What You Make Possible
Cisco Nexus 5000/5500 and 2000 Switch Architecture
BRKARC-3452
Session Goal

- This session presents an in-depth study of the architecture of the Nexus 5000/5550 family of Data Center switches and the Nexus 2000 Fabric Extender. Topics include internal architecture of the Nexus 5000, 5500 and 2000, the architecture of fabric and port extenders as implemented in the Nexus 2000 and Adapter FEX, Unified I/O, and 10G cut-thru Layer 2 and Layer 3 Ethernet. This year's content will include more focus on the QoS, SPAN, port channels and ACL's HW/SW implementation. This session is designed for network engineers involved in network switching design and Data Center architecture.

- Related sessions:
  
  BRKARC-3470 - Cisco Nexus 7000 Switch Architecture
  BRKCRS-3144 - Troubleshooting Cisco Nexus 7000 Series Switches
  BRKCRS-3145 - Troubleshooting Cisco Nexus 5000 / 2000 Series Switches
  BRKARC-3472 - NX-OS Routing & Layer 3 Switching
  BRKDCT-2023 - Evolution of the Data Centre Access Architecture *
  BRKSAN-2047 - FCoE Design, Operations and Management Best Practices *

* This session is focusing on the Hardware Architecture of the Nexus 5000, 5500 and 2000, please see the DC Access and FCoE design sessions for a detailed discussion of the best practices design options for N2K/N5K and FCoE
Nexus 5000/5500 and 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Layer 3 Forwarding
- Nexus 2000 Architecture
  - FEXLink Architecture
- Nexus 5000/5500
  - SPAN & ERSPAN
  - Multicast
  - Port Channels
  - QoS
Nexus 5000/5500 and 2000 Architecture
Virtualized Data Center Access

Generation 1 - 5000

Nexus 5010 & Nexus 5020
- 20 or 40 Fixed Ports 10G/FCoE/IEEE DCB
- 1/2/4/8G FC Expansion Module Ports
- Line-rate, Non-blocking 10G
- 1 or 2 Expansion Module Slots

Generation 2 - 5500

Nexus 5548UP & 5596UP
- 32/48 Fixed Ports – SFP+ 1/10G Ethernet or 1/2/4/8 FC
- 48 Fixed 10GBaseT – RJ45
- Line-rate, Non-blocking 10G FCoE/IEEE DCB
- 1/3 Expansion Module Slot
- IEEE 1588, FabricPath & Layer 3 Capable

Generation 1, 2 & 3 Nexus 2000

Nexus 2000 Fabric Extender
- 48 Fixed Ports 100M/1G Ethernet (1000 BASE-T)
- 32 Fixed ports 1G/10G/FCoE/IEEE DCB
- 4-8 Fixed Port 10G Uplink
- Distributed Virtual Line Card

Nexus 5596T
- Generation 1 - 5000
- Generation 2 - 5500
- Generation 1, 2 & 3 Nexus 2000

Cisco Public
BRKARC-3452
© 2012 Cisco and/or its affiliates. All rights reserved.
Nexus 5500 Hardware

Nexus 5548 (5548P & 5548UP)

- 32 x Fixed Unified Ports 1/10 GE or 1/2/4/8 FC
- Fabric Interconnect Not Active on Nexus
- Out of Band Mgmt 10/100/1000
- USB Flash
- Console
- Fan Module
- Expansion Module
- Fan Module
- Power Entry
- Power Entry
- N + N Redundant FANs
- N + N Power Supplies
Nexus 5500 Hardware
Nexus 5596UP

- 3 Expansion Modules
- 48 x Fixed Unified Ports 1/10 GE or 1/2/4/8 FC
- Fabric Interconnect Not Active on Nexus
- Out of Band Mgmt 10/100/1000
- Console
- USB Flash
- Power Supply
- N + N Power Supplies
- Fan Module
- N + N Redundant FANS
Nexus 5500 Hardware
Nexus 5500 Expansion Modules

- Nexus 5500 expansion slots
  - Expansion Modules are hot swappable (Future support for L3 OIR)
  - Contain forwarding ASIC (UPC-2)
Nexus 5500 Hardware

Nexus 5500 Reversible Air Flow and DC Power Supplies

- Nexus 2000, 5548UP and 5596UP will support reversible airflow (new PS and fans)
- Nexus 2000, 5548UP and 5596UP will support DC power supplies (not concurrent with reversible airflow)
- Note: 5548UP and 5596UP **ONLY**, not 5010/5020/5548P

<table>
<thead>
<tr>
<th></th>
<th>Nexus 2000</th>
<th>Hardware Availability</th>
<th>Nexus 5000</th>
<th>Hardware Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Front-to-Back Airflow, AC Power</strong></td>
<td>Nexus 2148T Nexus 2200 Series</td>
<td>Today</td>
<td>Nexus 5010/5020 Nexus 5548P/5548UP/5596UP</td>
<td>Today</td>
</tr>
<tr>
<td><strong>Back-to-Front Airflow, AC Power</strong></td>
<td>Nexus 2200 Series</td>
<td>Today</td>
<td>Nexus 5548UP/5596UP</td>
<td>Today</td>
</tr>
<tr>
<td><strong>Front-to-Back Airflow, DC Power</strong></td>
<td>Nexus 2200 Series</td>
<td>Today</td>
<td>Nexus 5548UP/5596UP</td>
<td>Today</td>
</tr>
<tr>
<td><strong>Back-to-Front Airflow, DC Power</strong></td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Nexus 5500 Hardware Overview

Data and Control Plane Elements

Expansion Module

Unified Crossbar Fabric
Gen 2

Gen 2 UPC

Gen 2 UPC

Gen 2 UPC

Gen 2 UPC

CPU Intel Jasper Forest

South Bridge

Flash

Memory

NVRAM

Serial

Console

PCIe x4

PEX 8525
4 port PCIe Switch

PCIe x8

10 Gig

12 Gig

L1

L2

Mgmt 0

PCIe Dual Gig
0 1

PCIe Dual Gig
0 1

PCIe Dual Gig
0 1

DRAM DDR3

Flash

Memory

NVRAM

Serial

CPU Intel Jasper Forest

South Bridge

Flash

Memory

NVRAM

Serial

Console
Nexus 5000/5500 Hardware Overview

Data Plane Elements – Distributed Forwarding

- Nexus 5000/5500 use a distributed forwarding architecture
- Unified Port Controller (UPC) ASIC interconnected by a single stage Unified Crossbar Fabric (UCF)
- Unified Port Controllers provide distributed packet forwarding capabilities
- All port to port traffic passes through the UCF (Fabric)
- Cisco Nexus 5020: Layer 2 hardware forwarding at 1.04 Tbps or 773.8 million packets per second (mpps)
- Cisco Nexus 5596: Layer 2 hardware forwarding at 1.92 Tbps or 1428 mpps
Nexus 5000/5500 Hardware Overview

Data Plane Elements - Unified Crossbar Fabric

- **Nexus 5000 (Gen-1)**
  - 58-port packet based crossbar and scheduler
  - Three unicast and one multicast crosspoint per egress port

- **Nexus 5550 (Gen-2)**
  - 100-port packet based crossbar and new schedulers
  - 4 crosspoints per egress port dynamically configurable between multicast and unicast traffic

- **Central tightly coupled scheduler**
  - Request, propose, accept, grant, and acknowledge semantics
  - Packet enhanced iSLIP scheduler
  - Distinct unicast and multicast schedulers (see slides later for differences in Gen-1 vs. Gen-2 multicast schedulers)
  - Eight classes of service within the Fabric
Nexus 5500 Hardware Overview
Data Plane Elements - Unified Port Controller (Gen 2)

- Each UPC supports eight ports and contains,
- Multimode Media access controllers (MAC)
  - Support 1/10 G Ethernet and 1/2/4/8 G Fibre Channel
  - All MAC/PHY functions supported on the UPC (5548UP and 5596UP)
- Packet buffering and queuing
  - 640 KB of buffering per port
- Forwarding controller
  - Ethernet (Layer 2 and FabricPath) and Fibre Channel Forwarding and Policy (L2/L3/L4 + all FC zoning)
Nexus 5500 Hardware Overview
Control Plane Elements – Nexus 5500

- **CPU** - 1.7 GHz Intel Jasper Forest (Dual Core)
- **DRAM** - 8 GB of DDR3 in two DIMM slots
- **Program Store** - 2 GB of eUSB flash for base system storage and partitioned to store image, configuration, log.
- **Boot/BIOS Flash** - 8 MB to store upgradable and golden version of (Bios + bootloader) image
- **On-Board Fault Log (OBFL)** - 64 MB of flash to store hardware related fault and reset reason
- **NVRAM** - 6 MB of SRAM to store Syslog and licensing information
- **Management Interfaces**
  - **RS-232 console port:** console0
  - **10/100/1000BASE-T:** mgmt0 partitioned from inband VLANs
Nexus 5000/5500 Hardware Overview

Control Plane Elements - CoPP

- In-band traffic is identified by the UPC and punted to the CPU via two dedicated UPC interfaces, 5/0 and 5/1, which are in turn connected to eth3 and eth4 interfaces in the CPU complex
  - Receive – Dest Mac == Switch Mac
  - Copy – Copy of the packet needed by SUP
  - Exception - Needs exception handling
  - Redirected – Snooped or needed by the SUP
  - Glean – NextHop Mac not available
  - Multicast
  - Broadcast

- Eth3 handles Rx and Tx of **low** priority control pkts
  IGMP, CDP, TCP/UDP/IP/ARP (for management purpose only)

- Eth4 handles Rx and Tx of **high** priority control pkts
  STP, LACP, DCBX, FC and FCoE control frames (FC packets come to Switch CPU as FCoE packets)
Nexus 5000/5500 Hardware Overview
Control Plane Elements - CoPP

- CPU queuing structure provides strict protection and prioritization of inbound traffic
- Each of the two in-band ports has 8 queues and traffic is scheduled for those queues based on control plane priority (traffic CoS value)
- Prioritization of traffic between queues on each in-band interface
  - CLASS 7 is configured for strict priority scheduling (e.g. BPDU)
  - CLASS 6 is configured for DRR scheduling with 50% weight
  - Default classes (0 to 5) are configured for DRR scheduling with 10% weight
- Additionally each of the two in-band interfaces has a priority service order from the CPU
  - Eth 4 interface has high priority to service packets (no interrupt moderation)
  - Eth3 interface has low priority (interrupt moderation)
Nexus 5500 Hardware Overview
Control Plane Elements - CoPP

- On **Nexus 5500** an additional level of control invoked via polices on UPC-2
- Software programs a number of egress polices on the UPC-2 to avoid overwhelming the CPU (partial list)
  - STP: 20 Mbps
  - LACP: 1 Mbps
  - DCX: 2 Mbps
  - Satellite Discovery protocol: 2 Mbps
  - IGMP: 1 Mbps
  - DHCP: 1 Mbps
  - ... 
  - CLI exposed to tune CoPP exposed in 5.1(3)N1
Nexus 5500 Hardware Overview
Control Plane Elements - CoPP

- CoPP is enabled by default and can not be disabled
- CLI is a subset of the N7K CoPP
- Class-maps specify the matching protocol/packet
- Policy-map has groups of classes each with a policer rate and burst size
- All class-maps and policy-maps are currently predefined and new ones can not be defined.
- Three fixed policy-maps
  - Default (copp-system-policy-default)
  - Scaled-L2 (copp-system-policy-scaled-l2)
  - Scaled-L3 (copp-system-policy-scaled-l3)
- One “custom” policy which will have modifiable rates/burst values (copp-system-policy-custom)

```
control-plane
[no] service-policy input <policy_name>
```
Nexus 5000/5500 Hardware Overview

Control Plane Elements

- Monitoring of in-band traffic via the NX-OS built-in ethanalyzer
  Eth3 is equivalent to ‘inbound-lo’
  Eth4 is equivalent to ‘inbound-hi’

CLI view of in-band control plane data

```
dc11-5020-3# ethanalyzer local sniff-interface ?
  inbound-hi  Inbound(high priority) interface
  inbound-low Inbound(low priority) interface
  mgmt        Management interface
```

```
dc11-5020-4# sh hardware internal cpu-mac inband counters
eth3  Link encap:Ethernet  HWaddr 00:0D:EC:B2:0C:83
      UP BROADCAST RUNNING PROMISC ALLMULTI MULTICAST  MTU:2200  Metric:1
      RX packets:3 errors:0 dropped:0 overruns:0 frame:0
      TX packets:630 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:252 (252.0 b)  TX bytes:213773 (208.7 KiB)
      Base address:0x6020 Memory:fa4a0000-fa4c0000
eth4  Link encap:Ethernet  HWaddr 00:0D:EC:B2:0C:84
      UP BROADCAST RUNNING PROMISC ALLMULTI MULTICAST  MTU:2200  Metric:1
      RX packets:85379 errors:0 dropped:0 overruns:0 frame:0
      TX packets:92039 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:33960760 (32.3 MiB)  TX bytes:25825826 (24.6 MiB)
      Base address:0x6000 Memory:fa440000-fa460000
```
Nexus 5500 Hardware Overview
Nexus 5500 – UPC (Gen 2) and Port Mapping

- UPC-2 interfaces are indirectly mapped to front panel ports
- Mapping of ports to UPC-2 ASIC
  - The left column identifies the Ethernet interface identifier, xgb1/8 = e1/8
  - Column three and four reflect the UPC port that is associated with the physical Ethernet port

```
ateurs-5548# show hardware internal carmel all-ports
Carmel Port Info:
name |log|car|mac|flag|adm|opr|m:s:l|ipt|fab|xcar|xpt|if_index|diag|ucVer
-------+---+---+---+---+---+---+---+---+---+---+---+---+---+---
xgb1/2 |1 0 0 - |b7 |dis|dn |10:0:f|10 |92 |0 |0 |1a001000|pass| 4.0b
xgb1/3 |1 0 1 - |b7 |dis|dn |11:1:f|11 |88 |0 |0 |1a000000|pass| 4.0b
xgb1/4 |1 0 2 - |b7 |dis|dn |12:2:f|12 |93 |0 |0 |1a003000|pass| 4.0b
xgb1/5 |1 0 3 - |b7 |dis|dn |13:3:f|13 |89 |0 |0 |1a002000|pass| 4.0b
xgb1/6 |1 0 4 - |b7 |dis|dn |14:4:f|14 |90 |0 |0 |1a005000|pass| 4.0b
xgb1/7 |1 0 5 - |b7 |dis|dn |15:5:f|15 |94 |0 |0 |1a004000|pass| 4.0b
xgb1/8 |1 0 6 - |b7 |dis|dn |16:6:f|16 |95 |0 |0 |1a007000|pass| 4.0b
sup0 |32 |4 4 - |b7 |en |dn |4:4:0|4 |62 |0 |0 |15020000|pass| 0.00
sup1 |33 |4 5 - |b7 |en |dn |5:5:1|5 |59 |0 |0 |15010000|pass| 0.00
```
Nexus 5500 Hardware Overview
5548UP/5596UP – UPC (Gen-2) and Unified Ports

- All versions of 5500 support 1/10G on all ports
- 5548UP, 5596UP and N55-M16UP (Expansion Module) support Unified Port capability on all ports
  - 1G Ethernet Copper/Fibre
  - 10G DCB/FCoE Copper/Fibre
  - 1/2/4/8G Fibre Channel

5548UP, 5596UP & N55-M16UP

5548P
Unified Port Controller 2
Ethernet PHY
SFP+ Cage

5548UP, 5596UP & N55-M16UP
Unified Port Controller 2
PHY removed, all MAC and PHY functions performed on UPC-2
1/10G Ethernet ‘and’
1/2/4/8G FC capable on all ports
**Nexus 5500 Hardware Overview**  
**5548UP/5596UP – UPC (Gen-2) and Unified Ports**

- With the 5.0(3)N1 and later releases each module can define any number of ports as Fibre Channel (1/2/4/8 G) or Ethernet (either 1G or 10G)
- Initial SW releases supports only a continuous set of ports configured as Ethernet or FC within each ‘slot’
  - Eth ports have to be the first set and they have to be one contiguous range
  - FC ports have to be second set and they have to be contiguous as well
- Future SW release will support per port dynamic configuration

```
  n5k(config)# slot <slot-num>
  n5k(config-slot)# port <port-range> type <fc | ethernet>
```

<table>
<thead>
<tr>
<th>Slot 2 GEM</th>
<th>Slot 3 GEM</th>
<th>Slot 4 GEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eth Ports</td>
<td>Eth</td>
<td>FC</td>
</tr>
<tr>
<td>slot 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eth Ports</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Nexus 5000/5500 and 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Layer 3 Forwarding
- Nexus 2000 Architecture
  - FEXLink Architecture
  - Extending FEXLink – Adapter FEX
- Nexus 5000/5500
  - SPAN & ERSPAN
  - Multicast
  - Port Channels
  - QoS
Nexus 5000 & 5500 Packet Forwarding
UPC Details

Network Ports

Transmit MAC
Receive MAC

Packet Forwarding and Header Re-Write Engine

Congestion Control
Egress In-flight Packet Queues
Packet Buffer
Virtual Output Queues

Crossbar Fabric

Packet Buffer
Virtual Output Queues
Egress In-flight Packet Queues
Congestion Control
Packet Forwarding and Header Re-Write Engine
Receive MAC
Transmit MAC
Network Ports

Crossbar Fabric
Nexus 5000 & 5500 Packet Forwarding
Packet Forwarding Overview

1. Ingress MAC - MAC decoding, MACSEC processing (not supported currently), synchronize bytes
2. Ingress Forwarding Logic - Parse frame and perform forwarding and filtering searches, perform learning apply internal DCE header
3. Ingress Buffer (VoQ) - Queue frames, request service of fabric, dequeue frames to fabric and monitor queue usage to trigger congestion control
4. Cross Bar Fabric - Scheduler determines fairness of access to fabric and determines when frame is de-queued across the fabric
5. Egress Buffers - Landing spot for frames in flight when egress is paused
6. Egress Forwarding Logic - Parse, extract fields, learning and filtering searches, perform learning and finally convert to desired egress format
7. Egress MAC - MAC encoding, pack, synchronize bytes and transmit
Nexus 5000 Hardware Overview
Nexus 5000 UPC (Gen 1) Forwarding Details
Nexus 5500 Hardware Overview

Nexus 5500 UPC (Gen-2) Forwarding Details

- Changes in the 5500 UPC-2
- Larger tables and more functions

Parsed Packet

Collect Interface Configuration and State

Vlan Translation Table (16K)

Network Ports

Vlan State Table (4K)

Transmit MAC & PHY

Fibre Channel Switch Table (8K)

Receive MAC & PHY

Multicast Vector Table (8K)

Packet Forwarding and Header Re-Write Engine

FabricPath Switch Table (8K)

Congestion Control

Packet Buffer

Egress In-flight Packet Queues

Crossbar Fabric

Virtual Output Queues

Editing Instructions & Virtual Output Queue List

Virtual Interface Table (4K)

Network Ports

Station Table (32K)

Ethernet Learning

ACL Search Engine (4K)

Determine Destination (ingress only)

Fibre Channel Multipath Table (1K)

SGACL Label Table (4K)

Policy Enforcement

Binding Table (4K)

Multipath Expansion (ingress only)

Zoning Table (4K)

PortChannel Table (48)

Multicast Vector Table (8K)

Fabric Path Multipath Table (1K)

Multicast Vector Table (8K)

Station Table (32K)

Editing Instructions & Virtual Output Queue List

Crossbar Fabric

Notice of Trademark:
Cisco and the Cisco logo are trademarks or registered trademarks of Cisco Systems, Inc. and its affiliates in the United States and certain other countries.
All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company.

Editors: IBM

Cisco Public

BRKARC-3452 © 2012 Cisco and/or its affiliates. All rights reserved.
Nexus 5000
Station (MAC) Table allocation

- Nexus 5000 has a 32K Station table entries
  - 16K shadow entries used for vPC
  - 1k reserved for multicast (Multicast MAC addresses)
  - 1.2k assumed for hashing conflicts (conservative)
  - 13.8k effective Layer 2 unicast MAC address entries
Nexus 5500
Station (MAC) Table allocation

- Nexus 5500 has a 32K Station table entries
  - 4k reserved for multicast (Multicast MAC addresses)
  - 3k assumed for hashing conflicts (very conservative)
  - 25k effective Layer 2 unicast MAC address entries
Nexus 5000 & 5500 Packet Forwarding

DCE – Internal Nexus 5000/5500 Forwarding Header

- All frames forwarded internally using Cisco DCE Header after parsing the packet header
Nexus 5000 & 5500 Packet Forwarding
Packet Forwarding—Fibre Channel and FCoE

- Nexus 5000 and 5500 operate as both an Ethernet switch and a Fibre Channel switch
- Supports native FC as well as FCoE interfaces
- Internally within the switching fabric all Fibre Channel frames are forwarded as DCE/FCoE frames
  - FC to FCoE
  - FC to FC
  - FCoE to FC
  - FCoE to FCoE

All FC/FCoE Packets within the switching fabric are forwarded as DCE/FCoE frames

Forwarding logic either forwards FCoE frame or removes header and forwards FC frame

Forwarding logic adds Ethernet, FCoE and internal DCE headers

Unified Crossbar Fabric

FC 3/1
FC 2/3
Ethernet Header
FCoE Header
FC Header
FC Payload
CRC
EOF
FCS
Native FCoE frame
Native FC frame
Nexus 5000 & 5500 Packet Forwarding

vPC peer-link 5000/5500 Forwarding

- Nexus 5000 uses a different mechanism to identify vPC forwarded frames sent across the vPC peer-link.
- Nexus 5010/5020 leverages a shadow VLAN and MAC address to identify ‘vPC’ frames received on the peer switch to prevent looping frames.
- Nexus 5548/5596 leverages a DCE header to identify the vPC topology for each frame to prevent looping frames.
- 16 bytes of header added to all traffic crossing vPC peer link.

![Diagram of Nexus 5000 & 5500 Packet Forwarding](image)
**Nexus 5500 FabricPath**

Standards Based + Cisco Extensions

- Nexus 5500 UPC-2 supports two forms of mac-in-mac encapsulation for FabricPath
  - TRILL (RFC 6325, RFC 6327, RFC 6439)
  - Cisco FabricPath (DCE)
- Advertise up to 64 RBridge ‘or’ DCE Switch IDs
- Support 8K forwarding entries
- Support up to 16 equal cost forwarding path
- Support shared/source based multicast tree

![Cisco Forwarding Diagram](image)

![TRILL Forwarding Diagram](image)
Nexus 5000 & 5500 Packet Forwarding
Packet Forwarding—Cut Thru Switching

- Nexus 5000 & 5500 utilize a Cut Thru architecture when possible
- Bits are serialized in from the ingress port until enough of the packet header has been received to perform a forwarding and policy lookup
- Once a lookup decision has been made and the fabric has granted access to the egress port bits are forwarded through the fabric
- Egress port performs any header rewrite (e.g. CoS marking) and MAC begins serialization of bits out the egress port

Packet Header is serialized into UPC
Packet is serialized across Fabric once forwarding decision is made
Unified Crossbar Fabric
Forwarding
Egress Queue is only used if Pause Frame Received while packet in-flight
Packet Header Re-Write, MAC Learning and then serialized out egress port
Nexus 5000 & 5500 Packet Forwarding
Packet Forwarding—Cut-Through Switching

- Nexus 5000 and 5500 utilize both cut-through and store and forward switching.
- Cut-through switching can only be performed when the ingress data rate is equivalent or faster than the egress data rate.
- The X-bar fabric is designed to forward 10G packets in cut-through which requires that 1G to 1G switching also be performed in store and forward mode.
Nexus 5000 & 5500 Packet Forwarding

Minimizing Latency ‘and’ Loss

- Why Cut-Through Switching?
- It is only one variable in overall fabric optimization
- Designs target consistency of performance under variable conditions
- A balanced fabric is a function of maximal throughput ‘and’ minimal loss => “Goodput”

Data Center Design Goal: Optimizing the balance of end-to-end fabric latency with the ability to absorb traffic peaks and prevent any associated traffic loss
# Nexus 5000 & 5500 Packet Forwarding

Forwarding Mode Behavior (Cut-Through or Store and Forward)

<table>
<thead>
<tr>
<th>Source Interface</th>
<th>Destination Interface</th>
<th>Switching Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 GigabitEthernet</td>
<td>10 GigabitEthernet</td>
<td>Cut-Through</td>
</tr>
<tr>
<td>10 GigabitEthernet</td>
<td>1 GigabitEthernet</td>
<td>Cut-Through</td>
</tr>
<tr>
<td>1 GigabitEthernet</td>
<td>1 GigabitEthernet</td>
<td>Store-and-Forward</td>
</tr>
<tr>
<td>1 GigabitEthernet</td>
<td>10 GigabitEthernet</td>
<td>Store-and-Forward</td>
</tr>
<tr>
<td>FCoE</td>
<td>Fibre Channel</td>
<td>Cut-Through</td>
</tr>
<tr>
<td>Fibre Channel</td>
<td>FCoE</td>
<td>Store-and-Forward</td>
</tr>
<tr>
<td>Fibre Channel</td>
<td>Fibre Channel</td>
<td>Store-and-Forward</td>
</tr>
<tr>
<td>FCoE</td>
<td>FCoE</td>
<td>Cut-Through</td>
</tr>
</tbody>
</table>
Nexus 5000 & 5500 Packet Forwarding
Packet Forwarding - Cut Through Switching

- In Cut-Through switching frames are not dropped due to bad CRC
- Nexus 5000/5500 implements a CRC ‘stomp’ mechanism to identify frames that have been detected with a bad CRC upstream
- A packet with a bad CRC is “stomped”, by replacing the “bad” CRC with the original CRC exclusive-OR’d with the STOMP value (a 1’s inverse operation on the CRC)
- In Cut Through switching frames with invalid MTU (frames with a larger MTU than allowed) are not dropped
- Frames with a “> MTU” length are truncated and have a stomped CRC included in the frame
Nexus 5000 & 5500 Packet Forwarding

Packet Forwarding—Cut Through Switching

- Corrupt or Jumbo frames arriving inbound will count against the Rx Jumbo or CRC counters
- Corrupt or Jumbo frames will be identified via the Tx output error and Jumbo counters

```
dc11-5020-4# sh int eth 1/39
RX
576 unicast packets  4813153 multicast packets  55273 broadcast packets
4869002 input packets  33150983 bytes
31 jumbo packets <0 storm suppression packets
0 runts 0 giants 0 CRC 0 no buffer
0 input error 0 short frame 0 overrun 0 underrun 0 ignored
0 watchdog 0 bad etype drop 0 bad proto drop 0 if down drop
0 input with dribble 0 input discard
0 Rx pause

```

```
dc11-5020-4# sh int eth 2/4
TX
112 unicast packets  349327 multicast packets  56083 broadcast packets
405553 output packets  53600658 bytes
31 jumbo packets
31 output errors <0 collision 0 deferred 0 late collision
0 lost carrier 0 no carrier 0 babble
0 Tx pause
```
Nexus 5000 & 5500 Packet Forwarding
Packet Forwarding—Cut Thru Switching

- CRC and ‘stomped’ frames are tracked internally between ASIC’s within the switch as well as on the interface to determine internal HW errors are occurring

```
dc11-5020-4# sh hardware internal gatos asic 2 counters interrupt
<snip>
Gatos 2 interrupt statistics:
Interrupt name                                |Count |ThresRch|ThresCnt|Ivl
-----------------------------------------------+-------+--------+--------+---
gat_bm_port0_INT_err_mtu_vio                  |1f     |0       |1f
```

```
dc11-5020-4# sh hardware internal gatos asic 13 counters interrupt
<snip>
Gatos 13 interrupt statistics:
Interrupt name                                |Count |ThresRch|ThresCnt|Ivl
-----------------------------------------------+-------+--------+--------+---
gat_aw2_INT_mtu_out_vio                       |1f     |0       |1f
```

**Note:** Please see session BRKCRS-3145 (Troubleshooting the Cisco Nexus 5000 / 2000 Series Switches) for more information on this type of troubleshooting
The table below indicates the forwarding behavior for a corrupt packet (CRC error) arriving on a port operating in cut-through mode:

<table>
<thead>
<tr>
<th>Source Interface Type</th>
<th>Destination Interface Type</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>10GE/DCE/FCoE</td>
<td>10GE/DCE/FCoE</td>
<td>The CRC frame is transmitted as is</td>
</tr>
<tr>
<td>10GE/DCE/FCoE</td>
<td>Native Fibre Channel</td>
<td>The FC CRC is stomped. Also the frame is transmitted with EOFa</td>
</tr>
<tr>
<td>Native Fibre Channel</td>
<td>Native Fibre Channel</td>
<td>The FC CRC is stomped. Also the frame is transmitted with EOFa</td>
</tr>
<tr>
<td>Native Fibre Channel</td>
<td>10GE/DCE/FCoE</td>
<td>The FC CRC is stomped. Also the frame is transmitted with EOFa. Also the Ethernet CRC is stomped</td>
</tr>
</tbody>
</table>
Nexus 5000/5500 and 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Layer 3 Forwarding
- Nexus 2000 Architecture
  - FEXLink Architecture
- Nexus 5000/5500/2000
  - SPAN & ERSPAN
  - Multicast
  - Port Channels
  - QoS
Nexus 5500 Series
Nexus 5500 with Layer 3 support

1) Remove Fans
2) Replace Daughtercard with L3 enabled daughtercard
3) Install License and enabled NX-OS features

Nexus 5548P/UP
- Ordered with L3 daughtercard installed or order a FRU for an L2 5548
- Daughtercard can be replaced while in the rack

Nexus 5596UP
- At FCS one Layer 3 Expansion Module
- Support for OIR of Layer 3 Expansion Module and/or up to three Layer 3 Expansion Modules (Future)

160Gbps (240Mpps) Layer 3 processing
### Nexus 5500 Series – 5.1(3)N1
N55-D160L3-V2 and N55-M160L3-V2

#### N55-D160L3 and N55-M160L3

<table>
<thead>
<tr>
<th>Capability</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4 Longest Prefix Match</td>
<td>8k (16K with uRPF disabled)</td>
</tr>
<tr>
<td>Routes</td>
<td></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>

#### 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</td>
<td>8k (16K with uRPF disabled)</td>
</tr>
<tr>
<td>Routes</td>
<td></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>

**NOTE:** Increased Host and MCAST Route scale is supported in SW in the 5.2(1)N1 release
Nexus 5500 Series
Nexus 5500 with Layer 3 support

- Layer 3 Forwarding Engine connects 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 Series
Nexus 5500 with Layer 3 support

- A single NX-OS CLI is used to configure, manage and troubleshoot the 5500 for all protocols (vPC, STP, OSPF, FCoE, ...)
- There is ‘NO’ need to manage the Layer 3 ASIC directly (no ‘session 15’ interface is required)
- Routing Protocols are consistently configured across all layer 3 enabled NX-OS switches (Nexus 7000, Nexus 5500, Nexus 3000)
- Interfaces supported for Layer 3
  - L3 routed interface (non-FEX ports)
  - L3 sub-interface
  - SVI (FEX ports could be members of VLANs)
  - Port channels

```
L3-5548-1# sh run ospf
!Command: show running-config ospf
version 5.0(3)N1(1)
feature ospf
router ospf 1
  router-id 100.100.100.1
  area 0.0.0.0 authentication message-digest
  log-adjacency-changes
router ospf 100
  graceful-restart helper-disable
router ospf 2
interface Vlan10
  ip ospf passive-interface
  ip router ospf 1 area 0.0.0.0
interface Vlan20
  ip ospf passive-interface
  ip router ospf 1 area 0.0.0.0
interface Vlan100
  ip ospf authentication-key 3 9125d59c18a9b015
  ip ospf cost 4
  ip ospf dead-interval 4
  ip ospf hello-interval 1
  ip router ospf 1 area 0.0.0.0
```
Nexus 5500 Series

Nexus Unicast Routing

- NX-OS software & hardware architecture consistent between Nexus 5500 and Nexus 7000

```
L3-5548-1# sh ip route
IP Route Table for VRF "default"
'*' denotes best unicast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
10.1.1.0/24, ubest/mbest: 1/0, attached
  *via 10.1.1.1, Vlan10, [0/0], 3d00h, direct
10.1.1.1/32, ubest/mbest: 1/0, attached
  *via 10.1.1.1, Vlan10, [0/0], 3d00h, local

L3-5548-1# sh forwarding route
IPv4 routes for table default/base

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Next-hop</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.1.0/24</td>
<td>Attached</td>
<td>Vlan10</td>
</tr>
<tr>
<td>10.1.1.0/32</td>
<td>Drop</td>
<td>Null0</td>
</tr>
<tr>
<td>10.1.1.1/32</td>
<td>Receive</td>
<td>sup-eth1</td>
</tr>
<tr>
<td>10.1.1.2/32</td>
<td>10.1.1.2</td>
<td>Vlan10</td>
</tr>
<tr>
<td>10.1.1.255/32</td>
<td>Attached</td>
<td>Vlan10</td>
</tr>
</tbody>
</table>
```
### Nexus 5500 Series

Nexus 5500 with Layer 3 support

- Layer 3 Forwarding Engine connects 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 Status
--- ---- -------------------------- ---- -------------------------------
<snip>
3 0 O2 Daughter Card with L3 ASIC N55-D160L3 ok

L3-5548-1# sh int port-channel 127
port-channel127 is up
<snip>
```
Nexus 5500 Series

Nexus 5500 with Layer 3 support

- Layer 3 Forwarding Tables can be tuned for specific design scenarios
- Similar to SDM templates used on Catalyst 3750/3650
- Three table space allocations
  - Host Routes (1 entry per /32) – Adjacent Hosts
  - LPM (1 entry per route) – Longest Prefix Match Routes
  - Multicast Routes (*2 entries per mcast route) – (S,G) and (*,G)

```plaintext
L3-5548-1# show hardware profile status
Reserved LPM Entries = 1024.
Reserved Host Entries = 4000.
Reserved Mcast Entries = 2048.
Used LPM Entries = 8.
Used Host Entries in LPM = 0.
Used Mcast Entries = 0.
Used Host Entries in Host = 21.

L3-5548-1(config)# hardware profile module 3 lpm-entries 2048
L3-5548-1(config)# hardware profile multicast max-limit 4096

L3-5548-1# show hardware profile status
Reserved LPM Entries = 2048.
Reserved Host Entries = 4000.
Reserved Mcast Entries = 4096.
Used LPM Entries = 8.
Used Host Entries in LPM = 0.
Used Mcast Entries = 0.
Used Host Entries in Host = 21.
```

Default Configuration

- 1K LPM Routes
- 7K Shared LPM & Host Routes
- 4K Host Routes
- 2K L3 Multicast Routes

Tuned Configuration

- 2K LPM Routes
- 6K Shared LPM & Host Routes
- 4K L3 Multicast Routes
- 2K LPM Routes
- 6K Shared LPM & Host Routes
- 4K L3 Multicast Routes
Nexus 5500 Series
Version 1 & Version 2 Layer 3 Module

- Multicast Routes (2 entries per mcast route) – (S,G) and (*,G)
- vPC requires 4 entries (Dual Active DR)
Nexus 5500 Series

RAACL Support

- RACLs can be configured on:
  - L3 Physical interface
  - L3 port-channel interface
  - L3 Sub-Interface
  - L3 Vlan Interface (SVI)
- RACLs and VACLs can not co-exist on the same SVI
  - First one configured is allowed
- Ingress – 2K ACE supported
- Egress – 1K ACE supported

Verifying the RACLs programming
L3-5548-1# show ip acc summ
IPV4 ACL acl01
  Total ACEs Configured: 1
  Configured on interfaces:
    Ethernet1/17 - ingress (Router ACL)
    Ethernet1/17 - egress (Router ACL)

<snip>
Nexus 5000/5500 and 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Layer 3 Forwarding
- Nexus 2000 Architecture
  - FEXLink Architecture
- Nexus 5000/5500
  - SPAN & ERSPAN
  - Multicast
- Port Channels
- QoS
Nexus Fabric Extender (FEX)
802.1BR (VNTAG) Port Extension

- The 802.1BR Architecture provides the ability to extend the bridge (switch) interface to downstream devices
- 802.1BR associates the Logical Interface (LIF) to a Virtual Interface (VIF)

NIV downlink ports are assigned a virtual identifier (VIF) that corresponds to a virtual interface on the bridge and is used to forward frames through NIV's.

Note: Not All Designs Supported in the Architecture Are Currently Implemented
Nexus 2000 Fabric Extender (FEX) VN-Tag Port Extension

- Nexus 2000 Fabric Extender operates as a remote line card and does **not** support local switching
- All forwarding is performed on the Nexus 5000/5500 UPC VNTag is a Network Interface Virtualization (NIV) technology that ‘extends’ the Nexus 5000/5500 port down (Logical Interface = LIF) to the Nexus 2000 VIF referred to as a Host Interface (HIF)
  - VNTag is added to the packet between Fabric Extender and Nexus 5000/5500
  - VNTag is stripped before the packet is sent to hosts
- VNTag allows the Fabric Extender to act as a data path of Nexus 5000/5500/7000 for all policy and forwarding

### VNTag

<table>
<thead>
<tr>
<th>VNTag Ethertype</th>
<th>d</th>
<th>p</th>
<th>destination virtual interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td></td>
<td></td>
<td>source virtual interface</td>
</tr>
</tbody>
</table>

Logical Interface (LIF) on the ingress UPC is used to forward the packet
Packet is forwarded over fabric link using a specific VNTag

N2K ASIC maps specific VNTag to HIF interface

|-------|-------|----------|-----------|-----------------
|       |       |          |           |
|       |       |          |           |
|       |       |          |           |
Nexus 5500 Adapter FEX
Association of a vNIC to a veth

- A-FEX is supported on UCS and Nexus 5500
- Virtual NIC (vNIC): Refers to a hardware partition of a physical NIC as seen by an Operating System (Virtual Interface = VIF)
- Virtual Ethernet interface (veth): Refers to a virtual network port” (vNIC) as seen by the Nexus 5500 (Logical Interface = LIF)
Nexus 5000/5500 and 2000
Packet Forwarding Overview

1. Frame received on N2K HIF port
2. Nexus 2000 appends VNTag and forwards frame to fabric uplink
3. Nexus 5000 UPC performs ingress forwarding and queuing
4. If required egress queuing and flow control
5. Nexus 5000 UPC appends destination VNTag and forwards frame on fabric link
6. VNTag stripped and frame forwarded out on N2K HIF port
Nexus 5000/5500 and 2000 Virtual Switch
Packet Forwarding Latency

- Nexus 2000 also supports Cut-Through switching
  - 1GE to 10GE on first N2K ingress is store and forward
  - All other stages are Cut Through (10GE N2K port operates in end to end cut through)
- Port to Port latency is dependent on a single store and forward operation at most
Nexus 5000/5500 and 2000
Switching Morphology - Is this Really Different?

- Nexus 2000 Architecture localizes the Forwarding ASIC in the parent switch (supervisor)
- Minimal latency due to cut-thru architecture
- De-coupled life cycle management (upgrade the supervisor without worrying about line card)
  - TCO advantages
  - Reduced SW/HW complexity
- Key Design consideration is over-subscription
Nexus 5000/5500 and 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Layer 3 Forwarding
- Nexus 2000 Architecture
  - FEXLink Architecture
- Nexus 5000/5500/2000
  - SPAN & ERSPAN
  - Multicast
  - Port Channels
  - QoS
Nexus 5000 SPAN Rx
SPAN Replication and Rate Limiting

- SPAN data packets are replicated at ingress port ASIC-Unified Port Controller (UPC) for Rx SPAN
- SPAN packets is queued at the SPAN destination port VOQ
- UPC to Fabric connection for each ingress port is clocked at 12Gbps (20% overspeed)
- Data packets and SPAN packets share the 12Gbps fabric connection at SPAN source
- On **Nexus 5000**
  - A rate limit CLI was introduced in order to limit the SPAN traffic 1 Gig
  - The CLI is configured on SPAN destination port
  - Once the CLI is configured the SPAN traffic is limited to 1 Gig independently of ingress data traffic

Original Packet is queued for egress port

SPAN Packet queued for SPAN destination port

12 Gbps

SPAN traffic is rate limited if exceeds 1Gbps to protect production traffic

Eth 1/8

Eth 1/20

Data Packet

Span Packet

Outgoing Traffic

Egress Traffic

© 2012 Cisco and/or its affiliates. All rights reserved.
**Nexus 5500 SPAN Rx**

**SPAN Replication and Rate Limiting**

- SPAN data packets are replicated at ingress port ASIC-Unified Port Controller (UPC) for Rx SPAN.
- SPAN packets is queued at the SPAN destination port VOQ.
- UPC to Fabric connection for each ingress port is clocked at 12Gbps (20% overspeed).
- Data packets and SPAN packets share the 12Gbps fabric connection at SPAN source.

**On Nexus 5500**

- When data rate is above 5Gbps, SPAN traffic is reduced to 0.75Gbps to avoid potential congestion over the link between ingress port and switch fabric.
- The aggregate SPAN traffic from all SPAN sources (including both RX and TX SPAN) can’t exceed 5Gbps per UPC.
- SPAN traffic won’t affect data traffic when SPAN destination port is congested.
Nexus 5000 & 5500 SPAN Tx
SPAN Replication and Rate Limiting

- SPAN data packets are replicated at egress port ASIC-Unified Port Controller (UPC) for Tx SPAN

  - On **Nexus 5000**
    - A rate limit CLI was introduced in order to limit the SPAN traffic 1 Gig
    - The CLI is configured on SPAN destination port
    - Once the CLI is configured the SPAN traffic is limited to 1 Gig independently of ingress data traffic

  - On **Nexus 5500**
    - When data rate is above 5Gbps, SPAN traffic is reduced to 0.75Gbps to avoid potential congestion over the link between ingress port and switch fabric
    - The aggregate SPAN traffic from all SPAN sources (including both RX and TX SPAN) can’t exceed 5Gbps per UPC
    - SPAN traffic won’t affect data traffic when SPAN destination port is congested
Nexus 5500 SPAN – Tracking Rate Limiting
SPAN Replication and Rate Limiting

- Find out the UPC ASIC and port number of SPAN source port (Carmel is the UPC ASIC name in 5500)

```
L3-N5548-2# show hardware internal carmel port ethernet 1/1
```

Carmel port xgb1/1 card-config info:
- if_index : 0x1a000000
- logical_port : 0
- front_port : 0
- carmel instance : 0
- mac port : 1

- Check SPAN packets drop due to SPAN policing

```
5548-1# show hard int carmelasic 0 registers match bm.*cnt.*span.*drop.*1$
```

Slot 0 Carmel 0 register contents:

<table>
<thead>
<tr>
<th>Register Name</th>
<th>Offset</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>car_bm_CNT_span0_drop_addr_1</td>
<td>0x523fc</td>
<td>0xee222553</td>
</tr>
<tr>
<td>car_bm_CNT_span1_drop_addr_1</td>
<td>0x52400</td>
<td>0</td>
</tr>
<tr>
<td>car_bm_CNT_span2_drop_addr_1</td>
<td>0x52404</td>
<td>0</td>
</tr>
<tr>
<td>car_bm_CNT_span3_drop_addr_1</td>
<td>0x52408</td>
<td>0</td>
</tr>
<tr>
<td>car_bm_CNT_span4_drop_addr_1</td>
<td>0x5240c</td>
<td>0</td>
</tr>
</tbody>
</table>

Done.
Nexus 5000 & 5500 ERSPAN
Encapsulated Remote SPAN

- Nexus 5000/5500 support local SPAN and ERSPAN source session
- Nexus 5548P/5548UP/5596UP – 4 SPAN/ERSPAN sessions
- Nexus 5010/5020 – 2 SPAN/ERSPAN session
- ERSPAN encapsulates SPAN traffic to IP-GRE frame format and allow remote monitoring traffic over IP network
- Both Nexus 5000 and Nexus 5500 platforms support ERSPAN
- Support ERSPAN source session only.
- N7K, Cat6K and Nexus 1010 NAM can de-capsulate ERSPAN.
- ERSPAN does not require L3 module and L3 license
Nexus 5000 & 5500 ERSPAN
Encapsulated Remote SPAN

- On N5K the ERSPAN destination is the IP address of the remote switch that will de-capaculate the ERSPAN traffic
- Software figures out the egress interface of the ERSPAN traffic by checking the destination IP address against its routing table
- No L3 module and license required.
- Without L3, user need to configure IP address under SVI and configure static route for VRF “default”

```cisco
feature interface-vlan
interface vlan 100
ip address 10.10.10.1/24
no shut
vrf context default
ip route 0.0.0.0/0 10.10.10.2

monitor erspan origin ip-address 10.10.10.1 global
monitor session 10 type erspan-source
  erspan-id 20
  vrf default
destination ip 65.65.65.2
  source interface port-channel1000 rx
  no shut
```

```
monitor session 1 type erspan-destination
  erspan-id 20
  vrf default
  source ip 65.65.65.2
destination interface Ethernet1/1
  no shut
```
Nexus 5000/5500 and 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Layer 3 Forwarding
- Nexus 2000 Architecture
  - FEXLink Architecture
- Nexus 5000/5500/2000
  - SPAN & ERSPAN
  - Multicast
- Port Channels
- QoS
Nexus 5000/5500 Multicast Forwarding
Fabric-Based Replication

- Nexus 5000 and 5500 use fabric based egress replication
- Traffic is queued in the ingress UPC for each MCAST group
- When the scheduler permits the traffic if forwarded into the fabric and replicated to all egress ports
- When possible traffic is super-framed (multiple packets are sent with a single fabric scheduler grant) to improve throughput
Nexus 5000 Multicast Forwarding
Multicast Queues and Multicast Group Fan-Out

- A “FAN-OUT” = is an Output Interface List (OIL)
- The Nexus 5000 currently supports 1000 fan-outs and 4000 Multicast Groups
- The multicast groups need to be mapped to the 1000 fan-outs
- There are eight multicast queues per UPC forwarding engine (no VoQ for multicast)
- Hardware needs to map fan-outs to the eight queues
- Multicast scheduler waits until all egress queues are free to accept a frame before traffic in that queue is replicated across the fabric
Nexus 5000 Multicast Forwarding
Multicast Queues and Multicast Group Fan-Out

- Overlap of multicast groups to fan-outs to queues can result in contention for the fabric for a specific group
- Tuning of the multicast traffic and fan-out mapping to queues can be used to prioritize specific groups access to the fabric
- Of the eight queues available for multicast two are reserved (FCoE and sup-redirect multicast) leaving six for the remainder of the multicast traffic
- By default the switch uses the frame CoS to identify the multicast queue for a specific group
- If more groups are mapped to one CoS group than another the system queuing for multicast may be non-optimal
Nexus 5000 Multicast Forwarding
Multicast Queues and Multicast-optimization

- “Multicast-optimize” when enabled for a class of traffic assigns multicast fan-outs in that class to any unused CoS queues on a round robin basis

- With multicast optimization, you can assign these classes of traffic to the unused queues
  - One ‘class of service’ (CoS-based)
  - IP multicast (traffic-based)
  - All flood (traffic-based)

```
class-map type qos class-ip-multicast
    policy-map type qos MULTICAST-OPTIMIZE
        class class-ip-multicast
            set qos-group 2
        class-map type network-qos class-ip-multicast
            match qos-group 2
        policy-map type network-qos MULTICAST-OPTIMIZE
            class type network-qos class-ip-multicast
                multicast-optimize
            class type network-qos class-default
        system qos
            service-policy type qos input MULTICAST-OPTIMIZE
            service-policy type network-qos MULTICAST-OPTIMIZE
```
Nexus 5500 Multicast Forwarding

Nexus 5500 Data Plane Changes

- Nexus 5500 supports 4000 IGMP snooping entries
- Dedicated Unicast & Multicast Queuing and Scheduling Resources
  - 128 MCAST VOQ per port
  - 8 for egress queues for unicast and 8 for multicast
  - 4 Egress cross-points (fabric buffer) per egress port
  - Out of 4 fabric buffer, one is used for unicast, one for multicast and two are shared between unicast and multicast
- Two configurable Multicast scheduler modes
  - Overloaded mode (Proxy Queue)
    - Congested egress ports are ignored
    - Multicast packets are sent to non-congested port only
  - Reliable mode
    - Packets are sent to switch fabric when all OIF ports are ready, i.e., have fabric buffer and egress buffer to accept the multicast packets
Nexus 5500 Multicast Forwarding

IP Multicast Forwarding Table

- Multicast IP address is mapped to multicast MAC address with prefix 01.00.5E
- Nexus 5500 checks the destination MAC against the multicast MAC address to make forwarding decision
- IP multicast MAC shares same 32K MAC address table as unicast MAC
- Support 4K groups currently
- Multicast Index Table keep tracks of the OIF (Outgoing Interface List) or fanout
- L3 and L4 headers are used for ACL and QoS processing
Nexus 5500 Multicast Forwarding
Nexus 5500 Data Plane Changes

- Proxy queues to detect congestion at egress
- One proxy queue for each hardware egress queue
- Bytes are added to proxy queue when packets arrive at egress hardware queue
- Proxy queues are drained at 98% of port speed using DWRR
- When proxy queue is full egress port sends "overload" message to central scheduler
- Central scheduler excludes the port in multicast scheduling calculation when overload bit is set AND there is no fabric buffer available. Multicast packet is sent over to non-congested port
- In case of congestion there is a delay for proxy queue to signal overload

N5k(config)#hardware multicast disable-slow-port-pruning
Nexus 5500 Multicast Forwarding
Layer 3 & Layer 2 Multicast Replication

- Layer 3 ASIC performs L3 Multicast replication
  - Forwards one copy of a frame to the egress L3 port channel for each egress subnet
- UPC ASIC perform L2 Multicast replication
  - Fabric based replication for all egress ports with IGMP state within each subnet
- No support for:
  - PIM dense (NX-OS does not support dense mode)
  - Bi-Dir (L3 ASIC is not capable of BiDir)
  - SSM with vPC (NX-OS does not support PIM-SSM with vPC)
  - PIM-SSM is supported with vPC+ with 5.1(3)N1 and later

L3 ASIC replicates for all L3 OIL's (one packet sent to fabric per VLAN)

Any required L2 replication within an egress subnet is performed in the Fabric
Nexus 5500 Multicast Forwarding
vPC and Layer 3 Interaction

- In a vPC when Nexus 5500 is running PIM both N5ks will forward multicast traffic to source tree (vPC leverages the concept of a pseudo DR)
- However only the real DR generate source registration toward RP (multicast routing behaviour)
- To ensure correct forwarding one post routed copy of multicast packet is sent to peer via reserved VLAN
- Following CLI must be configured, otherwise receivers in non-VPC VLAN (VLAN 13 in this example) and receivers behind L3 interface won’t be able to receive multicast traffic
- When N5k-1 receives multicast traffic from source it notifies N5k-2 about the source IP and group address via CFS message
- N5k-2 then generate source registration toward RP

N5596-L3-1(config)# vpc bind-vrf <vrf name> vlan <VLAN ID>
N5596-L3-1(config)# vpc bind-vrf default vlan 4000
Nexus Virtualized Access Switch
Nexus 2000 Multicast Forwarding

- Nexus 2000 supports egress based Multicast replication
- Each fabric link has a list of VNTag’s associated with each Multicast group
- A single copy of each multicast frame is sent down the fabric links to the Nexus 2000
- Extended Multicast VNTag has an associated flooding fan-out on the Nexus 2000 built via IGMP Snooping
- Nexus 2000 replicates and floods the multicast packet to the required interfaces
- Note: When the fabric links are configured using static pinning each fabric link needs a separate copy of the multicast packet (each pinned group on the Nexus 2000 replicates independently)
- Port Channel based fabric links only require a single copy of the multicast packet
Nexus 5000/5500 and 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Layer 3 Forwarding
- Nexus 2000 Architecture
  - FEXLink Architecture
- Nexus 5000/5500
  - SPAN & ERSPAN
  - Multicast
  - Port Channels
- QoS
Nexus 5000/5500 Port Channels

Nexus 5000/5500 Port Channel Types

- Nexus 5010/5020 supports 16 port channels of up to 16 links each
- Nexus 5548/5596 support 48 port channels of up to 16 links each
- Nexus 2200 FEX supports 24 port channels of up to 8 links each
- Port channels configured on FEX do not take any resource from the Nexus 5000/5500 switch
- Nexus 5500 LIF port channels (MLID) do not consume a HW port channel resource
- Nexus 5548/5596 support up 1152 vPC port channels

Nexus 5500 – 48 port channels (96/2) of up to 16 interfaces per port channel

Nexus 5500 – 1152 vPC port channels supported
Nexus 2000 Port Channels

Nexus 2248/2232 Port Channels

- Nexus 2200 series FEX support local port channels
- All FEX ports are extended ports (Logical Interfaces = LIF)
- A local port channel on the N2K is still seen as a single extended port
- Extended ports are each mapped to a specific VNTag
- HW hashing occurs on the N2K ASIC
- Number of ‘local’ port channels on each N2K is based on the local ASIC
  - 21xx – Do not support local port channels (2 port vPC only)
  - 22xx – Support up to 24 local port channels of up to 8 interfaces each as well as vPC (total of $2 \times 8 = 16$ ports)
Nexus 5000/5500 Port Channels

Nexus 5000/5500 Port Channel Efficiency

- Prior generations of Etherchannel load sharing leveraged eight hash buckets
- Could lead to non-optimal load sharing with an odd number of links
- Nexus 5000/5500 and 22xx utilize 256 buckets
- Provides better load sharing in normal operation and avoids in-balancing of flows in any link failure cases

```
dc11-5020-3# sh port-channel load-balance forwarding-path interface port-channel 100
dst-ip 10.10.10.10 src-ip 11.11.11.11
Missing params will be substituted by 0's.
Load-balance Algorithm: source-dest-ip
crc8_hash: 24   Outgoing port id: Ethernet1/37
```
Nexus 5000/5500 Port Channels

Nexus 5000/5500 Port Channel Efficiency

- Nexus 5500 increases potential randomization to hashing
  - VLAN added to hash input
  - Increased number of polynomials and two stage hashing

**Hashing Input**

- Ethernet DA
- Ethernet SA
- IP DA or FC D_ID
- IP SA or FC S_ID
- TCP DP
- TCP SP or FC OX_ID
- VLAN

**Polynomial Select**

- CRC-8a: $X^8+X^5+X^4+1$
- CRC-8b: $X^8+X^5+X^3+X^2+1$
- CRC-8c: $X^8+X^5+X^4+X^3+1$
- CRC-8d: $X^8+X^7+X^6+X^4+X^2+1$
- CRC-8e: $X^8+X^6+X^3+X^2+1$
- CRC-8f: $X^8+X^7+X^3+X^2+1$
- CRC-8g: $X^8+X^7+X^4+X^3+X+1$
- CRC-8h: $X^8+X^2+X+1$

**Field Select**

- XOR
- Modulo

**Number of Equal Paths**

- 256 Values

**CRC-3a**: $X^3+X+1$
**CRC-4a**: $X^4+X^3+X^2+X+1$

**Nexus 5000/5500 Port Channel Efficiency**

- Nexus 5500 increases potential randomization to hashing
- VLAN added to hash input
- Increased number of polynomials and two stage hashing

```
L3-5548-1(config)# port-channel load-balance ethernet source-dest-port CRC8b
```
Nexus 5000/5500 Port Channels

Nexus 5000/5500 Port Channel Symmetry

- Port Channel hash is symmetrical
- Traffic A -> B will be hashed over the same bundle member as B -> A
Nexus 5000/5500 and 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Layer 3 Forwarding
- Nexus 2000 Architecture
  - FEXLink Architecture
- Nexus 5000/5500/2000
  - SPAN & ERSPAN
  - Multicast
  - Port Channels
  - QoS
Data Center QoS Requirements
What do we trust, how do we queue and where do classify and mark?

- Data Center QoS requires some additions to classical Voice/Video QoS
- New PHB behaviours required
- New set of trust boundaries
- New traffic flows and new queuing requirements

### Data Center QoS Requirements

<table>
<thead>
<tr>
<th>Application Class</th>
<th>Per-Hop Behavior</th>
<th>Admission Control</th>
<th>Queuing &amp; Dropping</th>
<th>Application Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>VoIP Telephony</td>
<td>EF</td>
<td>Required</td>
<td>Priority Queue (PQ)</td>
<td>Cisco IP Phones (Q.711, Q.729)</td>
</tr>
<tr>
<td>Broadcast Video</td>
<td>CS5</td>
<td>Required</td>
<td>(Optional) PQ</td>
<td>Cisco IP Video Surveillance / Cisco Enterprise TV</td>
</tr>
<tr>
<td>Realtime Interactive</td>
<td>CS4</td>
<td>Required</td>
<td>(Optional) PQ</td>
<td>Cisco TelePresence</td>
</tr>
<tr>
<td>Multimedia Conferencing</td>
<td>AF4</td>
<td>Required</td>
<td>BW Queue + DSCP WRED</td>
<td>Cisco Unified Personal Communicator, WebEx</td>
</tr>
<tr>
<td>Multimedia Streaming</td>
<td>AF3</td>
<td>Recommended</td>
<td>BW Queue + DSCP WRED</td>
<td>Cisco Digital Media System (VoD)</td>
</tr>
<tr>
<td>Network Control</td>
<td>CS6</td>
<td></td>
<td>BW Queue</td>
<td>EIGRP, OSPF, BGP, HSRP, IKE</td>
</tr>
<tr>
<td>Call-Signaling</td>
<td>CS3</td>
<td></td>
<td>BW Queue</td>
<td>SCCP, SIP, H.323</td>
</tr>
<tr>
<td>Ops / Admin / Mgmt (QAM)</td>
<td>CS2</td>
<td></td>
<td>BW Queue</td>
<td>SNMP, SSH, Syslog</td>
</tr>
<tr>
<td>Transactional Data</td>
<td>AF2</td>
<td></td>
<td>BW Queue + DSCP WRED</td>
<td>ERP Apps, CRM Apps, Database Apps</td>
</tr>
<tr>
<td>Bulk Data</td>
<td>AF1</td>
<td></td>
<td>BW Queue + DSCP WRED</td>
<td>E-mail, FTP, Backup Apps, Content Distribution</td>
</tr>
<tr>
<td>Best Effort</td>
<td>DF</td>
<td></td>
<td>Default Queue + RED</td>
<td>Default Class</td>
</tr>
<tr>
<td>Scavenger</td>
<td>CS1</td>
<td></td>
<td>Min BW Queue (Differential)</td>
<td>YouTube, iTunes, BitTorrent, Xbox Live</td>
</tr>
</tbody>
</table>

### Application Classifications

- VoIP Telephony (EF)
- Broadcast Video (CS5)
- Realtime Interactive (CS4)
- Multimedia Conferencing (AF4)
- Multimedia Streaming (AF3)
- Network Control (CS6)
- Call-Signaling (CS3)
- Ops / Admin / Mgmt (QAM) (CS2)
- Transactional Data (AF2)
- Bulk Data (AF1)
- Best Effort (DF)
- Scavenger (CS1)

### QoS Requirements

- **VoIP Telephony (EF)**: Required for Cisco IP Phones (Q.711, Q.729)
- **Broadcast Video (CS5)**: Priority Queue (PQ) for Cisco IP Video Surveillance / Cisco Enterprise TV
- **Realtime Interactive (CS4)**: (Optional) PQ for Cisco TelePresence
- **Multimedia Conferencing (AF4)**: BW Queue + DSCP WRED for Cisco Unified Personal Communicator, WebEx
- **Multimedia Streaming (AF3)**: BW Queue + DSCP WRED for Cisco Digital Media System (VoD)
- **Network Control (CS6)**: BW Queue for EIGRP, OSPF, BGP, HSRP, IKE
- **Call-Signaling (CS3)**: BW Queue for SCCP, SIP, H.323
- **Ops / Admin / Mgmt (QAM) (CS2)**: BW Queue for SNMP, SSH, Syslog
- **Transactional Data (AF2)**: BW Queue + DSCP WRED for ERP Apps, CRM Apps, Database Apps
- **Bulk Data (AF1)**: BW Queue + DSCP WRED for E-mail, FTP, Backup Apps, Content Distribution
- **Best Effort (DF)**: Default Queue + RED for Default Class
- **Scavenger (CS1)**: Min BW Queue (Differential) for YouTube, iTunes, BitTorrent, Xbox Live
Data Center QoS Requirements

CoS or DSCP?

- We have non IP based traffic to consider again
  - FCoE – Fibre Channel Over Ethernet
  - RCoE – RDMA Over Ethernet
- DSCP is still marked but CoS will be required and used in Nexus Data Center designs

<table>
<thead>
<tr>
<th>PCP/COS</th>
<th>Network priority</th>
<th>Acronym</th>
<th>Traffic characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0 (lowest)</td>
<td>BK</td>
<td>Background</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>BE</td>
<td>Best Effort</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>EE</td>
<td>Excellent Effort</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>CA</td>
<td>Critical Applications</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>VI</td>
<td>Video, &lt; 100 ms latency</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>VO</td>
<td>Voice, &lt; 10 ms latency</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>IC</td>
<td>Internetwork Control</td>
</tr>
</tbody>
</table>
Switch Architectures - Three Approaches to Buffering

1. Ingress per port Buffer
   - Ingress
   - Crossbar
   - Egress

2. Shared Memory Buffer
   - Ingress
   - Shared Memory Buffer
   - Egress

3. Egress per port Buffer
   - Ingress
   - Egress
   - Crossbar
Nexus 5000 & 5500 QoS
Packet Forwarding—Ingress Queuing

- Nexus 5000 and 5500 use an 8 Queue QoS model for unicast and multicast traffic
- Nexus 5000 and 5500 utilize an Ingress Queuing architecture
- Packets are stored in ingress buffers until egress port is free to transmit
- Ingress queuing provides an additive effective
- The total queue size available is equal to [number of ingress ports x queue depth per port]
- Statistically ingress queuing provides the same advantages as shared buffer memory architectures

Traffic is Queued on all ingress interface buffers providing a cumulative scaling of buffers for congested ports

Egress Queue 0 is full, link congested
Traffic is Queued on the Ingress buffer until the egress port is free to transmit the packet

To prevent Head of Line Blocking (HOLB) Nexus 5000 and 5500 use a Virtual Output Queue (VoQ) Model

Each ingress port has a unique set of 8 virtual output queues for every egress port (on 5596 the system uses 794 Ingress VOQs = 98 destinations * 8 classes on every ingress port)

If Queue 0 is congested for any port traffic in Queue 0 for all the other ports is still able to be transmitted

Common shared buffer on ingress, VoQ are pointer lists and not physical buffers

5000/5500 support limiting buffer per VoQ, “not” recommended as a default configuration (limits the ability to absorb bursts on an individual port/queue)

#Enabling the per VoQ limit (not a recommended default)
5596(config)# hardware unicast voq-limit
**Nexus 5000 QoS**

**UPC (Gen 1) QoS Defaults**

- QoS is enabled by default (not possible to turn it off)
- Four default class of services defined when system boots up
  - Two for control traffic (CoS 6 & 7)
  - One for FCoE traffic (class-fcoe – CoS 3)
  - Default Ethernet class (class-default – all others)
- You can define up to four additional system classes for Ethernet traffic.
- Control traffic is treated as strict priority and serviced ahead of data traffic
- The two base user classes (class-fcoe and class-default) get 50% of guaranteed bandwidth by default

```
dc11-5020-2# sh policy-map system type qos input
<snip>
Class-map (qos): class-fcoe (match-any)
  Match: cos 3
  set qos-group 1

Class-map (qos): class-default (match-any)
  Match: any
  set qos-group 0
```
Nexus 5000 QoS
UPC (Gen 1) QoS Capabilities (*Not Currently Supported)
Nexus 5000 QoS
UPC (Gen 1) Buffering

- 480KB dedicated packet buffer per one 10GE port or per two FC ports
- Buffer is shared between ingress and egress with majority of buffer being allocated for ingress
  - Ingress buffering model
  - Buffer is allocated per system class
  - Egress buffer only for in flight packet absorption
- Buffer size of ingress queues for drop class can be adjusted using network-qos policy

<table>
<thead>
<tr>
<th>Class of Service</th>
<th>Ingress Buffer(KB)</th>
<th>Egress Buffer(KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class-fcoe</td>
<td>76.8</td>
<td>18.8</td>
</tr>
<tr>
<td>Sup-Hi &amp; Sup-Lo</td>
<td>18.0 &amp; 18.0</td>
<td>9.6 &amp; 9.6</td>
</tr>
<tr>
<td>User defined no-drop class of service with MTU&lt;2240</td>
<td>76.8</td>
<td>18.8</td>
</tr>
<tr>
<td>User defined no-drop class of service with MTU&gt;2240</td>
<td>81.9</td>
<td>18.8</td>
</tr>
<tr>
<td>Tail drop class of service</td>
<td>20.4</td>
<td>18.8</td>
</tr>
<tr>
<td>Class-default</td>
<td>All remaining buffer</td>
<td>18.8</td>
</tr>
</tbody>
</table>
Nexus 5500 QoS
UPC (Gen 2) QoS Defaults

- QoS is enabled by default (not possible to turn it off)
- Three default class of services defined when system boots up
  - Two for control traffic (CoS 6 & 7)
  - Default Ethernet class (class-default – all others)
- Cisco Nexus 5500 switch supports five user-defined classes and the one default drop system class
- FCoE queues are ‘not’ pre-allocated
- When configuring FCoE the predefined service policies must be added to existing QoS configurations

```
# Predefined FCoE service policies
service-policy type qos input fcoe-default-in-policy
service-policy type queuing input fcoe-default-in-policy
service-policy type queuing output fcoe-default-out-policy
service-policy type network-qos fcoe-default-nq-policy
```
Nexus 5500 QoS

UPC (Gen 2) QoS Capabilities (*Not Currently Supported)
Nexus 5500 QoS
UPC (Gen 2) Buffering

- 640KB dedicated packet buffer per one 10GE port
- Buffer is shared between ingress and egress with majority of buffer being allocated for ingress
  - Ingress buffering model
  - Buffer is allocated per system class
  - Egress buffer only for in flight packet absorption
- Buffer size of ingress queues for drop class can be adjusted using *network-qos* policy

<table>
<thead>
<tr>
<th>Class of Service</th>
<th>Ingress Buffer(KB)</th>
<th>Egress Buffer(KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class-fcoe</td>
<td>78</td>
<td>19</td>
</tr>
<tr>
<td>Sup-Hi &amp; Sup-Lo</td>
<td>18.0 &amp; 18.0</td>
<td>9.6 &amp; 9.6</td>
</tr>
<tr>
<td>User defined no-drop class of service with MTU&lt;2240</td>
<td>78</td>
<td>19</td>
</tr>
<tr>
<td>User defined no-drop class of service with MTU&gt;2240</td>
<td>88</td>
<td>19</td>
</tr>
<tr>
<td>User defined tail drop class of service with MTU&lt;2240</td>
<td>22</td>
<td>19</td>
</tr>
<tr>
<td>User defined tail drop class of service with MTU&gt;2240</td>
<td>29</td>
<td>19</td>
</tr>
<tr>
<td>Class-default</td>
<td>All remaining buffer</td>
<td>19</td>
</tr>
</tbody>
</table>
Nexus 5000/5500 QoS
QoS Configuration and Behaviour

- NX-OS uses the Cisco MQC (Modular QoS CLI) which defines a three-step configuration model
  - Define matching criteria via a `class-map`
  - Associate action with each defined class via a `policy-map`
  - Apply policy to entire system or an interface via a `service-policy`
- Nexus 5000/5500 leverage the MQC qos-group capabilities to identify and define traffic in policy configuration
- Ingress buffering and queuing (as defined by ingress queuing policy) occurs at VOQ of each ingress port
  - Ingress VOQ buffers are `primary congestion-management point` for arbitrated traffic
- Egress scheduling (as defined by egress queuing policy) enforced by egress port
  - Egress scheduling dictates manner in which egress port bandwidth made available at ingress
  - Per-port, per-priority grants from arbiter control which ingress frames reach egress port
Nexus QoS
Configuration Overview

- **QoS** policy defines how the system classifies traffic, assigned to qos-groups
- **Network-QoS** policy defines system policies, e.g. which COS values ALL ports treat as drop versus no-drop
- **Ingress queuing policy** defines how ingress port buffers ingress traffic for ALL destinations over fabric
- **Egress queuing policy** defines how egress port transmits traffic on wire
  - Conceptually, controls how all ingress ports schedule traffic toward the egress port over fabric (by controlling manner in which bandwidth availability reported to arbiter)

<table>
<thead>
<tr>
<th>Type (CLI)</th>
<th>Description</th>
<th>Applied To…</th>
</tr>
</thead>
<tbody>
<tr>
<td>QoS</td>
<td>Packet Classification based on Layer 2/3/4 (Ingress)</td>
<td>Interface or System</td>
</tr>
<tr>
<td>Network-QoS</td>
<td>Packet Marking (CoS), Congestion Control WRED/ECN (Egress), (drop or no-drop, MTU), Buffer size</td>
<td>System</td>
</tr>
<tr>
<td>Queuing</td>
<td>Scheduling - Queuing Bandwidth % / Priority Queue (Egress)</td>
<td>Interface or System</td>
</tr>
</tbody>
</table>
Nexus QoS
Configuration Overview

1. Define qos Class-Map
2. Define qos Policy-Map
3. Apply qos Policy-Map under “system qos” or interface

- qos-group range for user-configured system class is 2-5
- Policy under system qos applied to all interfaces
- Policy under interface is preferred if same type of policy is applied under both system qos and interface
**Nexus QoS**

Configuration Overview

- Match qos-group is the only option for network-qos class-map
- Qos-group value is set by qos policy-map in previous slide
- No action tied to this class indicates default network-qos parameters.
- Policy-map type *network-qos* will be used to configure no-drop class, MTU, ingress buffer size and 802.1p marking

---

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>N5k(config)# class-map type network-qos class-1</td>
<td>Define network-qos Class-Map</td>
</tr>
<tr>
<td>N5k(config-cmap-nq)# match qos-group 2</td>
<td></td>
</tr>
<tr>
<td>N5k(config-cmap-nq)# class-map type network-qos class-2</td>
<td></td>
</tr>
<tr>
<td>N5k(config-cmap-nq)# match qos-group 3</td>
<td></td>
</tr>
<tr>
<td>N5k(config-cmap-nq)# policy-map type network-qos policy-nq</td>
<td>Define network-qos Policy-Map</td>
</tr>
<tr>
<td>N5k(config-pmap-nq)# class type network-qos class-1</td>
<td></td>
</tr>
<tr>
<td>N5k(config-pmap-nq-c)# class type network-qos class-2</td>
<td></td>
</tr>
<tr>
<td>N5k(config-pmap-nq-c)# system qos</td>
<td>Apply network-qos policy-map under system qos context</td>
</tr>
<tr>
<td>N5k(config-sys-qos)# service-policy type network-qos policy-nq</td>
<td></td>
</tr>
<tr>
<td>N5k(config-sys-qos)#</td>
<td></td>
</tr>
</tbody>
</table>

---

© 2012 Cisco and/or its affiliates. All rights reserved.
Configuring QoS

Network-QoS Policies

- Define **global** queuing and scheduling parameters for all interfaces in switch
  - Identify drop/no-drop classes, instantiate specific default queuing policies, etc.
- One network-QoS policy per system, applies to all ports
- Assumption is network-QoS policy defined/applied consistently network-wide
  - Particularly for no-drop applications, end-to-end consistency mandatory
Nexus 5000/5500 QoS
MTU per Class of Service (CoS Queue)

- MTU can be configured for each class of service (no interface level MTU)
- No fragmentation since Nexus 5000 is a L2 switch
- When forwarded using cut-through, frames are truncated if they are larger than MTU
- When forwarded using store-and-forward, frames are dropped if they are larger than MTU

```
class-map type qos iSCSI
  match cos 2
class-map type queuing iSCSI
  match qos-group 2
policy-map type qos iSCSI
  class iSCSI
    set qos-group 2
class-map type network-qos iSCSI
  match qos-group 2
policy-map type network-qos iSCSI
  class type network-qos iSCSI
    mtu 9216
system qos
  service-policy type qos input iSCSI
  service-policy type network-qos iSCSI
```
Data Center Bridging Control Protocol

DCBX Overview - 802.1Qaz

- Negotiates Ethernet capability's: PFC, ETS, CoS values between DCB capable peer devices
- Simplifies Management: allows for configuration and distribution of parameters from one node to another
- Responsible for Logical Link Up/Down signaling of Ethernet and Fibre Channel
- DCBX is LLDP with new TLV fields
- The original pre-standard CIN (Cisco, Intel, Nuova) DCBX utilized additional TLV's
- DCBX negotiation failures result in:
  - per-priority-pause not enabled on CoS values
  - vfc not coming up – when DCBX is being used in FCoE environment

```
dc11-5020-3# sh lldp dcbx interface eth 1/40
Local DCBXP Control information:
Operation version: 00  Max version: 00  Seq no: 7  Ack no: 0
Type/Subtype    Version    En/Will/Adv Config
006/000        000        Y/N/Y
```


DCBX Switch

DCBX CNA Adapter
Priority Flow Control
FCoE Flow Control Mechanism – 802.1Qbb

- Enables lossless Ethernet using PAUSE based on a COS as defined in 802.1p
- When link is congested, CoS assigned to “no-drop” will be PAUSED
- Other traffic assigned to other CoS values will continue to transmit and rely on upper layer protocols for retransmission
- Not only for FCoE traffic

- B2B Credits
- Fibre Channel
- Ethernet Link
- Transmit Queues
- Receive Buffers
- Eight Virtual Lanes

Enables lossless Ethernet using PAUSE based on a COS as defined in 802.1p.

When link is congested, CoS assigned to “no-drop” will be PAUSED.

Other traffic assigned to other CoS values will continue to transmit and rely on upper layer protocols for retransmission.

Not only for FCoE traffic.

priority flow control

52

52

802.1Qbb

FCoE Flow Control Mechanism

Priority Flow Control
FCoE Flow Control Mechanism – 802.1Qbb

- Enables lossless Ethernet using PAUSE based on a COS as defined in 802.1p
- When link is congested, CoS assigned to “no-drop” will be PAUSED
- Other traffic assigned to other CoS values will continue to transmit and rely on upper layer protocols for retransmission
- Not only for FCoE traffic

- B2B Credits
- Fibre Channel
- Ethernet Link
- Transmit Queues
- Receive Buffers
- Eight Virtual Lanes

Enables lossless Ethernet using PAUSE based on a COS as defined in 802.1p.

When link is congested, CoS assigned to “no-drop” will be PAUSED.

Other traffic assigned to other CoS values will continue to transmit and rely on upper layer protocols for retransmission.

Not only for FCoE traffic.
Nexus 5000/5500 QoS
Priority Flow Control and No-Drop Queues

- Actions when congestion occurs depending on policy configuration
  - PAUSE upstream transmitter for lossless traffic
  - Tail drop for regular traffic when buffer is exhausted
- Priority Flow Control (PFC) or 802.3X PAUSE can be deployed to ensure lossless for application that can’t tolerate packet loss
- Buffer management module monitors buffer usage for no-drop class of service. It signals MAC to generate PFC (or link level PAUSE) when the buffer usage crosses threshold
- FCoE traffic is assigned to class-fcoe, which is a no-drop system class
- Other class of service by default have normal drop behavior (tail drop) but can be configured as no-drop
Nexus 5000/5500
Priority Flow Control – Configuration

- On Nexus 5000 once **feature fcoe** is configured, 2 classes are made by default

```
policy-map type qos default-in-policy
class type qos class-fcoe
    set qos-group 1
class type qos class-default
    set qos-group 0
```

- **class-fcoe** is configured to be **no-drop** with an MTU of 2158

```
policy-map type network-qos default-nq-policy
    class type network-qos class-fcoe
        pause no-drop
        mtu 2158
```

- Enabling the FCoE feature on Nexus 5548/96 does ‘**not**’ create no-drop policies automatically as on Nexus 5010/20
- Must add policies under system QOS:

```
system qos
    service-policy type qos input fcoe-default-in-policy
    service-policy type queuing input fcoe-default-in-policy
    service-policy type queuing output fcoe-default-out-policy
    service-policy type network-qos fcoe-default-nq-policy
```
Enhanced Transmission Selection (ETS)
Bandwidth Management – 802.1Qaz

- Prevents a single traffic class of “hogging” all the bandwidth and starving other classes
- When a given load doesn’t fully utilize its allocated bandwidth, it is available to other classes
- Helps accommodate for classes of a “bursty” nature

![Offered Traffic Diagram]

![10 GE Link Realized Traffic Utilization Diagram]
**Nexus 5500 and iSCSI - DCB**

PFC (802.1Qbb) & ETS 802.1Qaz

- **iSCSI TLV will be supported in the 5.2 release (CY12) – 3rd Party Adapters not validated until that release**

- Functions in the same manner as the FCoE TLV

- Communicates to the compatible Adapter using DCBX (LLDP)

- Steps to configure
  - Configure Class Maps to identify iSCSI traffic
  - Configure Policy Maps to identify marking, queueing and system behaviour
  - Apply policy maps

```plaintext
class-map type qos class-iscsi
  match protocol iscsi
  match cos 4

class-map type queuing class-iscsi
  match qos-group 4

policy-map type qos iscsi-in-policy
  class type qos class-fcoe
    set qos-group 1
  class type qos class-iscsi
    set qos-group 4
```

- iSCSI TLV will be supported in the 5.2 release (CY12) – 3rd Party Adapters not validated until that release
- Functions in the same manner as the FCoE TLV
- Communicates to the compatible Adapter using DCBX (LLDP)
- Steps to configure
  - Configure Class Maps to identify iSCSI traffic
  - Configure Policy Maps to identify marking, queueing and system behaviour
  - Apply policy maps

- Identify iSCSI traffic

- Functions in the same manner as the FCoE TLV
- Communicates to the compatible Adapter using DCBX (LLDP)
- Steps to configure
  - Configure Class Maps to identify iSCSI traffic
  - Configure Policy Maps to identify marking, queueing and system behaviour
  - Apply policy maps

- Identify iSCSI traffic

- Functions in the same manner as the FCoE TLV
- Communicates to the compatible Adapter using DCBX (LLDP)
- Steps to configure
  - Configure Class Maps to identify iSCSI traffic
  - Configure Policy Maps to identify marking, queueing and system behaviour
  - Apply policy maps

- Identify iSCSI traffic
Define policies to be signaled to CNA

Define switch queue BW policies

Define iSCSI MTU and 'if' single hop topology no-drop behaviour

Nexus 5500 and iSCSI - DCB

PFC (802.1Qbb) & ETS 802.1Qaz
Nexus 5000/5500 QoS
Mapping the Switch Architecture to ‘show queuing’

dc11-5020-4# sh queuing int eth 1/39

Interface Ethernet1/39 TX Queuing
qos-group sched-type oper-bandwidth
  0  WRR           50
  1  WRR           50

Interface Ethernet1/39 RX Queuing
qos-group 0
  q-size: 243200, HW MTU: 1600 (1500 configured)
  drop-type: drop, xon: 0, xoff: 1520

Statistics:
  Pkts received over the port : 85257
  Ucast pkts sent to the cross-bar : 930
  Mcast pkts sent to the cross-bar : 84327
  Ucast pkts received from the cross-bar : 249
  Pkts sent to the port : 133878
  Pkts discarded on ingress : 0
  Per-priority-pause status : Rx (Inactive), Tx (Inactive)

<snip - other classes repeated>

Total Multicast crossbar statistics:
  Mcast pkts received from the cross-bar : 283558

Egress (Tx) Queuing Configuration

Packets Arriving on this port but dropped from ingress queue due to congestion on egress port
Nexus 5500 Series
Layer 3 QoS Configuration

- Internal QoS information determined by ingress Carmel (UPC) ASIC is ‘not’ passed to the L3 ASIC
- Need to mark all routed traffic with a dot1p CoS value used to:
  - Queue traffic to and from the 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

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

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

```
class-map type network-qos nqcm-grp2
  match qos-group 2

class-map type network-qos nqcm-grp4
  match qos-group 4

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
```
Nexus 5500 Series
Layer 3 QoS Configuration

- Apply “type qos” and network-qos policy for classification on the L3 interfaces and on the L2 interfaces (or simply system wide)
- Applying “type queuing” policy at system level in egress direction (output)
- Trident has CoS queues associated with every interface
  - 8 Unicast CoS queues
  - 4 Multicast CoS queues
- The individual dot1p priorities are mapped one-to-one to the Unicast CoS queues
  - This has the result of dedicating a queue for every traffic class
- With the availability of only 4 multicast queues the user would need to explicitly map dot1p priorities to the multicast queues
- `wrr-queue cos-map <queue ID> <CoS Map>`
Nexus 2000 QoS
Tuning the Port Buffers

- Each Fabric Extender (FEX) has local port buffers (FEX leverages a shared memory model)
- You can control the queue limit for a specified Fabric Extender for egress direction (from the network to the host)
- You can use a lower queue limit value on the Fabric Extender to prevent one blocked receiver from affecting traffic that is sent to other non-congested receivers (“head-of-line blocking”)
- A higher queue limit provides better burst absorption and less head-of-line blocking protection

```bash
# Disabling the per port tail drop threshold
dc11-5020-3(config)# system qos
dc11-5020-3(config-sys-qos)# no fex queue-limit
dc11-5020-3(config-sys-qos)#

# Tuning of the queue limit per FEX HIF port
dc11-5020-3(config)# fex 100
dc11-5020-3(config-fex)# hardware N2248T queue-limit 356000
dc11-5020-3(config-fex)# hardware N2248T queue-limit ?
<CR>
<2560-652800> Queue limit in bytes
```
**Nexus 2248TP-E**

**32MB Shared Buffer**

- Speed mismatch between 10G NAS and 1G server requires QoS tuning
- **Nexus 2248TP-E** utilizes a 32MB shared buffer to handle larger traffic bursts
- Hadoop, NAS, AVID are examples of bursty applications
- You can control the queue limit for a specified Fabric Extender for egress direction (from the network to the host)
- You can use a lower queue limit value on the Fabric Extender to prevent one blocked receiver from affecting traffic that is sent to other non-congested receivers ("head-of-line blocking")

```
N5548-L3(config-fex)# hardware N2248TPE queue-limit 4000000 rx
N5548-L3(config-fex)# hardware N2248TPE queue-limit 4000000 tx

N5548-L3(config)#interface e110/1/1
N5548-L3(config-if)# hardware N2348TP queue-limit 4096000 tx
```

Tune 2248TP-E to support a extremely large burst (Hadoop, AVID, …)
**Nexus 2248TP-E**

**Enhanced Counters**

```
N5596-L3-2(config-if)# sh queuing interface e110/1/1
Ethernet110/1/1 queuing information:
  Input buffer allocation:
  Qos-group: 0
  frh: 2
  drop-type: drop
  cos: 0 1 2 3 4 5 6
  xon xoff
  +------------------+
  | 0 0              |
  | buffer-size       |
  | 65536             |

Queueing:
  queue qos-group cos priority bandwidth mtu
  +------------+----------+--------+---------+-------+
  | 2          | 0        | 0 1 2 3 4 5 6 | WRR | 100   | 9728 |

Queue limit: 2097152 bytes

Queue Statistics:
  +------------+----------+--------+---------+-------+------------+-------+
  | Que|Received / |Tail Drop |No Buffer|MAC Error|Multicast|Queue Depth|
  No |Transmitted|          |         |          |         |           |
  +------------+----------+--------+---------+-------+------------+-------+
  2rx| 5863073| 0 | 0 | - | 0| 0|
  2tx| 426378558047| 28490502| 0 | 0 | 0| 0|
```

- **Ingress queue limit (Configurable)**
- **Egress queue limit (Configurable)**
- **Egress queues**: CoS to queue mapping
- **Bandwidth allocation**
- **MTU**
- **Per port per queue counters**
- **Drop due to oversubscription**
Nexus 5000/5500 and 2000 Architecture

Data Center Switch

- The functional elements of the Nexus 5000/5500 and 2000 are familiar
  - Distributed forwarding—L2/L3 forwarding, ACL, QoS TCAM
  - Protected management and control plane
  - Non-blocking cross bar switching fabric
  - Flexible connectivity through multiple line cards
- Some new capabilities and physical form factor
  - QoS - DCB, per class MTU, no-drop queues and VoQ
  - Multiprotocol—Ethernet and FC/FCoE forwarding
  - Remote Line Cards (FEX & VNTag)
Conclusion

- You should now have a thorough understanding of the Nexus 5000/5500 Data Center switches and the Nexus 2000 Fabric Extender packet flows, and key forwarding engine functions…
- Any questions?
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
BUILT FOR THE HUMAN NETWORK

© 2012 Cisco and/or its affiliates. All rights reserved.
Cisco Public