prod-DB-subnets
  set vrf prod-db
!
route-map return-traffic-from-backup-vrf permit 20
  match ip address backup-to-prod-APP-subnets
  set vrf prod-app
!
route-map traffic-from-prod-APP-to-backup-vrf permit 10
  match ip address prod-APP-subnets-needing-backup
  set vrf backup
!
route-map traffic-from-prod-DB-to-backup-vrf permit 10
  match ip address prod-DB-subnets-needing-backup
  set vrf backup
```
Route Leaking in NX-OS

PBR VRF select

- Apply VRF membership and policy to interfaces:

```plaintext
interface Vlan225
  no shutdown
  vrf member prod-app
  ip address 10.64.20.1/23
  ip policy route-map traffic-from-prod-APP-to-backup-vrf
!
interface Vlan240
  no shutdown
  vrf member prod-db
  ip address 10.64.32.1/24
  ip policy route-map traffic-from-prod-DB-to-backup-vrf
!
interface Vlan628
  no shutdown
  vrf member backup
  ip address 10.66.254.1/23
  ip policy route-map return-traffic-from-backup-vrf
```
Route Leaking in NX-OS
VRF-lite Shared Service with Route Target Import / Export

- No RD needed
- No MPLS license needed!
- BGP without neighbors

```bash
feature bgp
vrf context Orange
  address-family ipv4 unicast
    route-target export 1:1
    route-target import 3:3
!
vrf context Green
  address-family ipv4 unicast
    route-target export 2:2
    route-target import 3:3
!
vrf context Shared
  address-family ipv4 unicast
    route-target export 3:3
    route-target import 1:1
    route-target import 2:2

router eigrp 100
  vrf Orange
    redistribute bgp 1 route-map x
  vrf Green
    redistribute bgp 1 route-map y
!
router bgp 1
  vrf Red
    address-family ipv4 unicast
      redistribute eigrp 100 route-map a
    vrf Green
      address-family ipv4 unicast
        redistribute eigrp 100 route-map b
    vrf Shared
      address-family ipv4 unicast
        redistribute connected
```
Route Leaking in NX-OS
MPLS Layer 3 Unicast VPN *intranet* with MP-BGP

- Unique RD per VRF and per PE for load-balancing
- Common RT per VRF on ALL PEs

```bash
vrf context test
  rd 118.48.20.162:100
  address-family ipv4 unicast
    route-target both 300:30

router bgp 65000
  address-family vpnv4 unicast
    router-id 118.48.20.162
  address-family ipv4 unicast
  neighbor 118.8.20.164 remote-as 65000
    update-source loopback0
    address-family vpnv4 unicast
      send-community extended

!Configuration under router bgp mode
vrf test
  address-family ipv4 unicast
    redistribute connected
```
# VRF Configuration

## NX-OS – IOS Differences

<table>
<thead>
<tr>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip vrf vrf-1</td>
<td>vrf context vrf-1</td>
</tr>
<tr>
<td>rd 1:1</td>
<td>[shutdown]</td>
</tr>
<tr>
<td>route-target import 1:1</td>
<td>ip route 10.0.0.0/8 1.1.1.1</td>
</tr>
<tr>
<td>route-target export 1:1</td>
<td>!not available with VRF-lite</td>
</tr>
<tr>
<td>!</td>
<td>rd 1:1</td>
</tr>
<tr>
<td>ip route vrf vrf-1 10.0.0.0 255.0.0.0 1.1.1.1</td>
<td>address-family ipv4 unicast</td>
</tr>
<tr>
<td>vrf definition vrf-1</td>
<td>route-target import 1:1</td>
</tr>
<tr>
<td>rd 1:1</td>
<td>route-target export 1:1</td>
</tr>
<tr>
<td>address-family ipv4</td>
<td>!</td>
</tr>
<tr>
<td>route-target import 1:1</td>
<td>ip route vrf vrf-1 10.0.0.0 255.0.0.0 1.1.1.1</td>
</tr>
<tr>
<td>route-target export 1:1</td>
<td></td>
</tr>
<tr>
<td>!</td>
<td></td>
</tr>
<tr>
<td>int GigEthernet2/1.10</td>
<td>int Ethernet2/1.10</td>
</tr>
<tr>
<td>encapsulation dot1q 10</td>
<td>encapsulation dot1q 10</td>
</tr>
<tr>
<td>vrf forwarding vrf-1</td>
<td>vrf member vrf-1</td>
</tr>
<tr>
<td>!</td>
<td></td>
</tr>
<tr>
<td>int GigEthernet2/2.10</td>
<td>int Ethernet2/2.10</td>
</tr>
<tr>
<td>encapsulation dot1q 11</td>
<td>encapsulation dot1q 10</td>
</tr>
<tr>
<td>vrf forwarding vrf-1</td>
<td>vrf member vrf-1</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# IGP Virtualization

- Multiple processes per VDC
- **Multiple VRFs per process (best practice)**
- Forward referencing of VRFs

<table>
<thead>
<tr>
<th>Protocol</th>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
</table>
| OSPF     | router ospf 10 vrf 1  
  router-id 1.1.1.1  
  !required for multi-VRF IOS CE  
  capability vrf-lite | router ospf 10  
  router-id 1.1.1.1  
  vrf 1  
  router-id 1.1.1.2 |
| IS-IS    | router isis 1  
  net <NSAP address>  
  !maximum 1 VRF per process  
  vrf 1 | router isis 1  
  net <NSAP address>  
  vrf 1  
  net <net name 2> |
| EIGRP    | router eigrp 1  
  eigrp router-id 1.1.1.1  
  address-family ipv4 vrf 1 autonomous-system 200  
  eigrp router-id 2.2.2.2 | router eigrp 1  
  router-id 1.1.1.1  
  vrf 1  
  autonomous-system 200  
  router-id 2.2.2.2 |
Layer 3 Virtualization (VRF)
Management and Troubleshooting

- Placement of “vrf” option in the end
- “routing-context” mode
  - Set VRF scope for show and exec commands
  - By default, CLI exec mode operates in VRF default

<table>
<thead>
<tr>
<th>Without “routing-context”</th>
<th>With “routing-context”</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch# ping 172.26.242.1 vrf management</td>
<td>switch# routing-context vrf management</td>
</tr>
<tr>
<td>switch# show routing vrf management</td>
<td>switch%management# show routing</td>
</tr>
<tr>
<td>switch# show ip arp vrf management</td>
<td>switch%management# show ip arp</td>
</tr>
</tbody>
</table>
Unicast Routing Architecture with VDCs and VRFs

- VDC 1
  - OSPF VRF 1
  - VRF 1
  - uRIB
  - uFDM

- VDC 2
  - OSPF VRF 1
  - VRF 1
  - uRIB
  - uFDM

- VDC 3
  - OSPF VRF 1
  - VRF 1
  - uRIB
  - uFDM

Supervisor Engine

- IP FIB
- Hardware
- I/O Module
Multicast Routing Architecture with VDCs

- VDC1
  - PIM
  - VRF 1
  - VRF 2
  - etc.
  - VRF 1
  - VRF 2
  - mRIB
  - MFDM

- VDC2
  - PIM
  - VRF 1
  - VRF 2
  - etc.
  - VRF 1
  - VRF 2
  - mRIB
  - MFDM

- VDC3
  - PIM
  - VRF 1
  - VRF 2
  - etc.
  - VRF 1
  - VRF 2
  - mRIB
  - MFDM

- Supervisor Engine

- IP FIB
  - Hardware
  - I/O Module

- IP FIB
  - Hardware
  - I/O Module

- IP FIB
  - Hardware
  - I/O Module
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- **Unicast Routing Protocols**
- Multicast Routing
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
# NX-OS Unicast Routing Protocols

## Platform Support

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Nexus 7000</th>
<th>Nexus 5500</th>
<th>Nexus 3000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IPv4</td>
<td>IPv6</td>
<td>IPv4</td>
</tr>
<tr>
<td>RIPv2</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
</tr>
<tr>
<td>IS-IS</td>
<td>✓</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>OSPFv2</td>
<td>✓</td>
<td>N/A</td>
<td>✓</td>
</tr>
<tr>
<td>OSPFv3</td>
<td>-</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>EIGRP</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>BGPv4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Static</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

**Notes:**

- Nexus 5500 supports IPv6 in hardware, software support in future release
- Check release notes and configuration guides for latest support information
NX-OS Unicast Routing Protocols

Highlights

- Modular architecture
- Extensive High Availability features
- IPv4 and IPv6 support
- CLI/features similar to IOS (Catalyst 6500)
NX-OS Unicast Routing Protocols

Configuration Highlights

- Interface-centric model for IGPs
  - Simplified configuration view
  - Less error prone
  - For EIGRP and OSPF, backward compatible with “network” statements (hidden command)

```
router ospf 1
  passive-interface GigEthernet1/1
  network 10.0.0.1 0.0.0.0 area 0
router eigrp 1
  network 10.0.0.1 0.0.0.0
```

- Enable feature first to be able to configure:
  ```
  feature ospf
  ```

- Protocol shutdown option (all protocols):
  ```
  router ospf foo
  shutdown
  ```
 NX-OS Unicast Routing Protocols
Management and Troubleshooting Highlights

- Modular approach:
  `show run ospf`

- Controlled restart:
  `restart ospf 1`

- Record adjacency changes to logfile:
  `router ospf foo log-adjacency-changes`

- Debug-filters
  - Restrict the amount of debug output: neighbor, interface, VRF etc.
  - Each protocol has its own set of debug-filters
  - Use `debug-filter <protocol> vrf all` to apply to all VRFs
Router-id selection process

- Best practice for IGPs and BGP – specify explicitly using “router-id” command
- If not specified:

<table>
<thead>
<tr>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Highest loopback interface IP</td>
<td>1) IP from loopback 0</td>
</tr>
<tr>
<td>2) Highest IP from any other</td>
<td>2) IP from first loopback interface</td>
</tr>
<tr>
<td>interface</td>
<td>3) IP from first physical interface</td>
</tr>
</tbody>
</table>
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- **Unicast Routing Protocols**
  - OSPF
  - IS-IS
  - EIGRP
  - BGP
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
OSPF in NX-OS

Highlights

- OSPF v2 (RFC 2328 with option for RFC 1583 compatibility)
- OSPF v3 (RFC 5340, IPv6 only)
- Dynamic and modern link-state protocol
- Extensive High-Availability features
  - Graceful Restart (Nexus 3000 / 5500 in helper mode only)
  - Stateful Restart
  - SSO / ISSU
# OSPF in NX-OS

## Key Features

<table>
<thead>
<tr>
<th>Sub-feature area</th>
<th>Details (defaults in bold)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Areas</td>
<td>Multi-area&lt;br&gt;Regular, Stub, NSSA (RFC 3101)</td>
</tr>
<tr>
<td>LSAs</td>
<td>Type 3 (summary), Type 5 (redistributed), Type 9 and 10</td>
</tr>
<tr>
<td>Multi-pathing</td>
<td><strong>8</strong> (up to 16)</td>
</tr>
<tr>
<td>Metric manipulation</td>
<td>Static cost&lt;br&gt;Auto-cost reference bandwidth</td>
</tr>
<tr>
<td>Link types</td>
<td>Broadcast, Point-to-point, Virtual link, Sham link</td>
</tr>
<tr>
<td>Summarization and</td>
<td>Area range; summary-address&lt;br&gt;Filter lists (using route-maps)</td>
</tr>
<tr>
<td>filtering</td>
<td></td>
</tr>
<tr>
<td>Authentication</td>
<td>Clear text, MD5, keychains</td>
</tr>
<tr>
<td>Traffic Engineering</td>
<td>Yes – TE extentions (RFC3630) for OSPF v2</td>
</tr>
</tbody>
</table>
**OSPF in NX-OS**

Configuration and Management Highlights

- **Area ID for OSPF v2 and v3:**
  - Configure as single decimal or dotted decimal
  - For consistency, always display in dotted decimal

```bash
switch(config-if)# ip router ospf 1 area ?
A.B.C.D or <0-4294967295>  Area Id as an integer or ip address
switch(config-if)# ip router ospf 1 area 2
```

- **OSPF v3 address family mode configuration for summarization / redistribution:**

```bash
router ospfv3 1
  address-family ipv6 unicast
  area 4 range 2006:2000::/32
```

---

Unicast Routing Protocols
OSPF in NX-OS

Configuration Examples

- **OSPF v2**

  ```
  router ospf foo
  passive-interface default

  interface Vlan10
    ip router ospf 100 area 0.0.0.0
  interface Ethernet2/6
    no ip ospf passive
  ip router ospf 100 area 0.0.0.1
  ```

- **OSPF v3**

  ```
  router ospfv3 foo
    router-id 60.60.60.60
    area 4 virtual-link 50.50.50.50
    address-family ipv6 unicast
      area 4 range 2006:2000::/32

  interface Ethernet3/1
    ipv6 address 2006:5101:0100::2003/33
  ipv6 router ospfv3 foo area 0
  ```
OSPF in NX-OS
Management and Troubleshooting

switch# show ip ospf
Routing Process pl with ID 40.40.40.40 VRF default
Stateful High Availability enabled
Graceful-restart is configured
  grace period: 60, state: (null)
  Last graceful restart exit status: None
Supports opaque LSA
This router is an area border and autonomous system boundary.
Redistributing External Routes from
  isis-one
Administrative distance 110
Reference Bandwidth is 40000 Mbps
<...>
Maximum paths to destination 8
Number of external LSAs 6, checksum sum 0x2d81e
Number of areas is 3, 2 normal, 0 stub, 1 nssa
Number of active areas is 2, 2 normal, 0 stub, 0 nssa
Area BACKBONE(0)
  Area has existed for 1d01h
  Interfaces in this area: 5 Active interfaces: 5
  Passive interfaces: 0  Loopback interfaces: 0
  No authentication available
  SPF calculation has run 18 times
# NX-OS / IOS OSPF Differences

<table>
<thead>
<tr>
<th>Feature</th>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graceful Restart</td>
<td>Yes (Disabled)</td>
<td>Yes (Enabled)</td>
</tr>
<tr>
<td></td>
<td>IETF (RFC3623) or Cisco NSF</td>
<td>IETF (RFC3623)</td>
</tr>
<tr>
<td>Auto-cost Reference</td>
<td>100 Mbps</td>
<td>40 Gbps</td>
</tr>
<tr>
<td>Bandwidth default</td>
<td></td>
<td></td>
</tr>
<tr>
<td>iSPF</td>
<td>Yes (Disabled)</td>
<td>No</td>
</tr>
<tr>
<td>Partial SPF</td>
<td>Yes (Always Enabled)</td>
<td>Yes (Always Enabled)</td>
</tr>
<tr>
<td>Flood Reduction</td>
<td>Yes (Disabled)</td>
<td>No</td>
</tr>
<tr>
<td>Neighbor logging</td>
<td>Yes (Enabled)</td>
<td>Yes (Disabled)</td>
</tr>
</tbody>
</table>

* Defaults in **bold**
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- **Unicast Routing Protocols**
  - OSPF
  - IS-IS
  - EIGRP
  - BGP
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
IS-IS in NX-OS

Highlights

- ISO 10589 Intermediate System To Intermediate System intra-domain routing exchange protocol
- IPv4 routing support only
- Extensive High-Availability features
  - Graceful Restart
  - Stateful Restart
  - SSO / ISSU
## IS-IS in NX-OS

### Key Features

<table>
<thead>
<tr>
<th>Sub-feature area</th>
<th>Details (defaults in bold)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-pathing</td>
<td>8 (up to 16)</td>
</tr>
<tr>
<td>Authentication</td>
<td>Clear text, MD5, keychains</td>
</tr>
<tr>
<td>Wide Metric</td>
<td>Yes</td>
</tr>
<tr>
<td>Traffic Engineering</td>
<td>Yes – TE extensions (RFC 3784)</td>
</tr>
</tbody>
</table>
IS-IS in NX-OS
Configuration Highlights and Best Practices

- IS-IS point-to-point link configuration:
  ```
  switch(config)# interface Ethernet 2/4
  switch(config-if)# medium p2p
  ```

- Metric style – “transition” option for interoperability:
  ```
  switch(config)# router isis 1
  switch(config-router)# metric-style ?
  transition Use both narrow and wide metric style
  ```
IS-IS in NX-OS
Configuration

```plaintext
router isis foo
  mpls ldp autoconfig level-1-2
  net 49.0000.0032.0001.0001.00
  is-type level-1-2
  metric-style transition
  log-adjacency-changes
  address-family ipv4 unicast
    default-information originate

interface Vlan90
  ip router isis 1
  isis passive level-1-2
interface loopback0
  ip router isis 1
interface Ethernet1/8
  ip router isis 1
```
# NX-OS / IOS IS-IS Differences

<table>
<thead>
<tr>
<th>Feature</th>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graceful Restart</td>
<td>Yes (Disabled)</td>
<td>Yes (Enabled)</td>
</tr>
<tr>
<td></td>
<td>IETF (RFC 5306) or Cisco NSF</td>
<td>IETF (RFC 5306)</td>
</tr>
<tr>
<td>Metric style options</td>
<td>Narrow</td>
<td>Wide Transition</td>
</tr>
<tr>
<td></td>
<td>Wide</td>
<td>Transition</td>
</tr>
<tr>
<td>Neighbor logging</td>
<td>Yes (Enabled)</td>
<td>Yes (Disabled)</td>
</tr>
</tbody>
</table>

* Defaults in **bold**
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- **Unicast Routing Protocols**
  - OSPF
  - IS-IS
  - EIGRP
  - BGP
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
EIGRP in NX-OS

Highlights

- Single process instance for IPv4 and IPv6 routing
- Support for only IP protocols
- Wide metric support with interoperability
- High-Availability features
  - Graceful Restart (Nexus 3000 / 5500 in helper mode only)
  - SSO / ISSU
# EIGRP on NX-OS

## Key Features

<table>
<thead>
<tr>
<th>Sub-feature area</th>
<th>Details (defaults in bold)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summarization and route-filtering</td>
<td>Summary address (on interface)</td>
</tr>
<tr>
<td></td>
<td>Distribute lists (using prefix list or route map)</td>
</tr>
<tr>
<td>Multi-pathing</td>
<td>8 (up to 16)</td>
</tr>
<tr>
<td>Authentication</td>
<td>MD5, keychains</td>
</tr>
</tbody>
</table>
EIGRP in NX-OS

Configuration Highlights

- "distribute-list" configuration is interface-centric only:

```plaintext
feature eigrp

interface Ethernet2/1
   ip address 10.1.48.4/24
   ip router eigrp 1
   ip distribute-list eigrp 1 route-map test in

router eigrp 1
   network 10.1.48.0 0.0.0.255
```

- Use "default-information originate" to accept and advertise default

```plaintext
switch(config)# router eigrp 1
switch(config-router)# default-information originate ?
<CR>
always Always advertise default route
route-map Use a route-map for default route metrics
```
EIGRP in NX-OS

Configuration

```conf
router eigrp foo
  autonomous-system 100
  router-id 31.3.3.3
  address-family ipv4 unicast
  address-family ipv6 unicast
    redistribute bgp 1 route-map test
  autonomous-system 100
vrf 101
  autonomous-system 101
  router-id 31.3.3.3
  address-family ipv4 unicast
  address-family ipv6 unicast
    redistribute bgp 1 route-map test
  autonomous-system 101

interface Vlan100
  ip router eigrp 1
  ip passive-interface eigrp 1

interface port-channel10.100
  ip router eigrp 1
  ip summary-address eigrp 1 101.0.0.0/16 255
```
EIGRP in NX-OS
Management and Troubleshooting Highlights

switch# show ip eigrp topology detail-links vrf 100
IP-EIGRP Topology Table for AS(1)/ID(31.1.1.1) VRF 100
Codes: P – Passive, A – Active, U – Update, Q – Query, R – Reply,
r – reply Status, s – sia Status
P 11.0.15.0/30, 5 successors, FD is 768, serno 603
  via 11.0.9.2 (768/320), Ethernet1/16.100
  via 11.0.6.2 (768/320), Ethernet1/13.100
  via 11.0.21.2 (768/320), Ethernet1/1.100
  via 11.0.22.2 (768/320), Ethernet1/2.100
  via 11.0.8.2 (768/320), Ethernet1/15.100
  via 11.0.10.2 (1024/768), port-channel10.100
<...>
show ip eigrp [<instance>]
show ip eigrp accounting
show ip eigrp interfaces [detail]
show ip eigrp neighbors [detail]
show ip eigrp route
show ip eigrp route-map statistics redistribute <protocol>
show ip eigrp topology
show ip eigrp traffic
show ip eigrp vrf
# NX-OS / IOS EIGRP Differences

<table>
<thead>
<tr>
<th>Feature</th>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graceful Restart</td>
<td>Yes (Disabled) - Cisco</td>
<td>Yes (Enabled) - Cisco</td>
</tr>
<tr>
<td>Auto-summary</td>
<td>Yes (Disabled)</td>
<td>No (Always Disabled)</td>
</tr>
<tr>
<td>Neighbor logging</td>
<td>Yes (Enabled)</td>
<td>Yes (Disabled)</td>
</tr>
<tr>
<td>Bandwidth configuration on interface</td>
<td>Bandwidth – 10 Gbps max</td>
<td>Bandwidth – 400 Gbps max</td>
</tr>
</tbody>
</table>

* Defaults in **bold**
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- **Unicast Routing Protocols**
  - OSPF
  - IS-IS
  - EIGRP
  - BGP
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
BGP

Highlights

- Full MP-BGP support
  - BGP version 4 (RFC 4271)
  - Multi-Protocol Extensions (RFC 2858)

- Integrated implementation for IPv4 and IPv6

- IPv6 support
  - IPv6 peers and prefixes
  - 6PE / 6VPE

- 4 byte ASN support and interoperability (RFC 4893)

- 1 BGP instance per VDC
# BGP in NX-OS

## Key Features

- BGP and MP-BGP features:

<table>
<thead>
<tr>
<th>Sub-feature area</th>
<th>Details (defaults in bold)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address Families</td>
<td>IPv4/ IPv6 unicast and multicast, MDT (MVPN), VPNv4, VPNv6 (6VPE), IPv6 labeled-unicast (6PE)</td>
</tr>
<tr>
<td>Load-balancing</td>
<td>Label, IP src-dst, TCP/UDP</td>
</tr>
<tr>
<td>Multipath</td>
<td>Yes – up to 16 for iBGP / eBGP / eiBGP</td>
</tr>
<tr>
<td>Route reflector</td>
<td>Yes for all AFs</td>
</tr>
<tr>
<td>Route Manipulation</td>
<td>Conditional Advertisement, routing policies</td>
</tr>
<tr>
<td>Hub and spoke</td>
<td>Yes</td>
</tr>
<tr>
<td>HA and Convergence</td>
<td><strong>Graceful Restart</strong> (RFC 4724)</td>
</tr>
<tr>
<td></td>
<td>PIC Core</td>
</tr>
<tr>
<td></td>
<td>Next-Hop Tracking</td>
</tr>
<tr>
<td></td>
<td>Low Memory Alert handling</td>
</tr>
<tr>
<td>Authentication</td>
<td>MD5</td>
</tr>
</tbody>
</table>
BGP in NX-OS
Configuration Highlights

- Neighbor-centric nested configuration model
- Address-family configuration required per neighbor

```plaintext
feature bgp
router bgp 1
  address-family ipv4 unicast
  network 20.20.20.0/24
neighbor 2.2.2.2 remote-as 2
  update-source loopback 0
  address-family ipv4 unicast
```

- Peer-templates (similar to peer-groups)
  - Can inherit session and policy templates

```plaintext
template peer-session SESSION1
  timers 30 90
template peer-policy POLICY1
  default-originate
template peer CUSTOMER-PEERS
  inherit peer-session SESSION1
  address-family ipv4 unicast
  inherit peer-policy POLICY1
neighbor 192.168.2.1 remote-as 10
  inherit peer CUSTOMER-PEERS
```
BGP in NX-OS

Configuration Best Practice: Summarization

- **Auto-summary is always disabled**
- Do not use `network` statement to summarize, use `aggregate-address` instead
- If `network` statement matches a static route to null, MPLS traffic to that route may be dropped

```
switch# show ip route vrf Prod
11.0.6.0/30, ubest/mbest: 1/0, attached
  *via 11.0.6.1, Eth1/13.100, [0/0], 6w5d, direct
11.0.7.0/30, ubest/mbest: 1/0, attached
  *via 11.0.7.1, Eth1/14.100, [0/0], 6w5d, direct

switch# show run
router bgp 65000
  vrf Prod
    address-family ipv4 unicast
      network 11.0.0.0/16
      aggregate-address 11.0.0.0/16

vrf context Prod
  ip route 11.0.0.0/16 Null0
```
BGP in NX-OS
Configuration Best Practices

- Keep idle peers shut

```plaintext
router bgp 1
  neighbor 1.1.1.1 remote-as 1
  shutdown
```

- BGP low memory avoidance – protect critical peers

```plaintext
router bgp 1
  neighbor 2.2.2.2 remote-as 2
  low-memory exempt
```

- Do not use prefix peering

```plaintext
router bgp 1
  neighbor 20.0.0.0/24 remote-as 2
```

```plaintext
router bgp 1
  neighbor 20.0.0.1 remote-as 2
```
BGP in NX-OS
Management and Troubleshooting

- Session setup
  - `show ip bgp summary`
  - `show ip bgp neighbor`
  - `show bgp sessions`
  - `show tcp connection`
  - `show tcp statistics`

- BRIB (routes)
  - `show ip bgp [<prefix>]`
  - `show ip bgp next-hop-database`
  - `show ip route [<prefix>]`
  - `show bgp paths`
**BGP in NX-OS**

Management and Troubleshooting

```plaintext
show ip bgp neighbor 10.103.72.72
BGP neighbor is 10.103.72.72, remote AS 65001, ebgp link, Peer index 2
BGP version 4, remote router ID 3.3.3.3
BGP state = Established, up for 5d22h
Peer is directly attached, interface Ethernet2/1
Last read 00:00:21, hold time = 180, keepalive interval is 60 seconds
Last written 00:00:16, keepalive timer expiry due 00:00:43
Connections established 1, dropped 0
Last reset by us 00:00:10, due to bad peer AS error
Reset error value 6501
Last reset by peer 1d20h, due to holdtimer expired error
Neighbor capabilities:
4-Byte AS capability: advertised
Address family IPv4 Unicast: advertised received

Message statistics:          Sent      Rcvd
Opens:                        1          1
Notifications:                0          0
Updates:                      2          3
Keepalives:                   8555       8558
Total:                        8558       8562
Total bytes:                  162604     162773
Bytes in queue:               0          0

For address family: IPv4 Unicast
BGP table version 20, neighbor version 20
6 accepted paths consume 216 bytes of memory

Local host: 10.103.72.3, Local port: 20582
Foreign host: 10.103.72.72, Foreign port: 179
```
# NX-OS / IOS BGP Differences

<table>
<thead>
<tr>
<th>Feature</th>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graceful Restart</td>
<td>Yes (Disabled) – IETF (RFC 4724)</td>
<td>Yes (Enabled) – IETF (RFC 4724)</td>
</tr>
<tr>
<td>Synchronization</td>
<td>Yes (Disabled)</td>
<td>No (Always Disabled)</td>
</tr>
<tr>
<td>Auto-summary</td>
<td>Yes (Disabled)</td>
<td>No (Always Disabled)</td>
</tr>
<tr>
<td>Redistribution of iBGP to IGP</td>
<td>Disabled (use “redistribute internal” to enable)</td>
<td>Disabled (use route-map match statement to enable)</td>
</tr>
<tr>
<td>Deterministic MED</td>
<td>Yes (Disabled)</td>
<td>Yes (Enabled)</td>
</tr>
<tr>
<td>Neighbor logging</td>
<td>Yes (Enabled)</td>
<td>Yes (Disabled)</td>
</tr>
<tr>
<td>Next-hop tracking</td>
<td>Yes (Disabled)</td>
<td>Yes (Always Enabled)</td>
</tr>
</tbody>
</table>

* Defaults in **bold**
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- Unicast Routing Protocols
- **Multicast Protocols**
  - First Hop Redundancy Protocols
  - Routing Policy and Policy-based Routing (PBR)
  - Layer 3 High Availability and Fast Convergence
- Summary
# NX-OS Multicast Protocols

## Platform Support

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Nexus 7000</th>
<th>Nexus 5500</th>
<th>Nexus 3000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IPv4</td>
<td>IPv6</td>
<td>IPv4</td>
</tr>
<tr>
<td>IGMP v1 / v2 / v3</td>
<td>✓</td>
<td>N/A</td>
<td>✓</td>
</tr>
<tr>
<td>PIM SM</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>PIM SSM</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>PIM BiDir</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>MSDP</td>
<td>✓</td>
<td>N/A</td>
<td>✓</td>
</tr>
<tr>
<td>MLD v1 / v2</td>
<td>N/A</td>
<td>✓</td>
<td>N/A</td>
</tr>
</tbody>
</table>

1. Not supported in VPC mode
2. Not supported on F2 I/O module
3. Nexus 3000 / 5000 support IPv6 in hardware, software support in future release
PIM in NX-OS

Highlights

- PIM process responsibilities:
  - Form PIM relationship with peers
  - Processes inbound and outbound PIM protocol packets
  - Interface with MRIB to provide/learn multicast routes

- PIM support
  - IPv4 (PIM) and IPv6 (PIM6)
  - Sparse Mode - RFC 4601
  - SSM - RFC 3569 / RFC 4607
  - BiDir *

- Configurable ASM/SSM/Bidir group-ranges
- **One single** PIM process runs all PIM flavors (SSM, BiDir, ASM) for VDC

* Not supported on F2 I/O Module
PIM in NX-OS

Key Features

- Configurable options for when Auto-RP should listen versus forward Auto-RP messages.
- Configurable options for when BSR should listen versus forward Bootstrap and Candidate-RP messages.

<table>
<thead>
<tr>
<th>Sub-feature area</th>
<th>Details (defaults in bold)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RP selection and redundancy</td>
<td>➢ Static</td>
</tr>
<tr>
<td></td>
<td>➢ Boot Strap Router (BSR)</td>
</tr>
<tr>
<td></td>
<td>➢ Auto-RP listener and forwarder</td>
</tr>
<tr>
<td></td>
<td>➢ Anycast-RP (with PIM or MSDP) – RFC 4610 / 3436</td>
</tr>
<tr>
<td>Troubleshooting</td>
<td>mping, mtrace</td>
</tr>
<tr>
<td>Authentication</td>
<td>Clear text, MD5</td>
</tr>
</tbody>
</table>
PIM in NX-OS

Configuration Highlights

- Multicast routing enabled by default, no commands needed such as “ip multicast-routing”, only enable PIM with “feature pim”

- Group range taken inline without ACL (can supply a route-map)

- Configuring PIM Sparse Mode:
  
  ```
  switch(config)# ip pim rp-address 1.1.1.1 group-list 224.0.0.0/4
  switch(config)# interface Vlan101
  switch(config-if)# ip pim sparse-mode
  ```

- Configuring PIM SSM:
  
  ```
  switch(config)# interface Vlan101
  switch(config-if)# ip pim sparse-mode
  switch(config-if)# ip igmp version 3
  ```

- Configuring PIM BiDir:
  
  ```
  switch(config)# ip pim rp-address 1.1.1.1 group-list 224.0.0.0/4 bidir
  switch(config)# interface Vlan101
  switch(config-if)# ip pim sparse-mode
  ```
# NX-OS / IOS Multicast Routing Differences

<table>
<thead>
<tr>
<th>Feature</th>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIM version</td>
<td>Ver 1, 2</td>
<td>Ver 2 only</td>
</tr>
<tr>
<td>PIM modes</td>
<td>Sparse mode</td>
<td>Sparse mode</td>
</tr>
<tr>
<td></td>
<td>Dense mode</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sparse-dense mode</td>
<td></td>
</tr>
<tr>
<td>IGMP</td>
<td>Ver 1, 2, 3</td>
<td>Ver 2, 3</td>
</tr>
</tbody>
</table>

* Defaults in **bold**
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
First Hop Redundancy Protocols in NX-OS

Platform Support

- FHRPs - family of protocols (HSRP, VRRP and GLBP) designed to allow redundancy of first-hop IP gateway

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Nexus 7000</th>
<th>Nexus 5500</th>
<th>Nexus 3000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IPv4</td>
<td>IPv6</td>
<td>IPv4</td>
</tr>
<tr>
<td>HSRP v1</td>
<td>✓</td>
<td>N/A</td>
<td>✓</td>
</tr>
<tr>
<td>HSRP v2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>VRRP v2</td>
<td>✓</td>
<td>N/A</td>
<td>✓</td>
</tr>
<tr>
<td>VRRP v3</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>GLBP</td>
<td>✓</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
First Hop Redundancy Protocols in NX-OS

Key Features

<table>
<thead>
<tr>
<th>Sub-feature area</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocols</td>
<td>HSRP v1/v2, VRRP v2, GLBP</td>
</tr>
<tr>
<td>HA</td>
<td>Object tracking</td>
</tr>
<tr>
<td></td>
<td>Stateful restart / ISSU / SSO</td>
</tr>
<tr>
<td>Authentication</td>
<td>Clear text, MD5 (HSRP v2 only)</td>
</tr>
</tbody>
</table>
FHRPs with vPC / vPC+
Active/Active Mode

- All FHRP protocols in vPC / vPC+ environment operate in Active/Active mode
- No additional configuration required
- General VRRP / HSRP best practices still apply, except:
  - Since running in active/active mode, aggressive timers can be relaxed
  - No need to manipulate priorities / preemption on different devices to achieve load-balancing
**FHRPs in NX-OS**

Configuration and Management Highlights

- HSRP - use “hsrp” keyword instead of “standby”
- Config commands exist under HSRP/VRRP/GLBP sub-mode

```
switch(config)# int Ethernet 1/1
switch(config-if)# hsrp 10
switch(config-hsrp)# ?
     authentication     Authentication
     ip                 Enable HSRP IPv4 and set the virtual IP address
     mac-address        Virtual MAC address
     preempt            Overthrow lower priority Active routers
     priority           Priority level
     timers             Hello and hold timers
     track              Associates track object to HSRP group

... switch(config)# int Ethernet1/1
switch(config-if)# glbp 10
switch(config-glbp)# ?
     ip                 Set Virtual IP address

... switch# show hsrp brief
Interface     Grp Prio P State   Active addr   Standby addr   Group addr
Vlan100       100 120  P Active local  101.0.100.4    101.0.100.1
Vlan110       110 120  P Active local  101.0.110.4    101.0.110.1
```
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- **Routing Policy and Policy-based Routing (PBR)**
- Layer 3 High Availability and Fast Convergence
- Summary
Routing Policy in NX-OS

Highlights

- NX-OS makes extensive use of route-maps
- Route-maps are defined as in IOS
- **Redistribution requires a route-map to be configured**
- BGP regular expressions in policies undergo basic CPU hog check during configuration
Routing Policy options in NX-OS

Key Features

- Route-maps utilize:
  - MAC list
  - IPv4 and IPv6 Prefix List
  - BGP AS-Path Access List
  - Standard Community List
  - Expanded Community List
  - Access List (for PBR only)
  - Routing Protocol metric
  - VLAN ID
Route-maps and redistribution

Configuration Highlights

- Configure route map and use in redistribution
- Classless redistribution by default (no “subnets” option)

```
route-map STATIC_TO_IAMP permit 10
match interface Vlan122

router ospf 1
redistribute static route-map STATIC_TO_IAMP
```

- Double redistribution not allowed
BGP Community lists

- Standard and extended community lists supported
- BGP community is **always** configured and displayed in new format (aa:nn):

```
switch(config)# ip community-list expanded test2 permit "1:1"
```
```
switch# show ip community-list
Expanded Community List test
    permit "1:1"
```
Policy Based Routing

Highlights

- IPv4 and IPv6 support
- PBR allows routing redirection based on configured rules (i.e. route-maps)
- PBR is a function of the uRIB applying administratively defined rules for the final information to be written to hardware
- PBR is implemented in **hardware** on Nexus 7000 with ECMP support
- VRF-select option
- Forward referencing for route-maps used in policies
Policy Based Routing in NX-OS

Configuration

- Configure a route map to implement PBR:

```
switch(config)# feature pbr
switch(config)# route-map SELECT-PROVIDER permit 10
switch(config-route-map)# match ip address CUSTOMER-A
switch(config-route-map)# set ip next-hop 10.1.1.1
switch(config-route-map)# route-map SELECT-PROVIDER permit 20
switch(config-route-map)# match ip address CUSTOMER-B
switch(config-route-map)# set ip next-hop 10.2.2.2 10.2.2.3 load-share

switch(config)# interface ethernet 1/1
switch(config-if)# ip policy route-map SELECT-PROVIDER
```

**Note:** Packets that are denied by the route map, or packets for which no active next-hop can be found in the route map, will be forwarded through the normal destination-based routing process.
# NX-OS / IOS Routing Policy and PBR Differences

<table>
<thead>
<tr>
<th>Features</th>
<th>IOS</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classless redistribution</td>
<td>Yes – requires “subnets” option</td>
<td>Yes – no additional options required</td>
</tr>
<tr>
<td>Double redistribution</td>
<td>Allowed</td>
<td>Not-allowed</td>
</tr>
</tbody>
</table>
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
  - Layer 3 High Availability and Fast Convergence
- Summary
Layer 3 High Availability
NX-OS and Nexus switches

- Features and best practices:
  - Hardware High Availability
  - Software High Availability
  - Protocol Fast Convergence

More details: BRKARC-3471 – Cisco NX-OS Software Architecture

Building Highly Available Layer 3 Networks with NX-OS and Nexus 7000
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- **Layer 3 High Availability and Fast Convergence**
  - Software High Availability
  - Protocol Fast Convergence
- Summary
Software High Availability

Design Goal

- Non-Stop Forwarding (NSF) for all protocols and all address families:
  - Active supervisor component crash
  - Supervisor Switchover (SSO) (initiated by user or HA policy)
  - In-Service Software Upgrade (ISSU)
Software High Availability

- Protocols and services utilize HA mechanisms:
  - Non-Stop Routing (NSR) / Stateful Restart
  - Graceful Restart (GR)
  - Periodic Refresh (PR)

<table>
<thead>
<tr>
<th>Protocol</th>
<th>NSR</th>
<th>GR</th>
<th>PR</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF v2</td>
<td>✓</td>
<td>✓</td>
<td>N/A</td>
</tr>
<tr>
<td>OSPF v3</td>
<td>✓</td>
<td>✓</td>
<td>N/A</td>
</tr>
<tr>
<td>IS-IS</td>
<td>✓</td>
<td>✓</td>
<td>N/A</td>
</tr>
<tr>
<td>EIGRP</td>
<td>-</td>
<td>✓</td>
<td>N/A</td>
</tr>
<tr>
<td>BGP</td>
<td>-</td>
<td>✓</td>
<td>N/A</td>
</tr>
<tr>
<td>PIM / PIM6</td>
<td>-</td>
<td>-</td>
<td>✓</td>
</tr>
<tr>
<td>RIP</td>
<td>-</td>
<td>-</td>
<td>✓</td>
</tr>
<tr>
<td>FHRPs</td>
<td>✓</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Stateful Process Restart

PSS = Persistent Storage Service
PSS provides reliable persistent storage for the software components to ‘checkpoint’ their run-time state enabling stateful restart.

More details: BRKARC-3471 – Cisco NX-OS Software Architecture
Graceful Restart

- Controlled restart of operations and signaling between protocol neighbors to recover a control plane process.
- Network device notifies neighbors of its “GR” capability prior to event. That allows data to be forwarded during a restart operation and neighbor re-establishment.
- NX-OS IETF Graceful Restart is NOT compatible with Cisco NSF (OSPF, IS-IS).

<table>
<thead>
<tr>
<th>Protocol</th>
<th>NX-OS GR implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF v2</td>
<td>RFC 3623</td>
</tr>
<tr>
<td>OSPF v3</td>
<td>RFC 5187</td>
</tr>
<tr>
<td>IS-IS</td>
<td>RFC 3847</td>
</tr>
<tr>
<td>EIGRP</td>
<td>Cisco</td>
</tr>
<tr>
<td>BGP</td>
<td>RFC 4724</td>
</tr>
<tr>
<td>LDP</td>
<td>RFC 3478</td>
</tr>
</tbody>
</table>
Graceful Restart

Graceful Restart requires interaction with the neighbors to recover

OSPF has already restarted once in last 4 min. Let’s do a “Graceful Restart”
Layer 3 Process Fault Recovery Logic

Example: OSPF on Nexus 7000

- Monitor service process
- Process freezes/crashes
  - Process restarted recently?
    - Yes
    - Process restarted twice in 4 min?
      - Yes: Stateless Restart (use GR extensions)
      - No
    - No: Initiate Stateful Switchover (reload if single supervisor)
  - No: Stateful Restart
    - Process supports stateful restart?
      - Yes
      - No
Fault Recovery HA Policies for Layer 3 Protocols

- HA policies for some of the network protocols deployed in a typical Data Center. You can see the restart attempts and their types for single vs. dual supervisor

<table>
<thead>
<tr>
<th>L3 Protocol</th>
<th>Dual Supervisor</th>
<th>Single Supervisor</th>
</tr>
</thead>
</table>
| OSPFv2      | 1<sup>st</sup> – Stateful Restart  
              2<sup>nd</sup> – Graceful Restart (RFC)  
              3<sup>rd</sup> – Supervisor Switchover | 1<sup>st</sup> – Stateful Restart  
              2<sup>nd</sup> – Graceful Restart (RFC)  
              3<sup>rd</sup> – Reload the system |
| EIGRP       | 1<sup>st</sup> – Graceful Restart (RFC)  
              2<sup>nd</sup> – Cisco NSF  
              3<sup>rd</sup> – Supervisor Switchover | 1<sup>st</sup> – Graceful Restart (RFC)  
              2<sup>nd</sup> – Cisco NSF  
              3<sup>rd</sup> – Reload the system |
| IS-IS       | 1<sup>st</sup> – Stateful Restart  
              2<sup>nd</sup> – Graceful Restart (RFC)  
              3<sup>rd</sup> – Supervisor Switchover | 1<sup>st</sup> – Stateful Restart  
              2<sup>nd</sup> – Graceful Restart (RFC)  
              3<sup>rd</sup> – Reload the system |
| BGP         | 1<sup>st</sup> – Graceful Restart (RFC)  
              3<sup>rd</sup> – Supervisor Switchover | 1<sup>st</sup> – Graceful Restart (RFC)  
              3<sup>rd</sup> – Reload the system |
| RIP         | 1<sup>st</sup> – Stateless Restart  
              3<sup>rd</sup> – Supervisor Switchover | 1<sup>st</sup> – Stateless Restart  
              3<sup>rd</sup> – Reload the system |
| PIM         | 1<sup>st</sup> – Stateless Restart  
              3<sup>rd</sup> – Supervisor Switchover | 1<sup>st</sup> – Stateless Restart  
              3<sup>rd</sup> – Reload the system |
| HSRP, VRRP, GLBP | 1<sup>st</sup> – Stateful Restart  
                      2<sup>nd</sup> – Stateful Restart  
                      3<sup>rd</sup> – Stateful Restart  
                      4<sup>th</sup> – Supervisor Switchover | 1<sup>st</sup> – Stateful Restart  
                      2<sup>nd</sup> – Stateful Restart  
                      3<sup>rd</sup> – Stateful Restart  
                      4<sup>th</sup> – Reload the system |
Software High Availability
Best Practices and Recommendations

- Use routing protocols such as OSPF and IS-IS, which support stateful process restart
- Make sure Graceful Restart is always enabled on both peers regardless of stateful restart capability:
  - In heterogeneous network, make sure that peer supports standards-based GR (except EIGRP)
  - **Note**: Supervisor OIR will always trigger graceful restart!

```
!NX-OS
router ospf 1
  graceful-restart

!IOS
router ospf 1
  nsf ietf
```
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- **Layer 3 High Availability and Fast Convergence**
  - Software High Availability
  - Protocol Fast Convergence
- Summary
Protocol Fast Convergence

What is it?

- Collection of features to improve Layer 3 network convergence after various failures:
  - Fast Failure Detection (FFD) – how quickly can I detect a forwarding failure resulting in eventual neighbor down event
  - Fast Failure Reaction (FFR) – once I know about it, how quickly can I react (notify peers and recalculate SPF)
Is FFD tuning needed beyond defaults?

- FFD tuning is needed when:
  - Intermediate L2 hop over L3 link
  - Concerns over protocol software failures
  - Concerns over unidirectional failures on point-to-point physical L3 links

- FFD tuning may not be needed when:
  - Point-to-point physical L3 links with no concerns over unidirectional failures
  - Enough software redundancy to account for protocol software failures
  - FHRPs are running in active-active mode (VPC)
Fast Failure Detection tuning

Protocol Hello / Dead Timers

- Layer 3 protocols (FHRPs, BGP, EIGRP, OSPF etc) send Hellos / Keepalives at defined rate to:
  - Maintain adjacencies (pass protocol specific info)
  - **Check neighbour reachability and detect failure**

- Tuning down Hello / Keepalive and Dead / Hold timers **may** speed up failure detection, but it is not recommended:
  - Each interface may have 2-3+ protocols establishing adjacency (e.g. HSRP, PIM, OSPF on SVI)
  - Increased supervisor CPU utilization and waste of link bandwidth
  - Configuration complexity
  - **NX-OS does not support non-disruptive software upgrade and supervisor switchover with tuned down protocol timers**
What is BFD?

- **Lightweight** hello protocol over multiple transport protocols: 
  - IPv4, IPv6, MPLS

- Used for fast (often sub-second) communication failure detection

- Single, common & standardized mechanism

- Any interested application (OSPF, BGP, HSRP, TE-FRR etc.) registers with BFD and is notified as soon as BFD recognizes a neighbor loss

- All registered applications benefit from uniform failure detection

- UDP port 3784 / 3785 (for echo)

- **RFC 5880**
Fast Failure Detection with BFD

- Bidirectional Forwarding Detection (BFD) – recommended fast failure detection mechanism
- Advantages of BFD over lowered hello / dead timers:
  - Reduced control plane load and link bandwidth usage
  - Sub-second failure detection
  - In-flight timer negotiation
  - Stateful restart, SSO and ISSU
  - Distributed implementation – I/O module CPU transmits / receives BFD packets
BFD in NX-OS and Nexus switches

- Ver 1 only, single-hop only
- Packets sent with DSCP CS6 / CoS 6
- **Nexus 3000** with centralized implementation from NX-OS 5.0(3)U2(2) onwards
- **Nexus 7000** with distributed implementation from NX-OS 5.0(2a) onwards
  - All F2 and M-series I/O modules
  - F1 modules don’t support L3 ports, so they can’t host BFD sessions.
  - FEX ports do not support BFD
BFD Clients and Interface Types

Nexus 7000

- Supported BFD clients (IPv4 only):
  - OSPFv2
  - EIGRP
  - IS-IS
  - PIM
  - BGP
  - HSRP
  - VRRP
  - Static routes
  - MPLS TE FRR

- Supported Layer 3 interface types:
  - Switched Virtual Interface (SVI)
  - Physical Port and sub-interface
  - Port Channel (per-link and logical (1 session / port-channel)) and sub-interface

Note: check configuration guides and release notes for latest supported configuration
BFD Clients and Interface Types

Nexus 3000

- Minimum interval: 50 msec x 3
- Supported BFD clients (IPv4 only):
  - OSPFv2
  - BGP
- Supported Layer 3 interface types:
  - Switched Virtual Interface (SVI)
  - Physical Port and sub-interface
  - Port Channel (logical only – 1 session / port-channel) and sub-interface

*Note: check configuration guides and release notes for latest supported configuration*
BFD Architecture in NX-OS (Nexus 7000)

Distributed Processing

- **SUP-BFD** - BFD process running on Supervisor Engine
  - Interfaces with LC-BFD processes
  - **Interfaces with BFD clients** (OSPF, BGP etc)

- **LC-BFD** – BFD process running on CPU of each I/O module
  - Communicates with SUP-BFD process
  - **Generates BFD hellos** (echo and async)
  - **Receives BFD hellos** (async)

- Support for stateful process restart, SSO and ISSU
- BFD is VRF and VDC aware

---

**Diagram: Layer 3 High Availability and Fast Convergence**

- **Supervisor Engine**
  - OSPF, HSRP, PIM, BGP, IS-IS, Etc.
  - **SUP-BFD**
  - **EOBC**

- **LC-BFD**
  - Module Inband
  - Hardware
  - I/O Module

---

BRKARC-3472 © 2012 Cisco and/or its affiliates. All rights reserved. Cisco Public
Protocol Fast Convergence
Key Best Practices and Recommendations

1) Decide whether FFD tuning is needed. If not – do not use it!

2) If FFD tuning is needed, use BFD for all protocols:
   a) Use BFD echo (default) whenever possible
   b) On Layer 3 port-channels, use per-link mode and prefer that over echo
   c) On L3 interfaces running BFD configure `no ip redirects`

3) If BFD can’t be used:
   a) Prior to NX-OS 5.2(1), lowering hello / dead timers is not supported under any circumstances
   b) From NX-OS 5.2(1), lowering hello / dead timers is supported without SSO / ISSU (target small scale environments)
   c) From NX-OS 5.2(3a), lowering HSRP and GLBP timers is supported with SSO / ISSU using extended hold timer feature

4) Keep consistent FFR configuration on all network devices

Details on tested protocol timers and scalability: Cisco Nexus 7000 Series NX-OS Verified Scalability Guide
Agenda

- Layer 3 requirements
- NX-OS Layer 3 Software Architecture
- Unicast Routing Protocols
- Multicast Protocols
- First Hop Redundancy Protocols
- Routing Policy and Policy-based Routing (PBR)
- Layer 3 High Availability and Fast Convergence
- Summary
Summary

- NX-OS is a modular operating system that decouples control and data planes
- NX-OS Layer 3 CLI has numerous usability enhancements
- NX-OS supports a variety of IPv4 and IPv6 routing, multicast and gateway redundancy protocols to meet data center and campus core needs
- NX-OS and Nexus switches provide numerous features for Layer 3 high availability and protocol fast convergence
- VDCs, VRFs and MPLS VPN deliver a strong Layer 3 virtualization feature set on NX-OS
References & Additional information

- Nexus 7000 Configuration Guides
- Nexus 5000 Configuration Guides
- Nexus 3000 Configuration Guides
- Nexus 7000 Release Notes
- Nexus 5000 Release Notes
- Nexus 3000 Release Notes
- Cisco Nexus 7000 NX-OS / IOS Comparison Technotes - DocWiki
- Building Highly Available Layer 3 Networks with Cisco NX-OS Software and Cisco Nexus 7000 Switches

Follow us on Twitter @CiscoNexus7000
Acronyms

- ACL – Access Control List
- API – Application Programming Interface
- BFD – Bidirectional Forwarding Detection
- BGP – Border Gateway Protocol
- CNH – Connected Next Hop
- DCI – Data Center Interconnect
- DSCP – Differentiated Services Code Point
- EIGRP – Enhanced Interior Gateway Routing Protocol
- FEX – Fabric Extender
- FFD – Fast Failure Detection
- FFR – Fast Failure Reaction
- GR – Graceful Restart
- MPLS – Multi-Protocol Label Switching
- NSF – Non-Stop Forwarding
- NSR – Non-Stop Routing
- OSPF – Open Shortest Path First
- OTV – Overlay Transport Virtualization
- PIC – Protocol Independent Convergence
- QoS – Quality of Service
- RNH – Recursive Next Hop
- RPF – Reverse Path Forwarding
- SSO – Stateful SwitchOver
- URIB – Unified Routing Information Base
- UFIB – Unified Forwarding Information Base
- VDC – Virtual Device Context
- VoQ – Virtual Output Queue
- VPC – Virtual Port-Channel
- VPN – Virtual Private Network
- VRF – Virtual Routing and Forwarding
## Related Cisco Live 2012 Events
### Technical Breakout Sessions

<table>
<thead>
<tr>
<th>Session-ID</th>
<th>Session Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BRKARC-3470</td>
<td>Cisco <strong>Nexus 7000</strong> Switch Architecture</td>
</tr>
<tr>
<td>BRKARC-3452</td>
<td>Cisco <strong>Nexus 5000/5500</strong> and <strong>2000</strong> Switch Architecture</td>
</tr>
<tr>
<td>BRKARC-3471</td>
<td>Cisco <strong>NX-OS</strong> Software Architecture</td>
</tr>
<tr>
<td>BRKDCT-2121</td>
<td>Virtual Device Context (VDC) Design and Implementation Considerations with <strong>Nexus 7000</strong></td>
</tr>
<tr>
<td>TECRST-3190</td>
<td>Advanced IP <strong>Routing</strong> Fast Convergence</td>
</tr>
<tr>
<td>BRKIPM-3062</td>
<td><strong>Nexus Multicast</strong> Design Best Practices</td>
</tr>
<tr>
<td>TECDCT-3297</td>
<td>Operating and Deploying <strong>NX-OS Nexus</strong> Devices in the Network Infrastructure</td>
</tr>
<tr>
<td>TECVIR-2003</td>
<td>Enterprise Network Virtualization</td>
</tr>
</tbody>
</table>
## Related Cisco Live 2012 Events

### Labs

<table>
<thead>
<tr>
<th>Session-ID</th>
<th>Session Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTRCRT-5205</td>
<td>Configuring Nexus 7000 Virtualization</td>
</tr>
</tbody>
</table>
Complete Your Online Session Evaluation

- Give us your feedback and you could win fabulous prizes. Winners announced daily.
- Receive 20 Passport points for each session evaluation you complete.
- Complete your session evaluation online now (open a browser through our wireless network to access our portal) or visit one of the Internet stations throughout the Convention Center.

Don’t forget to activate your Cisco Live Virtual account for access to all session material, communities, and on-demand and live activities throughout the year. Activate your account at the Cisco booth in the World of Solutions or visit www.ciscolive.com.
Final Thoughts

- Get hands-on experience with the Walk-in Labs located in World of Solutions, booth 1042
- Come see demos of many key solutions and products in the main Cisco booth 2924
- Visit www.ciscoLive365.com after the event for updated PDFs, on-demand session videos, networking, and more!
- Follow Cisco Live! using social media:
  - Facebook: https://www.facebook.com/ciscoliveus
  - Twitter: https://twitter.com/#!/CiscoLive
  - LinkedIn Group: http://linkd.in/CiscoLI
## Layer 3 System Comparison

### Nexus 5000 / 7000

<table>
<thead>
<tr>
<th>L3 Functional Areas</th>
<th>Nexus 7000 with M-series Modules</th>
<th>Nexus 5500 + L3 Card / Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>Redundant Route Processors</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Control Plane Protection</td>
<td>Extensive CoPP Granularity</td>
<td>Extensive CoPP Granularity</td>
</tr>
<tr>
<td>Distributed Processing</td>
<td>Yes - Distributed Multicast replication and BFD</td>
<td>No</td>
</tr>
<tr>
<td>FEX Routed Port</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>ISSU</td>
<td>Yes – any L2 or L3 configuration</td>
<td>Yes - L2 STP edge ports only, no L3</td>
</tr>
<tr>
<td>Stateful Process Restart / NSR</td>
<td>Yes – OSPFv2, OSPF v3, IS-IS, FHRPs</td>
<td>Yes – FHRPs</td>
</tr>
<tr>
<td>L3 over VPC</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

**CY11 Roadmap. Not available in existing release**
# Layer 3 Functional Comparison

## Nexus 5000 / 7000

<table>
<thead>
<tr>
<th>L3 Functional Areas</th>
<th>Nexus 7000 with M-series Modules</th>
<th>Nexus 5500 + L3 Card / Module</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Routing Protocols</strong></td>
<td>OSPFv2, EIGRP, RIPv2, BGP, IS-IS, PIM, IGMP</td>
<td>Static, OSPFv2, RIPv2, PIM, IGMP, BGP, EIGRP</td>
</tr>
<tr>
<td><strong>IPv6</strong></td>
<td>Dual Stack, OSPFv3, EIGRP, BGP, HSRPv6</td>
<td>For Management only</td>
</tr>
<tr>
<td><strong>L3 Segmentation</strong></td>
<td>VRF Lite, VRF-aware features, VRF Import/Export, MPLS VPNs</td>
<td>VRF-Lite, VRF-aware features</td>
</tr>
<tr>
<td><strong>High Availability</strong></td>
<td>NSF/SSO/ISSU, Graceful Restart, NSR</td>
<td>ISSU – L2 STP edge ports only Graceful Restart – helper only</td>
</tr>
<tr>
<td><strong>Fast Convergence</strong></td>
<td>BFD, BGP NHT and PIC Core, MPLS TE FRR</td>
<td>No</td>
</tr>
<tr>
<td><strong>Monitoring</strong></td>
<td>Flexible Netflow, Sampled Netflow, ERSPAN</td>
<td>ERSPAN</td>
</tr>
<tr>
<td><strong>Layer 2 DCI</strong></td>
<td>Overlay Transport Virtualization (OTV)</td>
<td>No</td>
</tr>
<tr>
<td><strong>Traffic Steering</strong></td>
<td>Policy-Based Routing, VRF Select, WCCPv2, Static Multicast MAC, MPLS Traffic Engineering</td>
<td>No</td>
</tr>
<tr>
<td><strong>Tunneling / Mobility</strong></td>
<td>Unicast over GRE, LISP</td>
<td>No</td>
</tr>
</tbody>
</table>

*Assumes all licenses present*
## Layer 3 Scale Comparison

Nexus 5000 / 7000 hardware and software

<table>
<thead>
<tr>
<th>L3 Functional Areas</th>
<th>Nexus 7000 with M1-XL/M2-XL Modules</th>
<th>Nexus 5500 + L3 Card / Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>L3 Interfaces</td>
<td>4K</td>
<td>4K</td>
</tr>
<tr>
<td>IPv4 Unicast FIB</td>
<td>900K</td>
<td>8K / 16K (Ver 1 / Ver 2 module)</td>
</tr>
<tr>
<td>IPv4 Multicast FIB</td>
<td>32K</td>
<td>4K / 8K (Ver 1 / Ver 2 module)</td>
</tr>
<tr>
<td>L3 ECMP</td>
<td>16 Way</td>
<td>16 Way</td>
</tr>
<tr>
<td>ARP</td>
<td>50K</td>
<td>8K / 16K (Ver 1 / Ver 2 module)</td>
</tr>
<tr>
<td>Routing Adjacency</td>
<td>400K verified, 1M maximum</td>
<td>8K maximum</td>
</tr>
<tr>
<td>L3 ACL entries</td>
<td>128K</td>
<td>Ingress: 2K, Egress: 1K</td>
</tr>
<tr>
<td>Segmentation</td>
<td>1K VRFs</td>
<td>1K VRFs</td>
</tr>
<tr>
<td>FEX Scale with L3 features enabled</td>
<td>32</td>
<td>8</td>
</tr>
</tbody>
</table>

- **Hardware capabilities listed, numbers are subject to vary depending on software releases**
- **Assumes all licenses present**
Switching Platform Comparison

- More information on platform differences can be found here:
Nexus 7000 NX-OS Software Scalability

- Check Nexus 7000 NX-OS Verified Scalability guide for latest information

- These are single dimensional numbers listed for guidance only
- All numbers are system wide and can be within a single VDC or spread across VDCs
Nexus 7000 M1/M2 Forwarding Engines

- Hardware forwarding engine(s) integrated on every M-series I/O module
- IPv4 and IPv6 multicast support (SM, SSM, BiDir)
- Ingress and egress NetFlow (full and sampled)
- RACL/VACL/PACLs
- QoS remarking and policing policies
- Policy-based routing (PBR)
- Unicast RPF check and IP source guard

<table>
<thead>
<tr>
<th>Hardware Table</th>
<th>M1 Modules</th>
<th>M1/M2-XL Modules without XL License</th>
<th>M1/M2-XL Modules with XL License</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIB TCAM</td>
<td>128K</td>
<td>128K</td>
<td>900K</td>
</tr>
<tr>
<td>Classification TCAM</td>
<td>64K</td>
<td>64K</td>
<td>128K</td>
</tr>
<tr>
<td>(ACL/QoS)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAC Address Table</td>
<td>128K</td>
<td>128K</td>
<td>128K</td>
</tr>
<tr>
<td>NetFlow Table</td>
<td>512K</td>
<td>512K</td>
<td>512K</td>
</tr>
</tbody>
</table>
Nexus 5500 with Layer 3 support

- Layer 3 Forwarding Engine connects the the X-Bar via two UPC (160GBps)
- Optional two stage forwarding
- Stage 1 – Ingress UPC forwards to destination MAC address
- If MAC address is external packet directly forwarded to egress port across X-Bar fabric (single stage only)
- If MAC address is the router MAC address (e.g. HSRP vmac) packet is forwarded across fabric to layer 3 Engine
- Stage 2 – Layer 3 lookup occurs and packet is forwarded to egress port across X-Bar fabric
- Only ‘routed’ packets are forwarded through the Layer 3 engine
Nexus 5500 with Layer 3 support

- Layer 3 Forwarding Engine connects the the X-Bar via two UPC Gen-2 using a 16 x 10G internal port-channel (iPC)
- Traffic is load shared across the 16 fabric connections (iPorts)
- Recommendation configure L2/L3/L4 port channel hashing (global switch parameter)

```
L3-5548-1# sh port-channel load-balance
Port Channel Load-Balancing Configuration:
 System: source-dest-port

Port Channel Load-Balancing Addresses Used Per-Protocol:
  Non-IP: source-dest-mac
  IP: source-dest-port source-dest-ip source-dest-mac
```

```
L3-5548-1# sh mod
Mod Ports  Module-Type                      Model
<snip>
3    0     O2 Daughter Card with L3 ASIC     N55-D160L3
```

```
L3-5548-1# sh int port-channel 127
port-channel127 is up
```
Nexus 5500 Series
Layer 3 QoS Configuration

- Internal QoS information determined by ingress Carmel (UPC) ASIC is ‘not’ passed to the Lithium L3 ASIC
- Need to mark all routed traffic with a dot1p CoS value used to:
  - Queue traffic to and from the Lithium L3 ASIC
  - Restore qos-group for egress forwarding
- **Mandatory** to setup CoS for the frame in the network-qos policy, one-to-one mapping between a qos-group and CoS value
- Classification can be applied to *physical interfaces* (L2 or L3, including L3 port-channels) not to SVIs

On initial ingress packet QoS matched and packet is associated with a qos-group for queuing and policy enforcement

If traffic is congested on ingress to L3 ASIC it is queued on ingress UPC ASIC

Routed packet is queued on egress from Lithium based on dot1p

Packet qos-group is not passed to Lithium, leverages CoS dot1p

```
class-map type network-qos nqcm-grp2
  match qos-group 2
policy-map type network-qos nqpm-grps
  class type network-qos nqcm-grp2
    set cos 4
  class type network-qos nqcm-grp4
    set cos 2
```
Routing and vPC

- Don’t use L2 port channel to attach routers to a vPC domain
- If both routed and bridged traffic is required:
  - Individual L3 links for routed traffic
  - L2 port-channel (vPC) for bridged traffic

More details: BRKDCT-2048 - Deploying Virtual Port Channel in NX-OS
vPC Layer-3 Routing – Supported Topologies

- Dynamic routing between external L3 devices and vPC peers over L3 routed interfaces is supported

- Dynamic routing between vPC peers over point-to-point VLAN across vPC peer-link is supported

- Dynamic routing between external L3 devices and vPC peers over non-vPC VLANs is supported
vPC Layer-3 Routing – Supported Topologies

- Dynamic routing between external L3 devices across vPC is supported
  - No dynamic routing with vPC peer devices (transit only)
vPC Layer-3 Routing – Supported Topologies

- When routing to vPC peer devices over vPC or vPC VLAN, configure static routes to FHRP address.
  - If the static routes point to the hardware mac-address, ensure “vPC peer-gateway” is enabled.
vPC Layer-3 Routing – Not Supported Topologies

- Dynamic Routing between External L3 Devices and vPC Peers over vPC or vPC VLANs is not supported.
vPC Layer-3 Routing – Not Supported Topologies

- Dynamic routing between vPC peer devices over vPC interconnection is not supported
BGP Prefix Independent Convergence

Why PIC Core?

1. Core link or node goes down
2. IGP notices failure, computes new paths to PE1/PE2
3. On PE3, IGP notifies BGP that a path to a next-hop has changed
4. BGP identifies affected paths
5. BGP updates RIB/FIB for all affected BGP prefixes

More details: Cisco Live London 2012 - BRKIPM-2265 - Deploying BGP Fast Convergence
BGP Prefix Independent Convergence

PIC Core

- From NX-OS 5.2, RIB/FIB/BGP support **PIC Core**:
  - Support for AFs: IPv4, VPNv4, VPNv6 (6VPE), 6PE
  - Virtual Objects (VOBJs) in FIB track Recursive Next Hops (RNH) and associated VPN labels for BGP prefixes
  - **IGP reachability** change to RNH results in a **single update** of VOBJ for all BGP prefixes
  - **Result: significant convergence improvements**

Note: Per-VRF label allocation is required in MPLS VPN scenario
BGP Prefix Independent Convergence
Without PIC Core

- **Prfx1**: `100.0.0.0/24`
  - via 10.1.1.1 label 100 1
  - via 20.1.1.1 label 100 2

- **Prfx2**: `102.0.0.0/24`

- **Prfx3**: `103.0.0.0/24`

- **Prfx4**: `104.0.0.0/24`

- **Prfx0**: `1.1.1.1/32`
  - via 10.1.1.1
  - via 20.1.1.1

**IGP**

**I/O Module**

**FIB**
BGP Prefix Independent Convergence

With PIC Core

BGP

Prfx4: 104.0.0.0/24
Prfx3: 103.0.0.0/24
Prfx2: 102.0.0.0/24
via 1.1.1.1 label 101
Prfx1: 100.0.0.0/24
via 1.1.1.1 label 100

IGP

Prfx0: 1.1.1.1/32
via 10.1.1.1
via 20.1.1.1

RIB

Prfx: 100.0.0.0/24
via 1.1.1.1 label 100

I/O Module

Prfx0
via ...

Prfx1
via ...

Prfx2
via ...

Prfx3
via ...

VOBJ

[RNH, label]

FIB

Prfx0
via ...

Prfx0
via ...
VDC Resource Allocation – URIB/MRIB

- Certain resources can be allocated and limited to a given VDC:
  
m4route-mem Set ipv4 route memory limits
m6route-mem Set ipv6 route memory limits
module-type Controls which type of modules are allowed in this vdc
monitor-session Monitor local/erspan-source session
monitor-session-erspan-dst Monitor erspan destination session
port-channel Set port-channel limits
u4route-mem Set ipv4 route memory limits
u6route-mem Set ipv6 route memory limits
vlan Set VLAN limits
vrf Set vrf resource limits

- How much RAM do I allocate for my routing tables?
  
  • Routing table memory limits are in MB. For an idea of MB to routes you can use the command “show routing ipv4|ipv6 memory estimate routes <1000-1000000> next-hops <1-16>”

  • u4route-mem and u6route-mem limits are only applied after a switchover or reload – they are not hot updates.
VDC Resource Allocation – URIB/MRIB

- Default allocations allow for majority of deployment scenarios
  - 8MB of memory allows for approx 6000 routes with 16 next hops
  - Can be modified by using VDC templates as needed

```
N7K1-VDC1# show vdc N7K1-VDC2 resource

<table>
<thead>
<tr>
<th>Resource</th>
<th>Min</th>
<th>Max</th>
<th>Used</th>
<th>Unused</th>
<th>Avail</th>
</tr>
</thead>
<tbody>
<tr>
<td>vlan</td>
<td>16</td>
<td>4094</td>
<td>35</td>
<td>0</td>
<td>4059</td>
</tr>
<tr>
<td>monitor-session</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>monitor-session-erspan-dst</td>
<td>0</td>
<td>23</td>
<td>0</td>
<td>0</td>
<td>23</td>
</tr>
<tr>
<td>vrf</td>
<td>2</td>
<td>4096</td>
<td>2</td>
<td>0</td>
<td>4086</td>
</tr>
<tr>
<td>port-channel</td>
<td>0</td>
<td>768</td>
<td>0</td>
<td>0</td>
<td>752</td>
</tr>
<tr>
<td>u4route-mem</td>
<td>8</td>
<td>8</td>
<td>1</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>u6route-mem</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>m4route-mem</td>
<td>8</td>
<td>8</td>
<td>1</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>m6route-mem</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>
```