Cisco live!

What You Make Possible

TOMORROW starts here.

CISCO

Cisco Nexus 5500/2000 Switch Architecture

BRKARC-3452

Session Goal

- This session presents an in-depth study of the architecture of:
  - Nexus 5000/5550
  - Nexus 6004
  - Nexus 2000 Fabric Extender
- Topics include internal architecture of above platforms, 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 content will include more focus on the Nexus 6004 architecture. This session is designed for network engineers involved in network switching design and Data Centre architecture.
- Related sessions:

BRKARC-3470 - Cisco Nexus 7000 Switch Architecture

BRKSAN-2047 - FCoE Design, Operations and Management Best Practices

Nexus 5000/5500, 6004 & 2000 Architecture

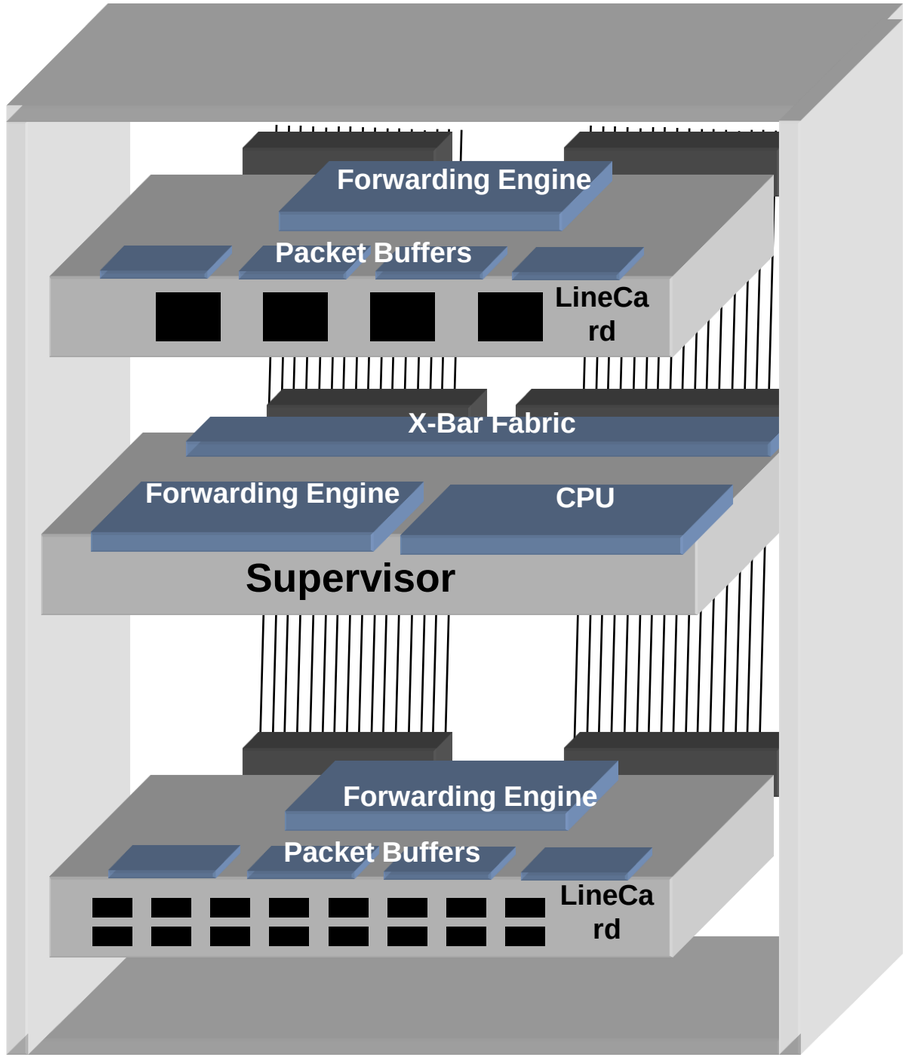
Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Architecture
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

Nexus 5000/5500 and 2000 Architecture

Data Centre 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)



Nexus 5000/5500 and 2000 Architecture

Virtualised Data Centre 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

NOTE: EoS announcement for 5010/5020: http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9670/eol_c51-709037.html

Generation 2 - 5500

Nexus 5548UP & 5596UP

Nexus 5596T

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

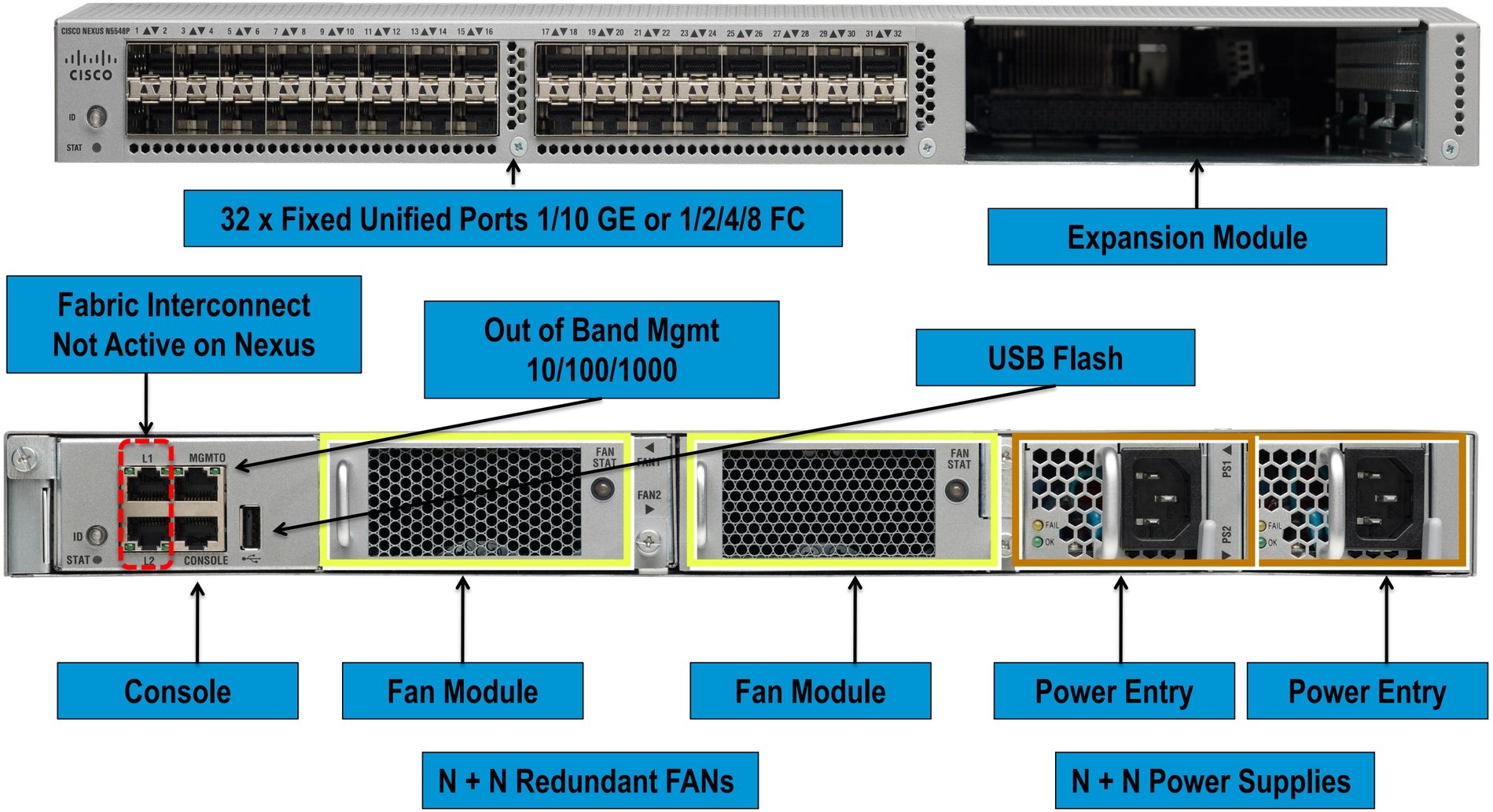
Generation 1, 2 & 3 Nexus 2000

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

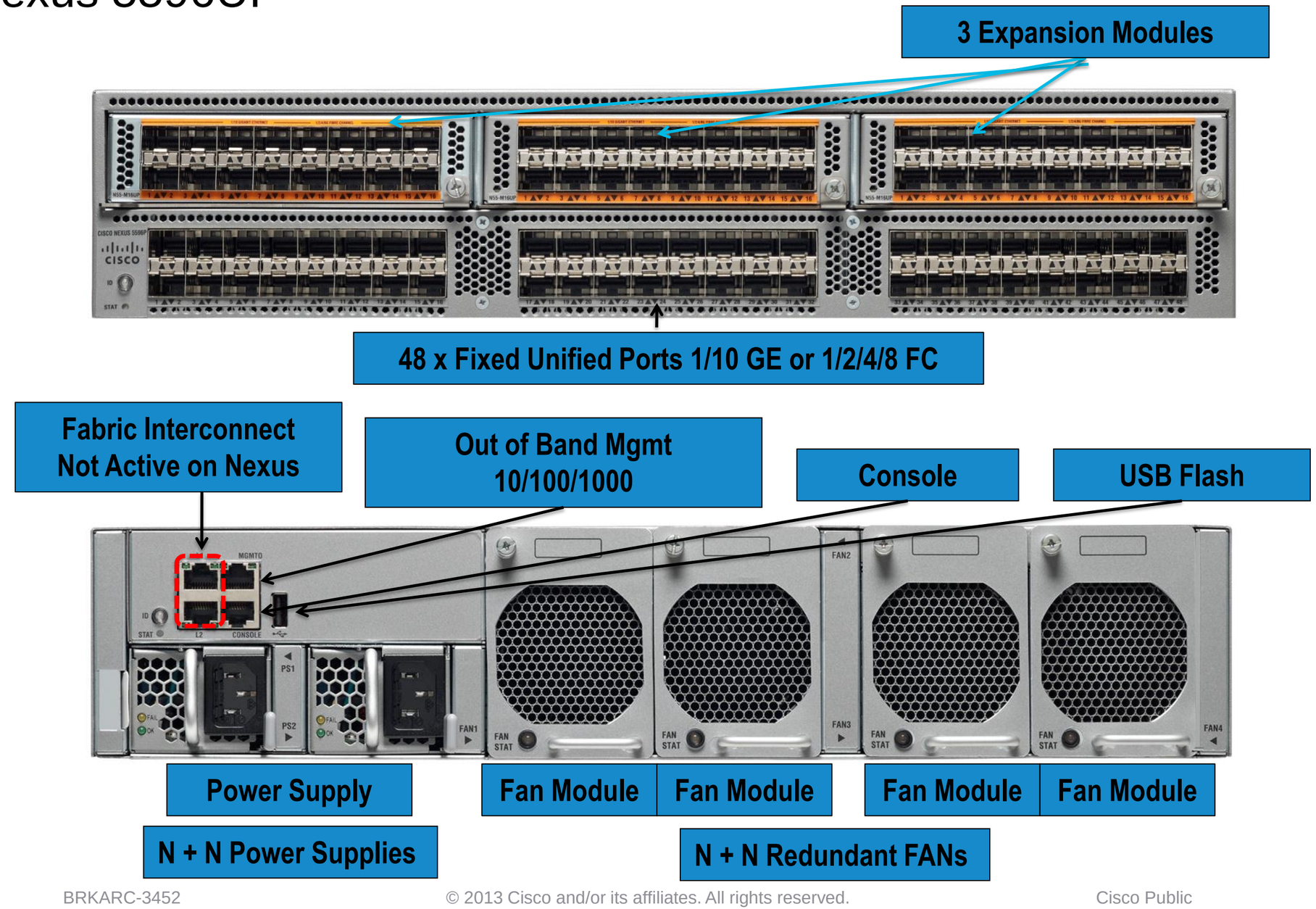
Nexus 5500 Hardware

Nexus 5548 (5548P & 5548UP)



Nexus 5500 Hardware

Nexus 5596UP



Nexus 5500 Hardware

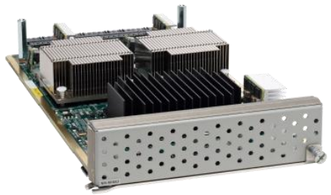
Nexus 5500 Expansion Modules

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

16 x 1/10GE

8 x 1/10GE +
8 x 1/2/4/8G FC

16 unified ports individually configurable as 1/10GE or 1/2/4/8G FC



L3 module for 160G of L3 I/O bandwidth

12 x 10 10GBase T

New

N55-M4Q GEM
(4 Port QSFP+)v

Nexus 5500 Hardware

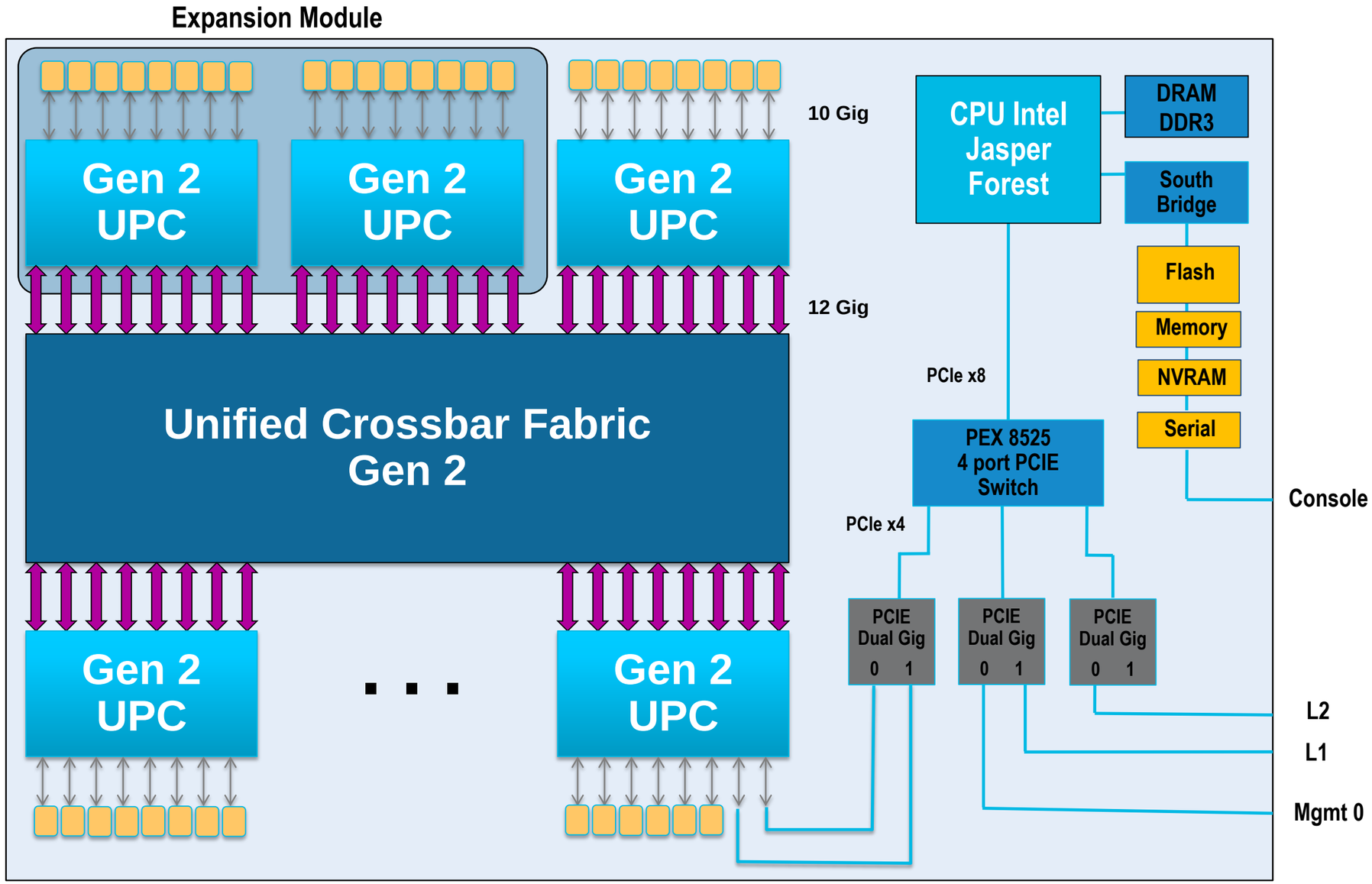
Nexus 5500 Reversible Air Flow and DC Power Supplies

- 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

	Nexus 2000	Hardware Availability	Nexus 5000	Hardware Availability
Front-to-Back Airflow, AC Power	Nexus 2148T Nexus 2200 Series	Today	Nexus 5010/5020 Nexus 5548P/5548UP/5596UP	Today
Back-to-Front Airflow, AC Power	Nexus 2200 Series	Today	Nexus 5548UP/5596UP	Today
Front-to-Back Airflow, DC Power	Nexus 2200 Series	Today	Nexus 5548UP/5596UP	Today
Back-to-Front Airflow, DC Power	N/A	N/A	N/A	N/A

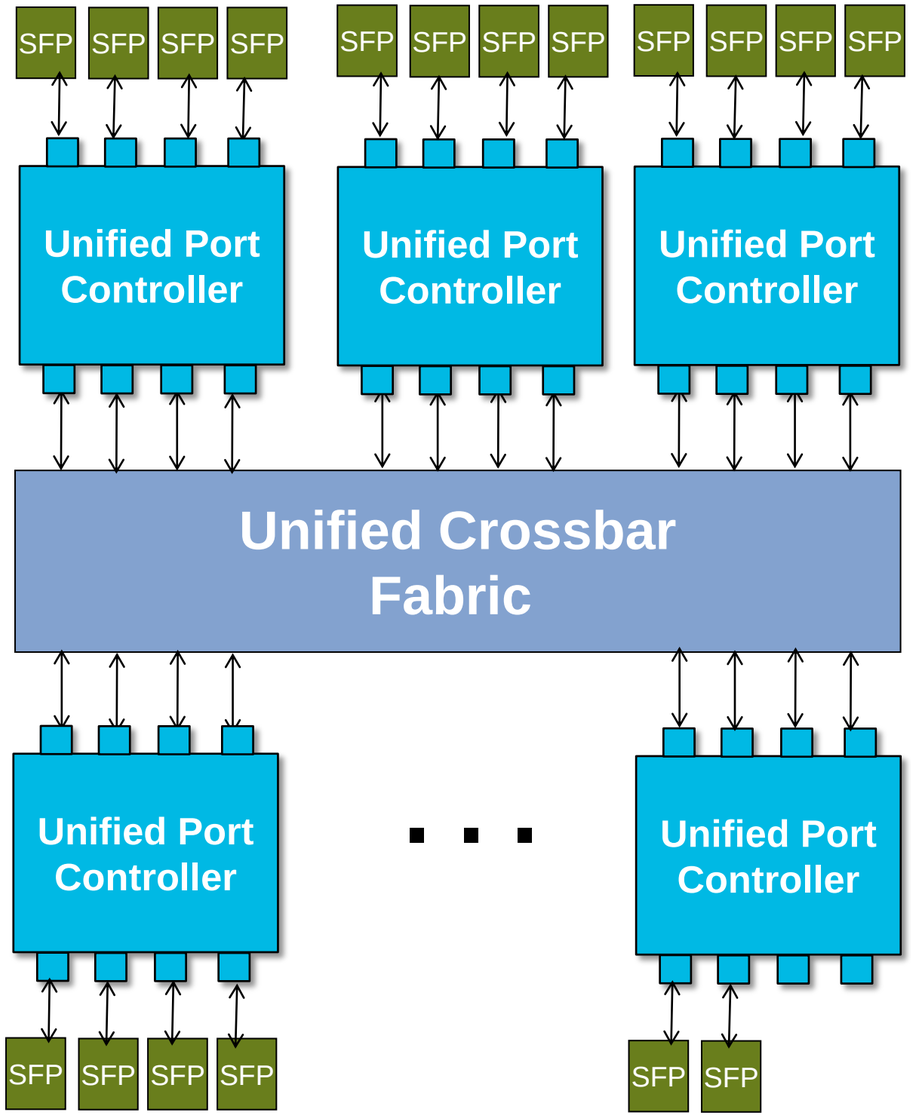
Nexus 5500 Hardware Overview

Data and Control Plane Elements



Nexus 5500 Hardware Overview

Data Plane Elements – Distributed Forwarding



- Nexus 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 1.92Tbps or 1428 mpps

Nexus 5500 Hardware Overview

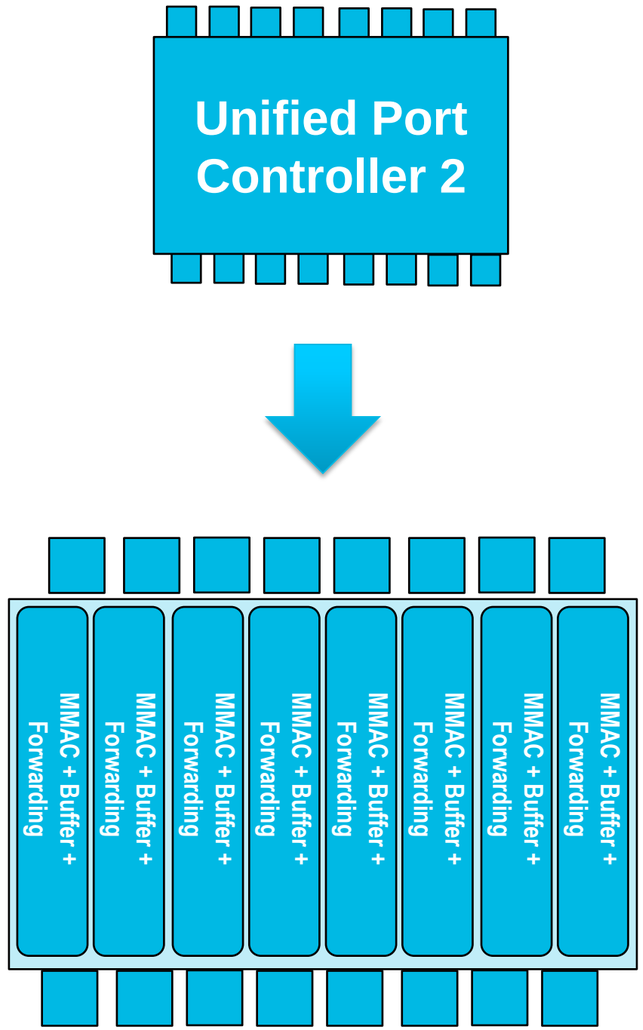
Data Plane Elements – Unified Crossbar Fabric

- 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 (MMAC)
  - 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)

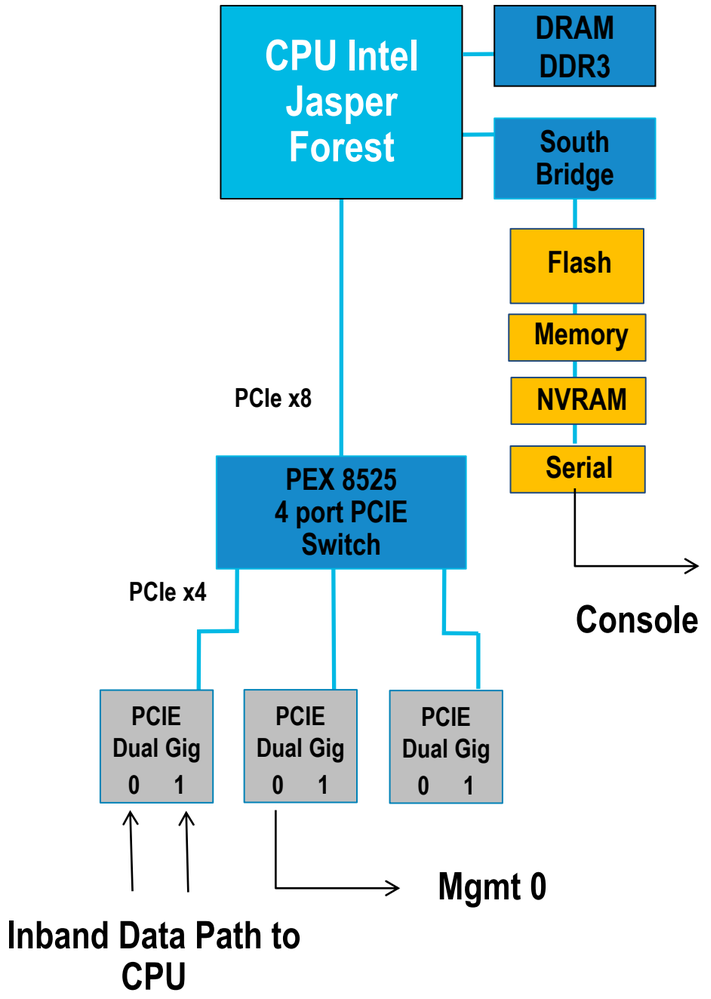


Nexus 5500 Hardware Overview
Control Plane Elements

- 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

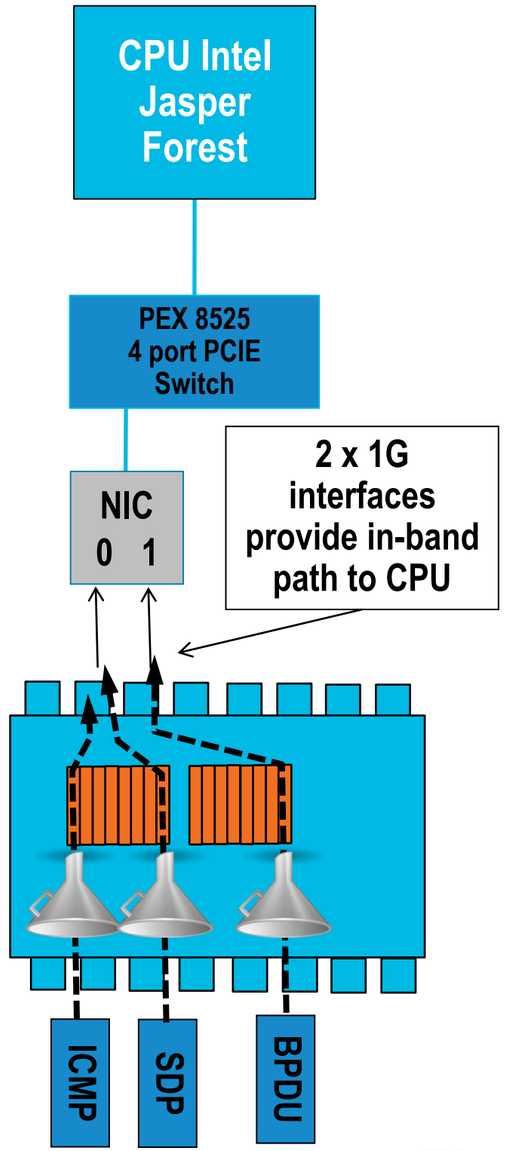
  10/100/1000BASE-T: mgmt0 partitioned from inband VLANs



Nexus 5500 Hardware Overview

Control Plane Elements - CoPP

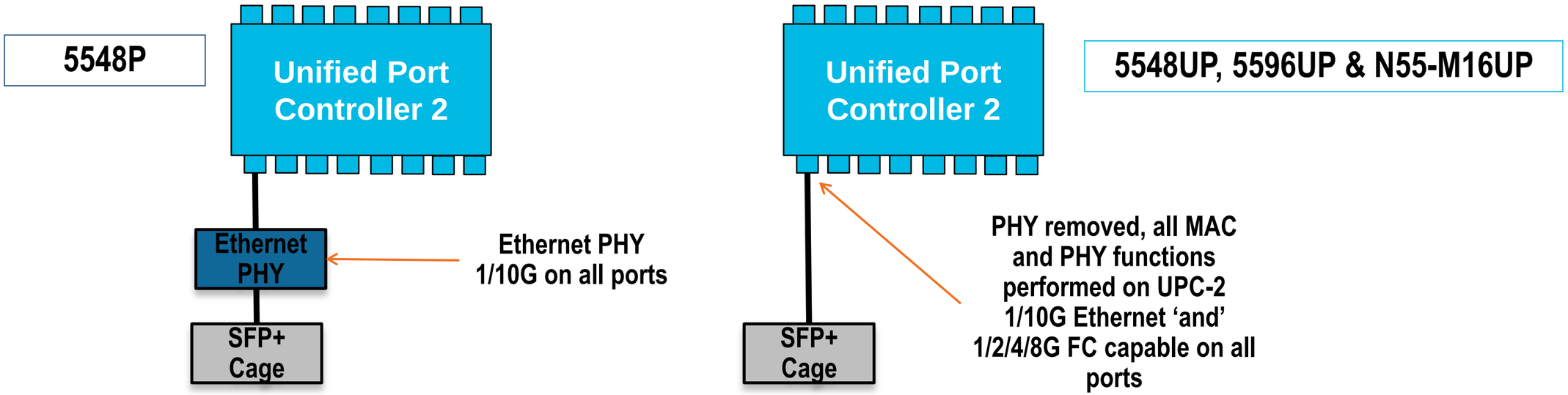
- In-band traffic is identified by the UPC and punted to the CPU via two dedicated UPC interfaces, 5/0 and 5/1, which are in turn connected to eth3 and eth4 interfaces in the CPU complex
  - Receive – Dest Mac == Switch Mac
  - Copy – Copy of the packet needed by SUP
  - Exception - Needs exception handling
  - 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 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



Nexus 5500 Hardware Overview

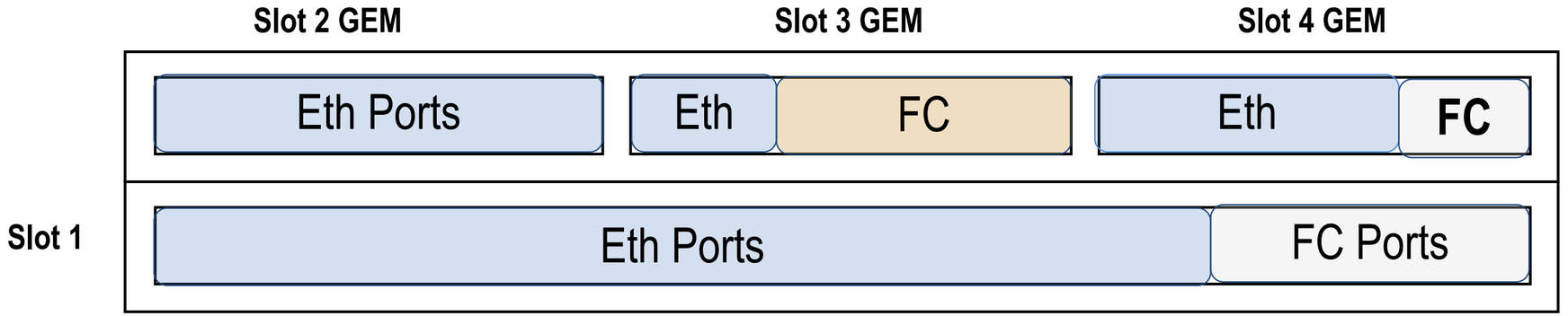
5548UP/5596UP – UPC (Gen-2) and Unified Ports

- 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>
```



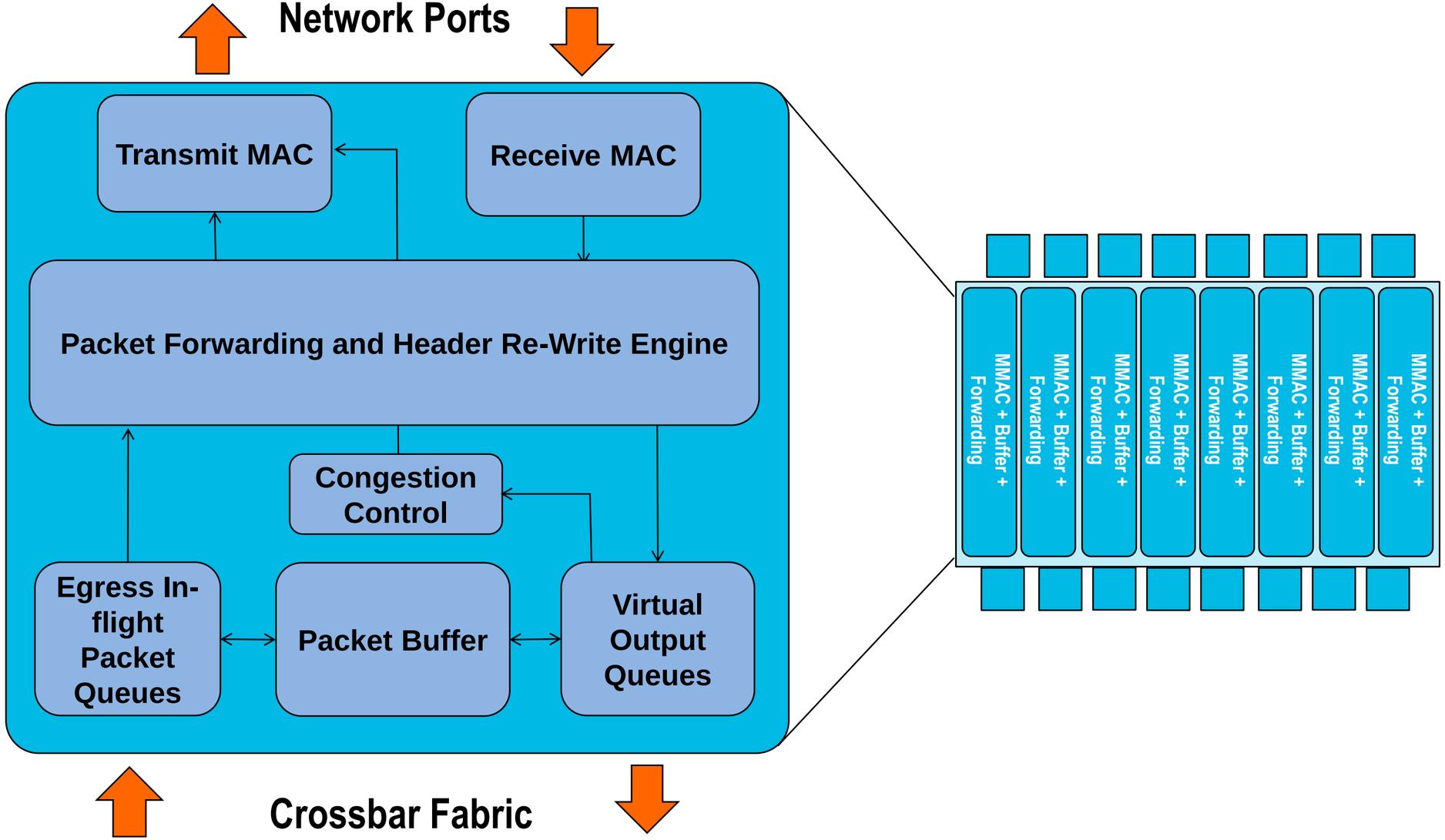
Nexus 5000/5500, 6004 & 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Architecture
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

Nexus 5500 Packet Forwarding

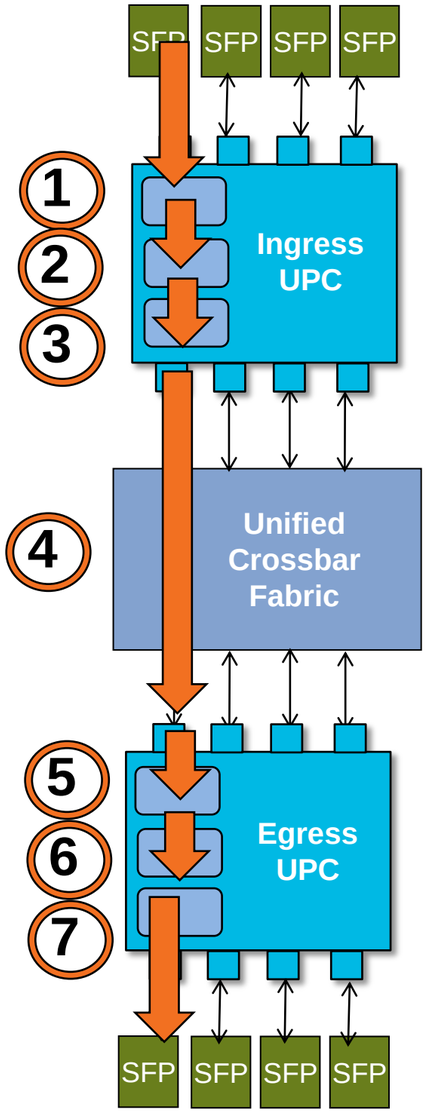
UPC Details



Nexus 5500 Packet Forwarding

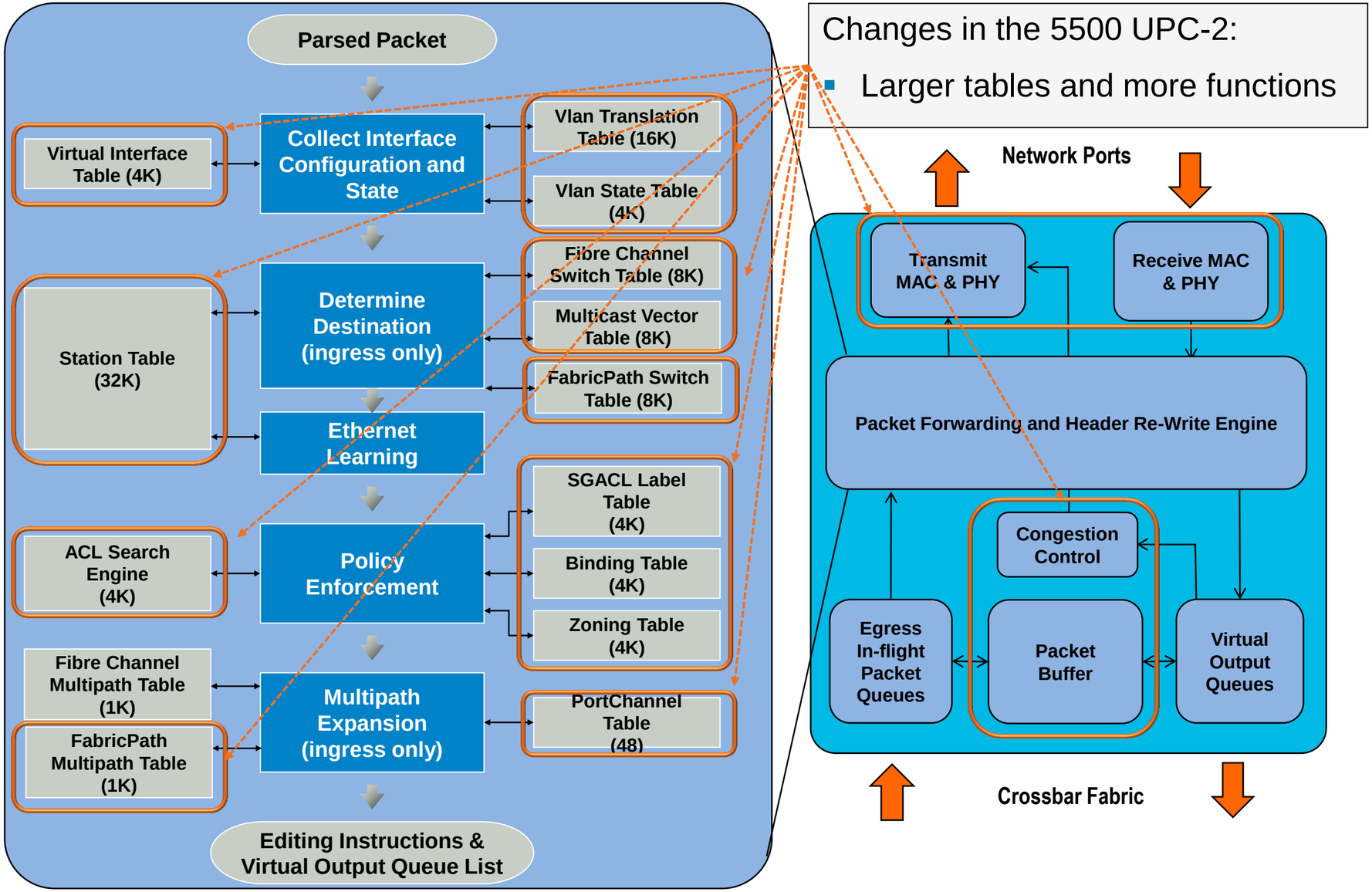
Packet Forwarding Overview

1. Ingress MAC - MAC decoding, MACSEC processing (not supported currently), synchronise 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, synchronise bytes and transmit



Nexus 5500 Packet Forwarding

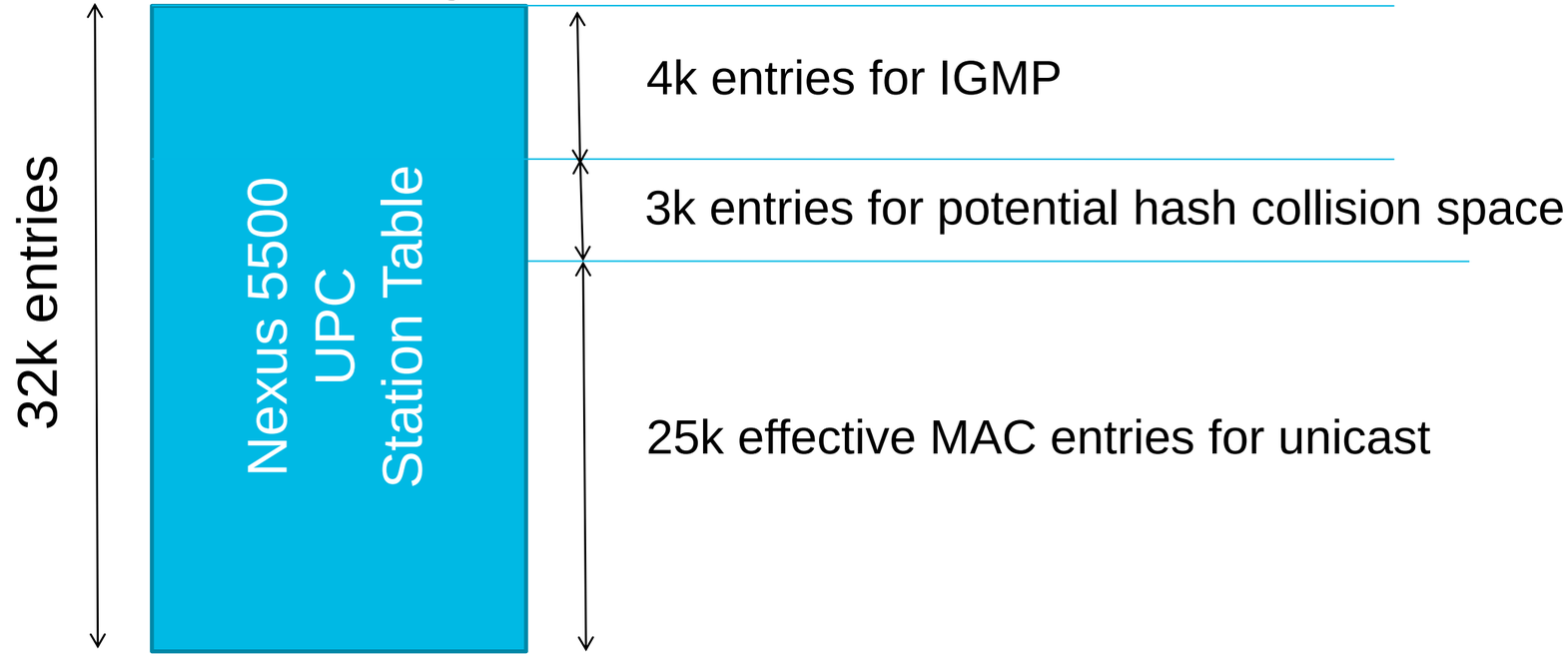
Nexus 5500 UPC (Gen 2) Forwarding Details



Nexus 5500 Packet Forwarding

Station (MAC) Table allocation

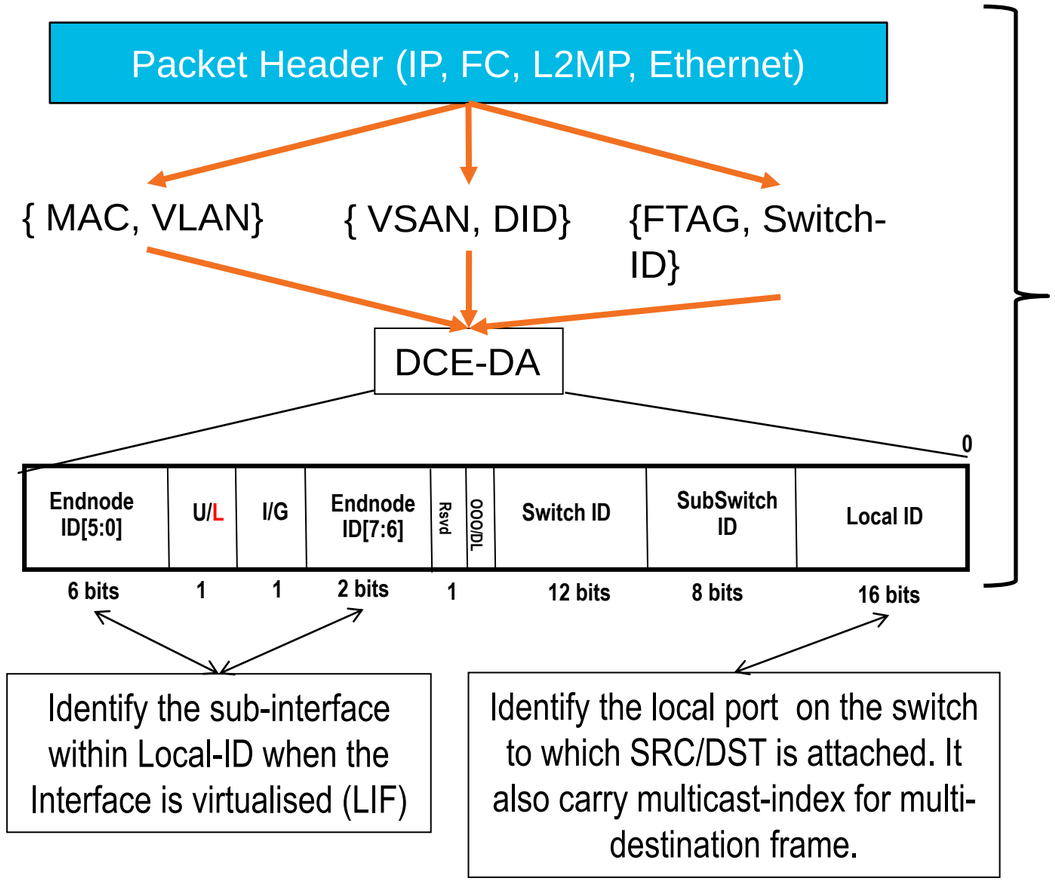
- Nexus 5500 has a 32K Station table entries
- 4k reserved for multicast (Multicast MAC addresses)
- 3k assumed for hashing conflicts (very conservative)
- 25k effective Layer 2 unicast MAC address entries

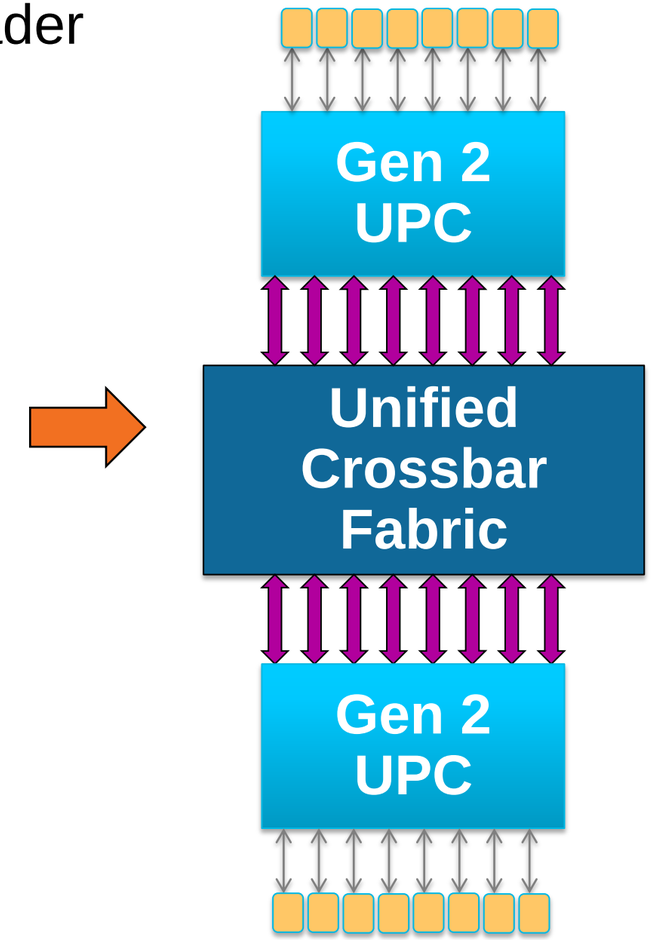


Nexus 5500 Packet Forwarding

DCE – Internal Nexus 5500 Forwarding Header

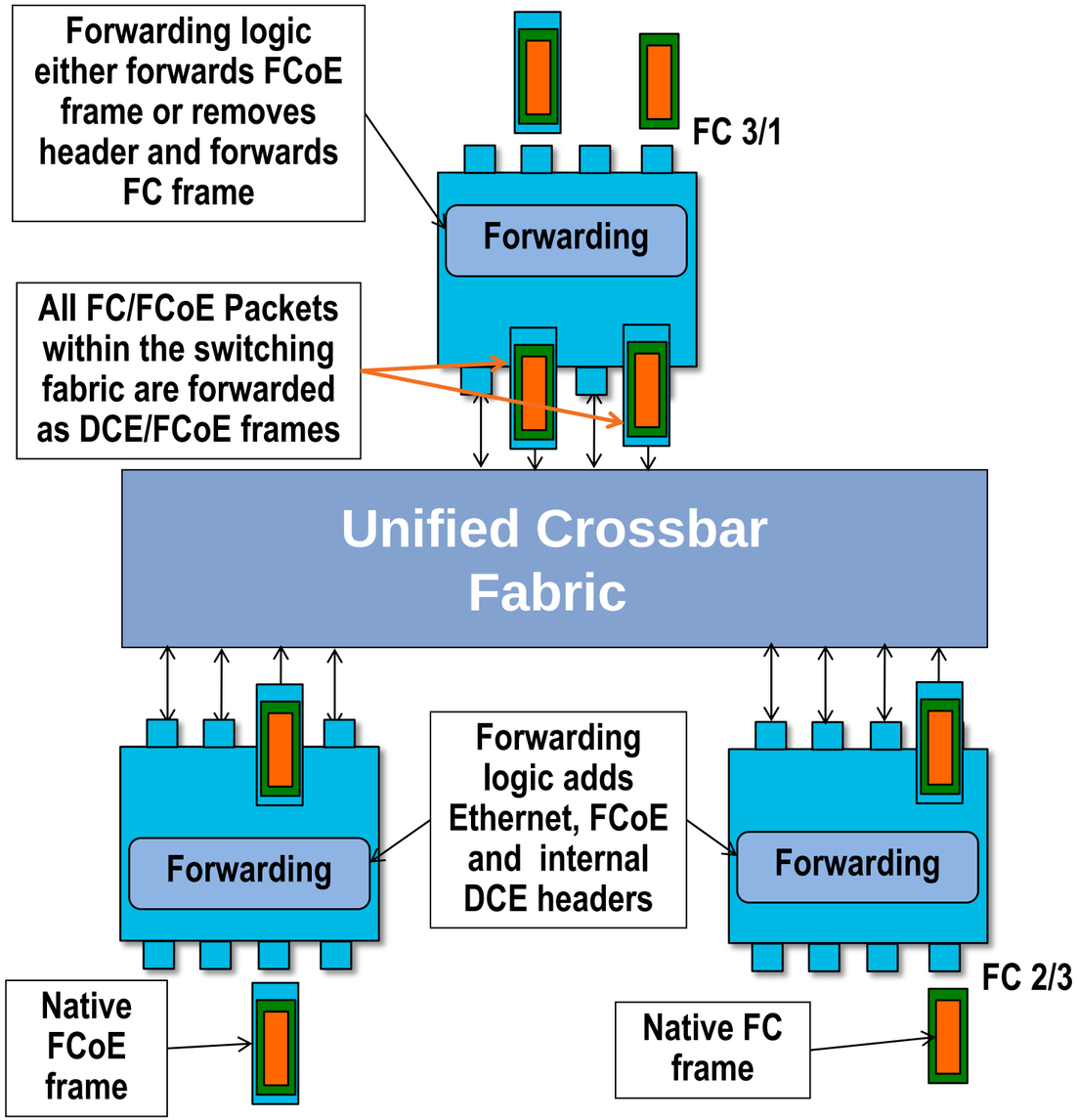
- All frames forwarded internally using Cisco DCE Header after parsing the packet header





Nexus 5500 Packet Forwarding

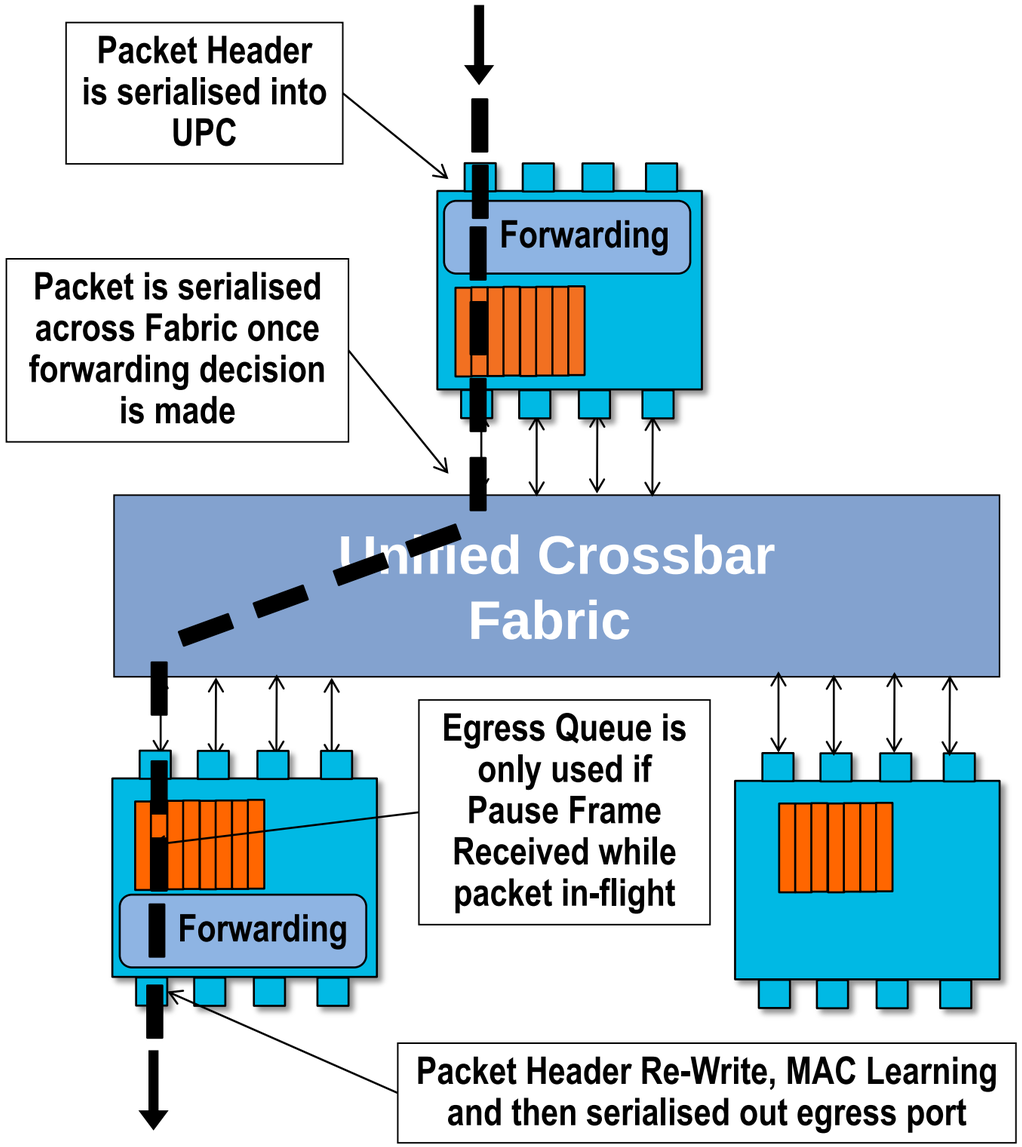
Packet Forwarding—Fibre Channel and FCoE



- Nexus 5500s 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

Nexus 5500 Packet Forwarding

Packet Forwarding—Cut-Through Switching

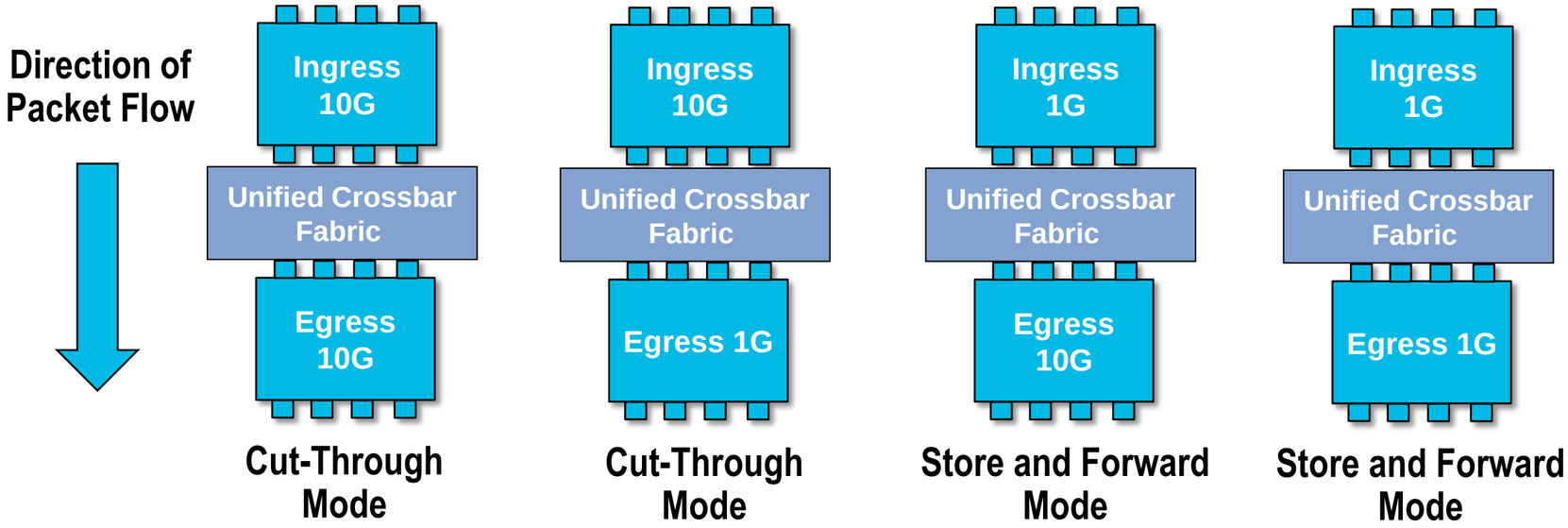


- Nexus 5500s utilise a Cut-Through architecture when possible
- Bits are serialised in from the ingress port until enough of the packet header has been received to perform a forwarding and policy lookup
- Once a lookup decision has been made and the fabric has granted access to the egress port bits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

Nexus 5500 Packet Forwarding

Packet Forwarding—Cut Thru Switching

- Nexus 5500 utilise 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 5500 Packet Forwarding

Cut-Through or Store-and-Forward)

For Your Reference

Source Interface	Destination Interface	Switching Mode
10 GigabitEthernet	10 GigabitEthernet	Cut-Through
10 GigabitEthernet	1 GigabitEthernet	Cut-Through
1 GigabitEthernet	1 GigabitEthernet	Store-and-Forward
1 GigabitEthernet	10 GigabitEthernet	Store-and-Forward
FCoE	Fibre Channel	Cut-Through
Fibre Channel	FCoE	Store-and-Forward
Fibre Channel	Fibre Channel	Store-and-Forward
FCoE	FCoE	Cut-Through

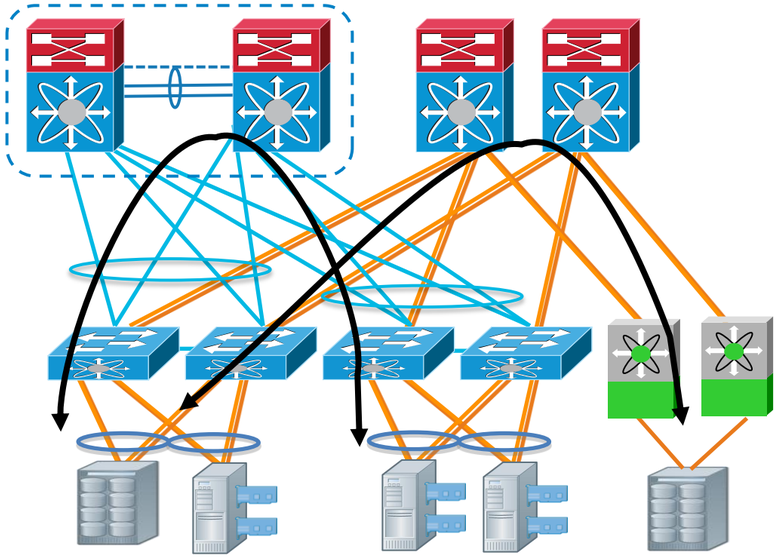
Nexus 5500 Packet Forwarding

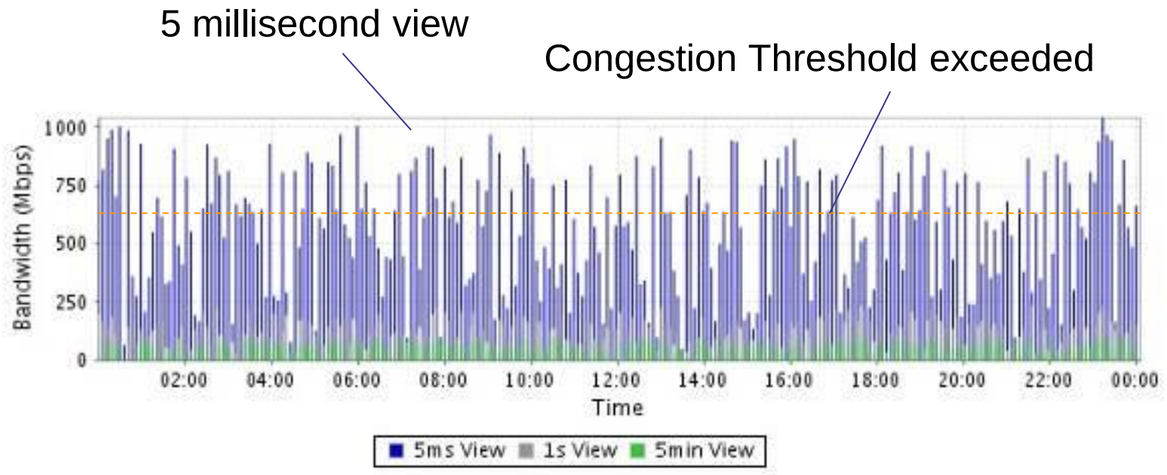
Minimising Latency ‘and’ Loss

- Why Cut-Through Switching?

  It is only one variable in overall fabric optimisation

- Designs target consistency of performance under variable conditions
- A balanced fabric is a function of maximal throughput ‘and’ minimal loss => “Goodput”





Data Centre Design Goal: Optimising the balance of end to end fabric latency with the ability to absorb traffic peaks and prevent any associated traffic loss

Nexus 5000/5500, 6004 & 2000 Architecture

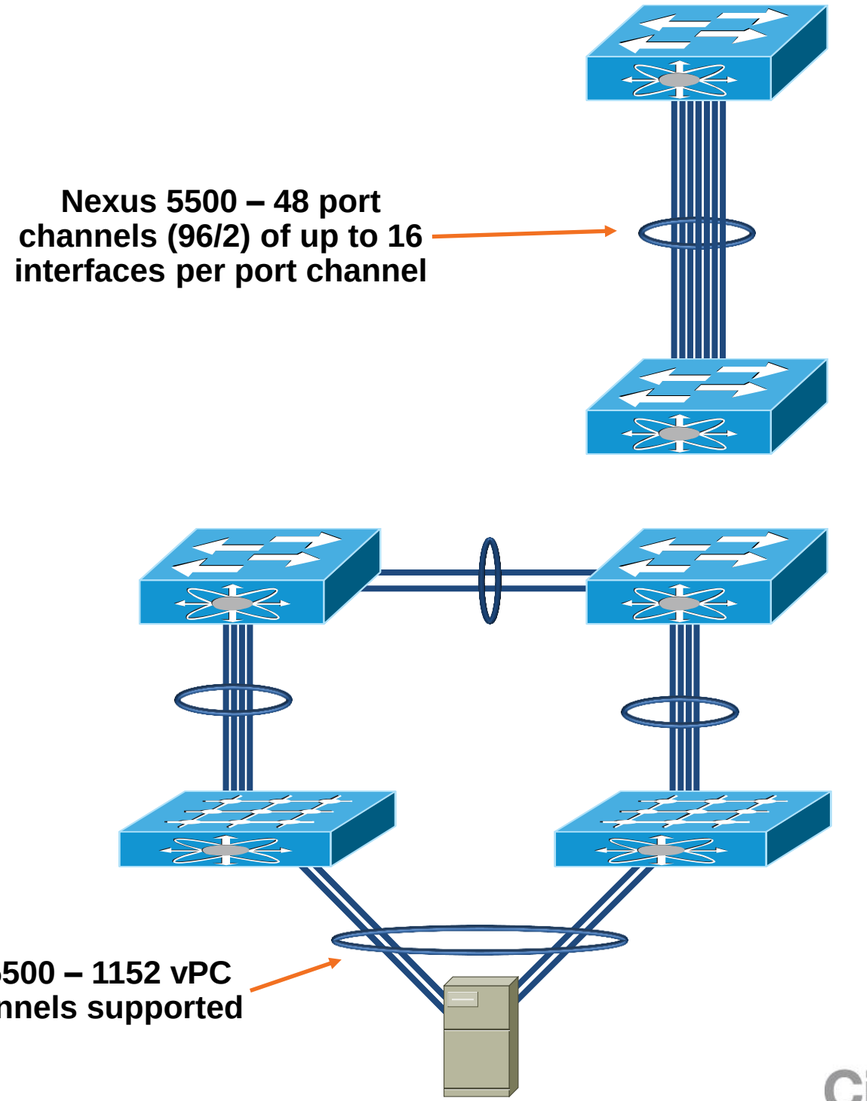
Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Architecture
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

Nexus 5000/5500 Port Channels

Nexus 5000/5500 Port Channel Types

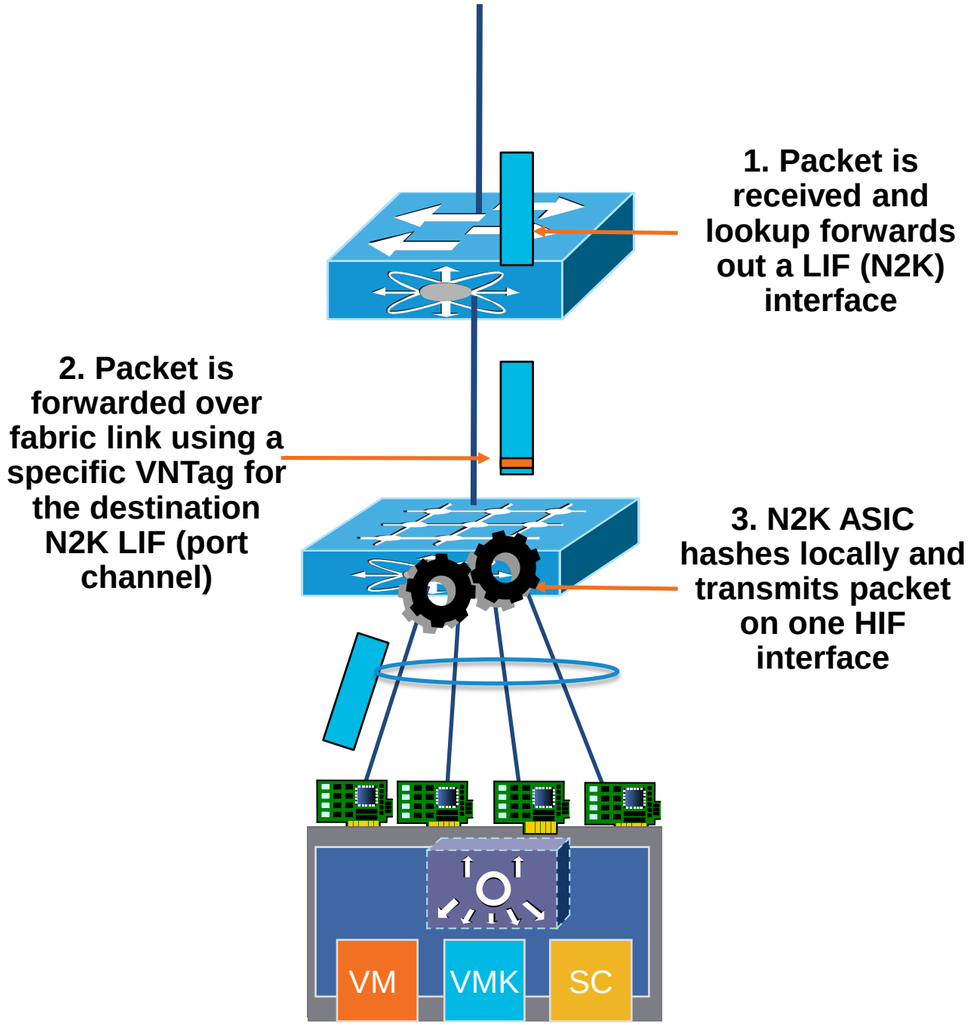
- Nexus 5010/5020 supports 16 port channels of up to 16 links each
- Nexus 5548/5596 support 48 port channels of up to 16 links each
- Nexus 2200 FEX supports 24 port channels of up to 8 links each
- Port channels configured on FEX do not take any resource from the Nexus 5000/5500 switch
- Nexus 5500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



Nexus 2000 Port Channels

Nexus 2248/2232 Port Channels

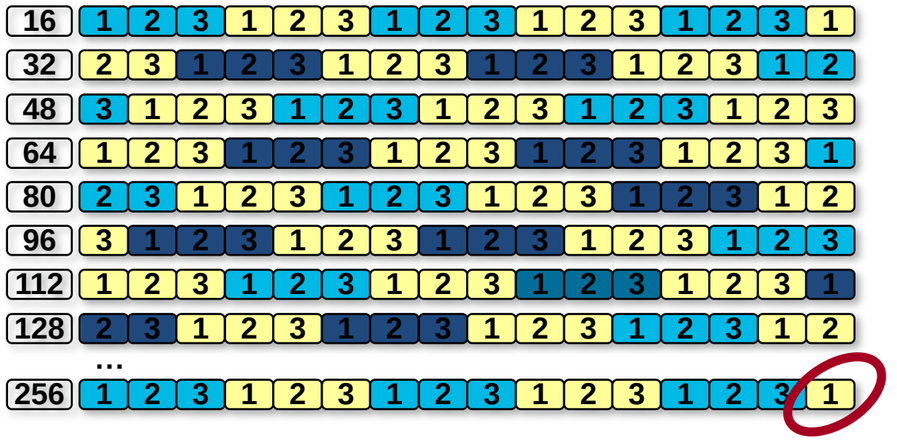
- Nexus 2200 series FEX support local port channels
- All FEX ports are extended ports (Logical Interfaces = LIF)
- A local port channel on the N2K is still seen as a single extended port
- Extended ports are each mapped to a specific VNTag
- HW hashing occurs on the N2K ASIC
- Number of ‘local’ port 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 x 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 utilise 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, 6004 & 2000 Architecture

Agenda

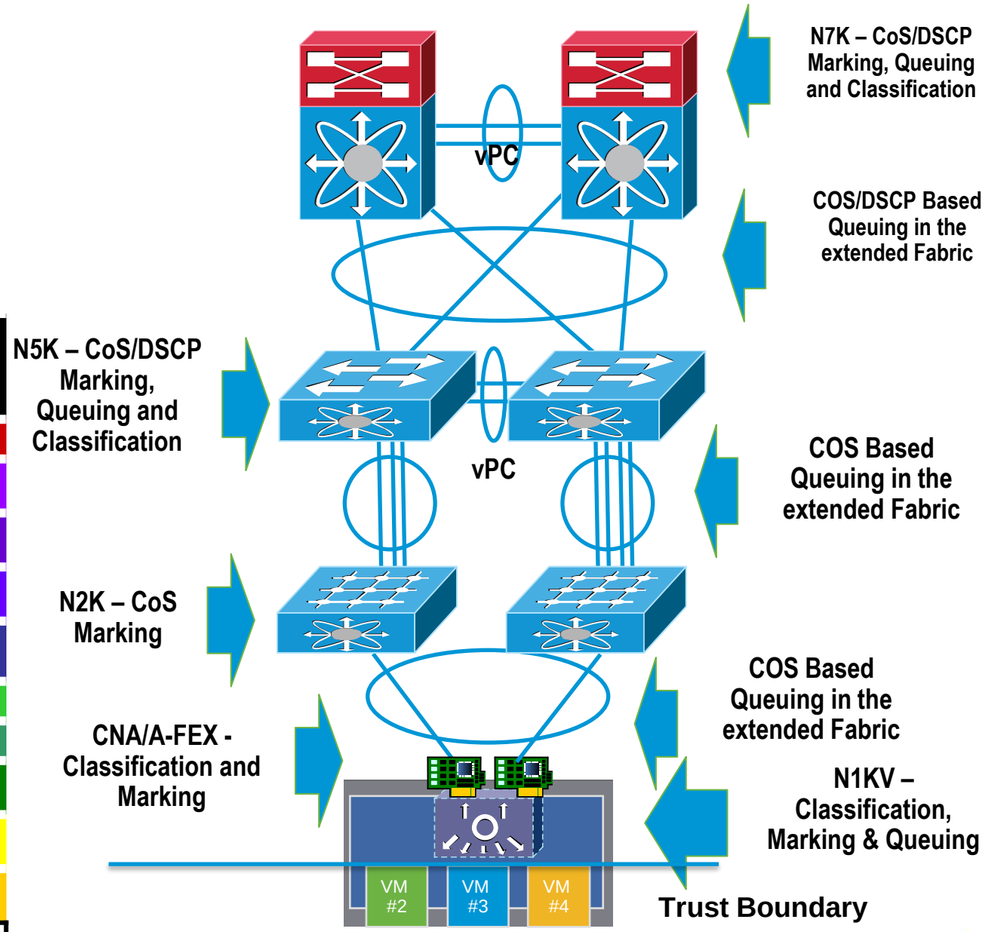
- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Architecture
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

Data Centre QoS Requirements

What do we trust, how do we queue and where do classify and mark?

- Data Centre 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

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



Data Centre QoS Requirements
CoS or DSCP?

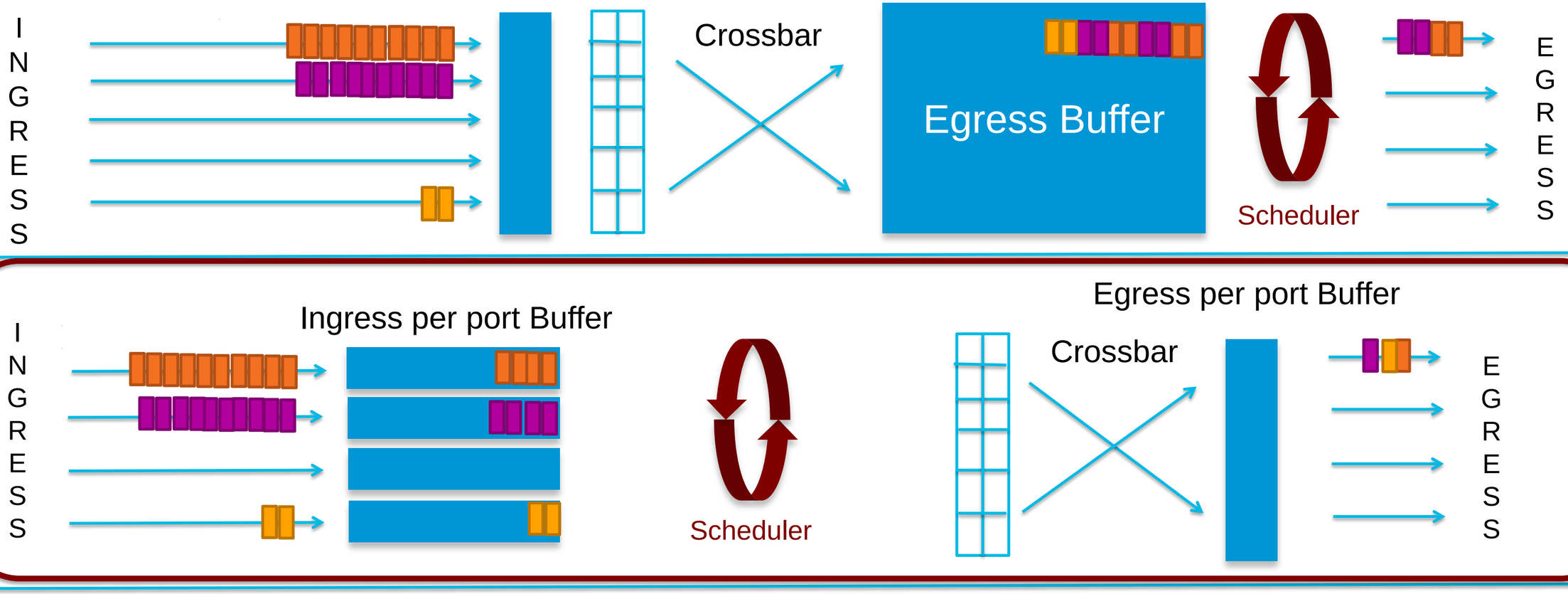
- We have non IP based traffic to consider again
  - FCoE – Fibre Channel Over Ethernet
  - RCoE – RDMA Over Ethernet
- DSCP is still marked but CoS will be required and used in Nexus Data Centre designs

PCP/COS	Network priority	Acronym	Traffic characteristics
1	0 (lowest)	BK	Background
0	1	BE	Best Effort
2	2	EE	Excellent Effort
3	3	CA	Critical Applications
4	4	VI	Video, < 100 ms latency
5	5	VO	Voice, < 10 ms latency
6	6	IC	Internetwork Control

IEEE 802.1Q-2005

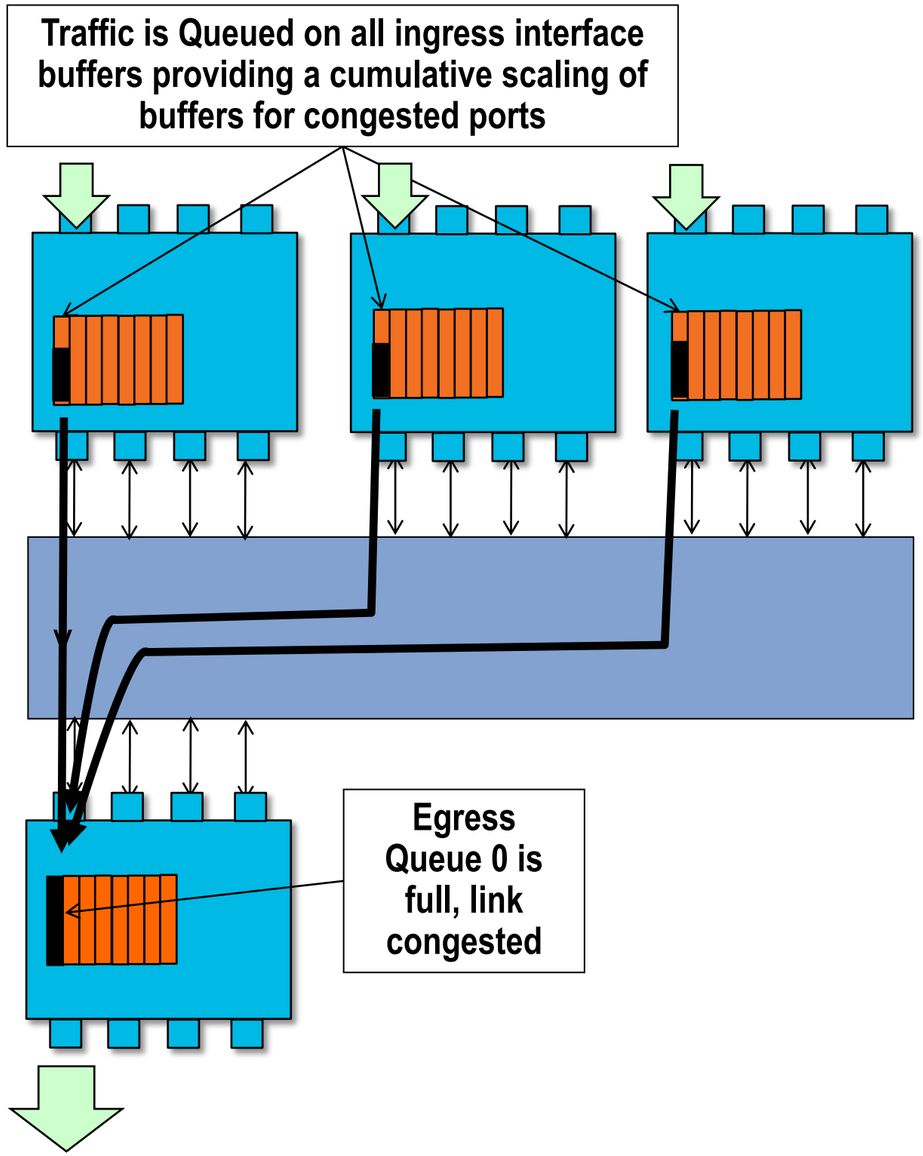
Switch Architectures

Three Approaches to Buffering



Nexus 5000 & 5500 QoS

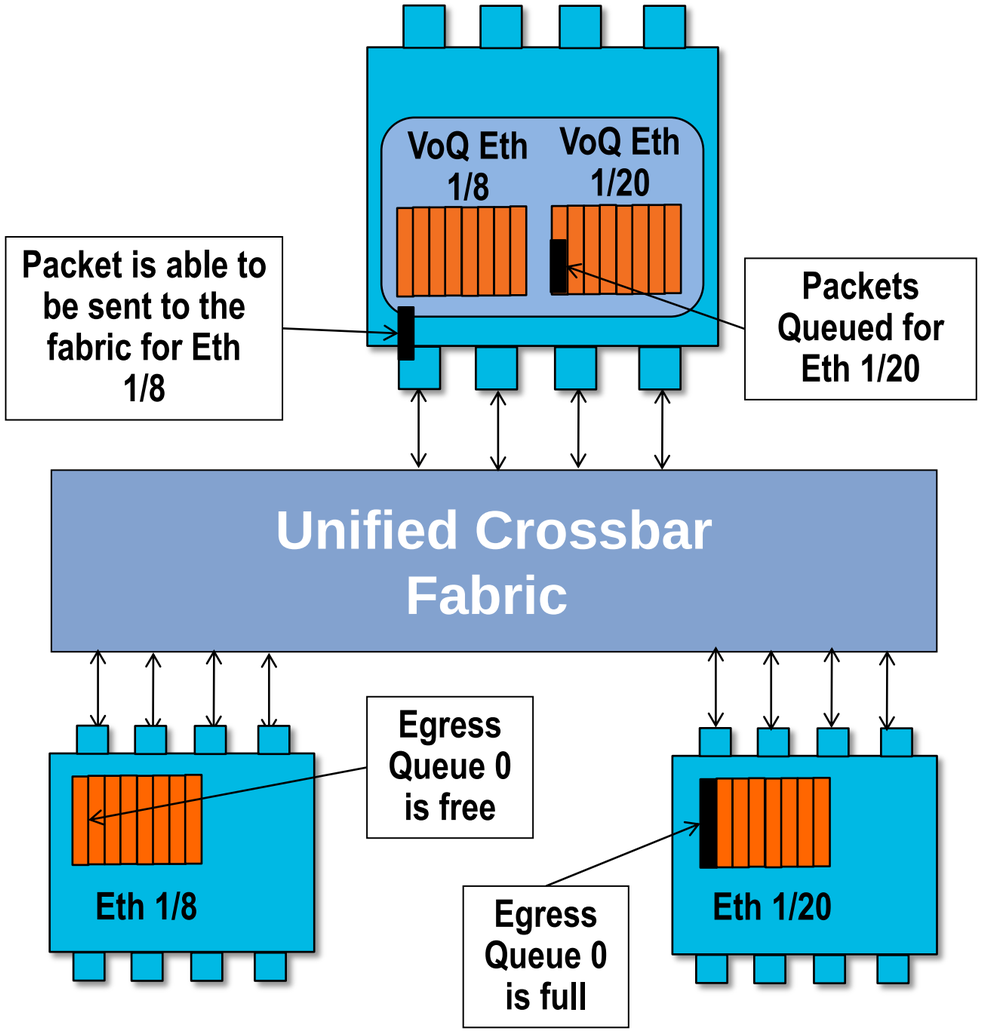
Packet Forwarding—Ingress Queuing



- Nexus 5000 and 5500 use an 8 Queue QoS model for unicast and multicast traffic
- Nexus 5000 and 5500 utilise 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

Nexus 5000 & 5500 QoS

Packet Forwarding—Virtual Output Queues

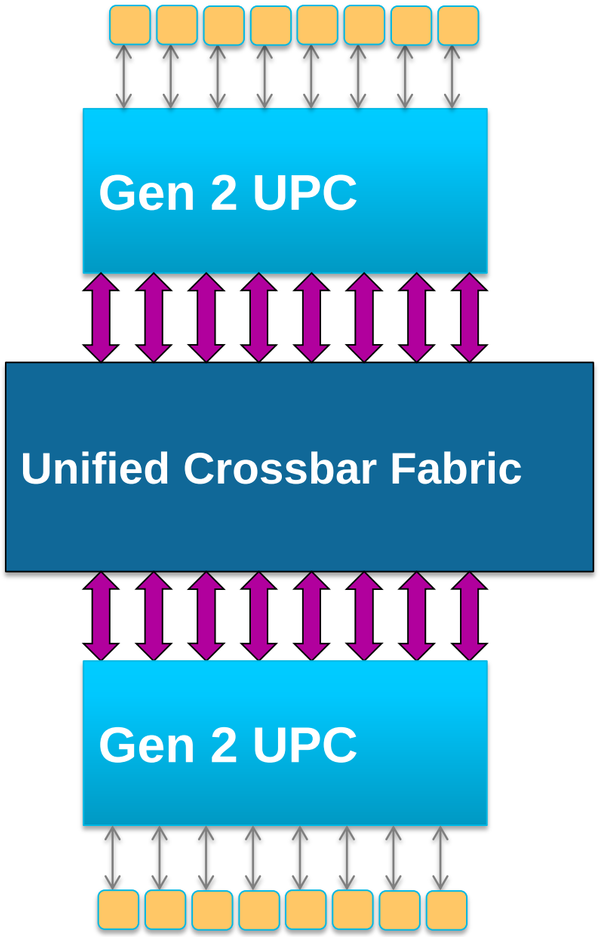


- Traffic is Queued on the Ingress buffer until the egress port is free to transmit the packet
- To prevent Head of Line Blocking (HOLB) Nexus 5000 and 5500 use a Virtual Output Queue (VoQ) Model
- Each ingress port has a unique set of 8 virtual output queues for every egress port (on 5596 there the 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 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) Buffering

For Your Reference

- 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

Class of Service	Ingress Buffer(KB)	Egress Buffer(KB)
Class-fcoe	78	19
Sup-Hi & Sup-Lo	18.0 & 18.0	9.6 & 9.6
User defined no-drop class of service with MTU<2240	78	19
User defined no-drop class of service with MTU>2240	88	19
User defined tail drop class of service with MTU<2240	22	19
User defined tail drop class of service with MTU>2240	29	19
Class-default	All remaining buffer	19

Default Classes

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 the manner in which bandwidth availability is reported to the arbiter)

Type (CLI)	Description	Applied To…
QoS	Packet Classification based on Layer 2/3/4 (Ingress)	Interface or System
Network-QoS	Packet Marking (CoS), Congestion Control WRED/ECN (Egress), (drop or no-drop, MTU), Buffer size	System
Queuing	Scheduling - Queuing Bandwidth % / Priority Queue (Egress)	Interface or System

Nexus QoS
Configuration Overview

```
N5k(config)# ip access-list acl-1
N5k(config-acl)# permit ip 100.1.1.0/24 any
N5k(config-acl)# exit
N5k(config)# ip access-list acl-2
N5k(config-acl)# permit ip 200.1.1.0/24 any
N5k(config)# class-map type qos class-1
N5k(config-cmap-qos)# match access-group name acl-1
N5k(config-cmap-qos)# class-map type qos class-2
N5k(config-cmap-qos)# match access-group name acl-2
N5k(config-cmap-qos)#
```

```
N5k(config)# policy-map type qos policy-qos
N5k(config-pmap-qos)# class type qos class-1
N5k(config-pmap-c-qos)# set qos-group 2
N5k(config-pmap-c-qos)# class type qos class-2
N5k(config-pmap-c-qos)# set qos-group 3
```

```
N5k(config)# system qos
N5k(config-sys-qos)# service-policy type qos input policy-qos
```

```
N5k(config)# interface e1/1-10
N5k(config-sys-qos)# service-policy type qos input policy-qos
```

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

```
N5k(config)# class-map type network-qos class-1
N5k(config-cmap-nq)# match qos-group 2
N5k(config-cmap-nq)# class-map type network-qos class-2
N5k(config-cmap-nq)# match qos-group 3
```

```
N5k(config)# policy-map type network-qos policy-nq
N5k(config-pmap-nq)# class type network-qos class-1
N5k(config-pmap-nq-c)# class type network-qos class-2
```

```
N5k(config-pmap-nq-c)# system qos
N5k(config-sys-qos)# service-policy type network-qos policy-nq
N5k(config-sys-qos)#
```

4. Define network-qos Class-Map
5. Define network-qos Policy-Map
6. Apply network-qos policy-map under *system qos* context

- 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

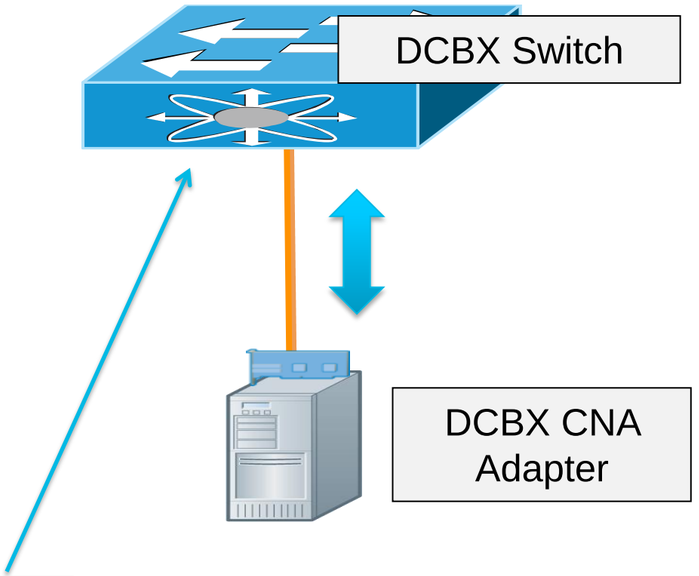
Data Centre Bridging Control Protocol

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 signalling of Ethernet and Fibre Channel
- DCBX is LLDP with new TLV fields
- The original pre-standard CIN (Cisco, Intel, Nuova) DCBX utilised 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



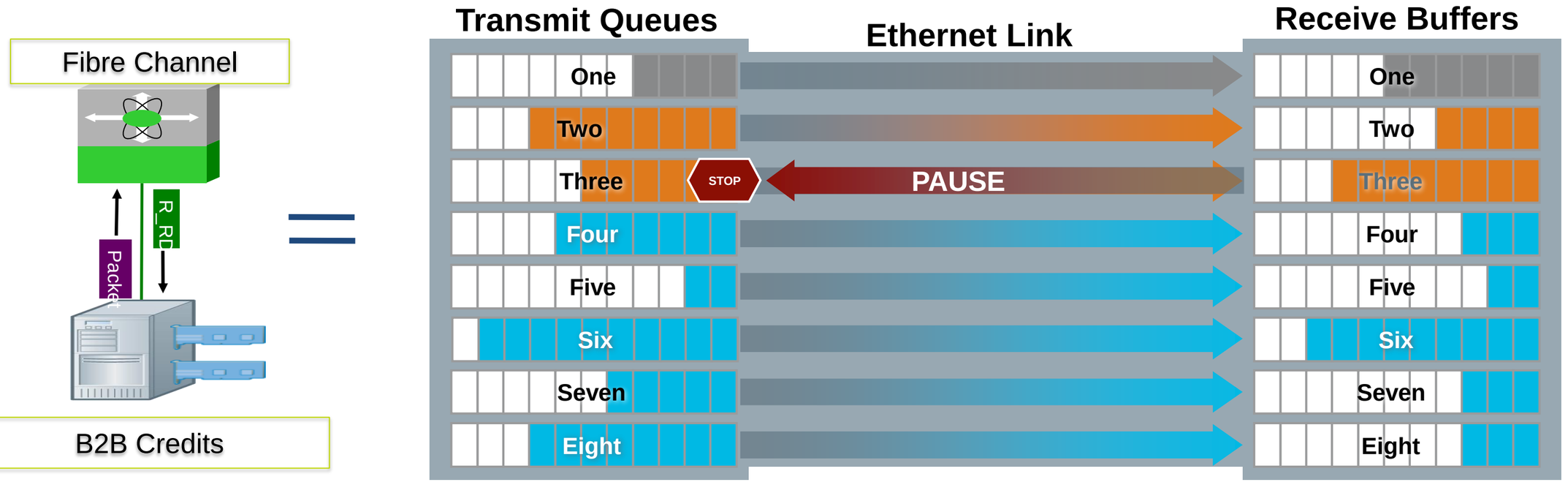

```
dc11-5020-3# sh lldp dcbx interface eth 1/40

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      00
<snip>
```

https://www.cisco.com/en/US/netsol/ns783/index.html

Priority Flow Control

FCoE Flow Control Mechanism – 802.1Qbb

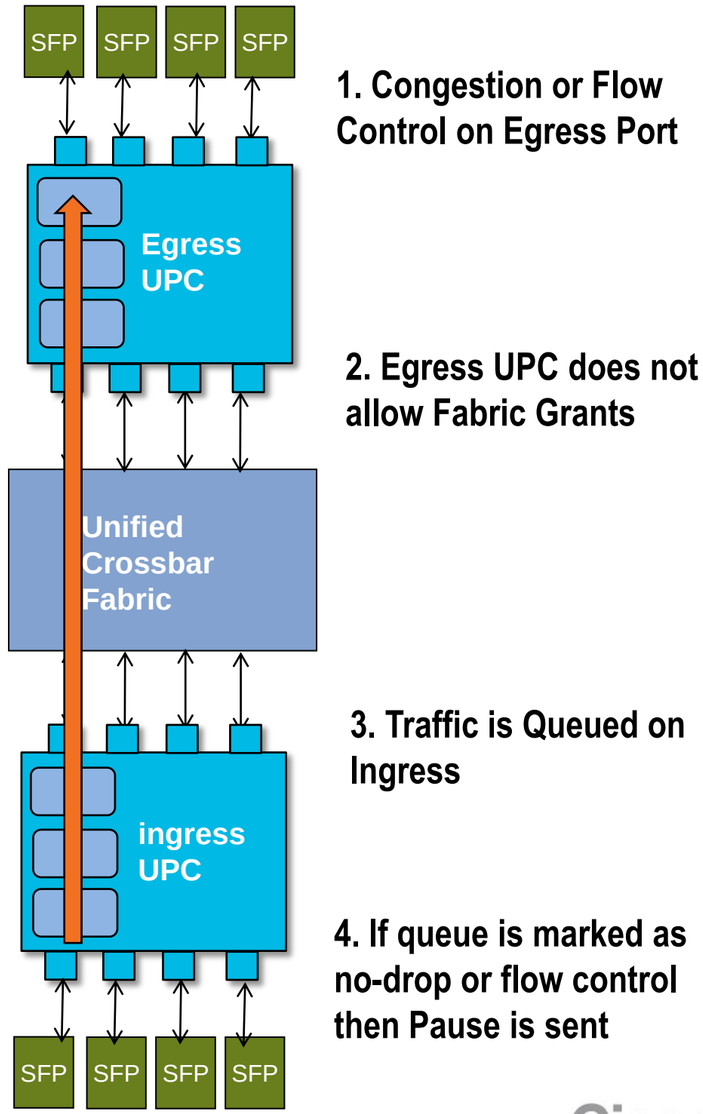
- Enables lossless Ethernet using PAUSE based on a COS as defined in 802.1p
- When link is congested, CoS assigned to “no-drop” will be PAUSED
- Other traffic assigned to other CoS values will continue to transmit and rely on upper layer protocols for retransmission
- Not only for FCoE traffic



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 behaviour (tail drop) but can be configured as no-drop



Nexus 5000/5500

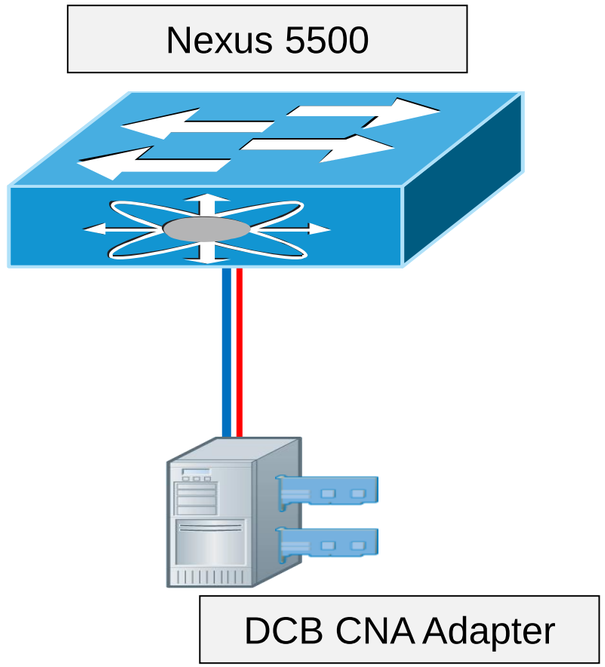
Priority Flow Control – Configuration

- 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
```



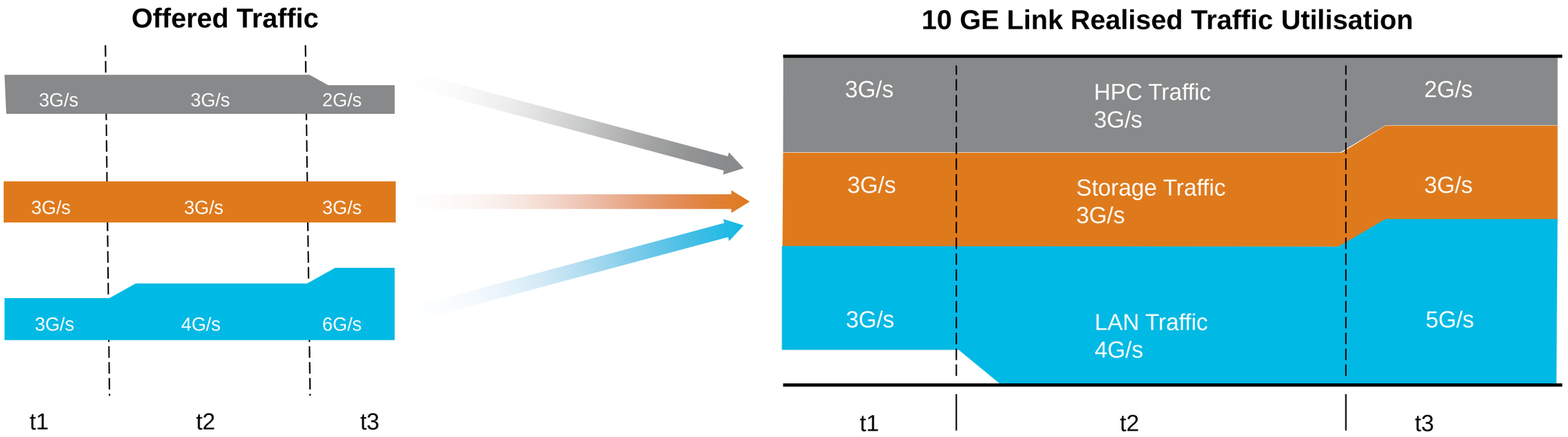
- Enabling the FCoE feature on Nexus 5548/96 does '***not***' create no-drop policies automatically as on Nexus 5010/20
- Must add policies under system QOS:

```
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 utilise its allocated bandwidth, it is available to other classes
- Helps accommodate for classes of a “bursty” nature



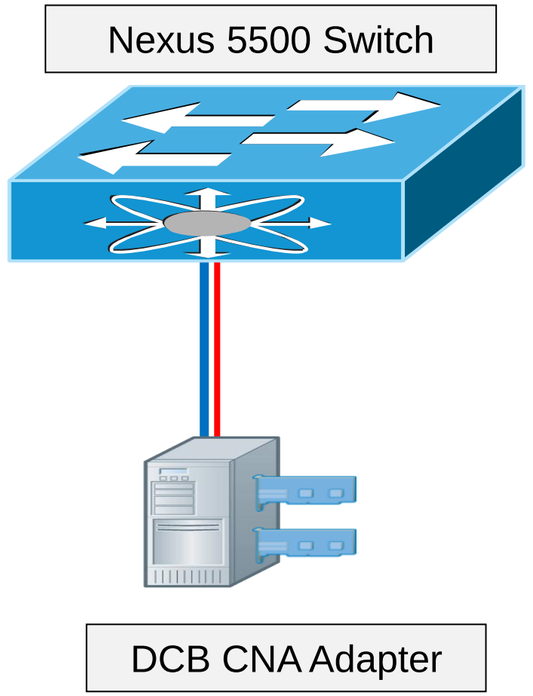
Nexus 5500 and iSCSI – DCB

PFC (802.1Qbb) & ETS 802.1Qaz

- **iSCSI TLV supported in the 5.2 release – 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, queuing and system behaviour
  - Apply policy maps

```
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
```

Identify iSCSI traffic



Nexus 5500 and iSCSI – DCB

PFC (802.1Qbb) & ETS 802.1Qaz

```
policy-map type queuing iscsi-in-policy
  class type queuing class-iscsi
    bandwidth percent 10
  class type queuing class-fcoe
    bandwidth percent 10
  class type queuing class-default
    bandwidth percent 80
```

Define policies to be signaled to CNA

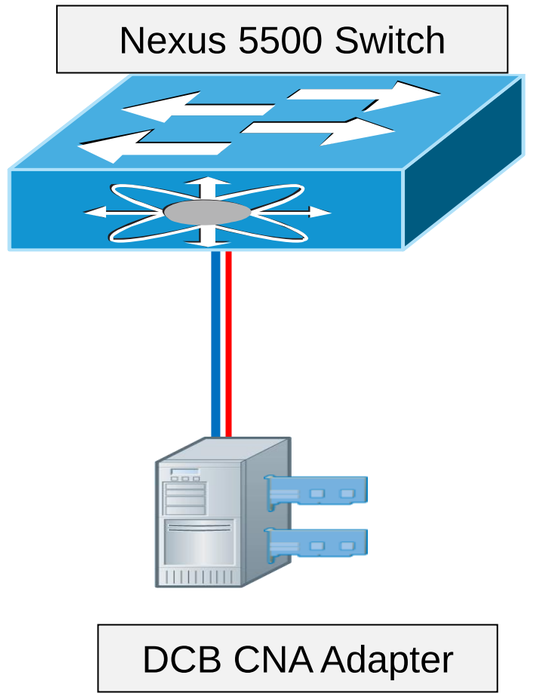
```
policy-map type queuing iscsi-out-policy
  class type queuing class-iscsi
    bandwidth percent 10
  class type queuing class-fcoe
    bandwidth percent 10
  class type queuing class-default
    bandwidth percent 80
```

Define switch queue BW policies

```
class-map type network-qos class-iscsi
  match qos-group 4

policy-map type network-qos iscsi-nq-policy
  class type network-qos class-iscsi
    set cos 4
    pause no-drop
    mtu 9216
class type network-qos class-fcoe
```

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

```
system qos
  service-policy type qos input iscsi-in-policy
  service-policy type queuing input iscsi-in-policy
  service-policy type queuing output iscsi-out-policy
  service-policy type network-qos iscsi-nq-policy
```

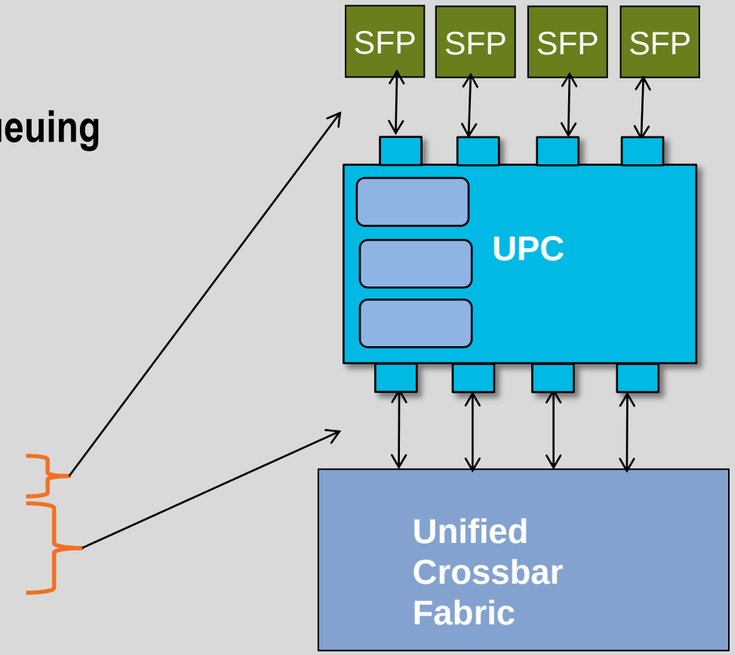


Nexus 5000/5500 QoS

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
```

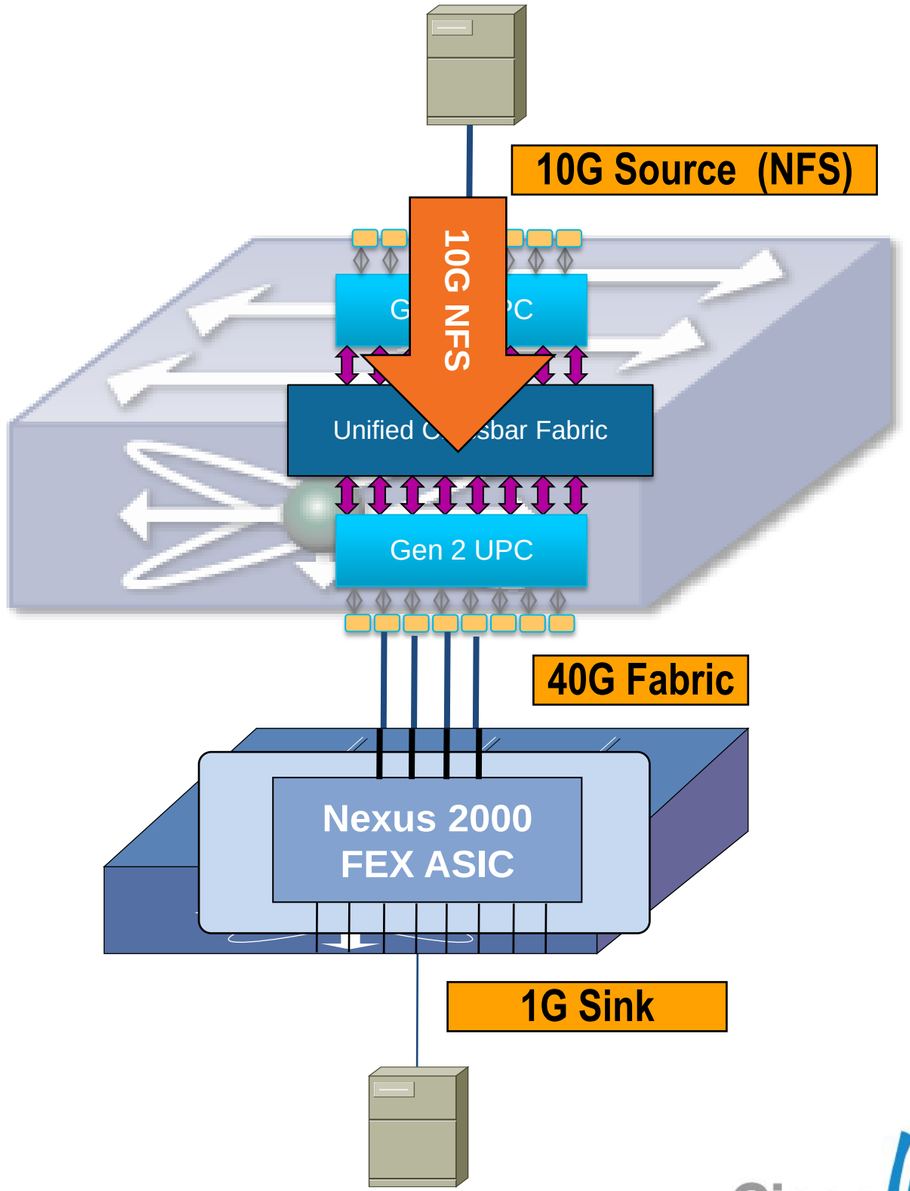
Egress (Tx) Queuing Configuration



Packets Arriving on this port but dropped from ingress queue due to congestion on egress port

Nexus 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

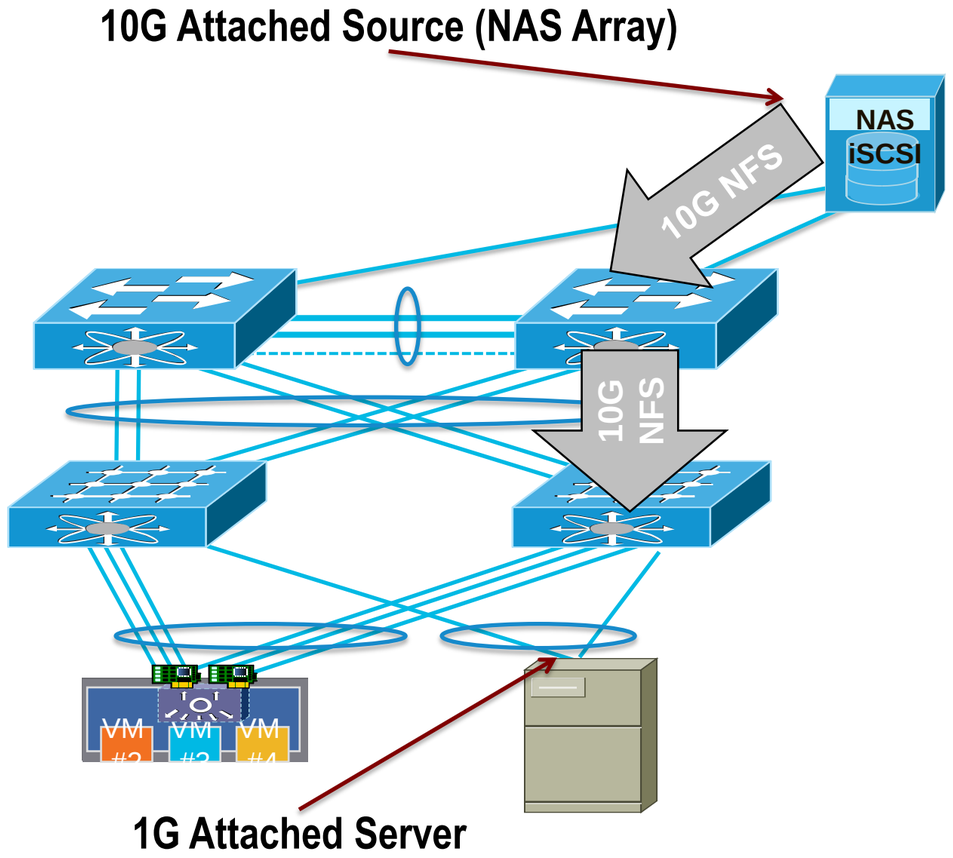



```
# Disabling the per port tail drop threshold
dc11-5020-3(config)# system qos
dc11-5020-3(config-sys-qos)# no fex queue-limit
dc11-5020-3(config-sys-qos)#

# 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*** utilises 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       buffer-size
  ---------+----------+-----------
  0          0          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
  No |Transmitted     |           |            |          |Tail Drop   |Depth
  ---+---------------+----------+-----------+---------+------------+-----
  2rx|          5863073|          0|           0|          0|     -      |    0
  2tx|    426378558047|   28490502|           0|          0|           0|    0
  ---+---------------+----------+-----------+---------+------------+-----

  <snip>
```

Ingress queue limit(Configurable)

Egress queues:
CoS to queue mapping
Bandwidth allocation
MTU

Egress queue limit(Configurable)

Per port per queue counters

Drop due to oversubscription

Introducing Nexus 6004

Nexus 5000/5500, 6004 & 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Overview
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

New! Nexus 6000 Series

2013

Cisco Nexus® 6000 Series

Nexus 6004

96 port 40G 4RU Switch

Nexus 5020
56-Port 2RU Switch

2008

Nexus 5548
48-Port 1RU Switch

2010

Nexus 5596T
10Gbase-T switch

2012

4-port QSFP+
GEM

2013

Cisco Nexus® 5000 Series

Nexus 5010
28-Port 1RU Switch

Nexus 5596
96-Port 2RU Switch

Nexus 6004

High Performance

- Line rate for L2 and L3 for all packet size
- Line rate SPAN
- 1 us port to port latency for all frame size
- Cut through switching at 10Gig and 40Gig
- FCoE at 40 Gig
- 25M buffer for 3 QSFP ports
- 31 active SPAN

High Scalability

- 96 ports at 40Gig
- 384 ports at 10 Gig
- Up to 256K MAC
- Up to 128K ARP
- 32K LPM routes
- 16K Bridge domain

Feature Rich

- L2 and L3 features
- vPC
- Fabric Path / TRILL
- Segment-ID
- Adapter FEX / VM FEX
- NAT

Visibility / Analytic

- Sampled Netflow
- Buffer monitoring
- Latency monitoring
- SPAN on drop
- SPAN on high latency
- Microburst monitoring

Data Centre Switching

Nexus 7000 and Nexus 6000: DC Considerations

Customer Requirements: Decision Points

- ✓ Virtualisation
- ✓ Scalability
- ✓ DCI/Mobility
- ✓ Environmentals
- ✓ L4-7 Services
- ✓ High Availability
- ✓ Latency
- ✓ Investment Protection

Decision Criteria in the Aggregation

Lead Platform: Modular, High-End Solution

Recommended when:

- ✓ Scale and Flexibility (100M/1G/10G/40G/100G/UP*)
- ✓ Highest Availability (HA)
- ✓ Investment Protection
- ✓ Multi-Protocol / Services VDC / OTV / MPLS / VPLS / LISP

Nexus 7000 Series

Up to 36 Tbps

Fixed, Mid-Range Solution

Recommended when:

- ✓ High density compact 10G/40G/100G*/UP*
- ✓ Low footprint & low power
- ✓ Low latency & jitter
- ✓ Extensive FEX Features

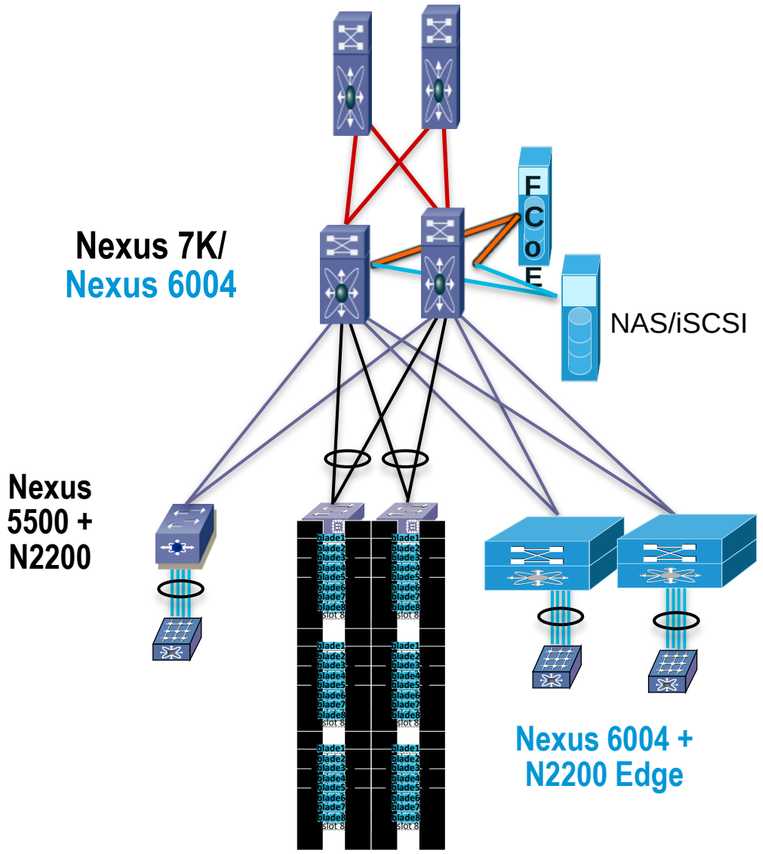
Nexus 6000 Series

Up to 7.68 Tbps

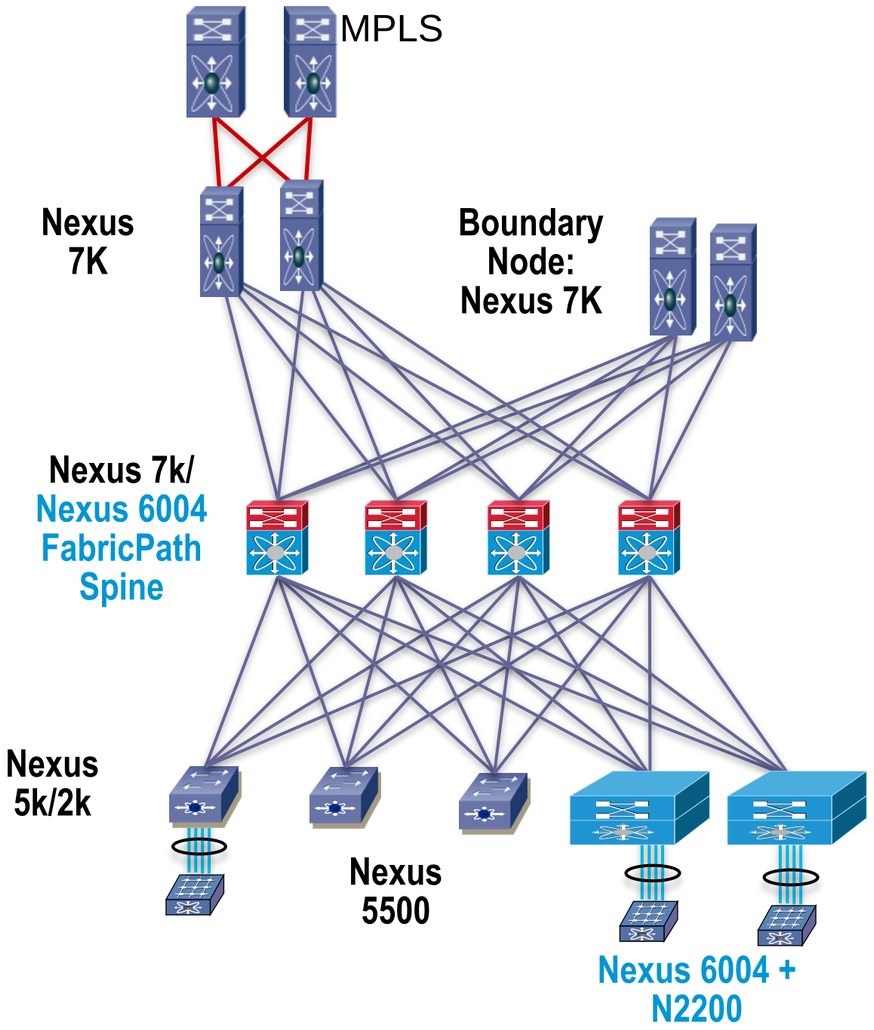
* Roadmap

Deployment Scenarios

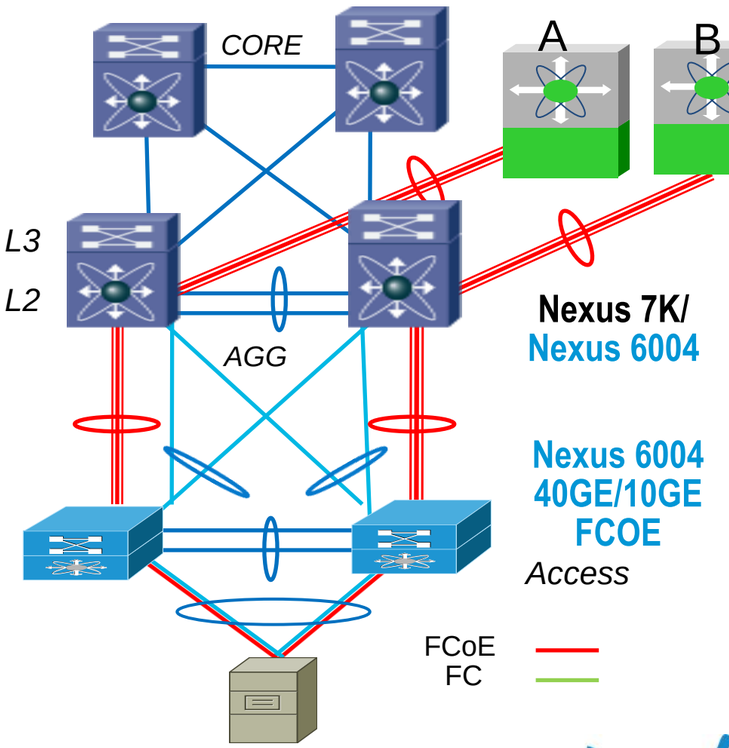
FEX Architecture



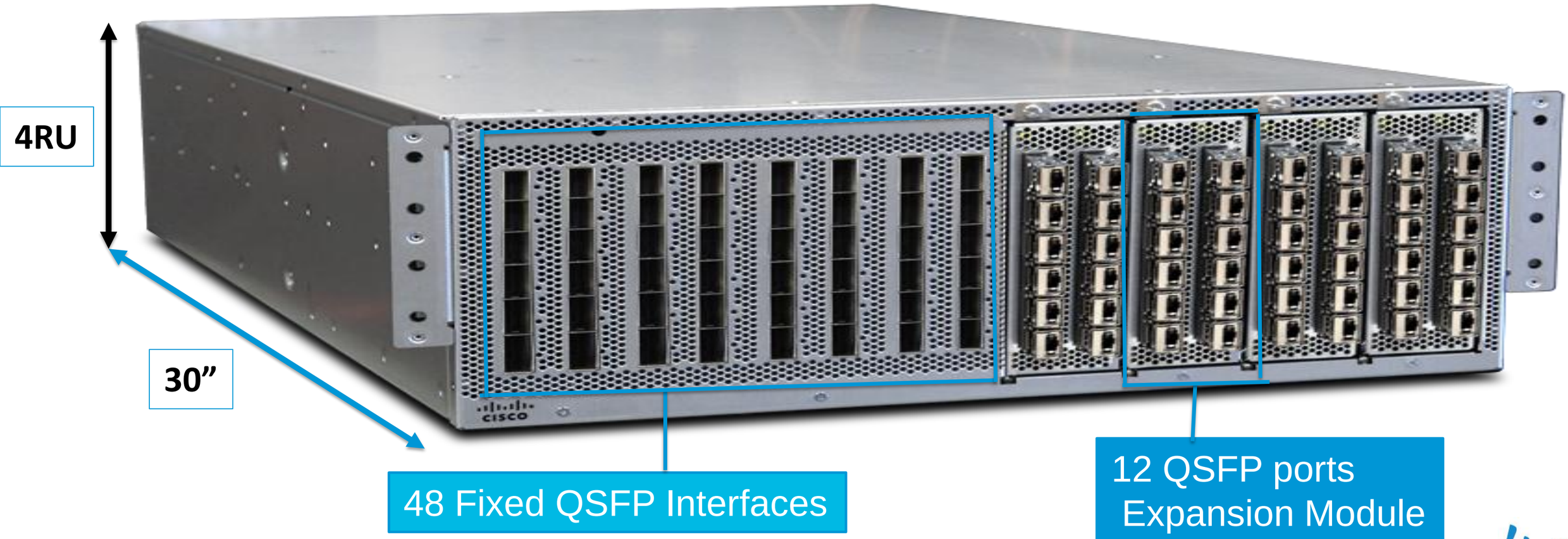
Large Scale Fabric (L2/L3)



Multi-Hop FCOE

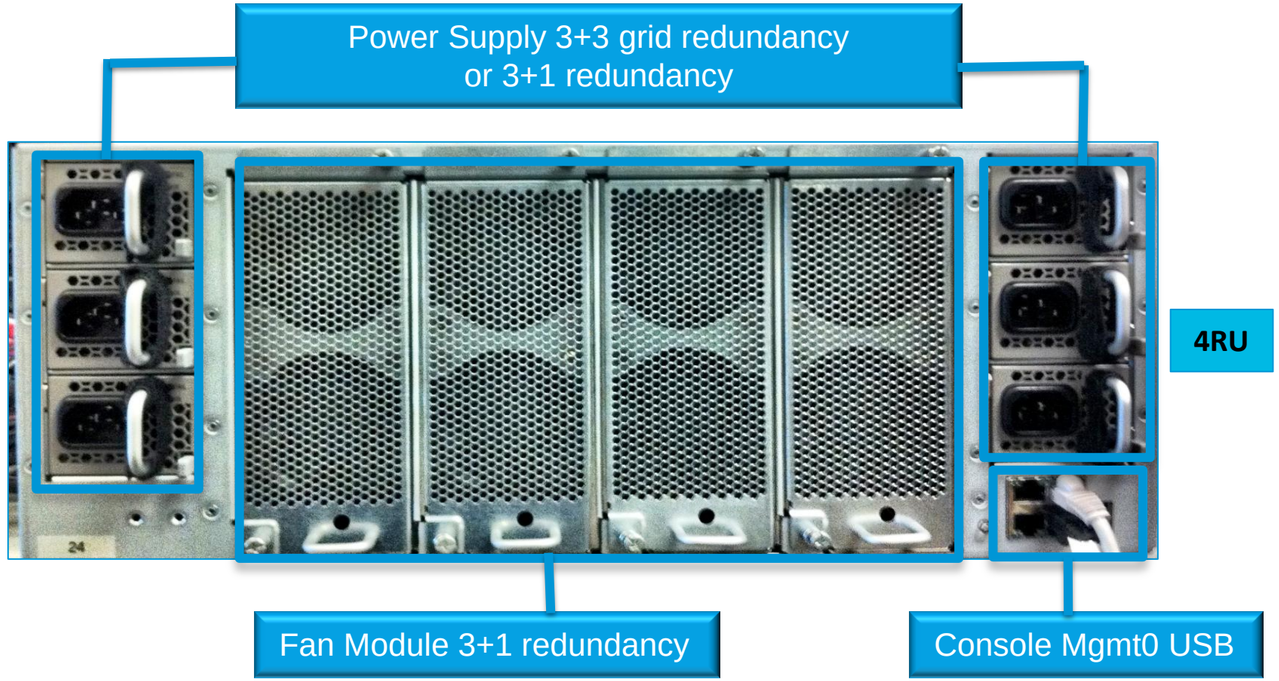


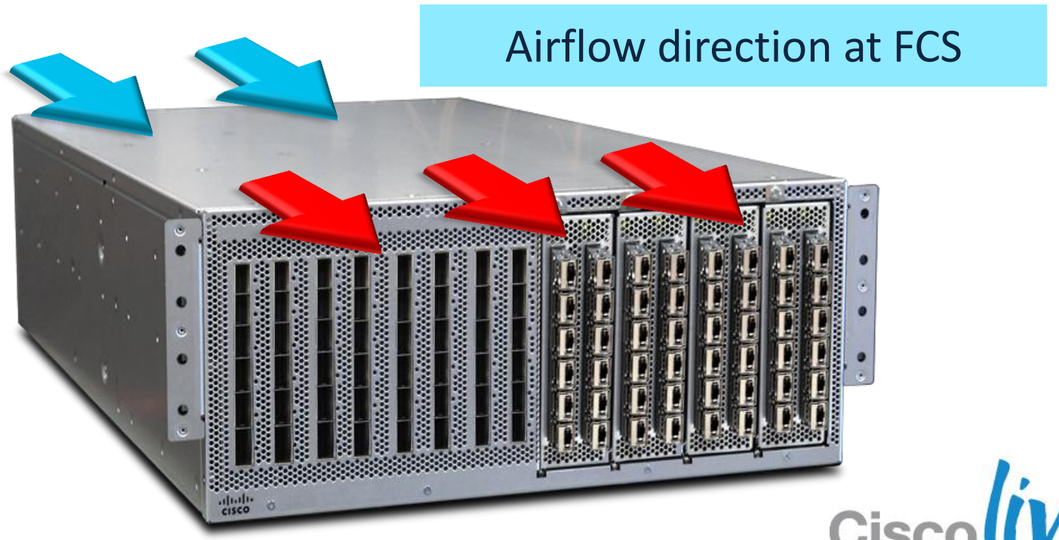
Nexus 6000 Chassis
Rear View



Nexus 6000 Chassis
Front View & Airflow

- Minimum 3 PS and 3 FAN are required
- Front to back or back to front air flow
- Port side exhaust at FCS
- Port side intake (Reversed airflow) with FCS + maintenance release - Q2CY13
- Different PS and Fan modules are required for different air flow directions





Nexus 6004 Physical Connections

- QSFP-SR4: 100m over OM3 MMF, 150m over OM4 MMF
- QSFP-CSR4: 300m over OM3 MMF, 400m over OM4 MMF

40G QSFP Interfaces

QSFP-SR4/QSFP-CSR4

QSFP-SR4/QSFP-CSR4

40G QSFP Interfaces

40G QSFP Interfaces

1M, 3M, 5M passive
7M, 10M Active

40G QSFP Interfaces

40G QSFP Interfaces

QSFP-SR4/QSFP-CSR4

10GE SFP+ interfaces

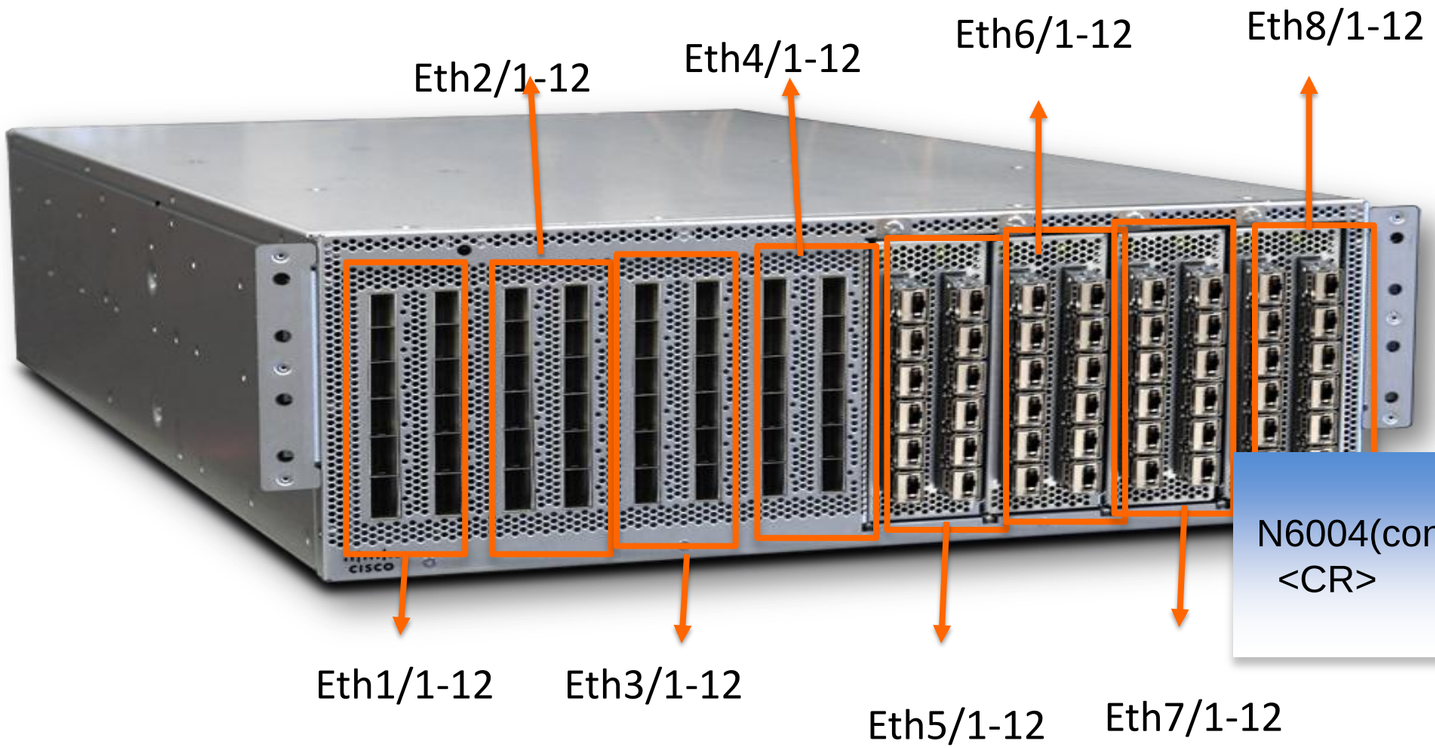
40G QSFP Interfaces

QSFP-4x10G-7M
QSFP-4x10G-10M

10GE SFP+ interfaces

Port Speed Configuration

- By default ports are in 40GE mode
- Port speed can be changed at group of 3 QSFP ports.
- The group of 12 QSFP ports need to be reset after port speed change.



40G interface
Ethernet 8/1

10G Interface
Ethernet 8/1/1
Ethernet 8/1/2
Ethernet 8/1/3
Ethernet 8/1/4

```
N6004(config)# interface breakout slot 8 port 1-12 map 10g-4x ?
  <CR>
```

40Gig to 10Gig Conversion

1. Apply global CLI to change interface types to 10GE

   Every three contiguous QSFP interfaces resides on one UPC ASIC

   The port range specified in the CLI has to include all ports on the ASIC.

2. Power off the affected modules
   - Every group of 12 QSFP interfaces are managed as one module, even for the fixed interfaces

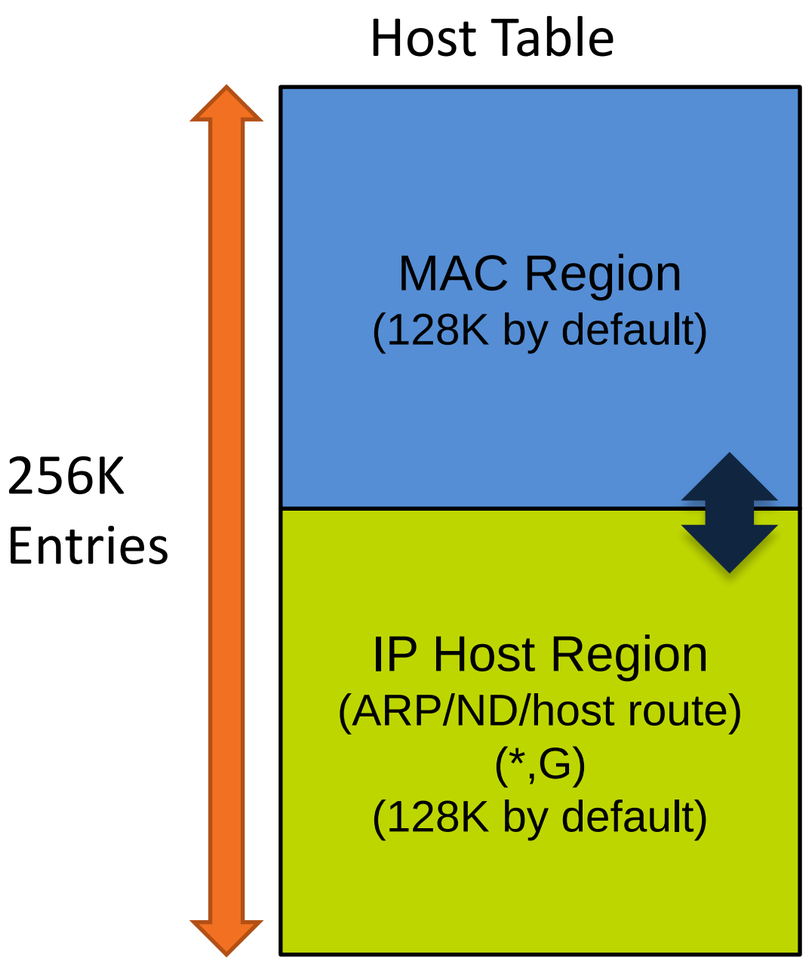
3. Power on the affected modules

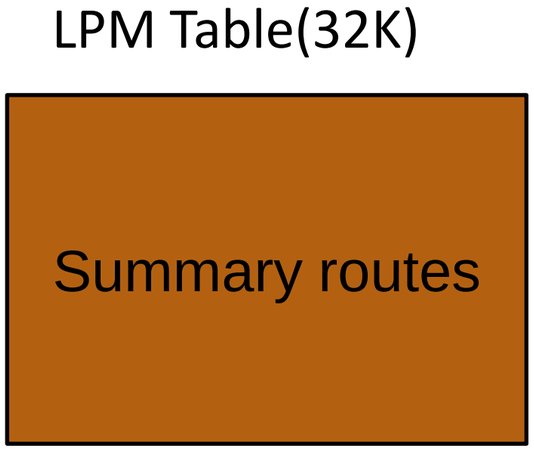
```
N6004(config)# interface breakout slot 1 port 1-6 map 10g-4x
```

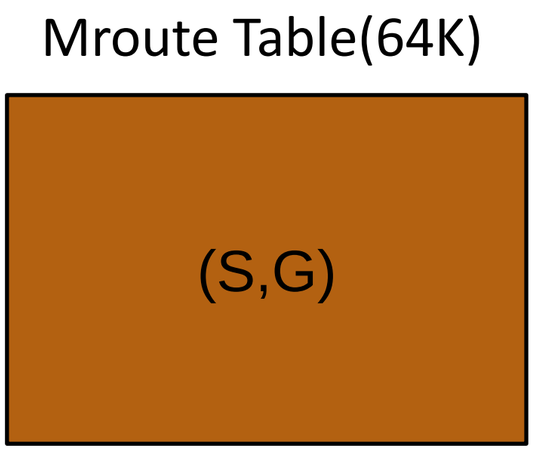
```
N6004(config)# poweroff module ?
  <1-8>  Please enter module number

N6004(config)# poweroff module 1
```

```
N6004(config)# no poweroff module 1
N6004(config)#
```

Nexus 6004 Key Forwarding Tables



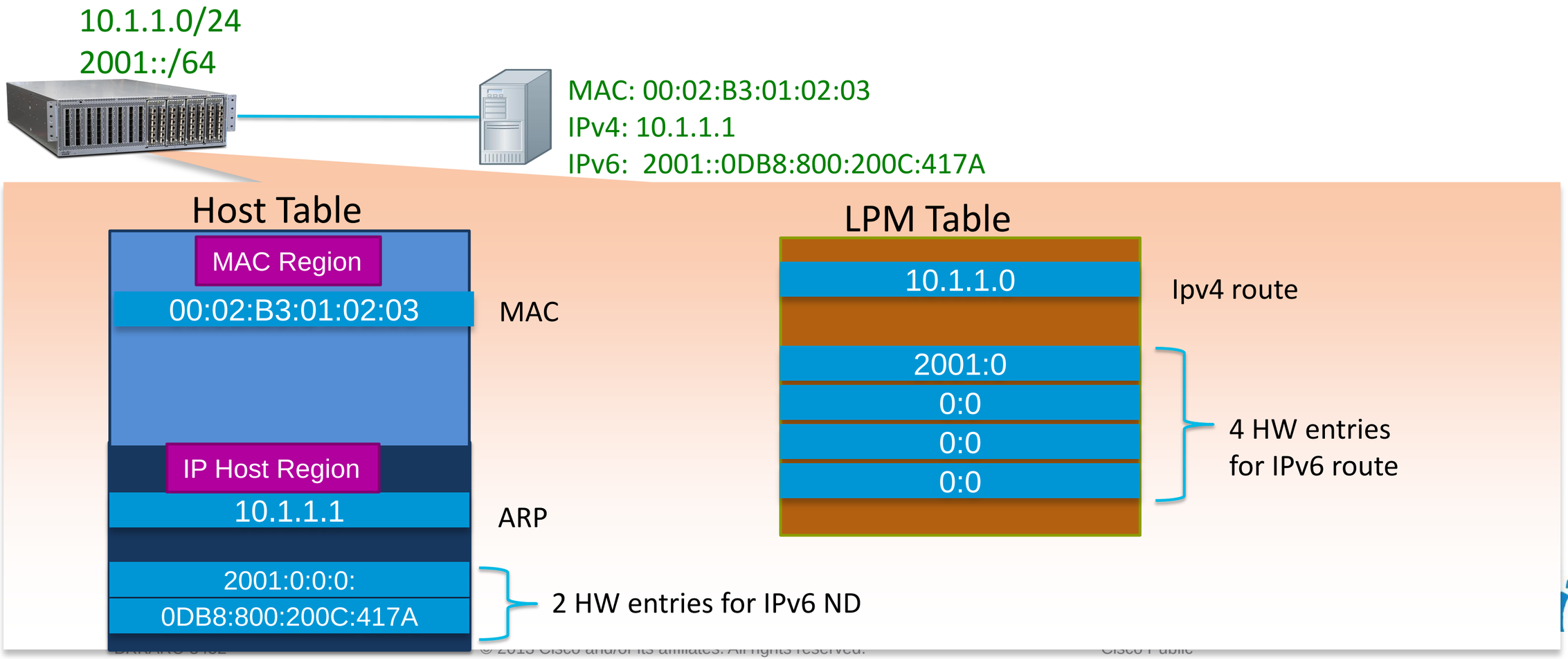




- Host Table: 256K entry hashing table. Actual capacity is slightly below 256K.
- Host Table is shared between MAC, ARP/ND and /32 host route and (*,G)
- Host Table default carving: 128K MAC, 128K IP host
- LPM Table: 32K entries. Summary routes
- Mroute table: 64K entries.

Nexus 6004 Unicast Table Scaling

- Each IPv6 ND consumes 2 entries in IP host region in host table
- Each IPv6 route consumes 4 entries in LPM table



Nexus 6004 Host Table Scaling

Deploy Scenario	Scalability
L2 switch	256K MAC
L2L3 Gateway with IPv4 only*	128K hosts
L2L3 Gateway with IPv6 only*	85K hosts
L2L3 Gateway with dual stack*	50K hosts
Leaf/Border Node	256K minus local MAC

- In L2 mode the IGMP snooping is stored at IP host region. So the actual MAC region will be less than 256K. At FCS software statically allocate 128K entries for MAC and 128K entries for IP host(ARP/ND/host route)
- Host table is hashing table. Actual capacity will be slightly below the number in the table.

* Assume one IPv4 or IPv6 per host. Hardware scaling number.

Nexus 6004 Control Plane

Nexus 5000

Nexus 5500

Nexus 6000

CPU Scaling – Feature Complexity and Corresponding Control Plane Growth

- CPU - 1.66 GHz Intel LV Xeon
- DRAM - 2 GB of DDR2 400 (PC2 3200) in two DIMM slots
- On-Board Fault Log - 64 MB of flash for failure analysis
- 1G Flash
- NVRAM - 2 MB of SRAM: Syslog and licensing information

- CPU - 1.7 GHz Intel Jasper Forest (Dual Core)
- DRAM - 8 GB of DDR3 in two DIMM slots
- On-Board Fault Log (OBFL) - 64 MB
- 2G Flash
- NVRAM - 6 MB of SRAM to store Syslog and licensing information

- Built-in single supervisor
- ISSU with L2 at FCS
- CPU – 4 Core Intel Gladden 2.0GHz
- DRAM - 4 DIMM slots 16GB by default.
- 8G Flash for NX-OS software and user files
- 6MB NVRAM 64MB OBLF(On-board Fault Logging)

Nexus 6004 vs. Nexus 5500
Performance and Scalability

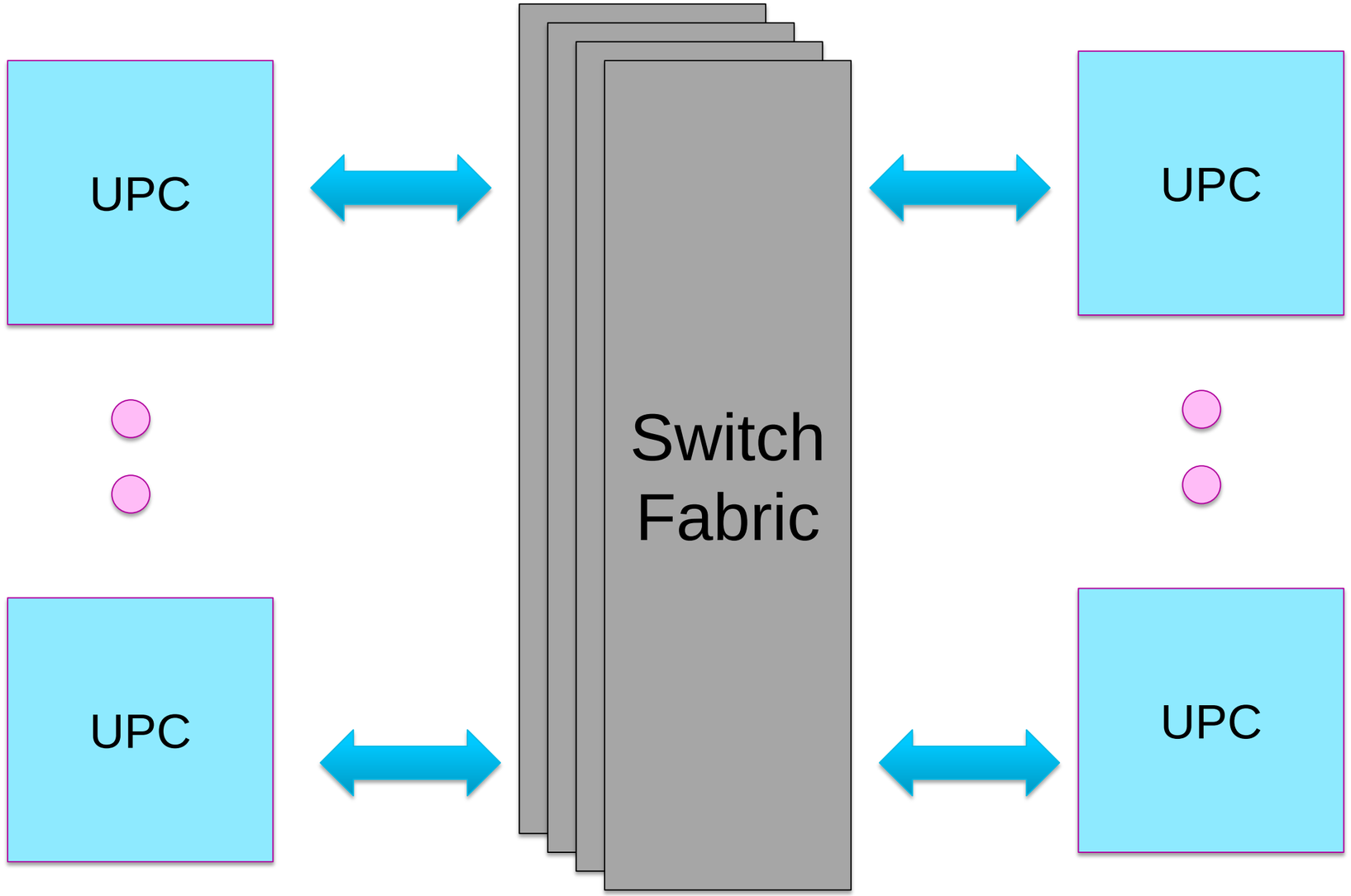
	Nexus 55xx	Nexus 6004
Latency	~1.8us	**~1us**
MAC table	32K	**256K MAC/ARP (flexible)**
LPM Routes	16K	**32K**
Hosts	16k	**128K**
Multicast route	8K	**32K**
Bridge Domains	4K	**16K**
ACLs	4K flexible	**4K flexible**
IGMP Snooping groups	4K	**32K**
ECMP	64 way	**1K**
VRFs	1K	**4K**
SPAN	4	**31, 16 can be ERSPAN**
Buffer	640K per port dedicated	**25MB per 3 QSFP**
VDC	N/A	**Possible**

Nexus 5000/5500, 6004 & 2000 Architecture

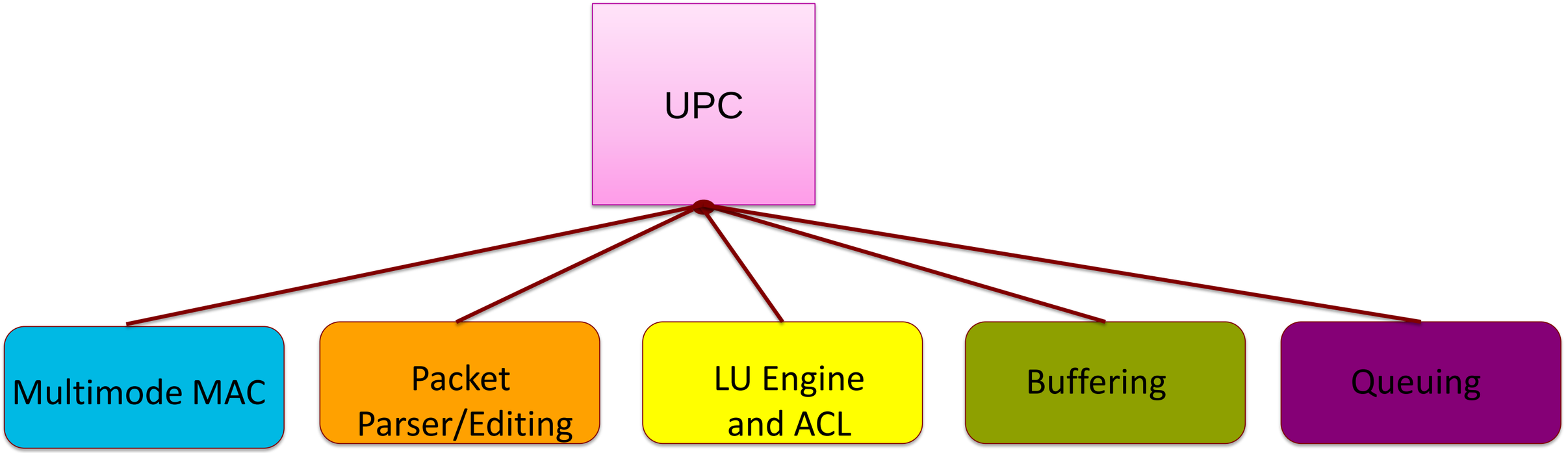
Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Overview
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

Nexus 6004 Architecture
Two Stage Fabric



Nexus 6004 Unified Port Controller (UPC)



- Packet protocol function: encoding/ decoding for physical medium dependent
- Flow control, PFC, 802.3x, B2B credit for Fibre Channel

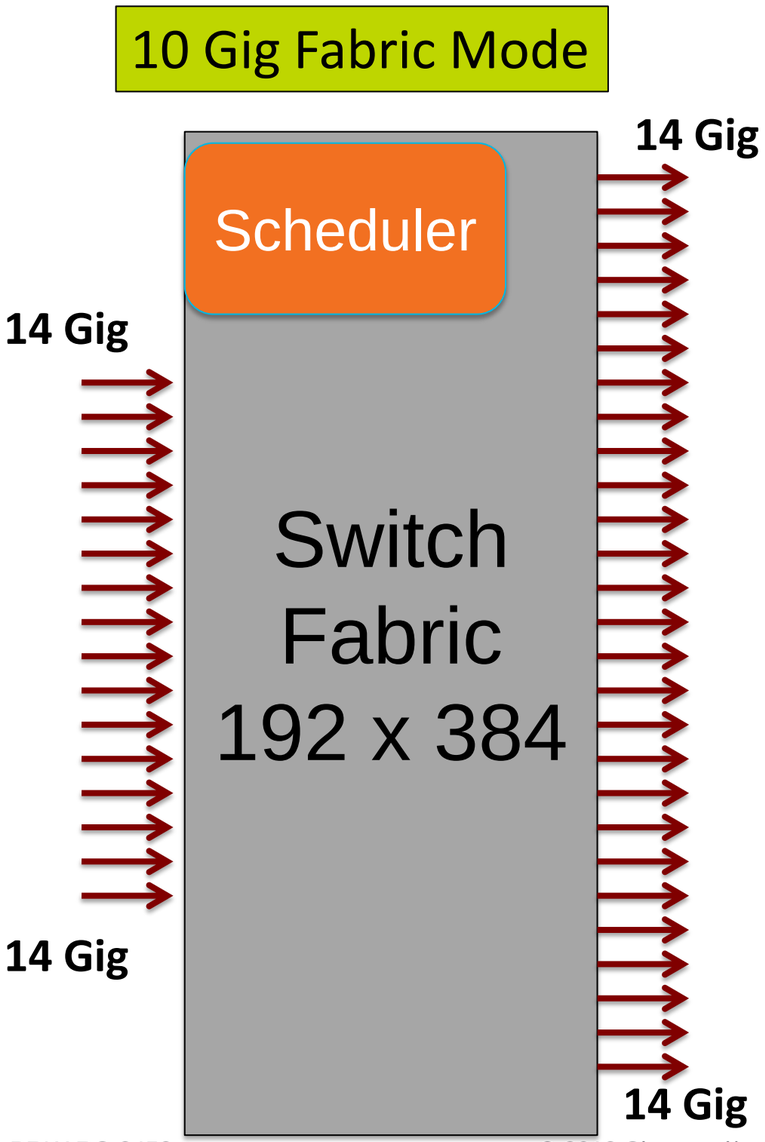
- Parse the packet, extract relevant fields, form a lookup vector and pass it to LU
- Create SPAN copy if needed
- Put 1588 clock

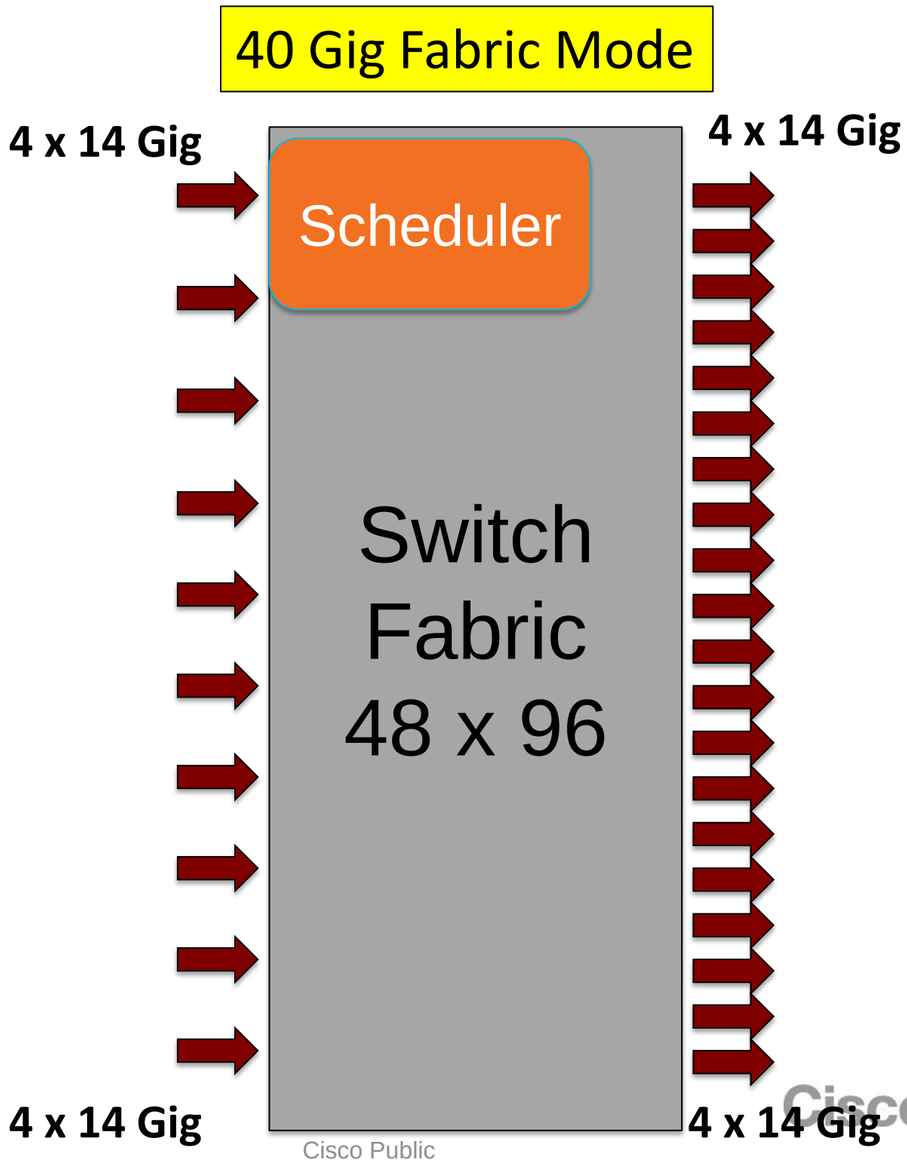
- Perform all the forwarding decision based on the vector received from FW
- AC module performs ACL, QoS classification, FC zoning, NAT, PBR

- Packet buffering on ingress and egress
- Pass queuing information to ingress QS
- Transmit frame to Fabric Interface
- Perform DWRR based on output queues.

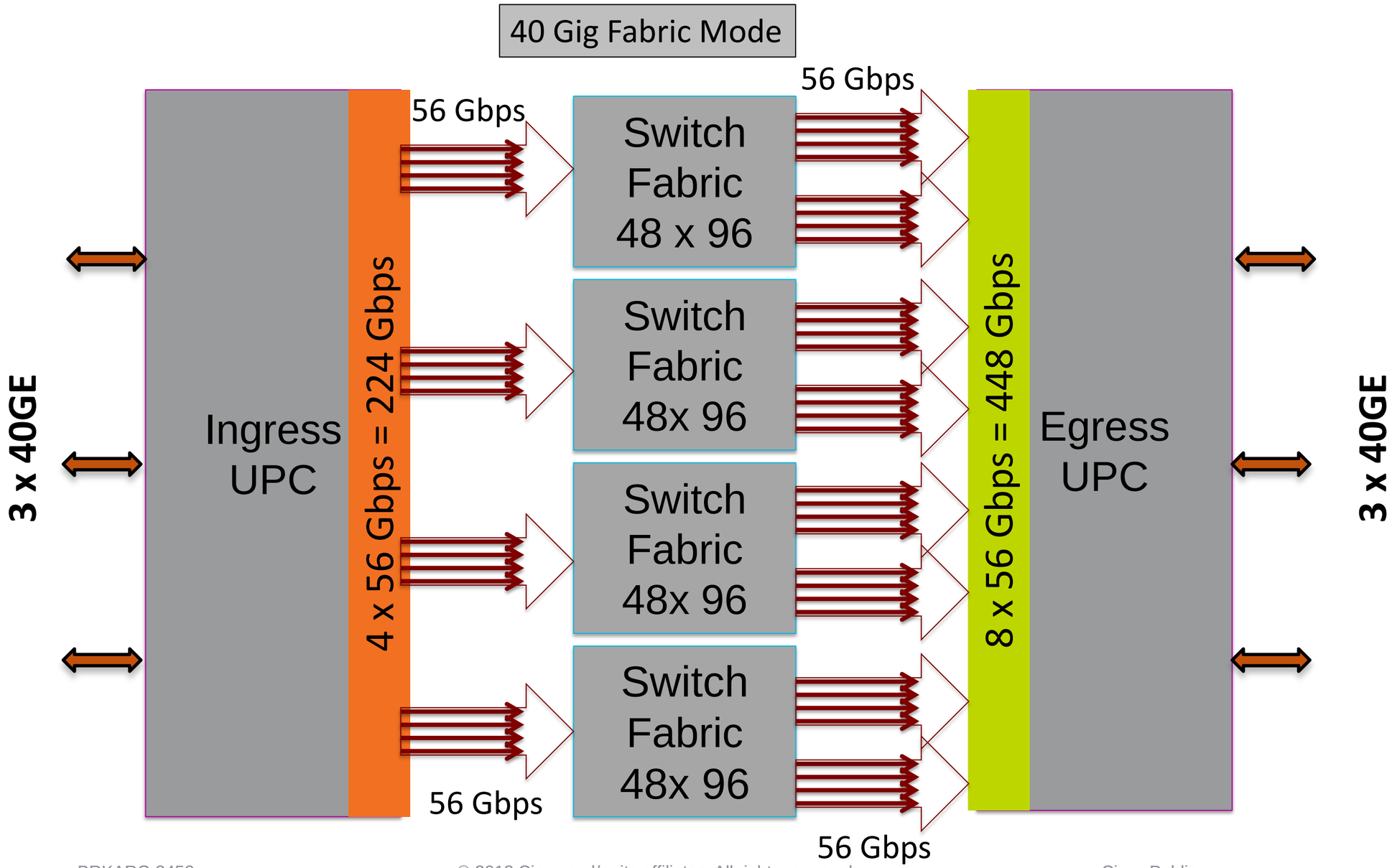
- Maintains ingress VoQs for unicast, multicast and SPAN.
- Maintains per port egress queues for unicast and multicast.
- Performs egress replications

Nexus 6004 Switch Fabric

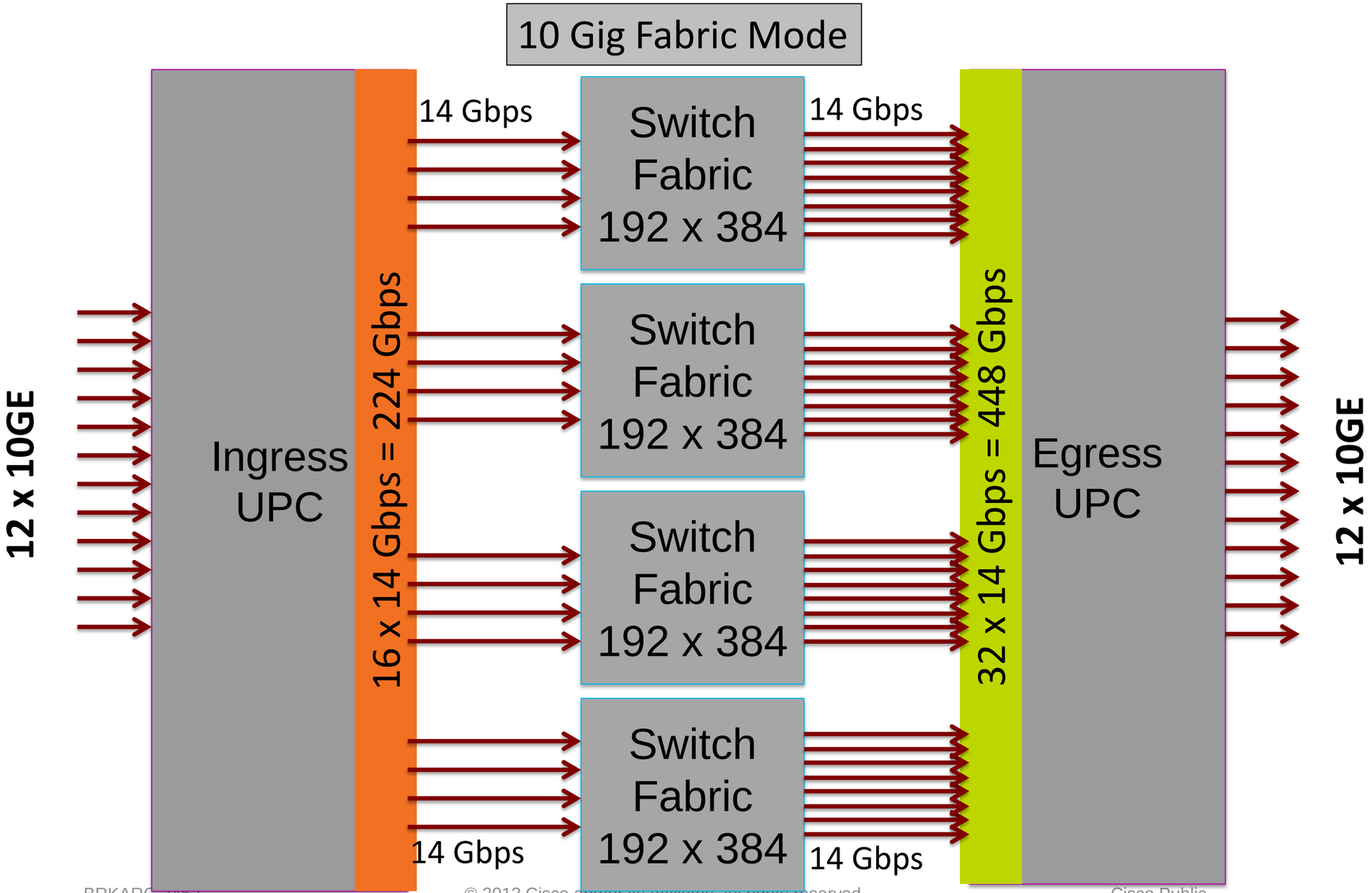




Nexus 6004 Internal Architecture
Fabric Mode 40 Gig



Nexus 6004 Internal Architecture
Fabric Mode 10 Gig



Cut Through vs. Store & Forward
Switch Fabric Versatility

- Depending on the port speed combination and switch fabric mode, Nexus 6004 performs either cut through switching or store& forward switching
- In **10 Gig fabric mode**, we do cut through switching when the egress is 10Gig

Ingress \ Egress	10GE	40GE
10GE	Cut-through	Store-N-Forwarding
40GE	Cut-through	Store-N-Forwarding

- In **40 Gig fabric mode**, we do cut through switching when the ingress is 40 Gig

Ingress \ Egress	10GE	40GE
10GE	Cut-through OR Store&Forwarding	Store-N-Forwarding
40GE	Cut-through	Cut-through

Fabric Mode, 40Gig or 10Gig?

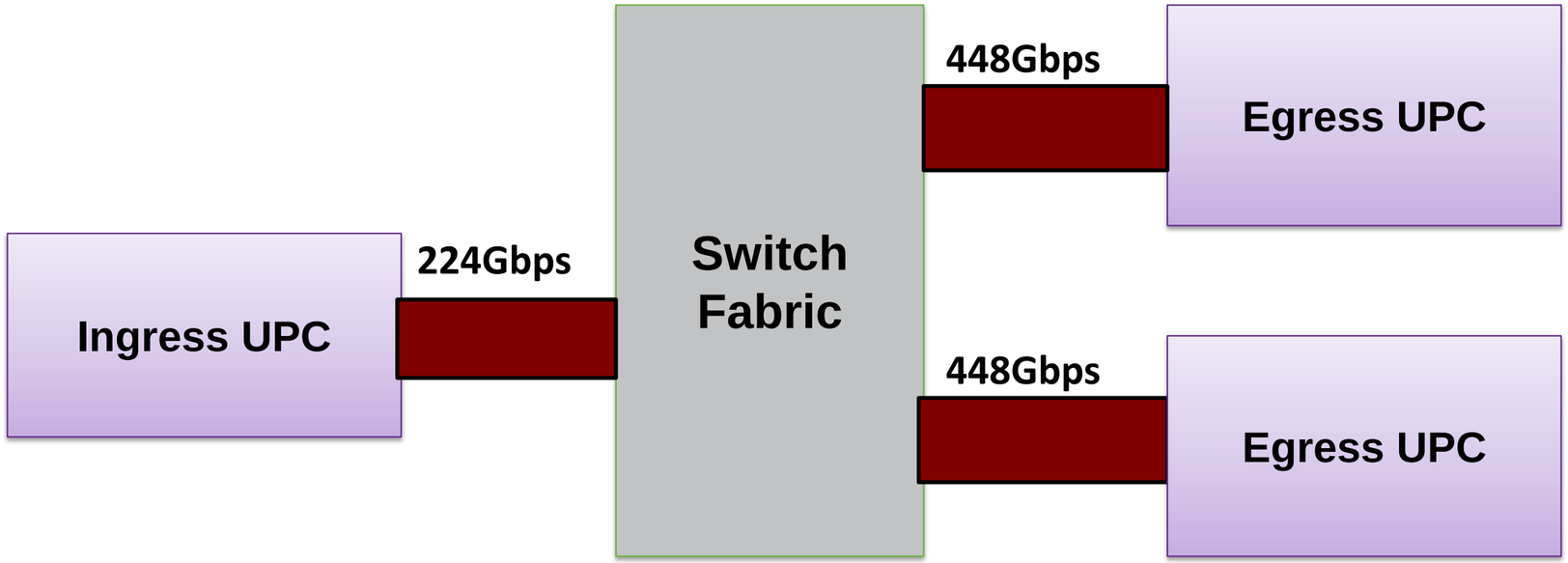
- If all ports are operating at10Gig, use the 10Gig fabric mode
- If all ports are operating at 40Gig, use 40 Gig fabric mode
- If there is a mix of 10Gig and 40Gig ports, use the cut through / S&F matrix in previous slides to see what traffic needs low latency, the following also needs to be considered
  - In 10Gig fabric mode, a 40GE interface can only carry 10Gbps flows
  - In 10Gig fabric mode, there could be a latency improvement of 200 ns for 10Gig ports
  - At FCS, ISSU is disabled when fabric mode is 10Gig mode. Post FCS, ISSU will be enabled in 10Gig fabric mode

Nexus 5000/5500, 6004 & 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Overview
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

Nexus 6004 SPAN Enhancements



- Total 31 active SPAN sessions. 16 ERSPAN sessions.
- Support ERSPAN termination
- Wire speed SPAN throughput. Extra fabric link bandwidth for SPAN traffic
- Best effort for SPAN traffic. Drop SPAN traffic in case of fabric link congestion
- Hardware support multiple SPAN destination ports per SPAN session
- Support PortChannel as SPAN destination port. Source port based hashing.

Nexus 6004 vs Nexus 5500 SPAN

SPAN Features	Nexus 6000	Nexus 5500
Total SPAN sessions	31	4
Local SPAN sessions	31	4
ERSPAN sessions	16	4
Prioritise data over SPAN	Yes(through scheduling)	SPAN policing
Line rate SPAN throughput	Yes	Yes for limited scenarios
ERSPAN destination session	Yes	No
Truncated SPAN/ERSPAN	Yes	Yes
ACL based SPAN/ERSPAN	Yes	Yes
SPAN on drop	Yes	No
SPAN on high latency	Yes	No
SPAN with multiple destination ports	Yes(each destination port burns one SPAN session)	Yes(each destination port burns one SPAN session)

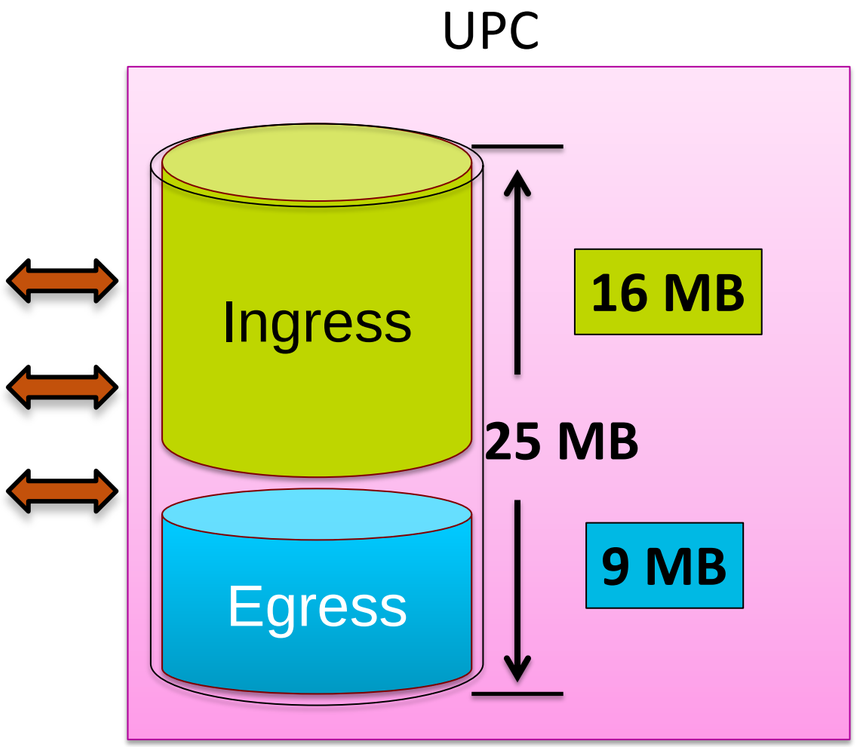
Nexus 5000/5500, 6004 & 2000 Architecture

Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Overview
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

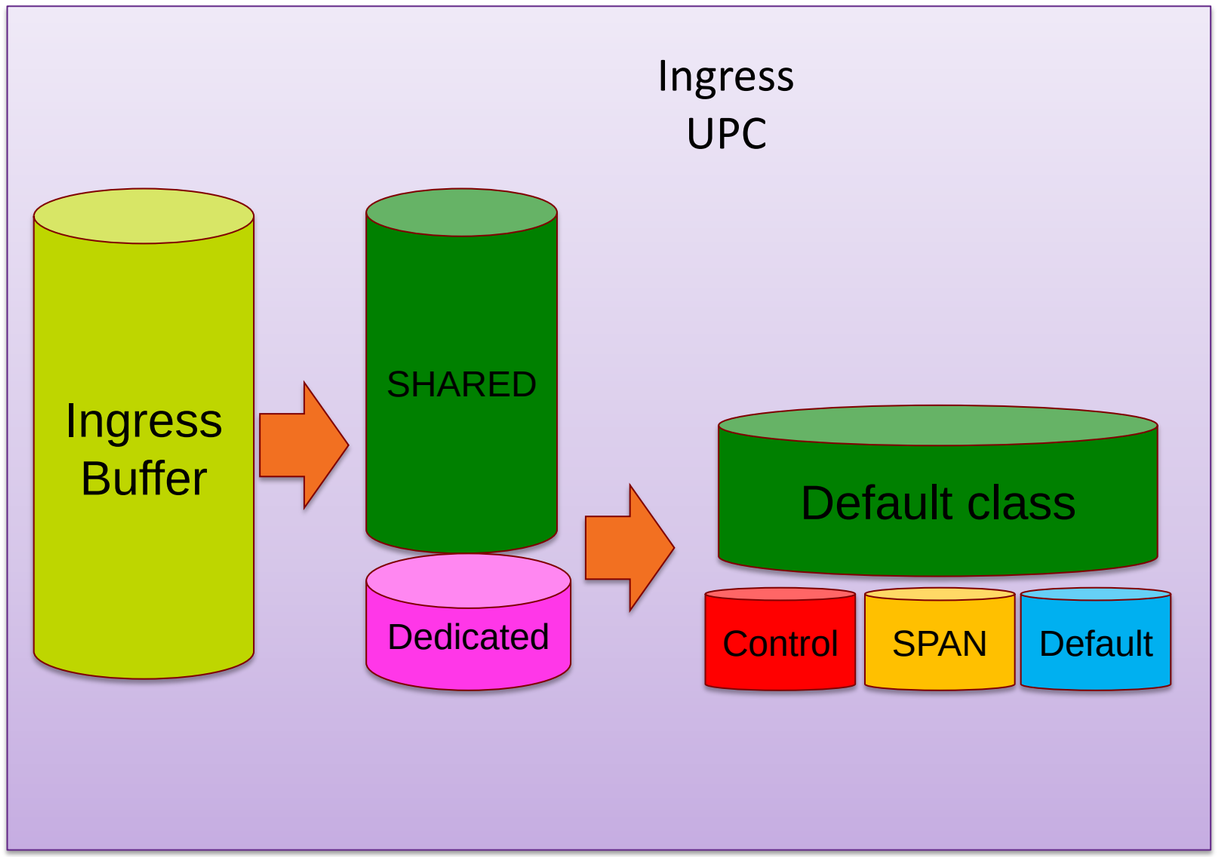
Nexus 6004 Buffer

- Each ports have 25 MB of buffers
- Shared between ingress and egress buffers by default
  - 16 MB ingress
  - 9 MB egress
- The total buffers space allocated for egress buffer is
  - Egress buffer + total of ALL buffer space of ingress ports sending to the egress port



Nexus 6004 Ingress Buffer Management

- By default all the shared buffer is allocated to default class
- A CLI is provided to change the amount of shared buffer, the remaining buffer is dedicated and divided between ports
- hardware shared-buffer-size <0-14.2 MB>
- A Drop class can take half of the shared buffer.
- The queue-limit change the fixed ingress buffer allocation for a class.



Nexus 6004 Ingress Buffer Allocation

40 Gig Port

Traffic Type	Buffer	
	Dedicated per 40Gig Port	Shared
Control Plane	67 KB	N/A
SPAN	152 KB	N/A
Default Class	100 KB	14.6 MB

10 Gig Port

Traffic Type	Buffer	
	Dedicated per 10Gig Port	Shared
Control Plane	64 KB	N/A
SPAN	38 KB	N/A
Default Class	100 KB	13.2 MB

Nexus 6004 FCoE Buffer Allocation

40 Gig Port

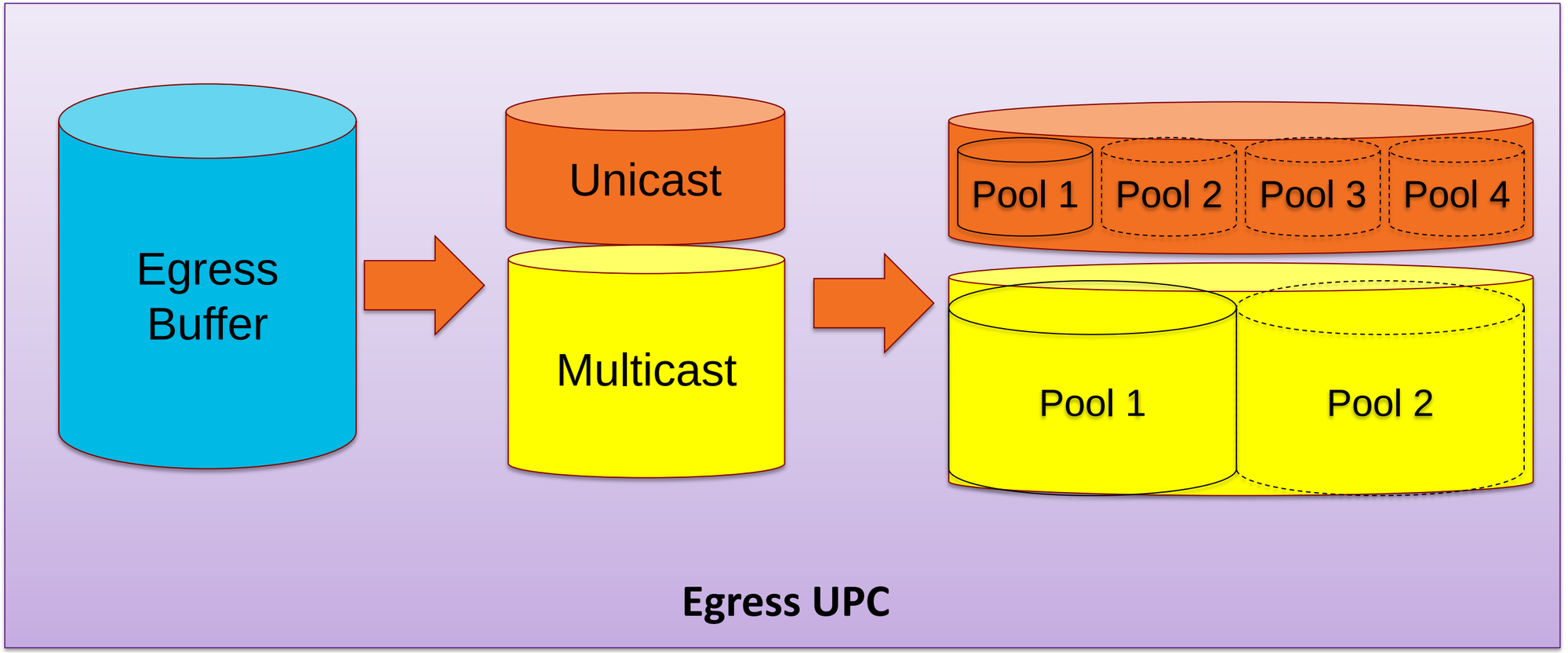
Traffic Type	Buffer		
	300m	**3Km**	**10Km**
FCoE	298 KB	565 KB	3.99 MB

10 Gig Port

Traffic Type	Buffer		
	300m	**1Km**	**10Km**
FCoE	165 KB	230 KB	1.08 MB

- By default no buffer is allocated to class FCoE
- The buffer allocation for FCoE is a function of port speed and distance
- There is enough buffer to support one port per UPC for FCoE over 100KM
- FCoE over 100KM requires 9.6 MB dedicated on a single port

Nexus 6004 Egress Buffer Management

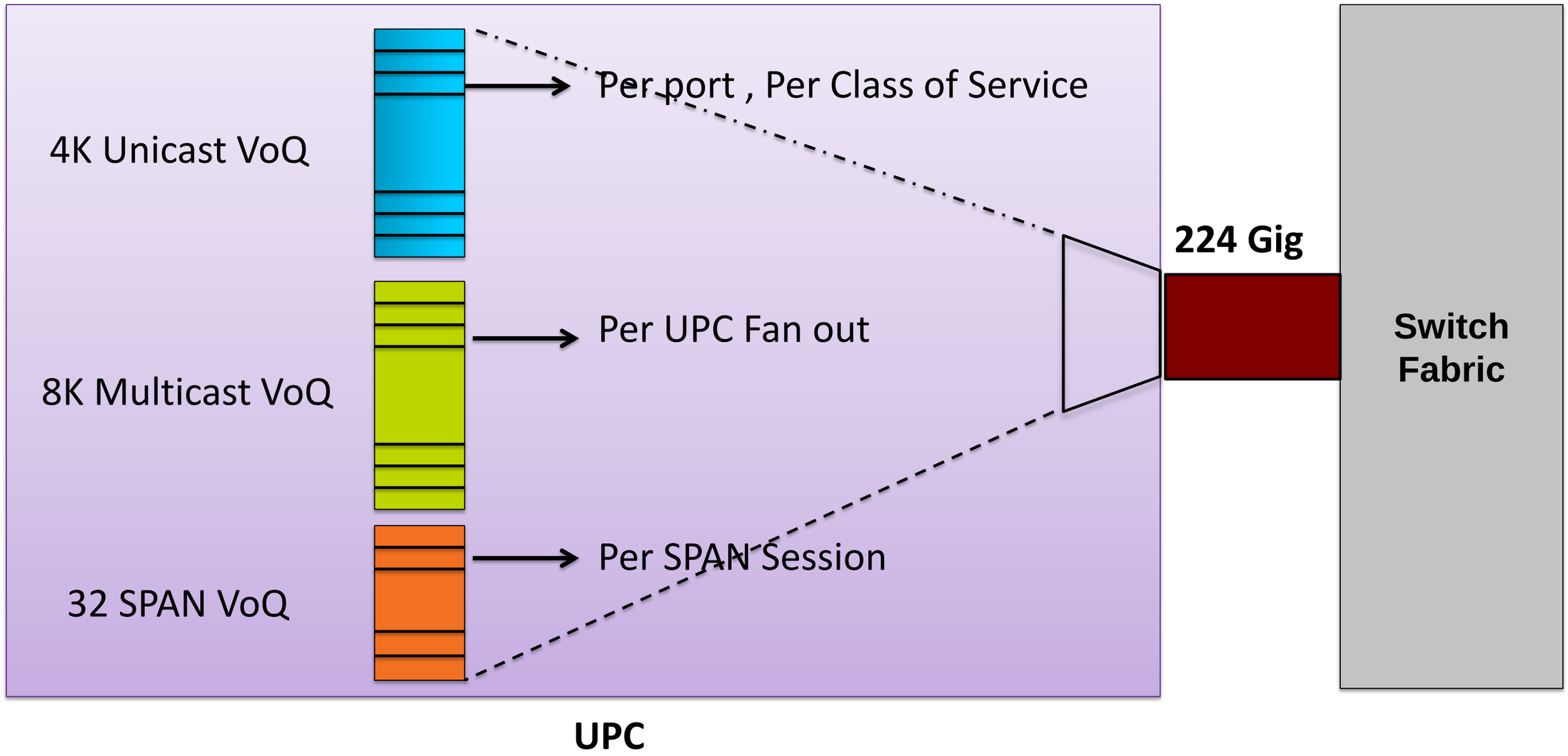


Nexus 6004 Egress Buffer Management

Buffers Depends on Fabric Speed (10G vs. 40G) & Port Mode Speed (10G vs. 40G)

Fabric Mode	Traffic Type	Buffer Size	
10 G		Dedicated – 10G	Shared
	Unicast	212 KB	None
	Multicast	None	6.3 MB
40G		Dedicated – 10G	Shared
	Unicast	212 KB	None
	Multicast	None	6 MB
10G		Dedicated – 40G	Shared
	Unicast	672 KB	None
	Multicast	None	6.1 MB
40G		Dedicated – 40G	Shared
	Unicast	795 KB	None
	Multicast	None	6.1 MB

Nexus 6004 Ingress Queuing



Nexus 5000/5500, 6004 & 2000 Architecture

Agenda

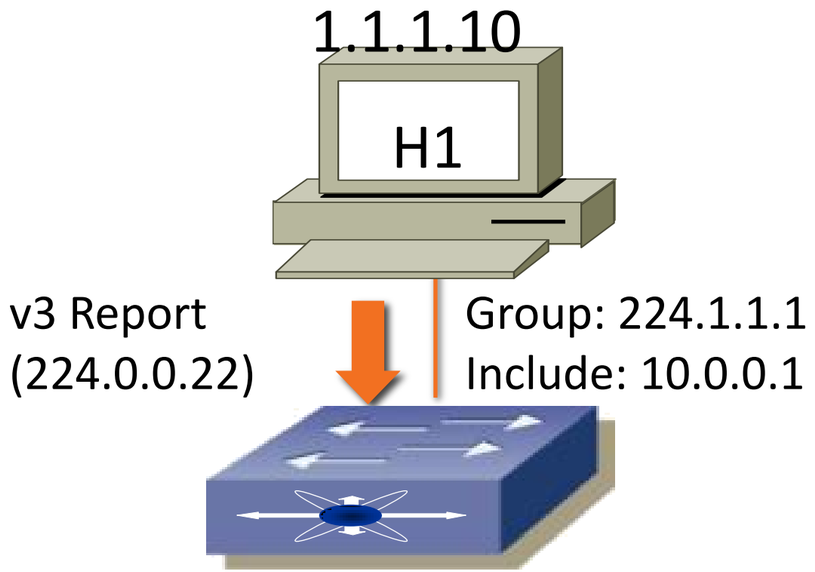
- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Overview
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

Nexus 6004 Multicast Highlights

- High Performance
  - Line rate L2 and L3 multicast throughput with all frame sizes
  - Low latency at scale
- High L3 Multicast Scalability
  - 64000 mroute table
- Larger buffer for burst absorption
- Optimised multicast replication
  - Fabric replication and egress replication
- Enhanced features
  - IP based forwarding for IGMP snooping
  - PIM-BiDir support
  - Flow based hashing for multicast over PortChannel
  - Better traffic visibility

IP Based Forwarding for IGMP Snooping

- Source IP and group address based forwarding for IGMPv3 snooping even when N6k is L2 switch
- Can filter traffic based on source IP for IGMPv3
- No concern of overlapping multicast MAC addresses



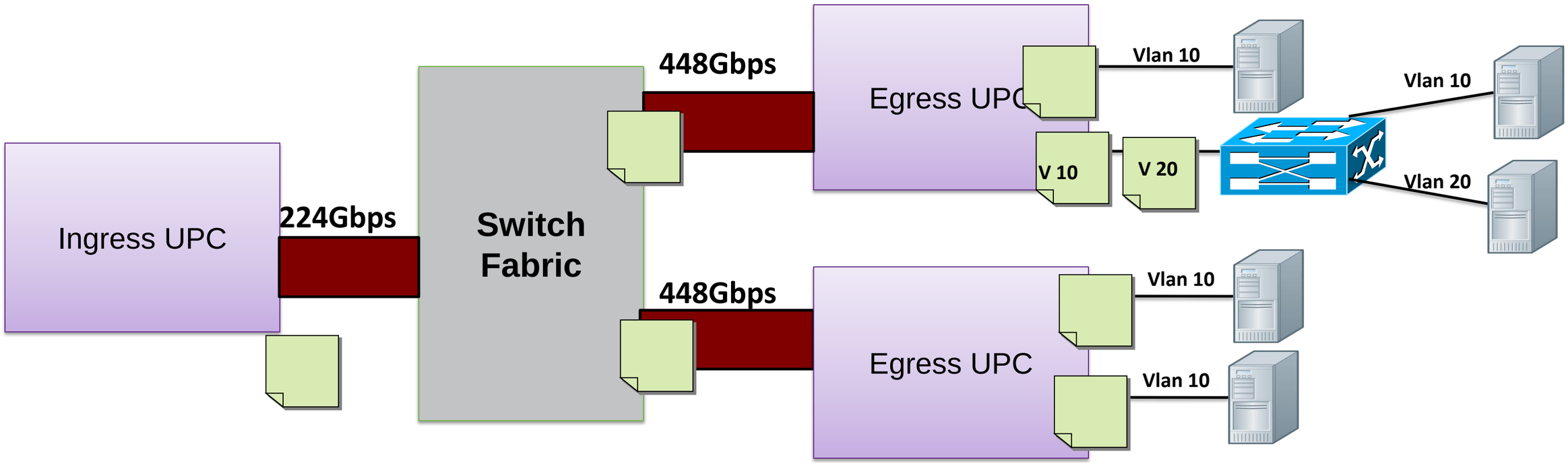
Multicast MAC based forwarding

Vlan10 0100.5E01.0101 eth1/1

IP based forwarding

Vlan10 10.0.0.1 224.1.1.1 eth1/1

Nexus 6004 Multicast Packet Replication



- Fabric replication: One copy is sent to each egress UPC that has at least one receiver
- Egress replication: UPC replicates packets locally to each port and multiple copies to same port if needed
- Egress buffering for microburst and oversubscription
- Drop multicast packet at egress on the per port per queue basis for congestion

Nexus 6004
Summary Key Enhancements

Performance	L2 and L3 line rate at 10Gig and 40Gig, Latency at 1us, 1K way ECMP, Line rate SPAN
Scalability	256K MAC/IP host routes, 32K LPM routes, 16K Bridge Domains, 4K VRF, 64K RPF, 64K Mroute table
Buffering	25M shared buffer per 3 QSFP ports
Multicast	Optimised multicast replication, flow based hashing, IP (S,G) / (*,G) lookup even for L2 multicast, egress midx translation
SPAN	Line rate SPAN, 31 SPAN sessions, SPAN on drops, SPAN on high latency
Analytics	Sampled Netflow, buffer monitoring, microburst monitoring, latency monitoring, SPAN on Drop, SPAN on High Latency

Nexus 5000/5500, 6004 & 2000 Architecture

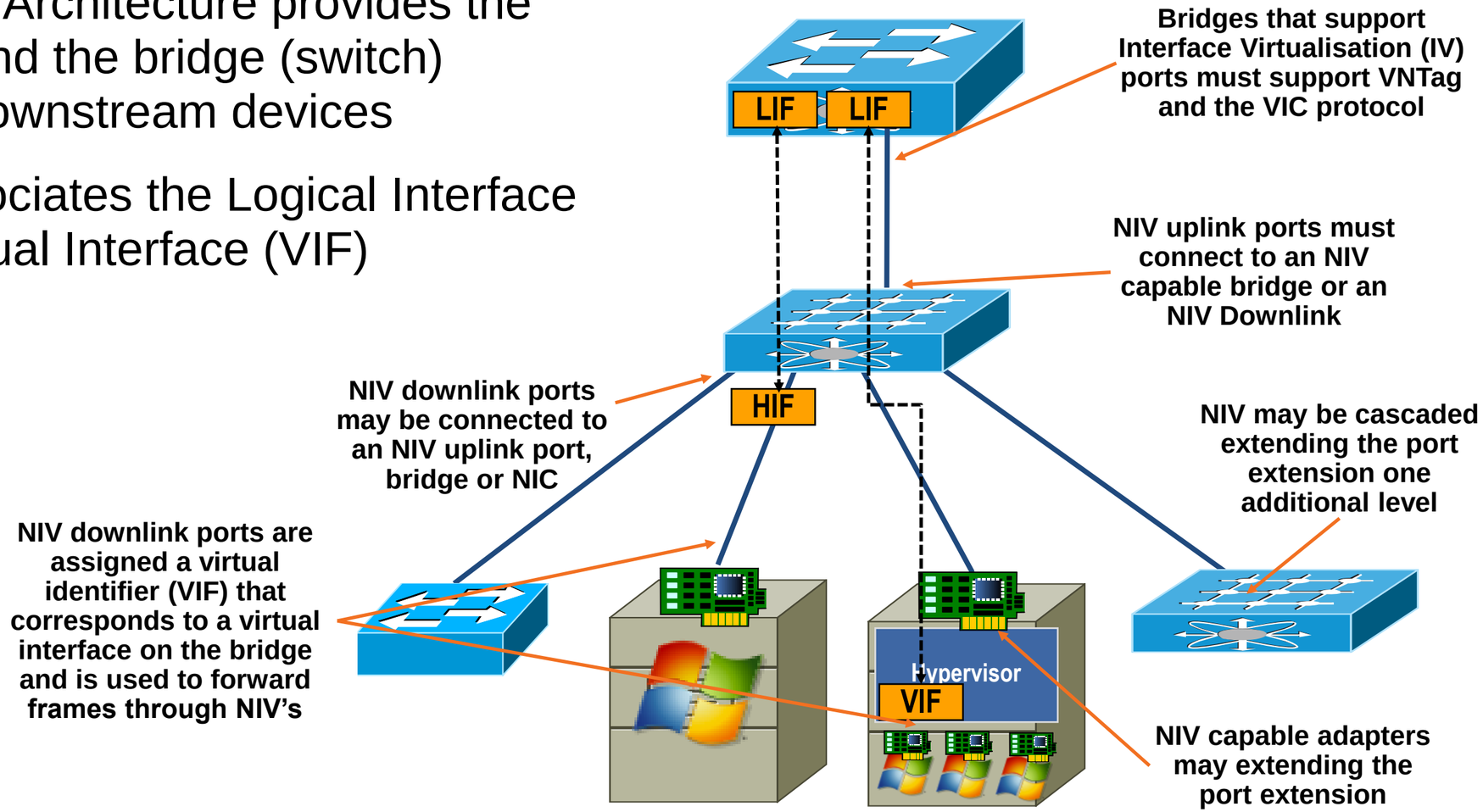
Agenda

- Nexus 5000/5500 Architecture
  - Hardware Architecture
  - Day in the Life of a Packet
  - Port Channels
  - QoS
- Nexus 6004 Overview
  - Architecture
  - SPAN
  - Buffering & QoS
  - Multicast
- Nexus 2000 Architecture
  - FEXLink Architecture

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)



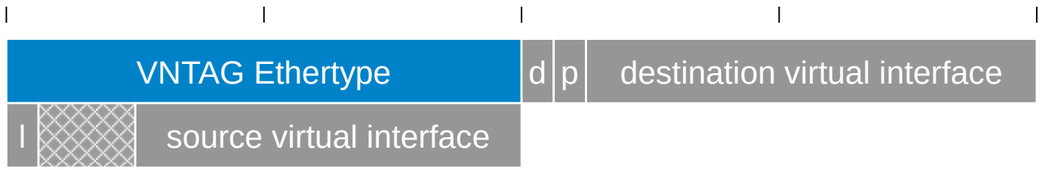
Nexus 2000 Fabric Extender (FEX)

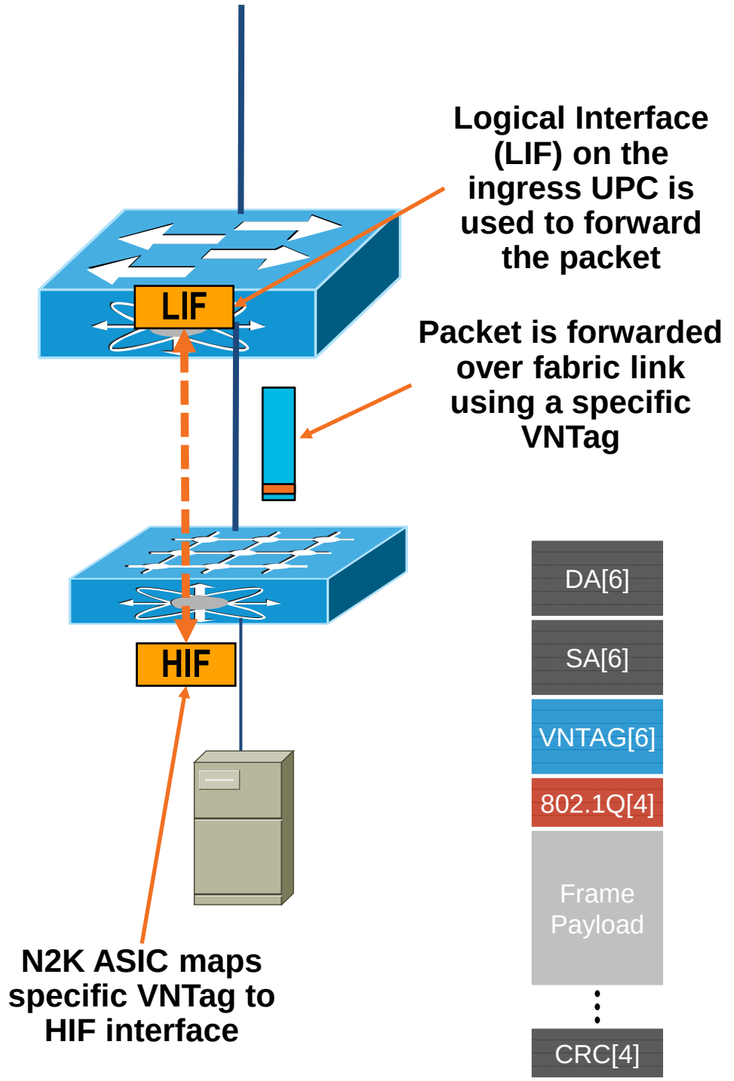
VN-Tag Port Extension

- 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 Virtualisation (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

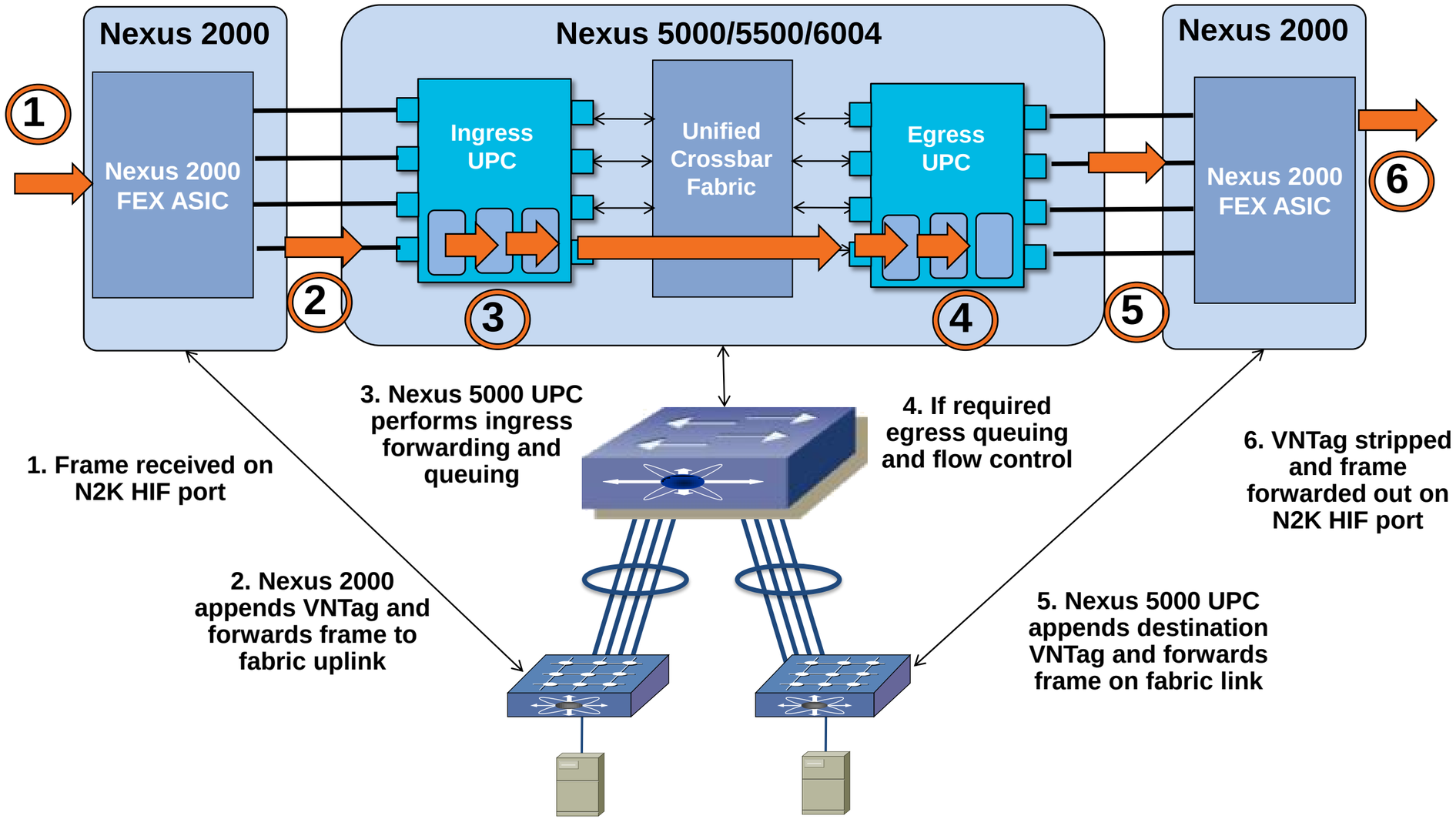
  VNTag is stripped before the packet is sent to hosts
- VNTag allows the Fabric Extender to act as a data path of Nexus 5000/5500/7000 for all policy and forwarding





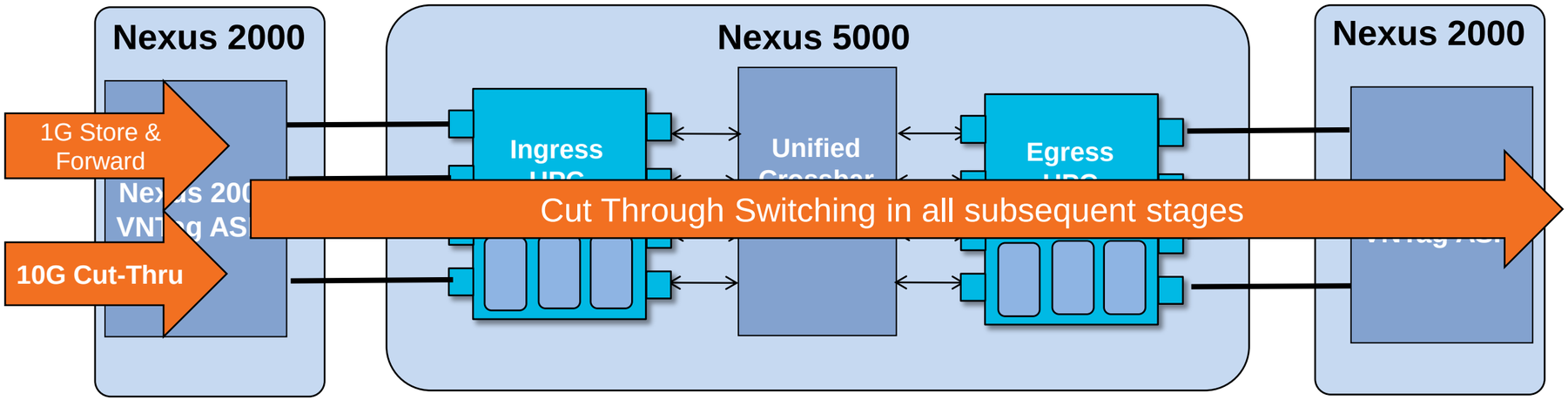
Nexus 5500/6004 and 2000

Packet Forwarding Overview

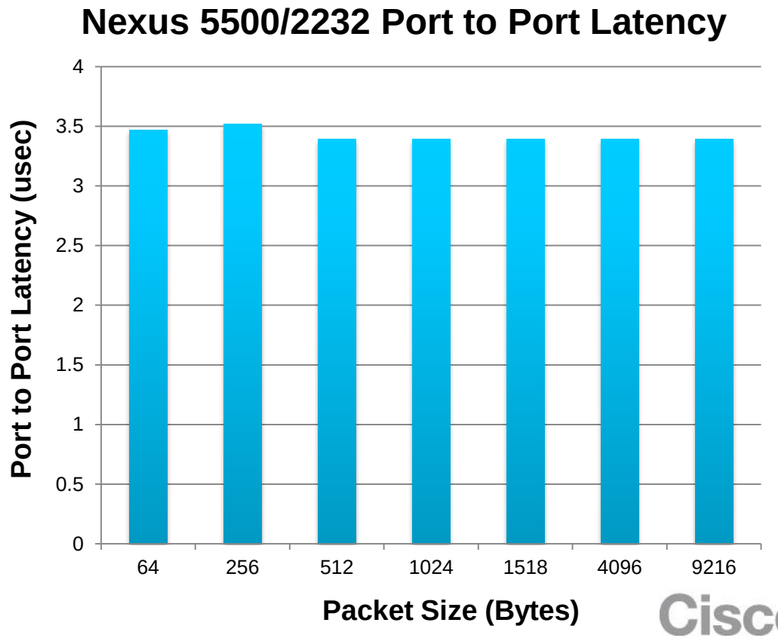


Nexus 5500/6004 and 2000

Packet Forwarding Latency



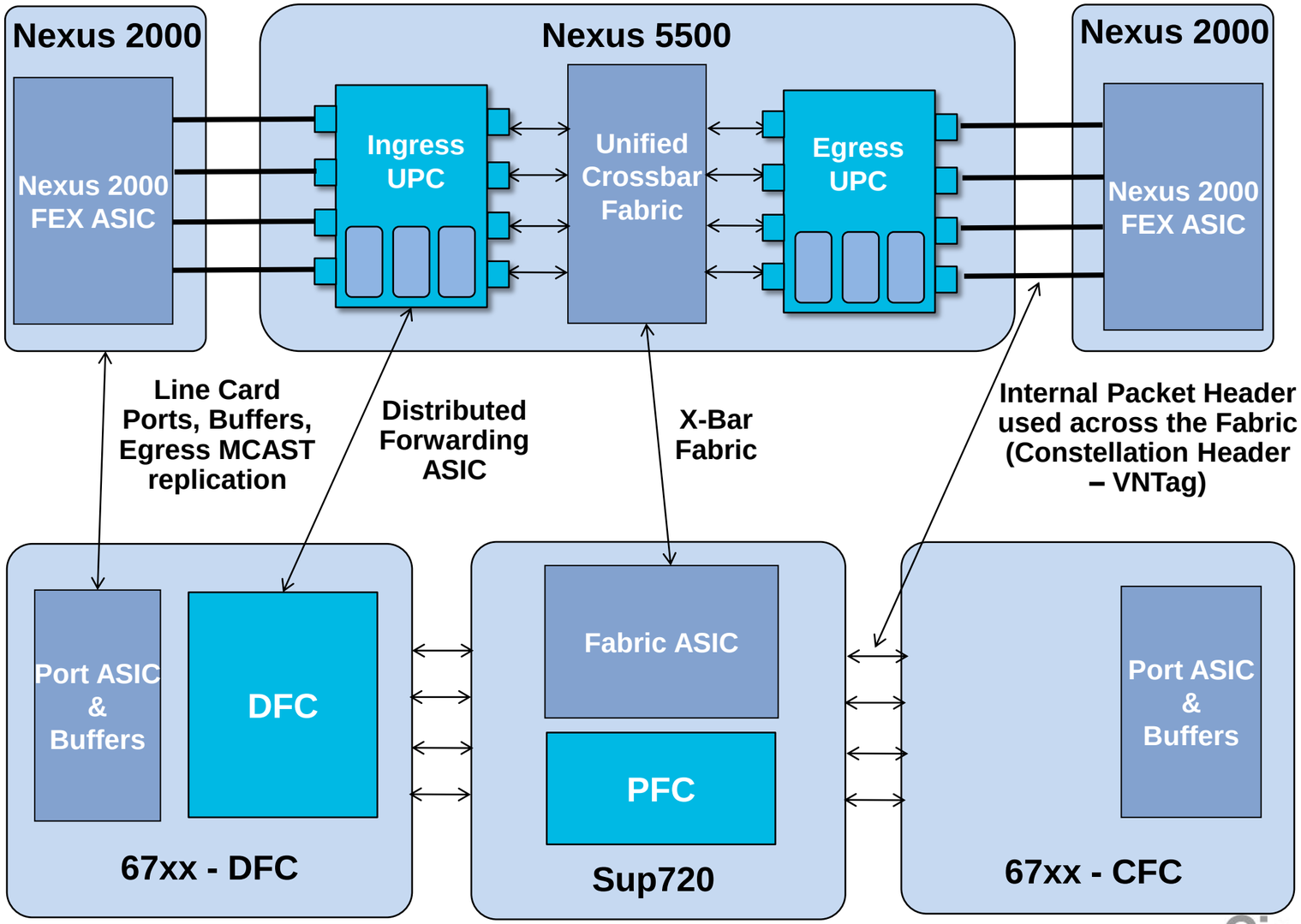
- Nexus 2000 also supports Cut -Through switching
  - 1GE to 10GE on first N2K ingress is store and forward
  - All other stages are Cut Through (10GE N2K port operates in end to end cut-through)
- Port to Port latency is dependent on a single store and forward operation at most
- Nexus 6004 allow lowest latency fabric ~ 1.2 micor-second



Nexus 5500/6004 and 2000

Switching Morphology - Is this Really Different?

- Nexus 2000 Architecture localises the Forwarding ASIC in the parent switch (supervisor)
- Minimal latency due to cut-through 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



Cisco Nexus 2000 Series
Platform Overview

New CY13

N2148T
48 Port 1000M Host Interfaces
4 x 10G Uplinks

N2248TP
48 Port 100/1000M Host Interfaces
4 x 10G Uplinks

N2232PP
32 Port 1/10G FCoE Host Interfaces
8 x 10G Uplinks

N2224TP
24 Port 100/1000M Host Interfaces
2 x 10G Uplinks

N2232TM
32 Port 1/10GBASE-T Host Interfaces
8 x 10G Uplinks (Module)

N2248TP-E
48 Port 100/1000M Host Interfaces
4 x 10G Uplinks
32MB Shared Buffer

FET-10G
Cost Effective Fabric Extender Transceiver

B22 HP
16 x 1/10G Host Interfaces
8 x 10G Uplinks

B22 FTS
16 x 1/10G Host Interfaces
8 x 10G Uplinks

Blade FEXs

Nexus 2248PQ-10GE

- 48 Port 1/10GE SFP+ Host Interfaces
- 4 x QSFP (16x10GE SFP+) Uplinks
- Additional uplink buffers (2x16MB)

B22 DELL
16 x 1/10G Host Interfaces
8 x 10G Uplinks

Q & A

Complete Your Online Session Evaluation

Give us your feedback and receive a Cisco Live 2013 Polo Shirt!

Complete your Overall Event Survey and 5 Session Evaluations.

- Directly from your mobile device on the Cisco Live Mobile App
- By visiting the Cisco Live Mobile Site www.ciscoliveaustralia.com/mobile
- Visit any Cisco Live Internet Station located throughout the venue

Polo Shirts can be collected in the World of Solutions on Friday 8 March 12:00pm-2:00pm

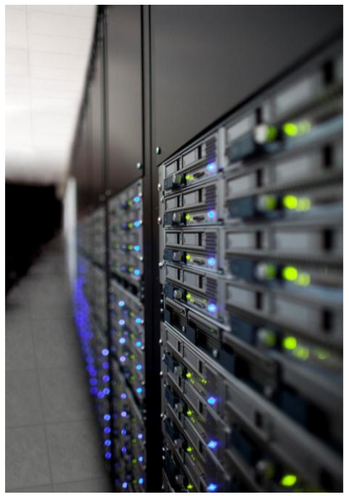
Cisco live! 365

Don’t forget to activate your Cisco Live 365 account for access to all session material, communities, and on-demand and live activities throughout the year. Log into your Cisco Live portal and click the "Enter Cisco Live 365" button.
www.ciscoliveaustralia.com/portal/login.ww

Appendix

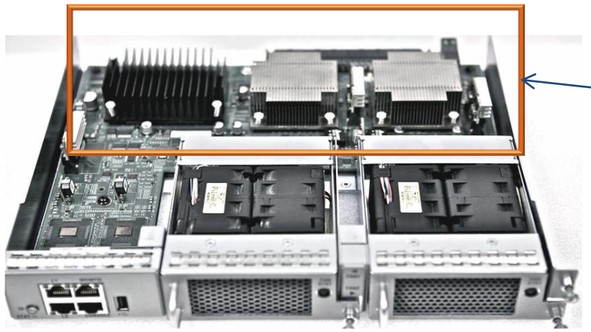
Agenda - Extras

- Nexus 5000
  - Layer 3 Module
  - SPAN & ERSPAN
  - Multicast

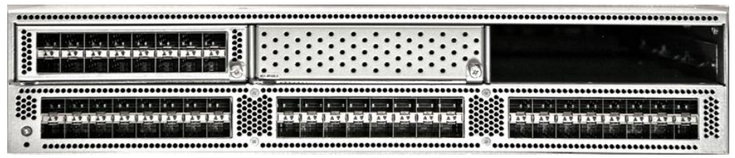


Nexus 5500 Series

Nexus 5500 with Layer 3 Support



160Gbps (240Mpps)
Layer 3 processing



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

1) Install L3 Expansion Module(s)
2) 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

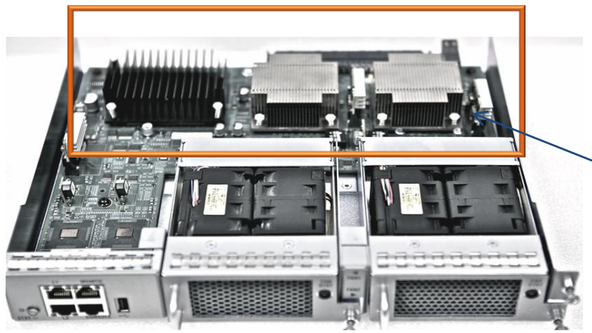
Nexus 5596UP

- 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)

Nexus 5500 Series – 5.1(3)N1

N55-D160L3-V2 and N55-M160L3-V2

Increased Scalability
16 FEXs per Nexus 5K



Version 2 Layer 3 Daughter-card and Module

N55-D160L3 and N55-M160L3

Capability	Scale
IPv4 Longest Prefix Match Routes	8k (16K with uRPF disabled)
IPv4 Host Table	8,000
IP Multicast Routes	4,000
L3 Interfaces	1K
VRF	1K

N55-D160L3-V2 and N55-M160L3-V2

Capability	Scale
IPv4 Longest Prefix Match Routes	8k (16K with uRPF disabled)
IPv4 Host Table	16,000
IP Multicast Routes	8,000
L3 Interfaces	1K
VRF	1K

NOTE: Increased Host and MCAST Route scale is supported in SW in the 5.2(1)N1 release

Nexus 5500 Series

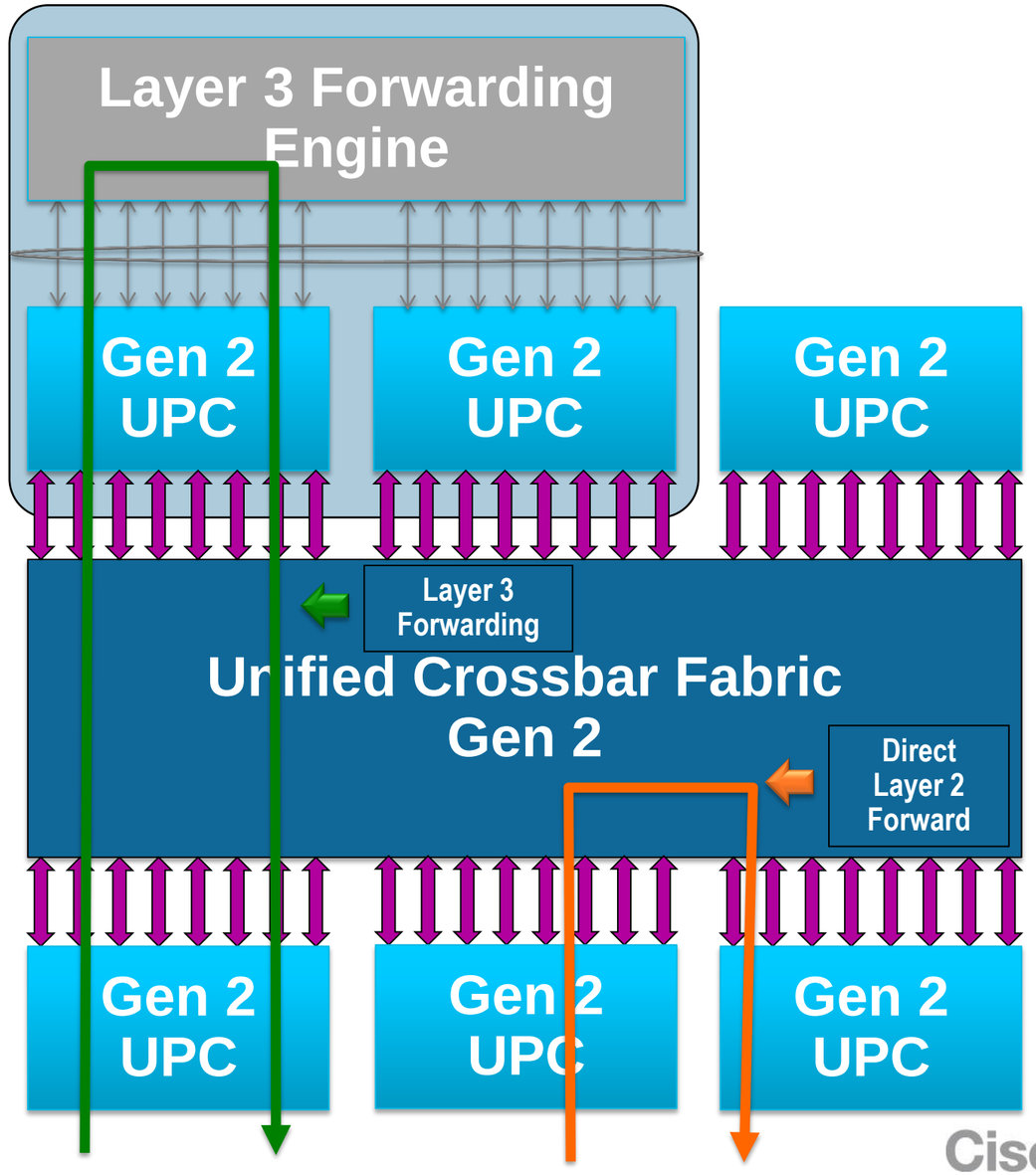
Nexus 5500 with Layer 3 support

- Layer 3 Forwarding Engine connects the the X-Bar via two UPC (160 Gbps)
- 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

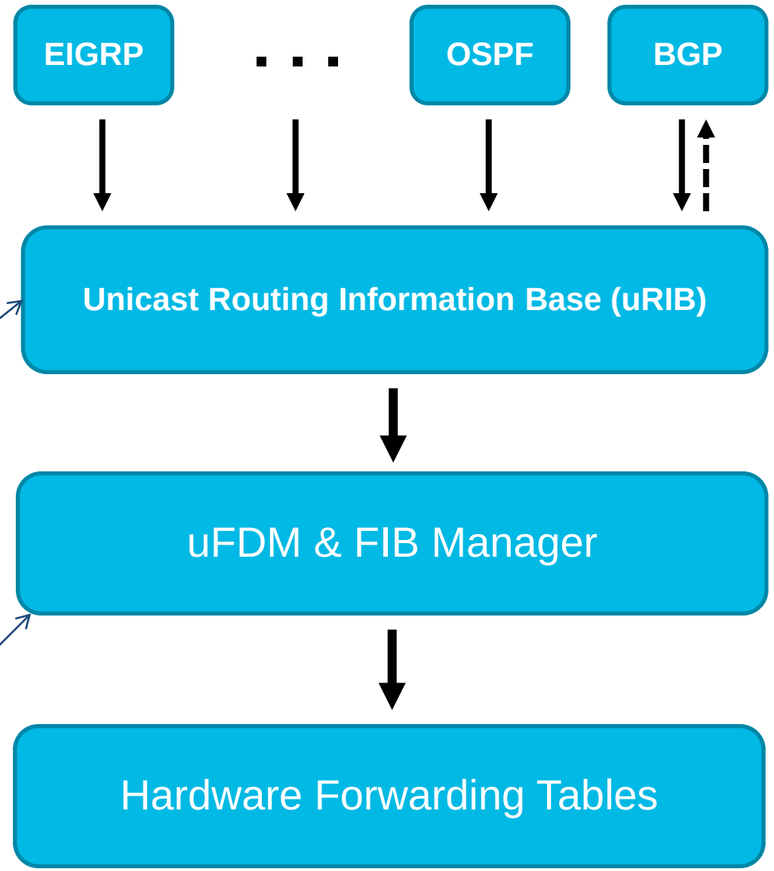
```
L3-5548-1# sh run ospf

!Command: show running-config ospf
!Time: Fri Mar 25 14:21:05 2011

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

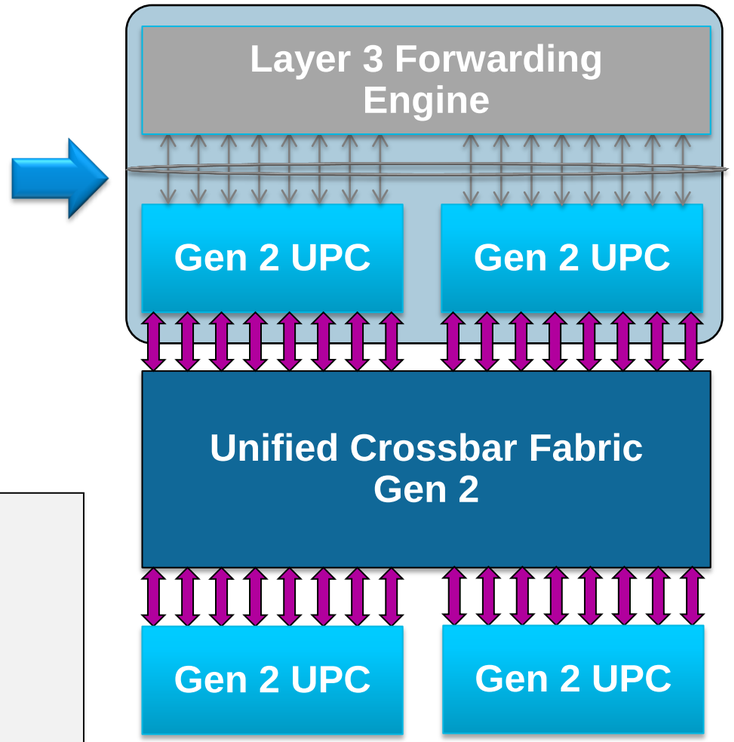
```
L3-5548-1# sh ip route
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]

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

------------------+------------------+---------------
Prefix            | Next-hop         | Interface
------------------+------------------+---------------
10.1.1.0/24         Attached           Vlan10
10.1.1.0/32         Drop               Null0
10.1.1.1/32         Receive            sup-eth1
10.1.1.2/32         10.1.1.2           Vlan10
10.1.1.255/32       Attached           Vlan10
```



Nexus 5500 Series

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                  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>
Members in this channel: Eth3/1, Eth3/2, Eth3/3, Eth3/4, Eth3/5, Eth3/6, Eth3/7, Eth3/8, Eth3/9,
Eth3/10, Eth3/11, Eth3/12, Eth3/13, Eth3/14, Eth3/15, Eth3/16
```

Nexus 5500 Series

Nexus 5500 with Layer 3 support

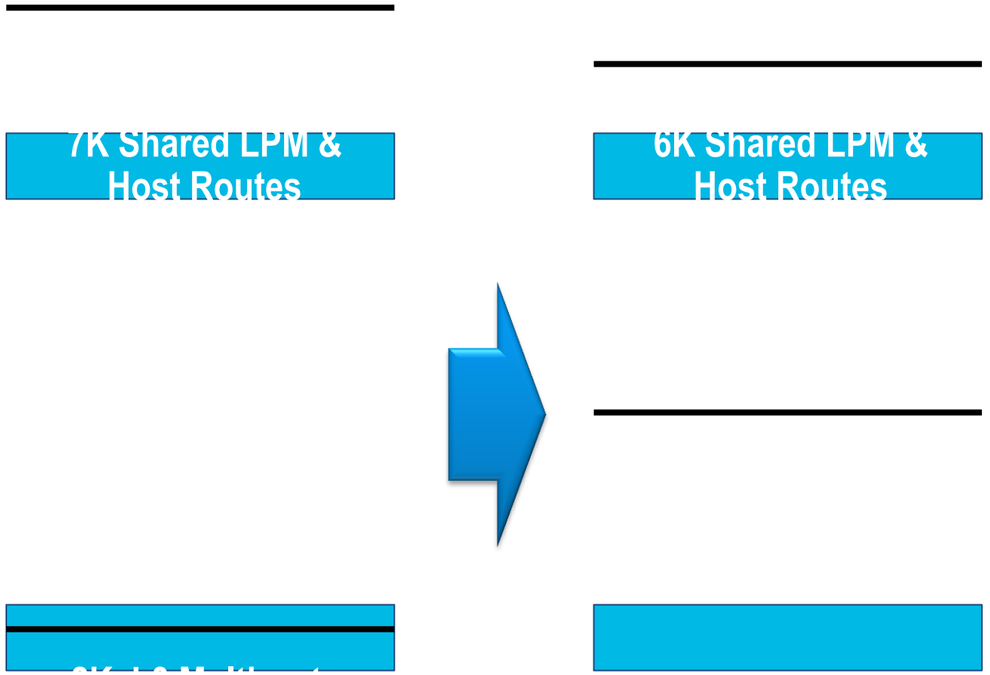
- Layer 3 Forwarding Tables can be tuned for specific design scenarios
- Similar to SDM templates used on Catalyst 3560/3750
- Three table space allocations

  Host Routes (1 entry per /32) – Adjacent Hosts

  LPM (1 entry per route) – Longest Prefix Match Routes

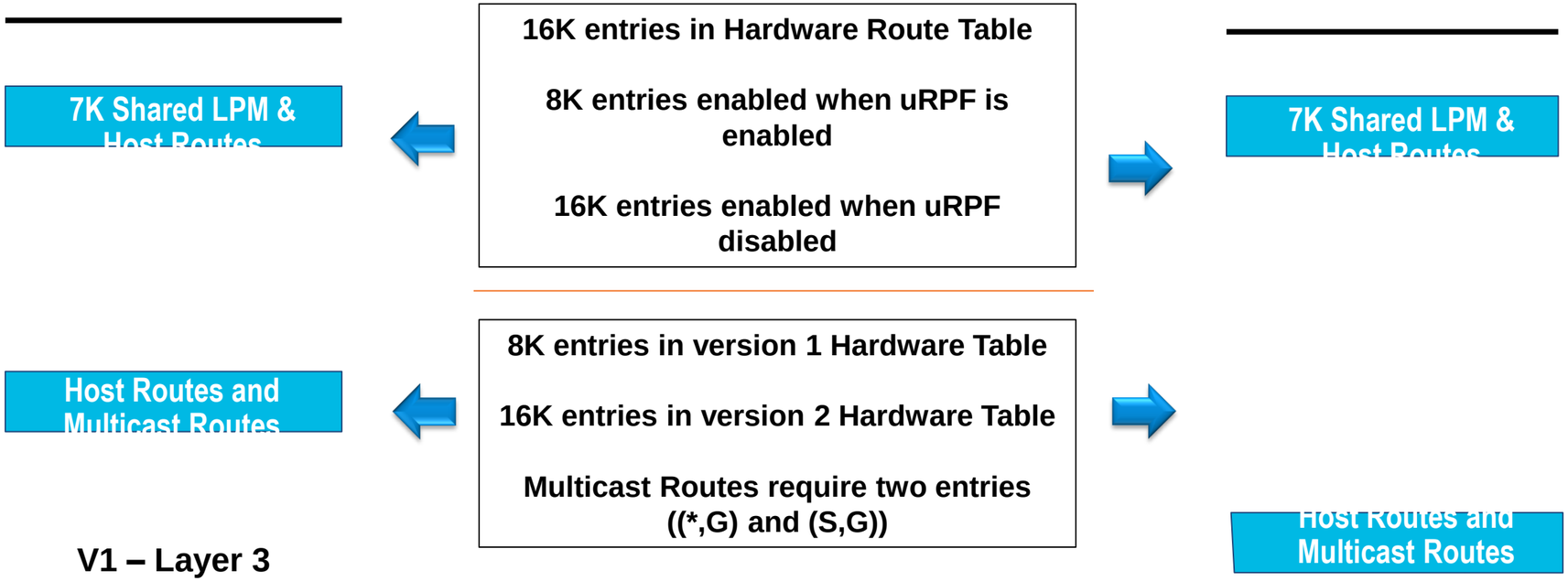
  Multicast Routes (*2 entries per mcast route) – (S,G) and (*,G)

```
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.
```



Nexus 5500 Series

Version 1 & Version 2 Layer 3 Module

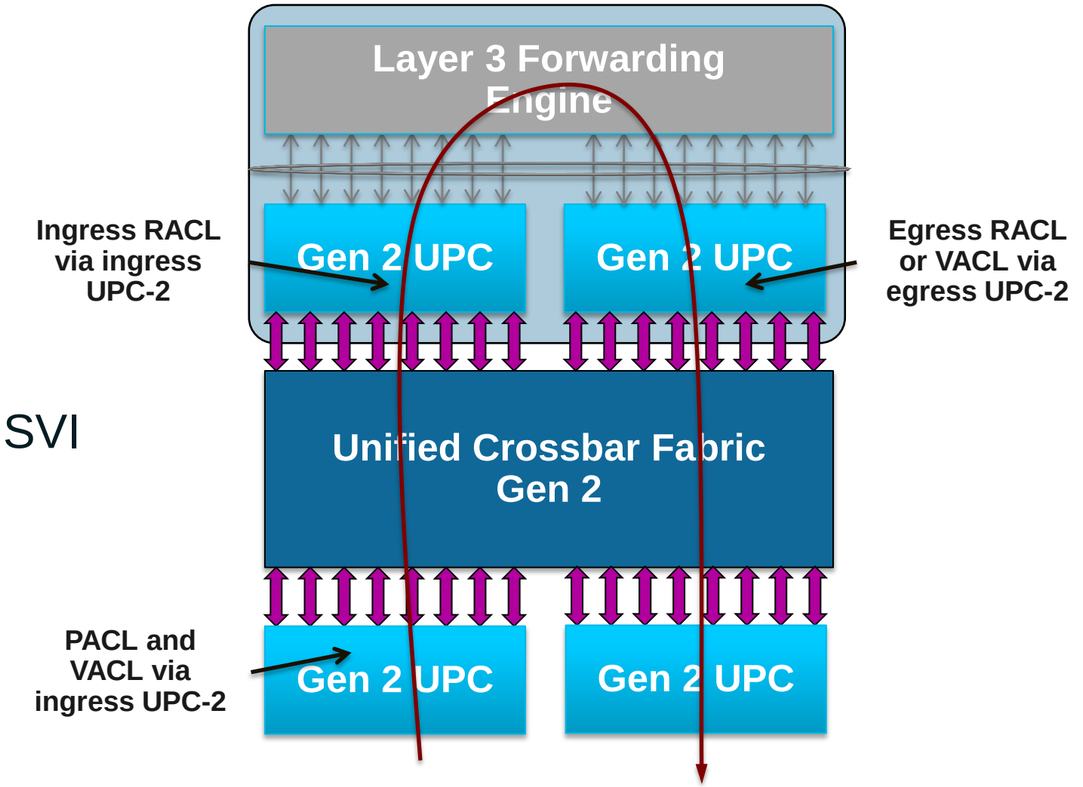


- Multicast Routes (2 entries per mcast route) – (S,G) and (*,G)
- vPC requires 4 entries (Dual Active DR)

Nexus 5500 Series

Access-Control List (ACL) 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

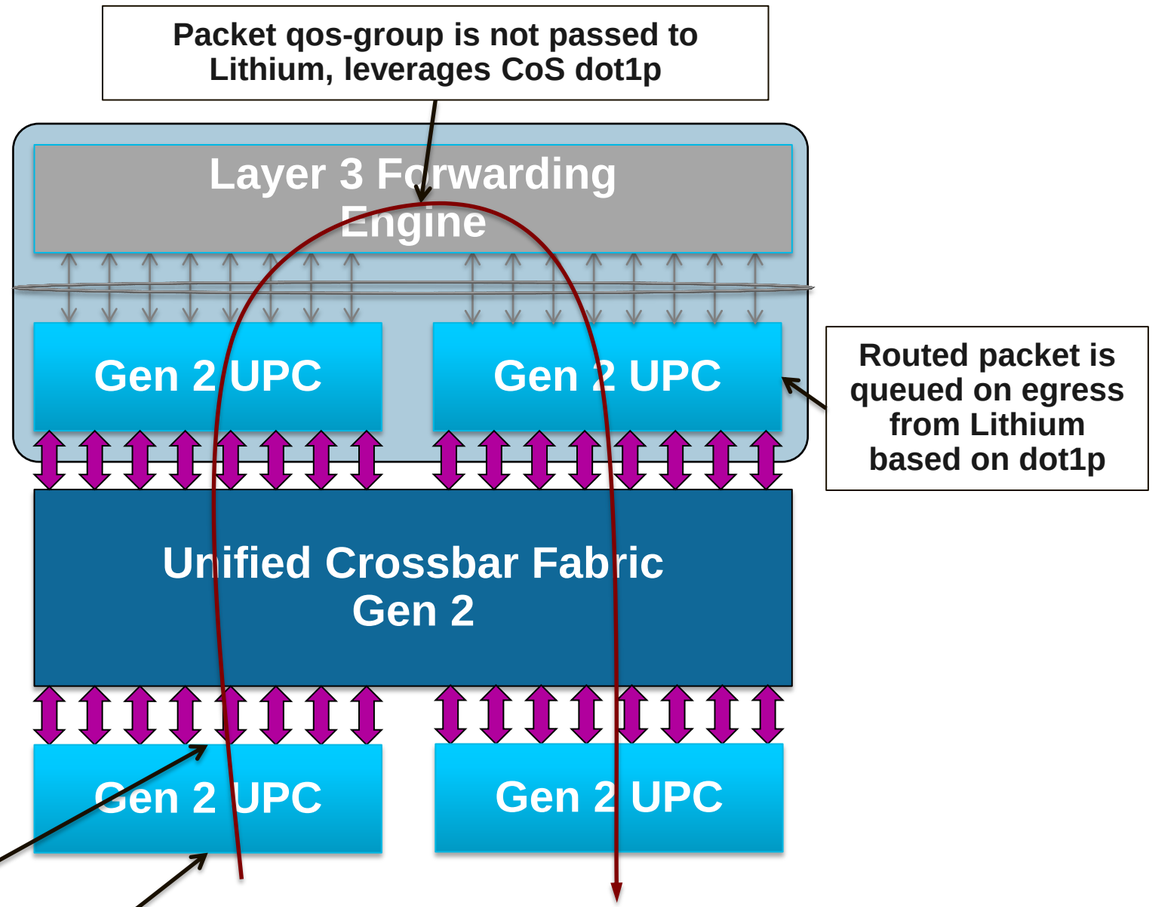



```
L3-5548-1(config)# interface ethernet 1/17
L3-5548-1(config-if)# ip access-group acl01 in
L3-5548-1(config-if)# ip access-group acl01 out

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 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



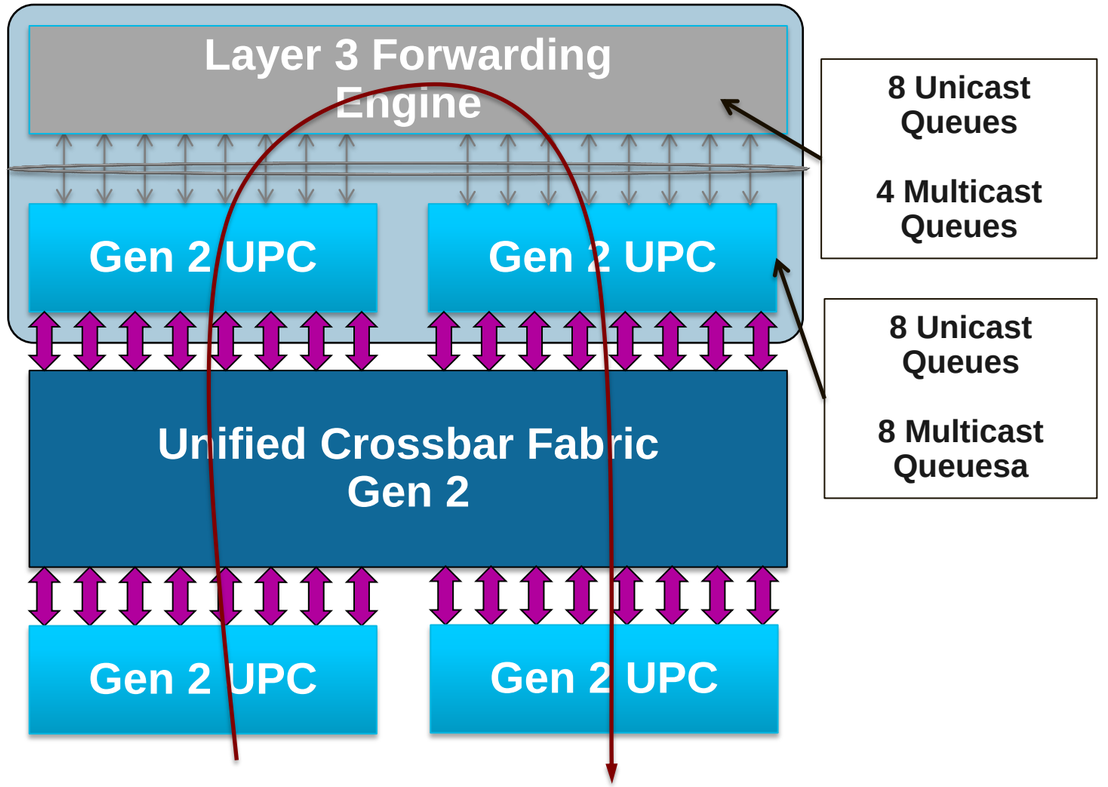
If traffic is congested on ingress to L3 ASIC it is queued on ingress UPC ASIC

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

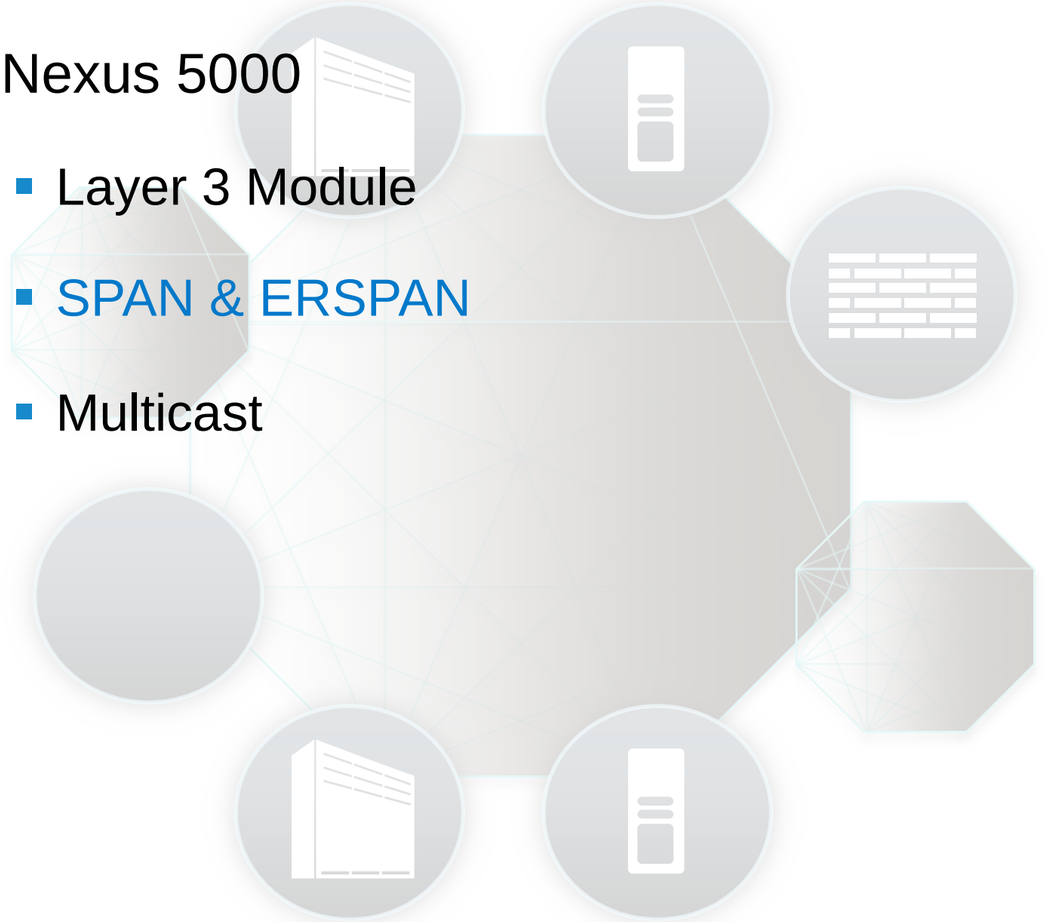
- Apply “type qos” and network-qos policy for classification on the L3 interfaces and on the L2 interfaces (or simply system wide)
- Applying “type queuing” policy at system level in egress direction (output)
- Trident has CoS queues associated with every interface
  - 8 Unicast CoS queues
  - 4 Multicast 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>

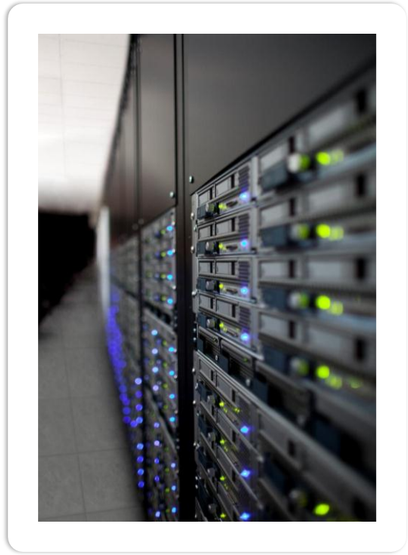



```
Nexus-5500(config)# wrr-queue cos-map 0 1 2 3

Nexus-5500(config)# sh wrr-queue cos-map
MCAST Queue ID            Cos Map
0                         0 1 2 3
1
2                         4 5
3                         6 7
```

Agenda - Extras

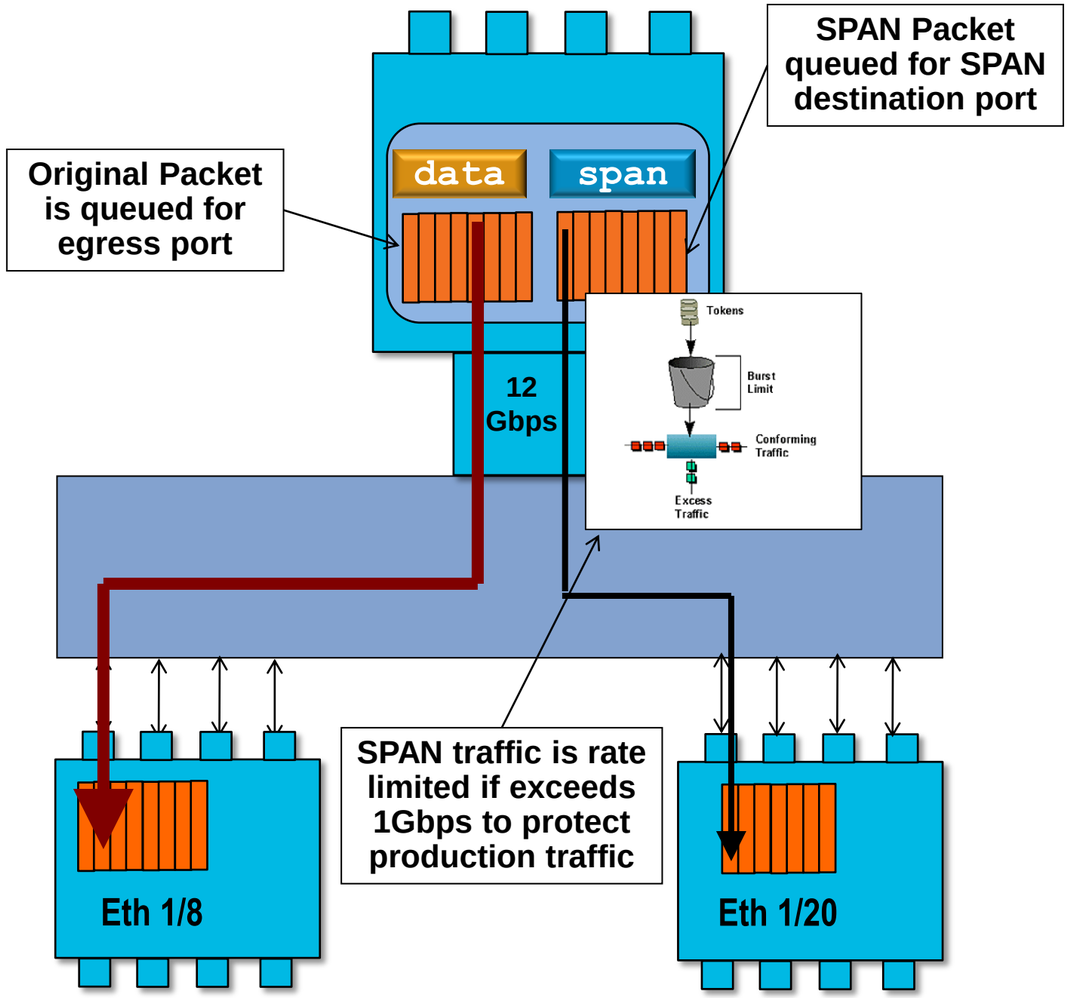
- Nexus 5000
  - Layer 3 Module
  - SPAN & ERSPAN
  - Multicast





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 sessions
- SPAN packets are 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***

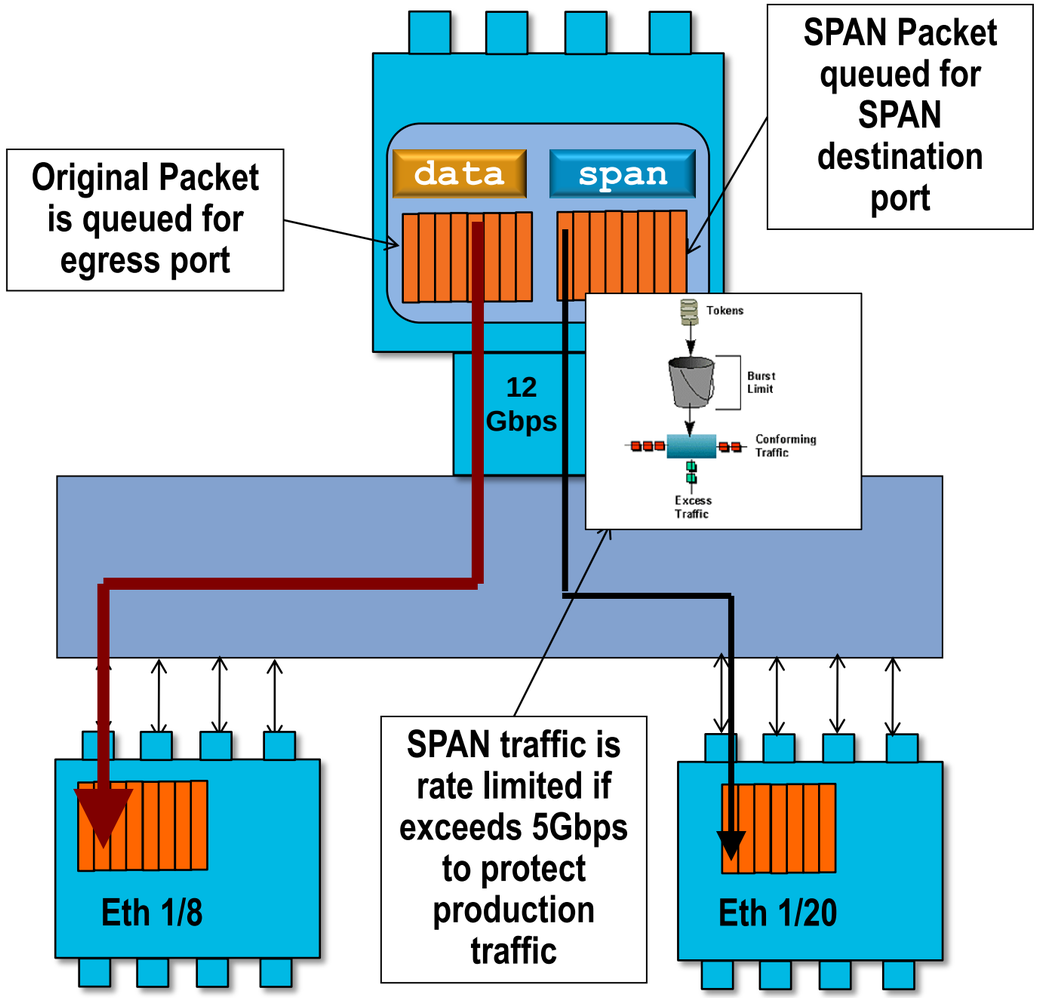
  A rate limit CLI was introduced in order to limit the SPAN traffic 1 Gig

  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

Nexus 5500 SPAN Rx

SPAN Replication and Rate Limiting



- SPAN data packets are replicated at ingress port ASIC-Unified Port Controller (UPC-2) for Rx SPAN sessions
- SPAN packets are 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***

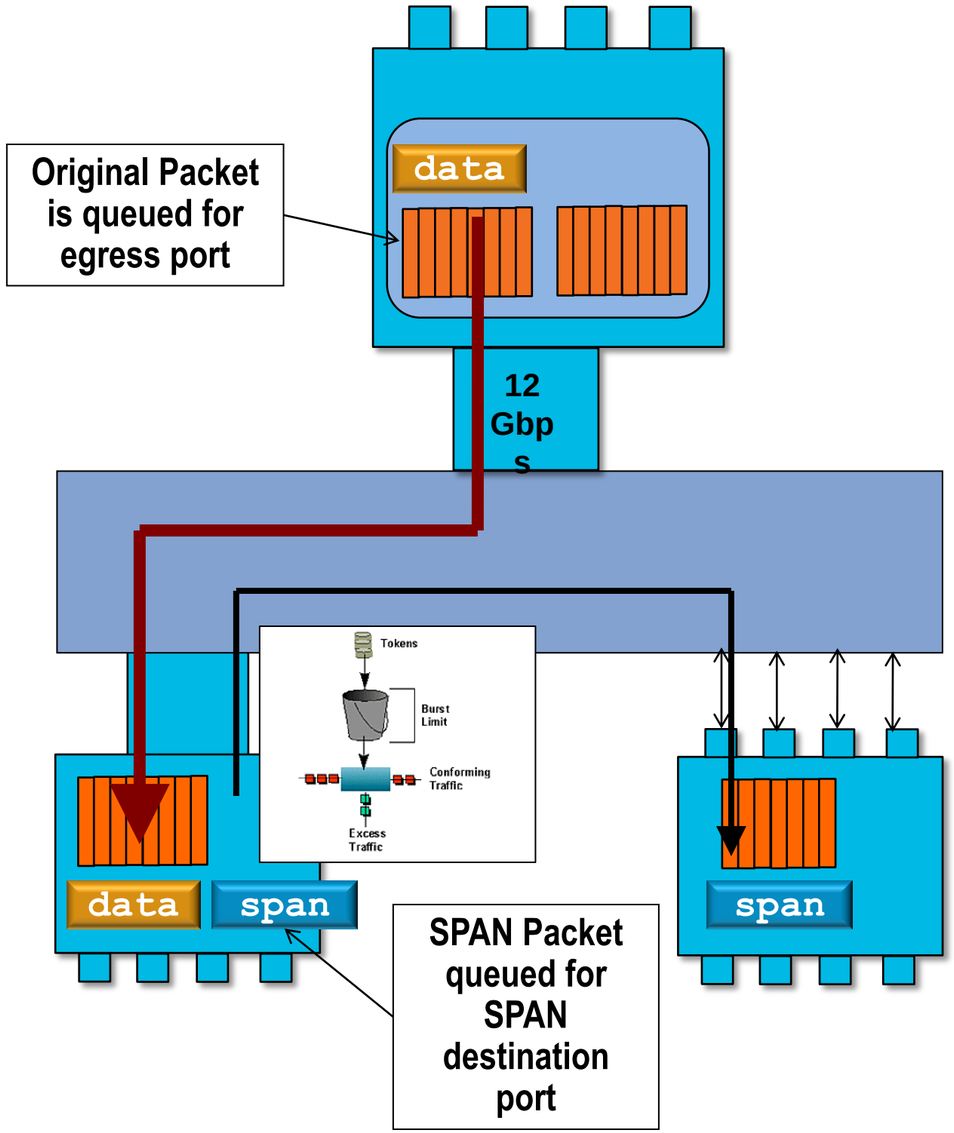
  When data rate is above 5Gbps, SPAN traffic is reduced to 0.75Gbps to avoid potential congestion over the link between ingress port and switch fabric

  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 sessions
- On ***Nexus 5000***

  A rate limit CLI was introduced in order to limit the SPAN traffic to 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***

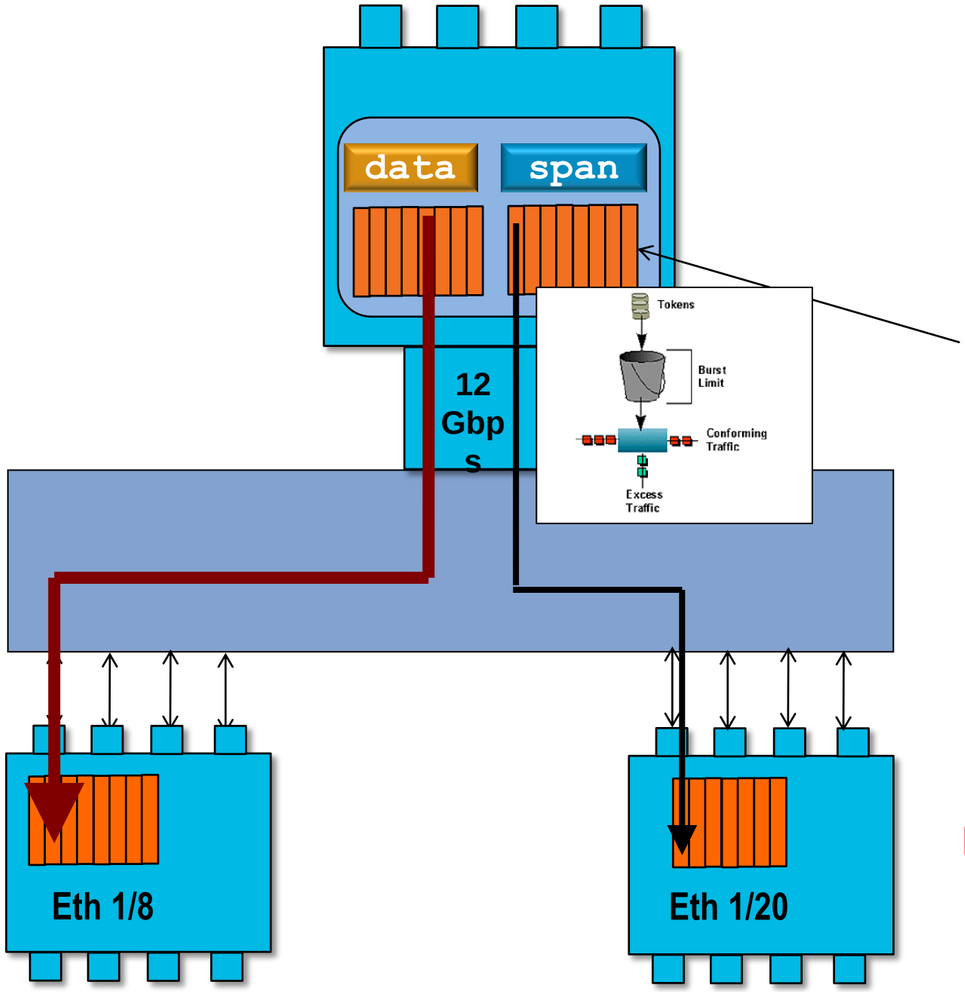
  When data rate is above 5Gbps, SPAN traffic is reduced to 0.75Gbps to avoid potential congestion over the link between ingress port and switch fabric

  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



- 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
```

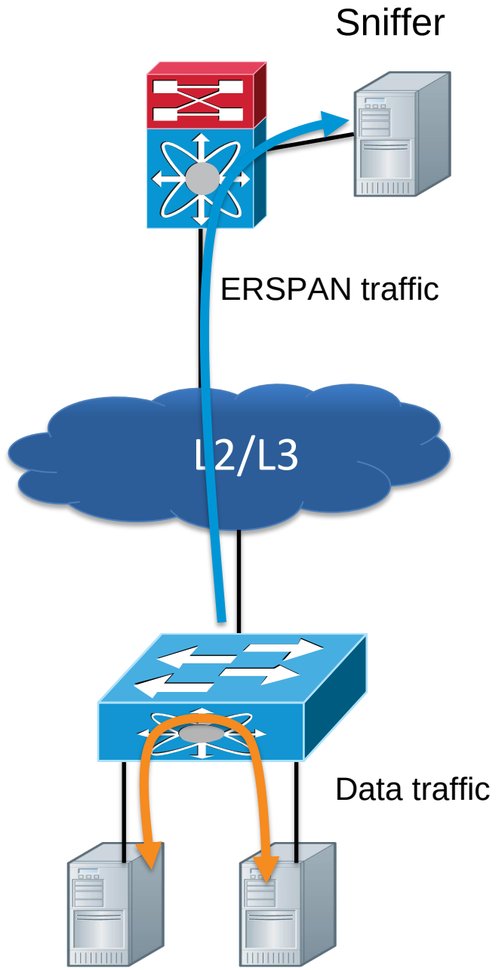
- Check SPAN packets drop due to SPAN policing

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

Slot 0 Carmel 0 register contents:
Register Name                                                  |  Offset   | Value
---------------------------------------------------------------+-----------+-----------
car_bm_CNT_span0_drop_addr_1                                   |  0x523fc  | 0xee222553
car_bm_CNT_span1_drop_addr_1                                   |  0x52400  | 0
car_bm_CNT_span2_drop_addr_1                                   |  0x52404  | 0
car_bm_CNT_span3_drop_addr_1                                   |  0x52408  | 0
car_bm_CNT_span4_drop_addr_1                                   |  0x5240c  | 0
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 for ERSPAN source sessions only
- N7K, Cat6K and Nexus 1110 NAM can de-capsulate ERSPAN
- ERSPAN does not require L3 module and L3 license



MAC header 14 bytes	IPv4 header 20 bytes	GRE header 8 bytes	ERSPAN header 8 bytes	Original packet (Ethernet frame)	CRC 4 bytes

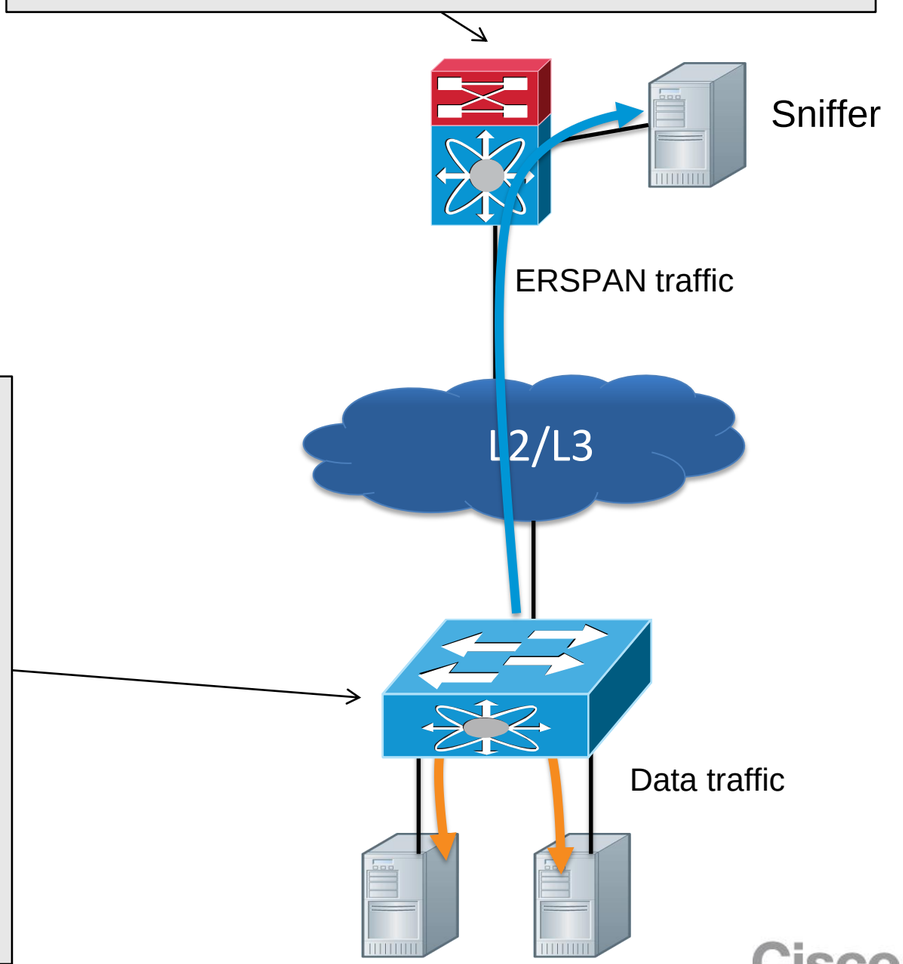
Nexus 5000 & 5500 ERSPAN

Encapsulated Remote SPAN

- On N5K the ERSPAN destination is the IP address of the remote switch that will de-capsulate the ERSPAN traffic
- Software figures out the egress interface of the ERSPAN traffic by checking the destination IP address against its routing table
- Without L3, user need to configure IP address under SVI and configure static route for VRF “default”

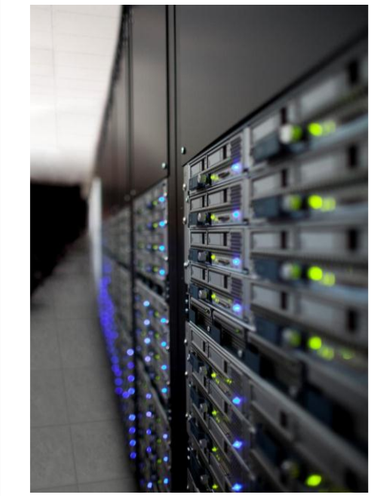
```
monitor session 1 type erspan-destination
  erspan-id 20
  vrf default
  source ip 65.65.65.2
  destination interface Ethernet1/1
  no shut
```

```
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
```



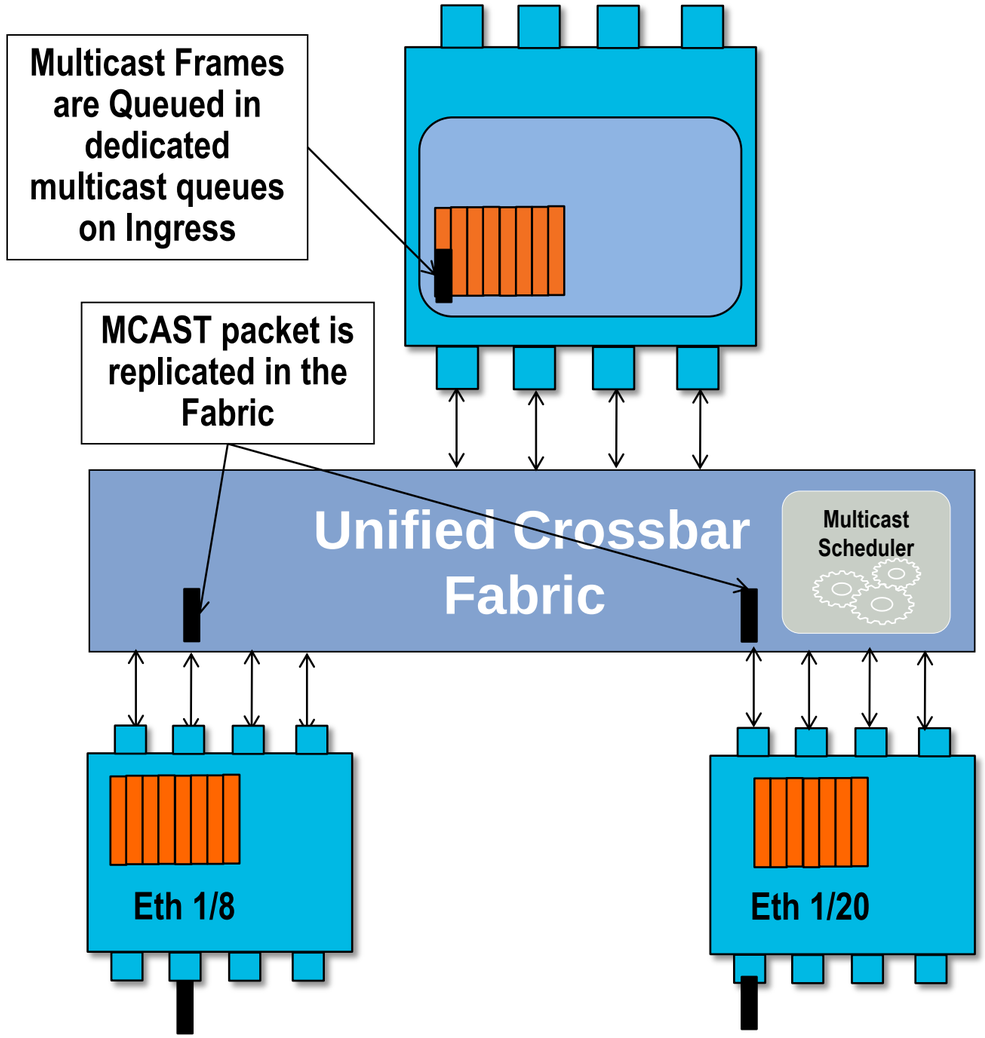
Agenda - Extras

- Nexus 5000
  - Layer 3 Module
  - SPAN
  - Multicast



Nexus 5500 Multicast Forwarding

Fabric-Based Replication

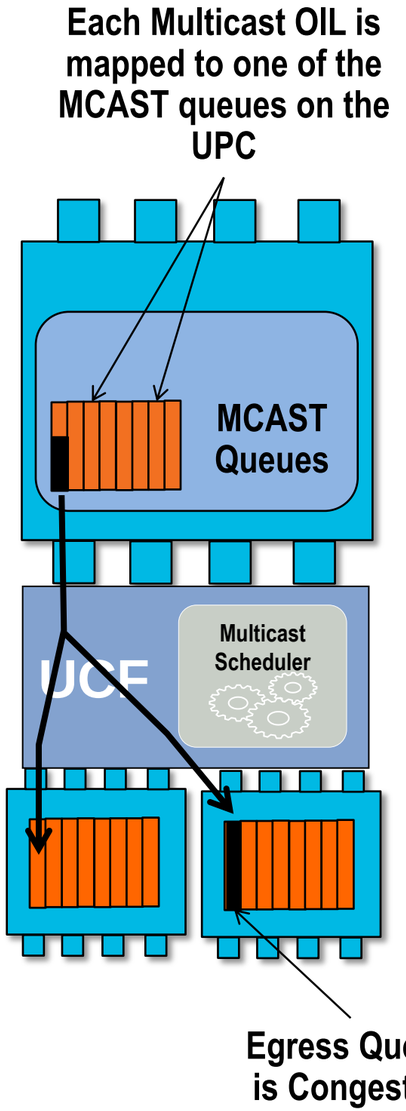


- Nexus 5500 use fabric based egress replication
- Traffic is queued in the ingress UPC for each MCAST group
- When the scheduler permits the traffic if forwarded into the fabric and replicated to all egress ports
- When possible traffic is super-framed (multiple packets are sent with a single fabric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

Nexus 5000 Multicast Forwarding

Multicast Queues and Multicast Group Fan-Out

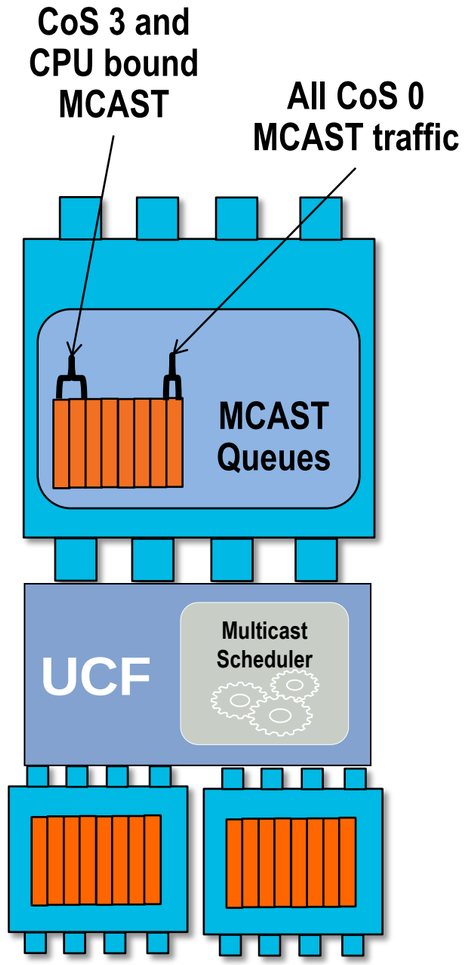
- “FAN-OUT” = an Output Interface List (OIL)
- The Nexus 5000 currently supports 1000 fan-outs and 4000 Multicast Groups
- The multicast groups need to be mapped to the 1000 fan-outs
- There are eight multicast queues per UPC forwarding engine (no VoQ for multicast)
- Hardware needs to map fan-outs to 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 prioritise 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 the system queuing for multicast may be non-optimal



Nexus 5000 Multicast Forwarding
Multicast Queues and Multicast Optimisation

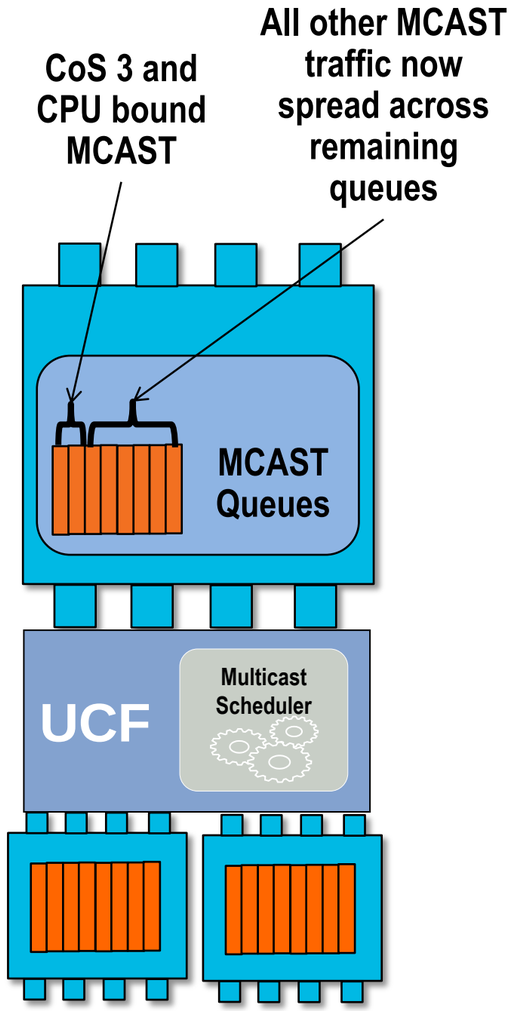
- "Multicast-optimise" when enabled for a class of traffic assigns multicast fan-outs in that class to any unused CoS queues on a round robin basis
- With multicast optimisation, you can assign these classes of traffic to the unused queues

  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
```

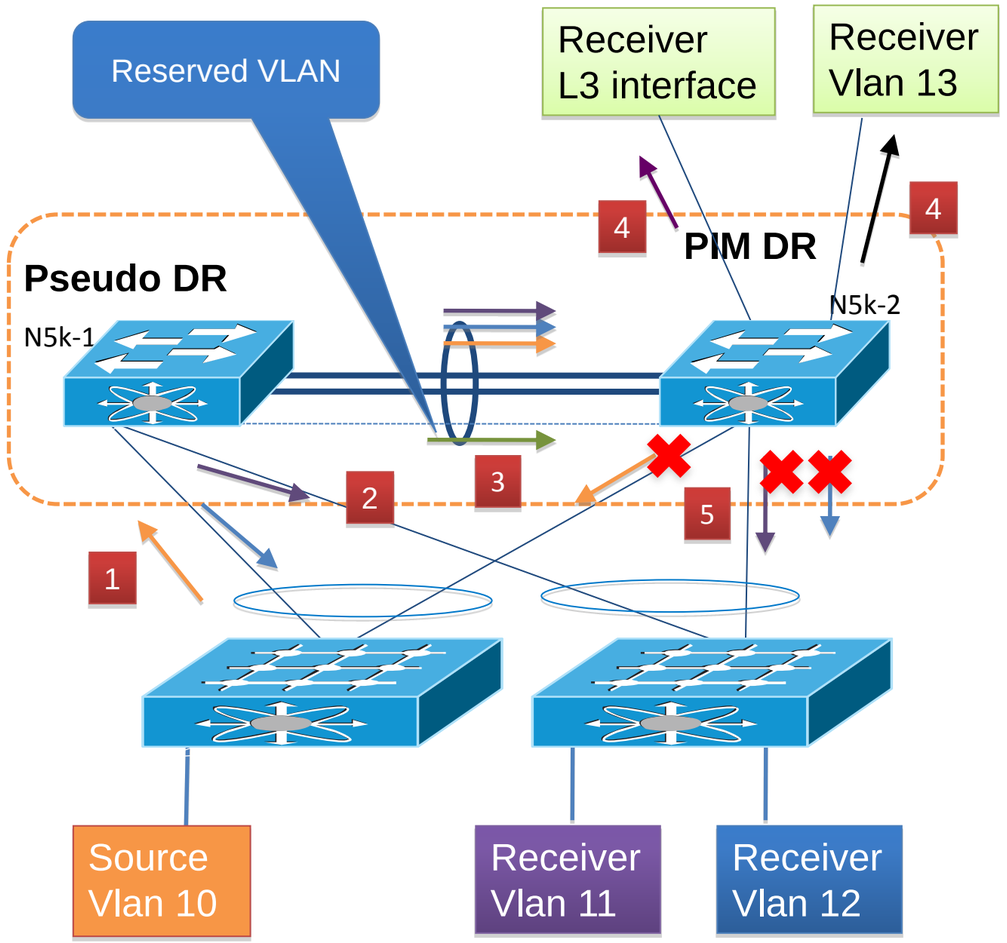


Nexus 5500 Multicast Forwarding

vPC and Layer 3 Interaction

- In a vPC when Nexus 5500 is running PIM both switches 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 Virtualised 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