Packet Journey inside Catalyst 9000 switches

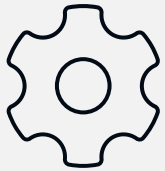
Jaroslaw Gawron
Principal Engineer
Cisco TAC

Ivan Shirshin
Technical Leader
Cisco TAC

BRKARC-2100

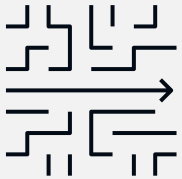
Session goals

Data and Control Plane Captures



Forwarding Verification

ASICs insights



Knowledge & Understanding

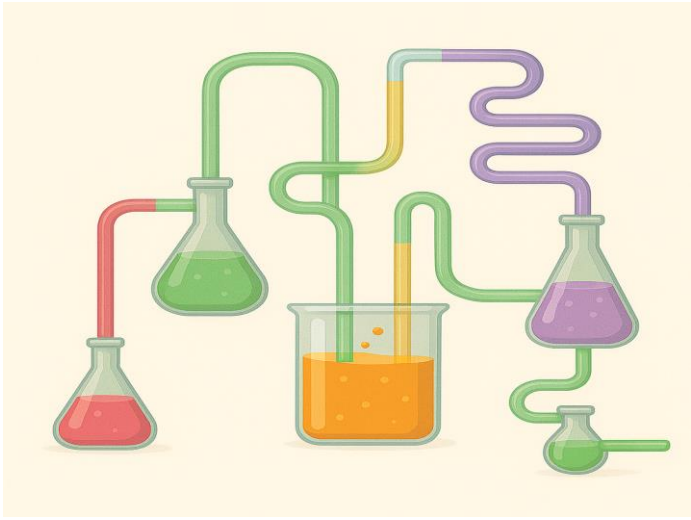
Toolbox

Packet Journey Mishaps

Are the packets getting dropped?

Or are the packets getting affected by one of the configured features?

Or maybe packets are processed by CPU instead of ASIC HW?



Cisco Webex App

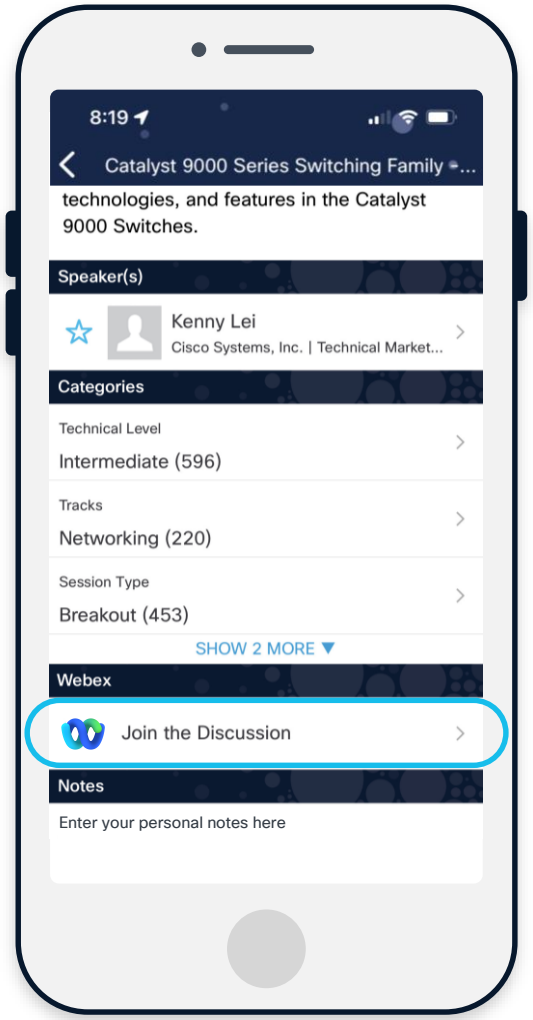
Questions?

Use Cisco Webex App to chat
with the speaker after the session

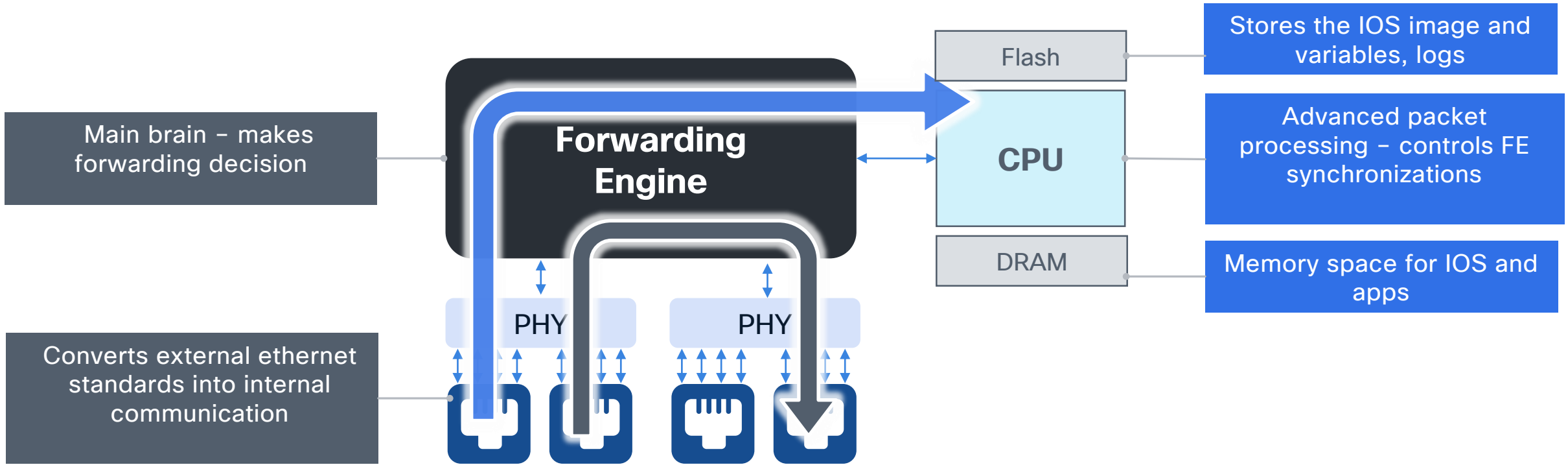
How

1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install the Webex App or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until June 13, 2025.



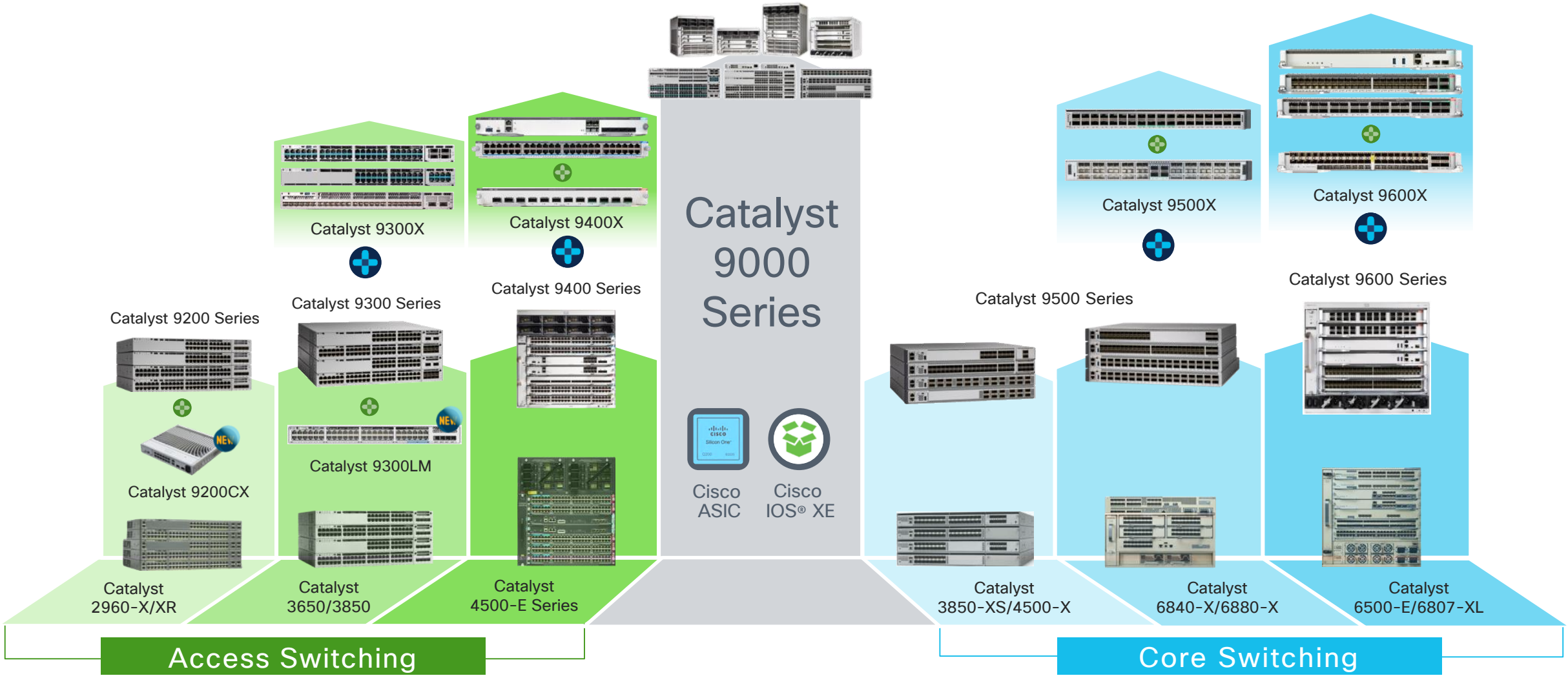
General Switch Architecture – Moon View



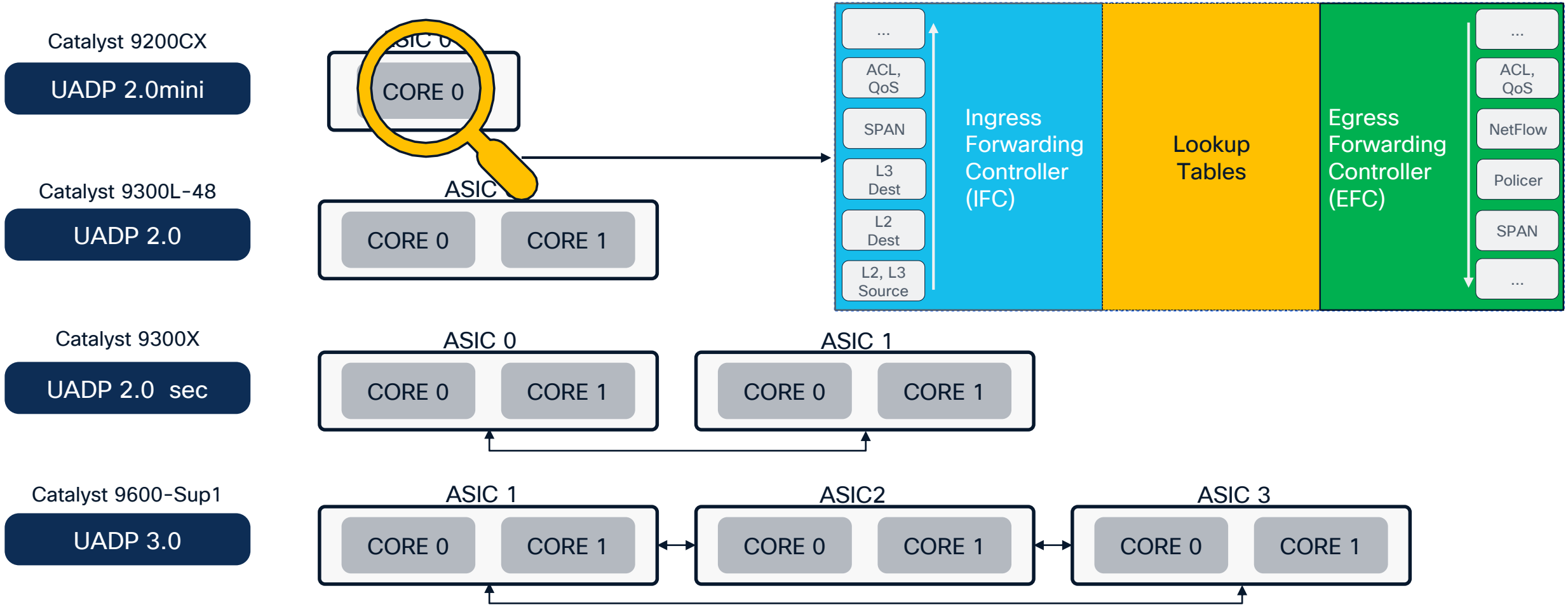
Catalyst 9000: UADP, S1

Landscape - Cisco Catalyst 9000 Switching Portfolio

One Family from Access to Core – Common Hardware & Software



Cisco UADP – One architecture, Multiple cores



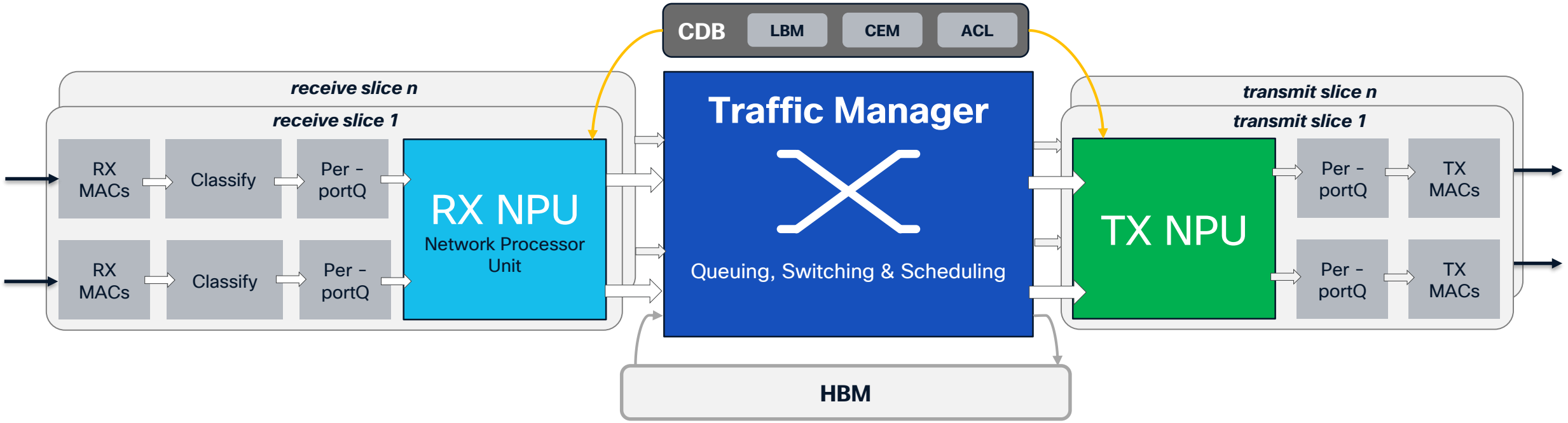
Multiplying ASICs on a switch and multiplying cores with in an UADP is a method to boost processing power

Cisco Silicon One – Q200, Single SOC

C9500X-28C8D
C9500X-60L4D
C9600X-SUP-2

ASIC 0

Q200



Multiplying slices on a single S1 ASIC is a method to boost processing power

Data Plane Captures

PHY Counters


```
show interface GigabitEthernet 1/0/1
--snip--
     1034 packets input, 124552 bytes, 0 no buffer
     Received 14 broadcasts (13 multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
```

IOS counters

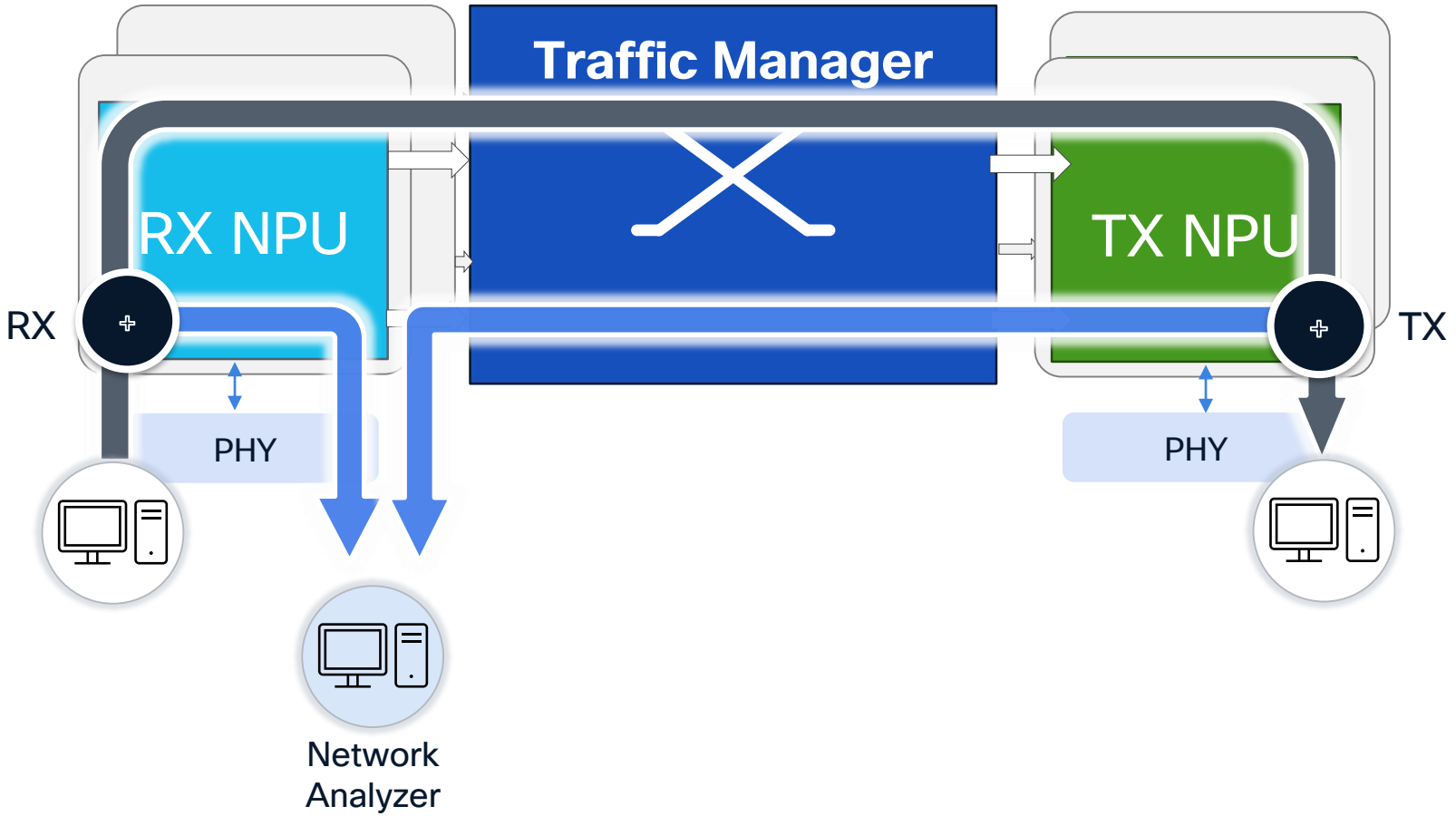
```
show controller ethernet-controller GigabitEthernet 1/0/1
Transmit                GigabitEthernet1/0/1                Receive
       132970 Total bytes                          125384 Total bytes
         1027 Unicast frames                         1027 Unicast frames
       119944 Unicast bytes                        119944 Unicast bytes
          128 Multicast frames                        14 Multicast frames
        12962 Multicast bytes                       5376 Multicast bytes
            1 Broadcast frames                         1 Broadcast frames
           64 Broadcast bytes                         64 Broadcast bytes
            0 System FCS error frames                  0 IpgViolation frames
--snip--
            0 Late collision frames                    0 SymbolErr frames
            0 Excess Defer frames                      0 Collision fragments
            0 Good (1 coll) frames                     0 ValidUnderSize frames
            0 Good (>1 coll) frames                    0 InvalidOverSize frames
            0 Deferred frames                          0 ValidOverSize frames
            0 Gold frames dropped                      0 FcsErr frames
```

PHY Counters

```
show controller ethernet-controller GigabitEthernet 1/0/1 phy
```

L1 Auto neg status

Data Plane Captures



- SPAN
- RSPAN
- ERSPAN
- EPC

SPAN | RSPAN | ERSPAN Configuration

	SPAN	RSPAN	ERSPAN
Switch 1	Define source and destination ports	Define source session and remote vlan	Define source interface and tunnel IPs

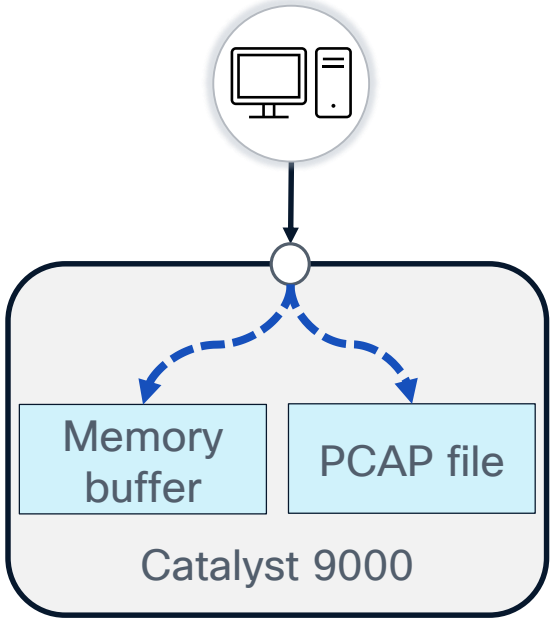
	`monitor session 1`
	` source interf gig1/0/1`
	`monitor session 1`
	` destination interf gig1/0/2`
		`vlan 100`
		` remote-span`
		`monitor session 1`
		` source interf gig1/0/1 tx`
		`monitor session 1`
		` destination remote vlan 100`
			`monitor session 1 type erspan-source`
			`  source interf gig1/0/1`
			`  destination`
			`  erspan-id 100`
			`  ip address 10.1.0.2`
			`  origin ip address 10.1.0.1`

Switch 2		Define destination session and remote vlan	Define destination interface and tunnel IP

		`vlan 100`
		` remote-span`
		`monitor session 1`
		` source remote vlan 100`
		`monitor session 1`
		` destination interf gig1/0/1`
			`monitor session 1 type erspan-destination`
			`  destination interf gig1/0/2`
			`  source`
			`  erspan-id 100`
			`  ip address 10.1.0.2`

EPC - Embedded Packet Capture

- Ability to capture packets in the device and store in local buffer
- Filter with ACL
- On-the-box analysis or Export to PCAP



Define a Capture Point

```
monitor capture mycap [interface INTERFACE | control-plane] <in|out|both>
```

Define buffer criteria

```
monitor capture mycap buffer [circular] [limit packets <1-10000>]
monitor capture mycap buffer size <1-10>
```

Define filters

```
monitor capture mycap access-list CAP_ACL
monitor capture mycap match { any | mac mac-match-string | ipv4 { any | host | protocol} { any | host} | ipv6 { any | host | protocol} { any | host} }
```

Start and stop the capture

```
monitor capture mycap start
monitor capture mycap stop
```

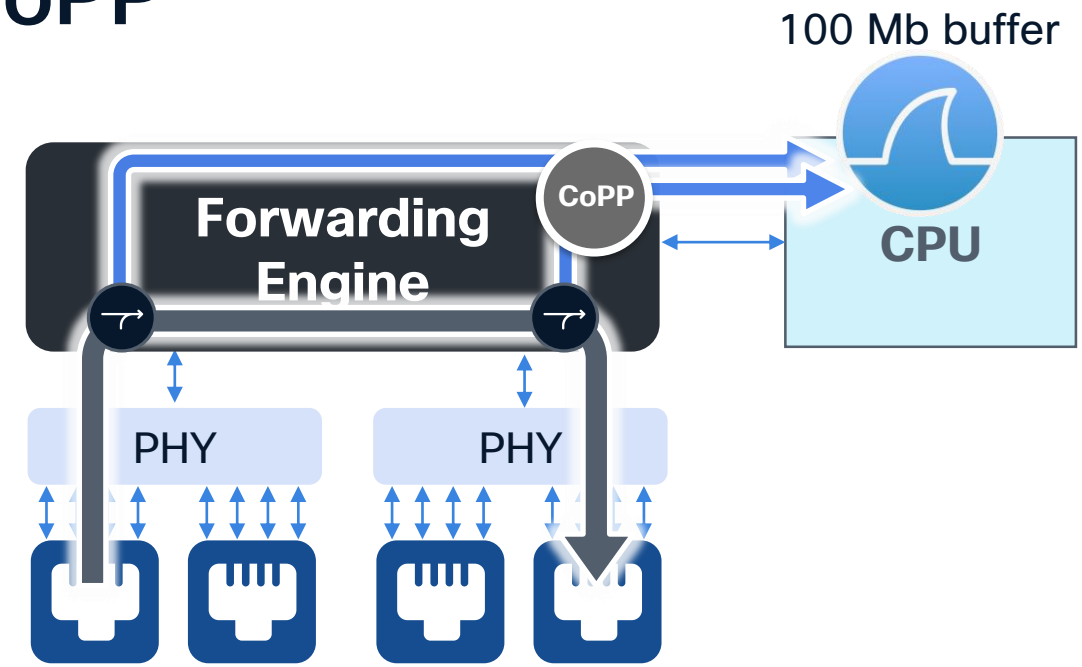
Embedded Packet Capture and CoPP

```
policy-map system-cpp-policy
  class  system-cpp-police-sw-forward
  police 1000
```

UADP

```
policy-map system-cpp-policy
  class  system-cpp-default
  police 2000
```

Silicon One



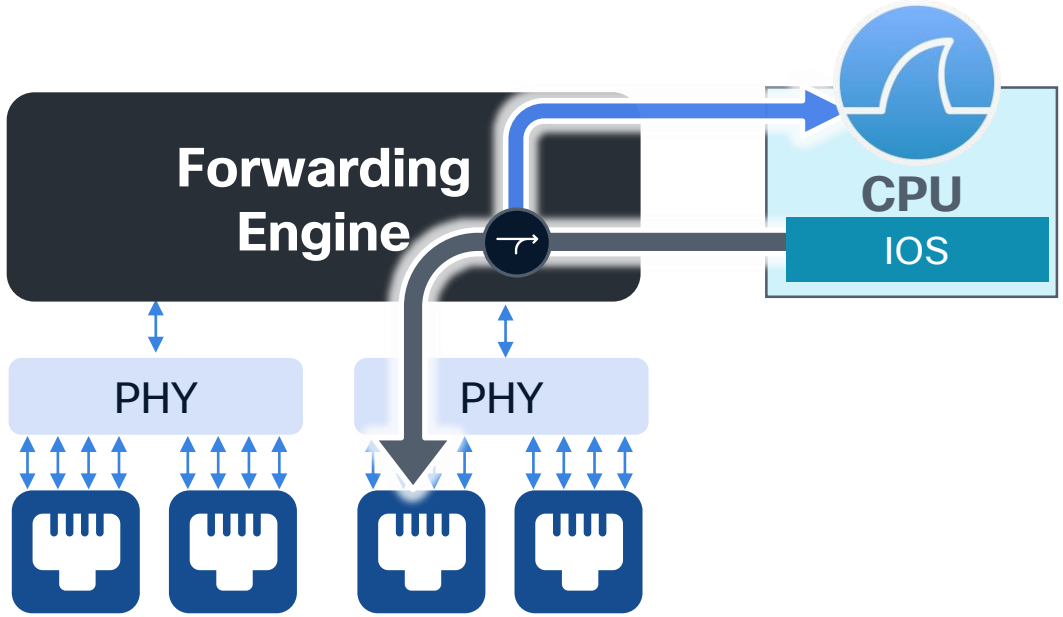

```
show platform hardware fed switch active qos queue stats internal cpu policer

                          CPU Queue Statistics
============================================================================================

                                                 (default) (set)     Queue         Queue
QId PlcIdx  Queue Name                  Enabled    Rate     Rate     Drop(Bytes)   Drop(Frames)
--------------------------------------------------------------------------------------------
<snip>
21   13     LOGGING                      Yes      1000      1000     0             0
```

UADP

Embedded Packet Capture and CPU-injected packets

".. CPU-injected packets are considered control plane packets. Therefore, these types of packets will not be captured on an interface egress capture... "

Network Management Configuration Guide, Cisco IOS XE (Catalyst 9300 Switches)



EPC in egress on physical interface direction does not capture all of CPU injected traffic:

```
monitor capture mycap gig 1/0/1 out match any start
```

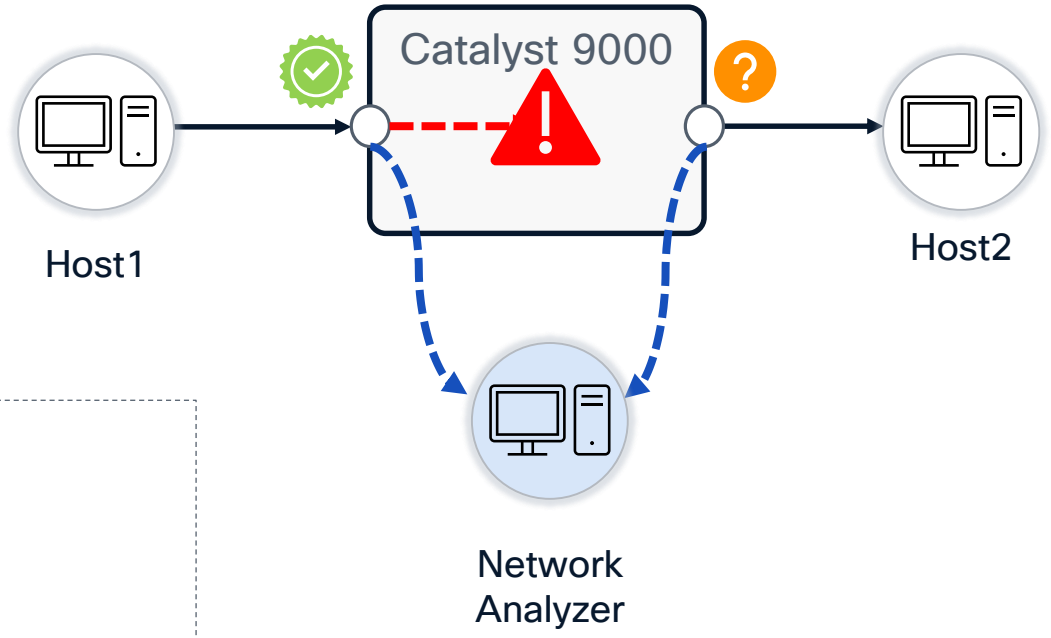
If you have a need to analyze CPU injected traffic use:

```
monitor capture mycap control-plane out match any start
```

ASICs Drops

Packet Drop Issues

- Connectivity issue between Hosts
- Traffic drop suspected on Catalyst 9000 switch




```
Host1# ping 10.1.20.2
PING 10.1.20.2 (10.1.20.2) from 10.1.10.1 : 56 data bytes
Request 0 timed out
Request 1 timed out
Request 2 timed out
Request 3 timed out
Request 4 timed out
```

How can we investigate the reason for packet drop?

ASIC Exceptions (aka Drops)

UADP & Silicon One

UADP

```
show platform hardware fed [active|1|2] fwd-asic drops exceptions
****EXCEPTION STATS ASIC INSTANCE 0 (asic/core 0/0)****
================================================================================
 Asic/core |                NAME                    |   prev   |  current  |  delta
================================================================================
0  0  NO_EXCEPTION                                   1653       1857        202
0  0  IPV4_CHECKSUM_ERROR                            54         61          7
0  0  ROUTED_AND_IP_OPTIONS_EXCEPTION                0          0           0
0  0  SIA_TTL_ZERO                                   0          0           0
--snip--
```

Identify correct ASIC instance

delta column helps to identify the name of the exception

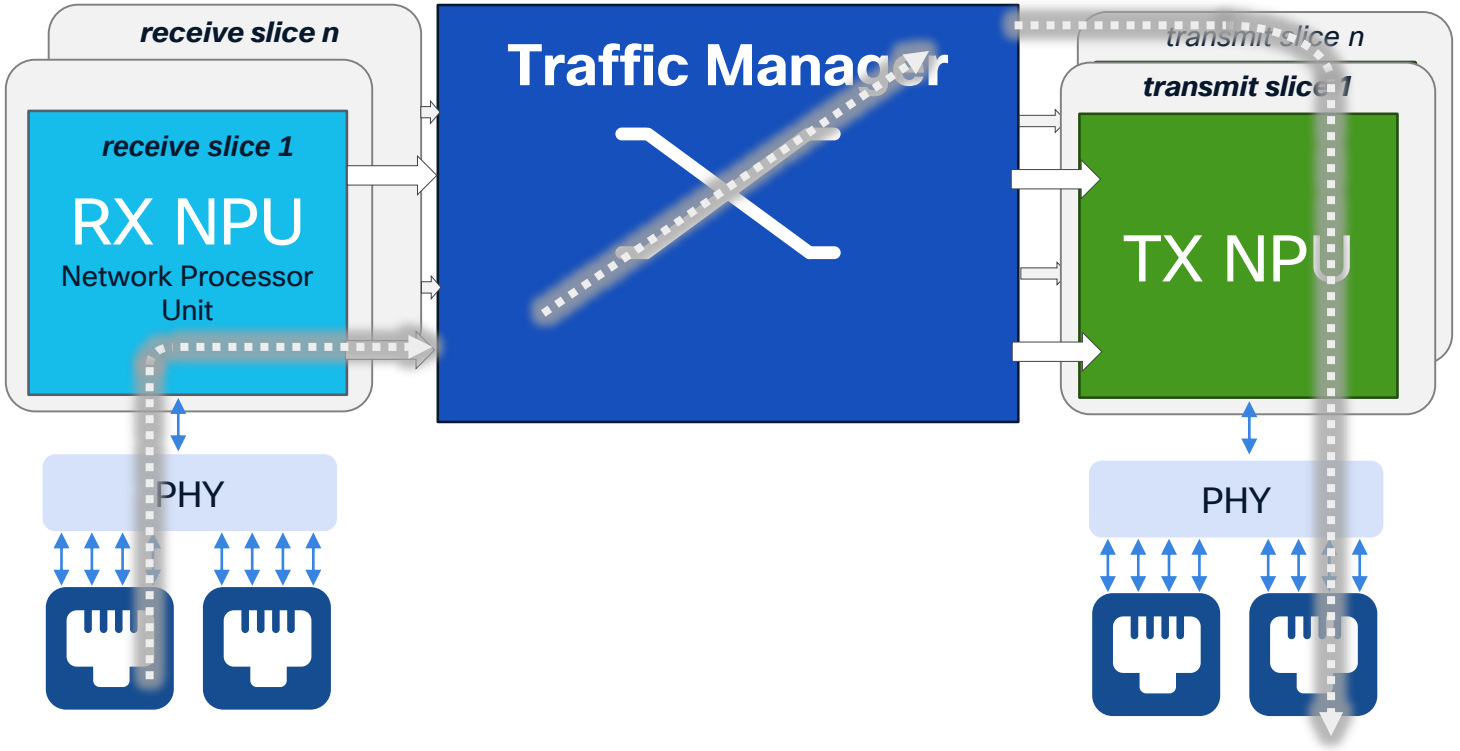
Exception name

Silicon One

```
show platform hardware fed active fwd-asic drops asic all
=============================================================================================================
|  #     |      Counters Name      |slice_number|   ifg_number   |  prev_value    |  current_value  |     delta      |
=============================================================================================================
|  3166 |PDVOQ Slice0 drop pkts    |           0|              -1|              97|               97|               0|
|  3167 |PDVOQ Slice0 drop bytes   |           0|              -1|           14065|            14065|               0|
|  3168 |PDVOQ Slice1 drop pkts    |           1|              -1|               0|                0|               0|
|  3169 |PDVOQ Slice1 drop bytes   |           1|              -1|               0|                0|               0|
--snip--
```

Is my ASIC/Core dropping any traffic ?

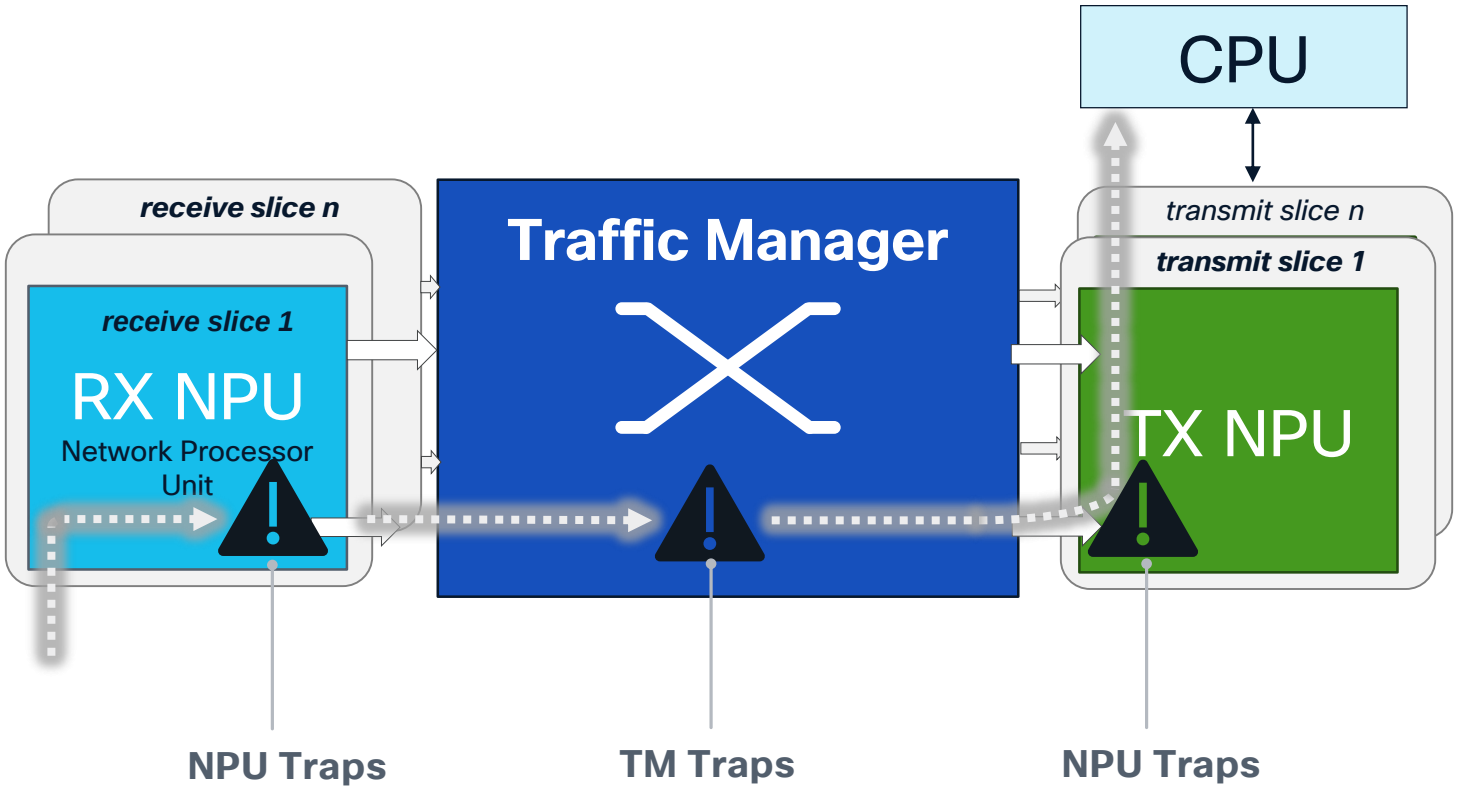
Mapping Interfaces to ASIC




```
show platform software fed active ifm mappings
Interface                IF_ID     Inst  Asic  Core  IFG
HundredGigE1/0/1         0x406     0     0     5
HundredGigE1/0/2         0x407     0     0     5
..
FourHundredGigE1/0/17    0x416     0     0     4
FourHundredGigE1/0/18    0x417     0     0     4
FourHundredGigE1/0/19    0x418     0     0     4
FourHundredGigE1/0/20    0x419     0     0     4
FourHundredGigE1/0/21    0x41a     0     0     3
```

Q200 - Single SOC (Switch on Chip)

ASIC Investigation for Drop Reason



S1 TRAP Mechanism

- Traffic based
- Triggered based on predefined set of events / type of packets (Expected as well as non-expected behaviours)
- Available for NPUs and TM components
- Allows to redirect traffic to CPU instead of silently dropping it

Ability to understand dropped traffic

S1 Traps Insights – Trap Descriptions and Actions

```
show platform hardware fed active fwd-asic insight s1_trap_status()
+-----------------------------------------------+-------------------------------------------------------------------
| Trap Type                | Event Description                        | Action         | Direction | <snip>
+-----------------------------------------------+-------------------------------------------------------------------
| ETHERNET_ACL_DROP        | The packet matched an ingress security   | DROP           | BOTH      |
                             ACL rule with a drop action.
|                          |                                          |                |           |
| ETHERNET_ACL_FORCE_PUNT  | The packet matched an ingress security   | DROP           | BOTH      |
                              ACL rule with a punt to host action.
|                          |                                          |                |           |
| ETHERNET_CISCO_PROTOCOLS | Trap incoming packet if packet is        | PUNT           | INGRESS   |
                              Cisco-specific control protocol
                              (e.g. CDP, VTP, DTP, PAgP, UDLD, PVSTP+).
|                          |                                          |                |           |
| ETHERNET_DA_ERROR        | Incoming packet's MAC DA equals zero.    | DROP           | INGRESS   |
|                          |                                          |                |           |
| ETHERNET_DHCPV4_CLIENT   | DHCP packet over IPv4, destined for      | PUNT           | INGRESS   |
                              a client (destination UDP port 68).
```

Action to the traffic forwarding

Traffic direction the trap is applied to

EPDA – Enhanced Packet Drop Analyzer

Silicon One

Identify traffic

```
show platform hardware fed active fwd-asic traps npu-traps asic all
```

Trap ID	Asic	NPU Trap Name	Prev	Current	Delta
1	0	la_event_e_ETHERNET_ACL_DROP	0	0	0
2	0	la_event_e_ETHERNET_ACL_FORCE_PUNT	0	0	0
4	0	la_event_e_ETHERNET_NO_TERMINATION_ON_L3_PORT	0	0	0
5	0	la_event_e_ETHERNET_CISCO_PROTOCOLS	2	2	0
6	0	la_event_e_ETHERNET_DA_ERROR	0	0	0
7	0	la_event_e_ETHERNET_DHCPV4_CLIENT	0	0	0
<snip>					
149	0	la_event_e_L3_NULL_ADJ	30255	34850	4595

Enable selected trap

```
debug platform software fed active drop-capture set-trap npu-traps l3 l3-null-adj
```

Capture traffic

```
debug platform software fed active drop-capture start
```

```
debug platform software fed active drop-capture stop
```

EPDA Continued

Silicon One

Display packets

```
show platform software fed active drop packet-capture brief
DropPackets packet capturing: disabled. Buffer wrapping: disabled
Total captured so far :  2313 packet(s)
Capture capacity      :  4096 packet(s)
Max. Meta header size :    88 byte(s)
Max. Packet data size :   128 byte(s)

------ DropPackets Packet Number: 1, Timestamp: 2024/03/25 15:04:46.823 ------
 interface : phy:  [if-id: 0x00000000], pal:  [if-id: 0x00000000]
 misc info : cause: 0 [Reserved ], sub-cause: 0, linktype: UNKNOWN [0]
 CE    hdr : dest mac: 4e41.5000.0111, src mac: 4e41.5000.0111, ethertype: 0x7106
 meta  hdr : Nxt. Hdr: 0x1, Fwd. Hdr: 0x2, SSP: 0x19
 meta  hdr : DSP: 0xffff, SLP: 0xe, DLP: 0x95
 ether hdr : dest mac: 341b.2d76.fd02, src mac: 6c29.d29d.36c3
 ether hdr : vlan: 3012, ethertype: 0x8100
 ipv4  hdr : dest ip: 172.16.10.11, src ip: 192.168.100.18
 ipv4  hdr : packet len: 100, ttl: 254, protocol: 1 (ICMP)
 icmp  hdr : icmp type: 8, code: 0
```

Default buffer of 4k (can be changed to 16k)

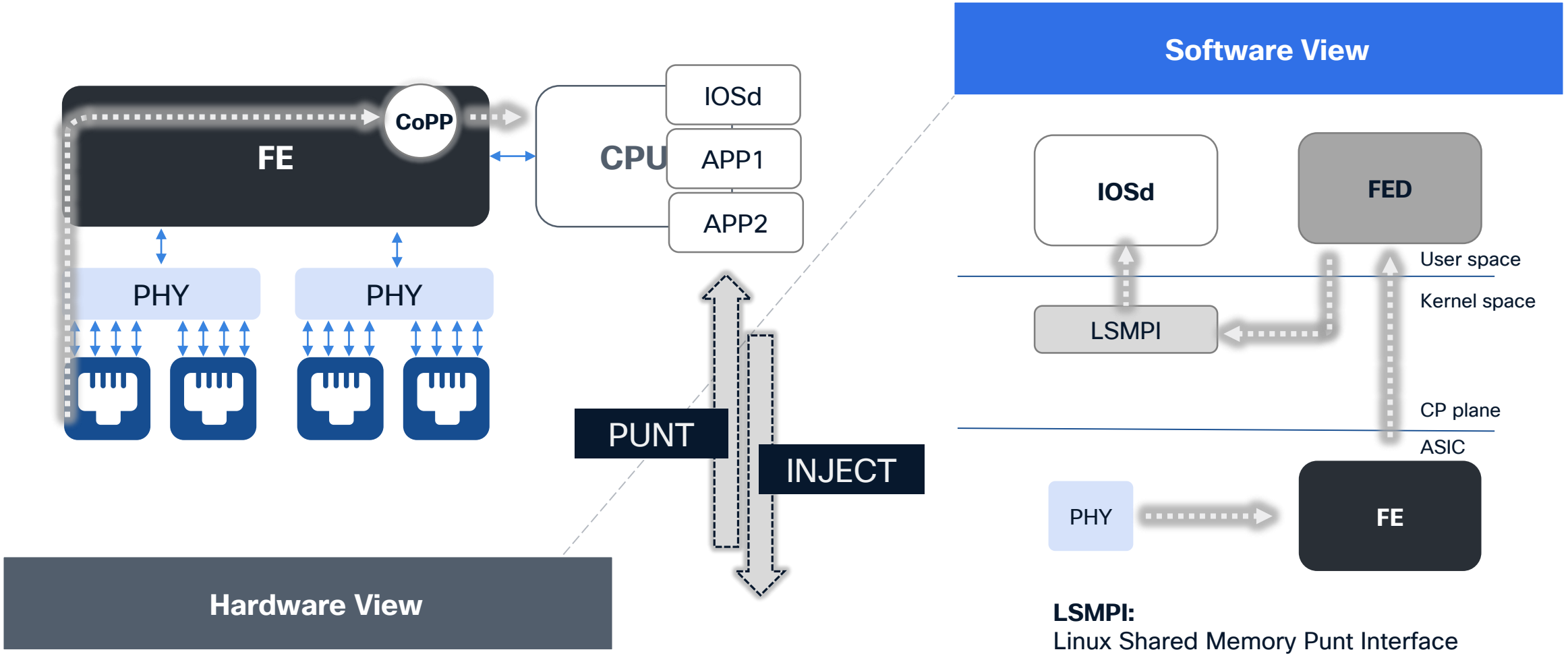
Details of dropped packet

Clear trap

```
debug platform software fed active drop-capture clear-trap npu-traps l3 l3-null-adj
```

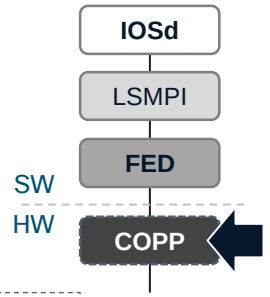
Control Plane Captures

Control Plane – Packet Journey in Software



LSMPI:
Linux Shared Memory Punt Interface

CoPP Overview UADP



UADP

```
show platform hardware fed [switch] active qos queue stats internal cpu policer

                        CPU Queue Statistics
============================================================================================

                                                    (default) (set)     Queue         Queue
QId PlcIdx  Queue Name                    Enabled     Rate     Rate     Drop(Bytes)   Drop(Frames)
--------------------------------------------------------------------------------------------
0    11     DOT1X Auth                     Yes       1000     1000     0             0
1    1      L2 Control                     Yes       2000     2000     1222          314
2    14     Forus traffic                  Yes       4000     4000     0             0
3    0      ICMP GEN                       Yes       750      750      0             0
4    2      Routing Control                Yes       5500     5500     0             0
<snip>
11   13     L2 LVX Data Pack               Yes       1000     1000     0             0
12   0      BROADCAST                      Yes       750      750      0             0
13   10     Openflow                       Yes       250      250      0             0
14   13     Sw forwarding                  Yes       1000     1000     0             0
<snip>
```

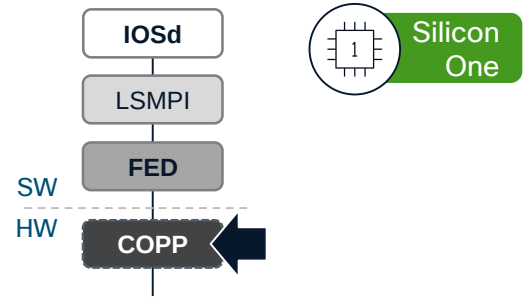
Non-zero counters indicate a loss of punted traffic

Queue name

Policing Values

Drops

CoPP Policer S1



show platform software fed active punt entries

Punject Punt Entries

Source	Name	Pri	TC	Policy	CIR-SW	CIR-HW	Pkts(A)	Bytes(A)	Pkts(D)	Bytes(D)
TRAP	ACL Drop(ETH)	1	0	system-cpp-default	2000	1931	0	0	0	0
MIRROR	ARP	4	4	system-cpp-police-arp	1000	965	0	0	0	0
TRAP	CISCO Protocols	3	5	system-cpp-police-l2-control	16000	15449	2	636	0	0
TRAP	**DHCP Client(v4)**	**3**	**4**	**system-cpp-police-dhcp-v4**	**6000**	**5793**	**3245**	**2332525**	**245**	**3532**
TRAP	DHCP Server(v4)	3	4	system-cpp-police-dhcp-v4	6000	5793	0	0	0	0
TRAP	DHCP Client(v6)	3	4	system-cpp-police-dhcp-v6	6000	5793	0	0	0	0
TRAP	DHCP Server(v6)	3	4	system-cpp-police-dhcp-v6	6000	5793	0	0	0	0
TRAP	ETH HOP-OPT	88	3	system-cpp-police-sw-forward	2000	1931	0	0	0	0
MIRROR	ISIS(L2)	3	5	system-cpp-police-isis	1000	965	4	6284	0	0
TRAP	LLDP	4	5	system-cpp-police-l2-control	16000	15449	2	636	0	0

<snip>

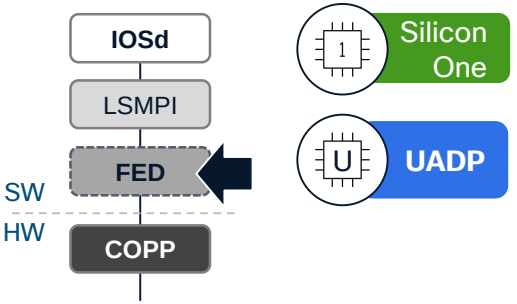
Configuration Class

Policing Values

Accepted

Dropped

Forwarding Engine Driver

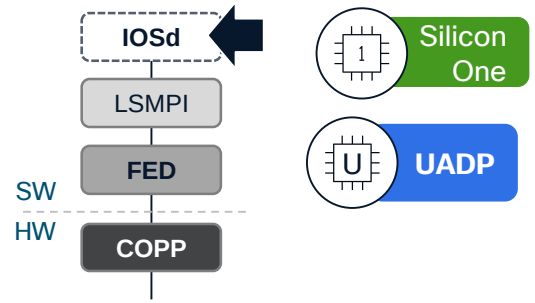



```
debug platform software fed active punt packet-capture start
Punt packet capturing started.
```

```
debug platform software fed active punt packet-capture stop
Punt packet capturing stopped. Captured 30 packet(s)
```

```
show platform software fed active punt packet-capture brief
Punt packet capturing: disabled. Buffer wrapping: disabled
Total captured so far :    30 packet(s)
Capture capacity      :  4096 packet(s)
Max. Meta header size :    88 byte(s)
Max. Packet data size :   128 byte(s)

------ Punt Packet Number: 1, Timestamp: 2024/12/16 10:43:38.526 ------
 interface : phy: HundredGigE1/0/3 [if-id: 0x000004a7], pal: HundredGigE1/0/3 [if-id: 0x000004a7]
 misc info : cause: 5 [CLNS IS-IS Control], sub-cause: 0, linktype: LAYER2 [10]
 CE    hdr : dest mac: 4e41.5000.0111, src mac: 4e41.5000.0111, ethertype: 0x7106
 meta  hdr : Nxt. Hdr: 0x1, Fwd. Hdr: 0, SSP: 0x1b
 meta  hdr : DSP: 0xffff, SLP: 0x4, DLP: 0x84
 ether hdr : dest mac: 0900.2b00.0005, src mac: 6c29.d2b2.59c6
 ether hdr : length: 1494
<snip>
```

Packet capture at IOSd Interface




```
debug ip packet detail
debug arp
debug bgp …
```

```
monitor capture CPU control-plane both match any start
```

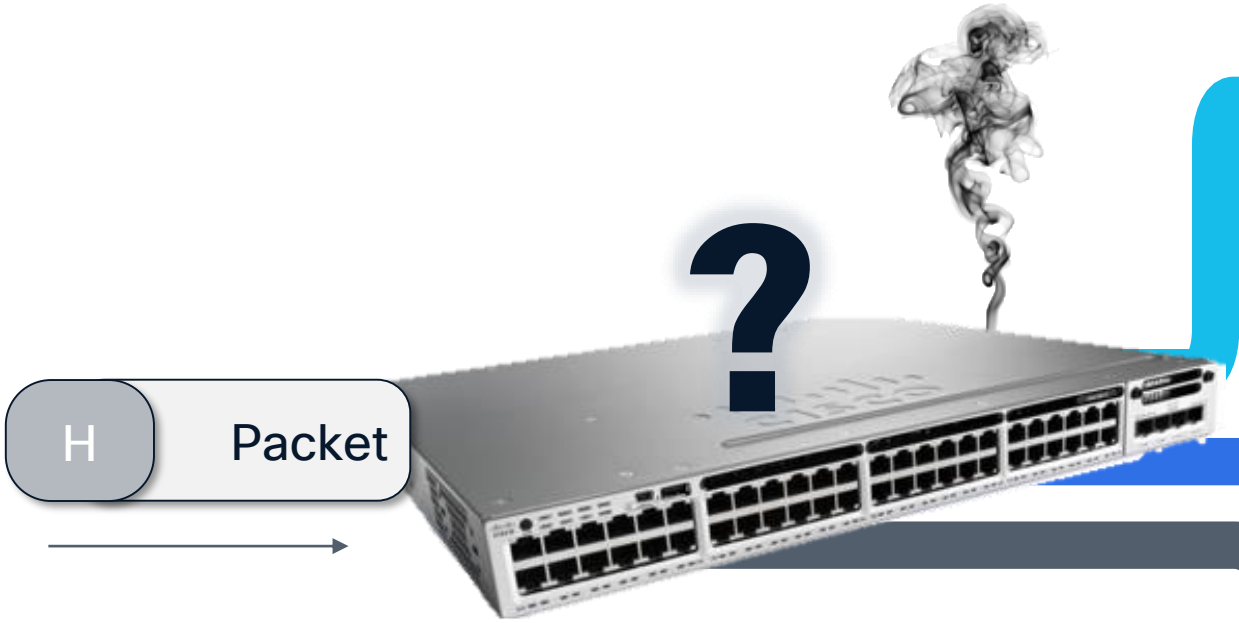
```
monitor capture CPU stop
```

```
show monitor capture CPU buffer brief
Starting the packet display ........ Press Ctrl + Shift + 6 to exit

    1   0.000000 6c:29:d2:b2:59:c6 -> 09:00:2b:00:00:05 ISIS HELLO 1508 P2P HELLO, System-ID: 0100.9800.4001
    2   0.508553 6c:29:d2:93:6d:46 -> 09:00:2b:00:00:05 ISIS HELLO 1508 P2P HELLO, System-ID: 0100.9800.4010
    3   0.617116 192.168.40.197 -> 224.0.0.13   PIMv2 72 Hello
    4   0.685620 192.168.40.3 -> 100.64.0.1   Syslog 120 LOCAL7.INFO: 65477: *Dec 16 10:53:56.213:
    5   1.010079 34:1b:2d:76:fc:01 -> 01:00:0c:cc:cc:cd STP 64 RST. Root = 32768/1/34:1b:2d:76:fc:00  Cost = 0  Port = 0x8001
```

```
monitor capture CPU export location flash:capture_cpu.pcap
Export Started Successfully
```

Packet Tracing

How to track forwarding decisions ?



Forwarded

- What is the egress interface?
- How does the packet look like at the egress?

Dropped

- Why is the packet dropped?

Punted

- Why is the traffic a subject for control-plane processing?

My packet is addressed to 192.168.40.1

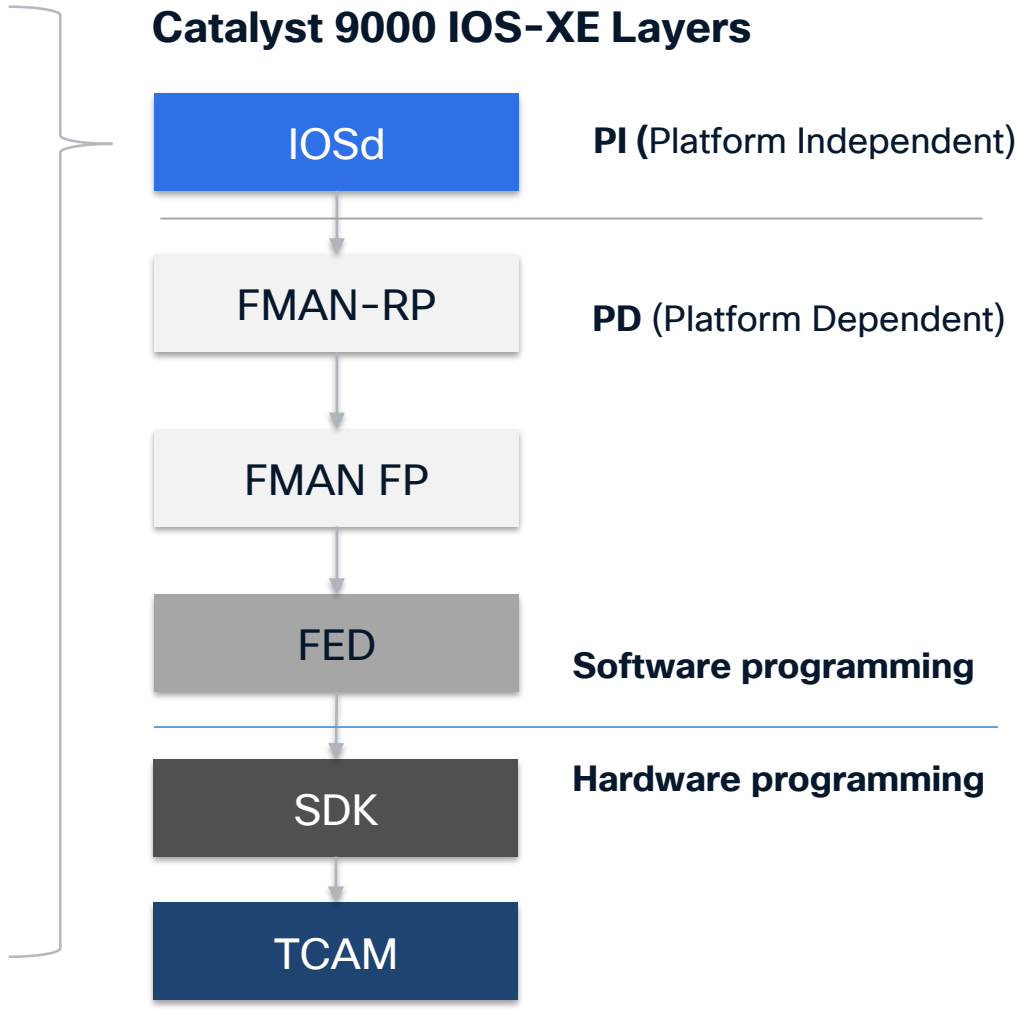
How will it egress my switch?

```
show ip route 192.168.40.1
Routing entry for 192.168.40.1/32
  Known via "isis", distance 115, metric 30, type level-2
  Redistributing via isis
  Last update from 192.168.40.222 on Vlan412, 4d19h ago
  Routing Descriptor Blocks:
  * 192.168.40.222, from 192.168.40.1, 4d19h ago, via Vlan412
      Route metric is 30, traffic share count is 1
```

```
show ip arp 192.168.40.222
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  192.168.40.222        174   6c29.d289.a15b  ARPA   Vlan412
```

```
show mac address-table address 6c29.d289.a15b
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
 412    6c29.d289.a15b    DYNAMIC     Hu1/0/1
```

Will my packet egress out via the Hu1/0/1 ?



Forwarding Tracking Tools

Catalyst 9000 Family

Simulates the arrival of a requested packet at a specified interface by the CPU. Based on the current state of the TCAM, it determines how the switch would process the traffic if it were received on the simulated interface. This is a software-based solution

SPF | **CPU Simulation** | **UADP 2.0** / **UADP 3.0** / **Q200**

Show Platform Forward

```
show platform hardware fed active forward …
```

Triggered by the first packet that meets the specified conditions, it collects forwarding data from the lookup stages without impacting live traffic. This solution relies on ASIC support and is hardware-based

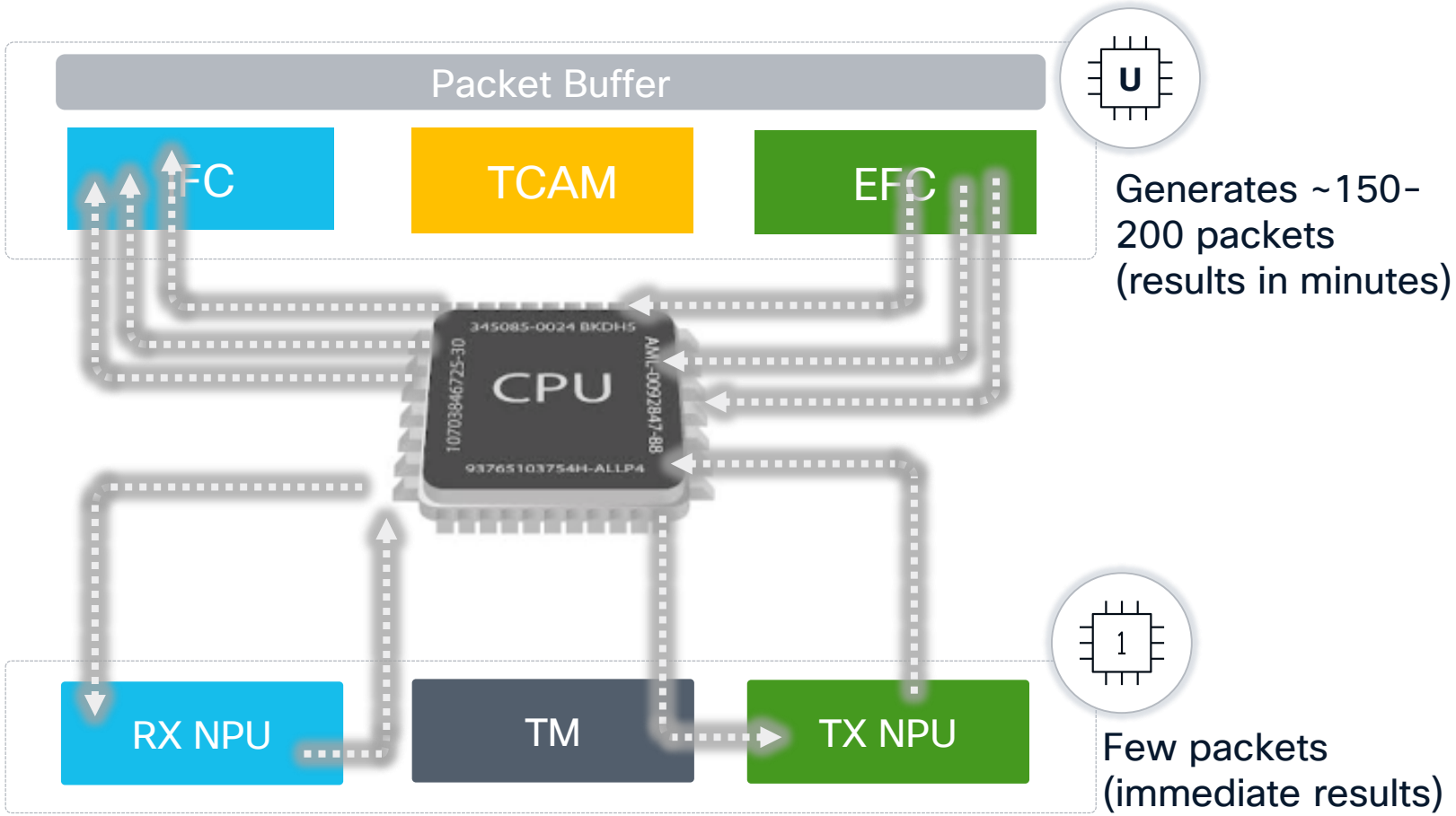
PSV | **Live tracing** | **UADP 3.0**

Packet State Vector

```
debug platform hardware fed active capture …
```

Forwarding tracing tools depend on the actual content of TCAM memory not only on the PI layer

Show Platform Forward



- Works on all UADP and S1 based platforms
- Offline mode
- Typically generates multiple packets to capture each stage
- The packets generated in SPF don't get forwarded
- Show forward cannot indicate QOS/Policer drops
- Input manual or in pcap format
- Adjusted to advanced forwarding scenarios

Show Platform Forward (SPF)

Manual Header Definition

Define packet header.
Specify L2/L3/L3 headers

UADP & Silicon One

```
show platform hardware fed switch [1|2|..] forward interface GigabitEthernet 1/0/22
0011.9267.b370 58bf.eab6.7fe2 ipv4 10.200.1.100 10.201.1.100 tcp 65000 80 0
```

Incoming interface

L2 SRC | L2 DST | L3 SRC | L3 DST | L4

~ 2min

Wait for results. Time depends on the type of used encapsulation

UADP

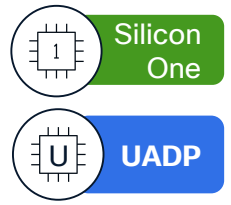
```
*Jun 12 10:50:49.075: %SHFWD-6-PACKET_TRACE_DONE:Switch 1 R0/0: fed:
Show fwd is completed. The capture file can be found at /flash/shfwd+timestamp.log
(ie. shfwdxxxxxx-xxxxxx.log) .
```

Verify forwarding results

UADP & Silicon One

```
show platform hardware  fed switch [1|2|..] forward last summary
show platform hardware  fed switch [1|2|..] forward last detail
```

Show Platform Forward Leveraging EPC



Define EPC filters and start capture

```
monitor capture TAC interface Gig 1/0/2 in match any
monitor capture TAC start
monitor capture TAC stop
Capture statistics collected at software:
        Capture duration - 8 seconds
        Packets received - 11
        Packets dropped - 0
        Packets oversized - 0
Capture buffer will exists till exported or cleared
Stopped capture point : TAC
```

Verify Content of EPC buffer and find the interesting frame

```
show monitor capture TAC buffer
Starting the packet display ........ Press Ctrl + Shift + 6 to exit

1 0.000000 192.168.100.100 -> 192.168.100.1 ICMP 114 Echo (ping) request
2 0.000011 192.168.100.1 -> 192.168.100.100 ICMP 114 Echo (ping) reply
```

Export EPC buffer to a file

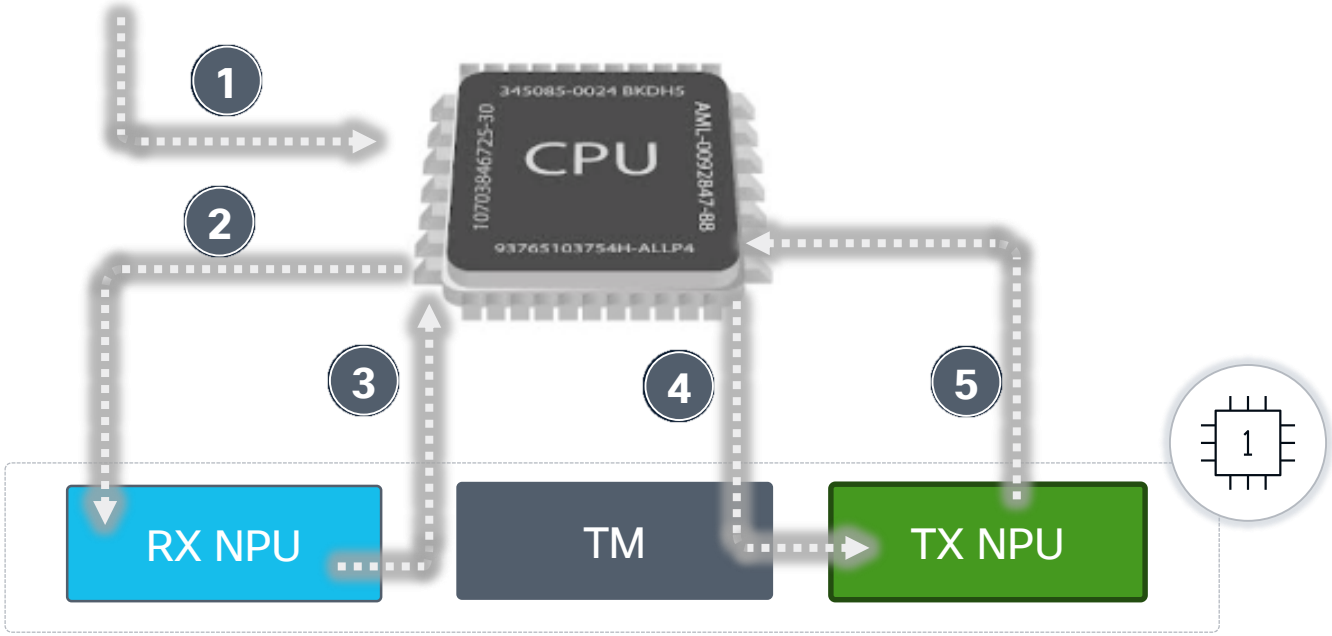
```
monitor capture TAC export location flash:capture2.pcap
```

Use the EPC pcap as the input for SPF

```
show platform hardware fed switch 1 forward interface GigabitEthernet 1/0/2 pcap flash:capture2.pcap number 2 data
```

Show Platform Forward

SiliconOne Platforms



1. Simulated packet definition
2. Info on how the packet is injected in NPU RX
3. NPU RX Forwarding results (Ingress resolution)
4. Info on how the packet is injected in NPU TX
5. NPU TX Forwarding results (Egress resolution + Rewrite Info)

By default only stages 1,3,5 presented

```
sh plat hardw fed act forward last summary
```

In detail mode all 5 stages presented

```
sh plat hardw fed act forward last detail
```

Results available immediately - no need to wait

Last 16 results stored

```
sh plat hardw fed act forward last list
```

Available since 17.15.1

Understanding the outputs

Silicon One

By default, the outputs present the most relevant information to help understand the forwarding decision:

Input packet

```
Packet-trace Id: spf1284569044
   Input packet:
      ###[ Ethernet ]###
        dst       = 98:a2:c0:7e:35:02
        src       = 20:20:30:30:40:40
        type      = IPv4
      ###[ IP ]###
           version   = 4
           ihl       = 5
           tos       = 0x0
           len       = 28
           id        = 1
           flags     =
           frag      = 0
           ttl       = 64
           proto     = icmp
           chksum    = 0xa831
           src       = 192.168.40.141
           dst       = 192.168.40.209
           \options   \
      ###[ ICMP ]###
              type      = echo-reply
              code      = 0
              chksum    = 0xffff
              id        = 0x0
              seq       = 0x0
              unused    = ''
```

Ingress resolution

```
  (Ingress):
  Ingress Interface : HundredGigE1/0/4
  traces:
      - # 1

   Decision:
      #
      # dsp: HundredGigE1/0/2
      # fwd_hdr_type: ipv4
      # fwd_relay_id_or_pwe_id: '0x0'
      # l3_dlp_id: 0x8 (Vlan411)
      # rx_nw_app_or_lb_key: '0xe746'
      # tm_hdr_type: '0x1'
      #
```

Egress resolution and rewrite

```
TX (Egress):
   traces:
       - # 1

        hierarchical view:
        #
        # code: ethernet_acl_force_punt
        # destination_sp: HundredGigE1/0/2
        # source: outbound_mirror
        #

            ###[ Ethernet ]###
              dst   = 6c:29:d2:9d:36:ee
              src   = 98:a2:c0:7e:35:02
              type      = VLAN
            ###[ 802.1Q ]###
                 prio      = 0
                 id        = 0
                 vlan      = 411
                 type      = IPv4
            ###[ IP ]###
                version   = 4
                ihl       = 5
                tos       = 0x0
                len       = 28
                id        = 1
                flags     =
                frag      = 0
                ttl       = 63
                proto     = icmp
                chksum    = 0xa931
                src       = 192.168.40.141
                dst       = 192.168.40.209
                \options   \
           ###[ ICMP ]###
                   type      = echo-reply
                   code      = 0
                   chksum    = 0xffff
                   id        = 0x0
                   seq       = 0x0
                   unused    = ''
           ###[ Padding ]###
                   load      =
0000000000000000000000000000
```

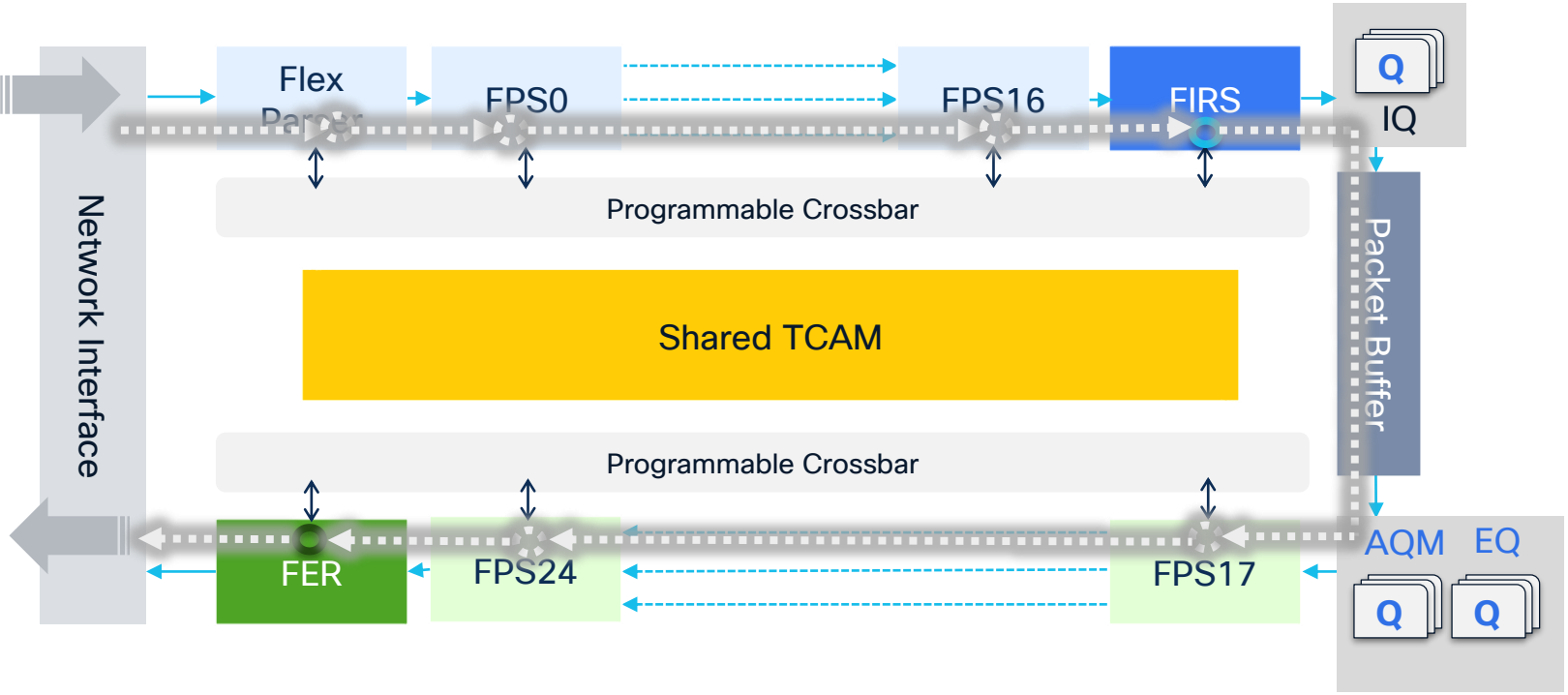

cisco

Packet State Vector

UADP 3.0

Catalyst 9500 High Performance

Catalyst 9600



- Allows for tracing packet flow in UADP 3.0 ASIC
- Uses the live traffic received by the switch
- Captures first packet which matches the capture criteria
- Independent of any feature interaction
- No effect on the switch functionality

Configuring PSV Debug Capture Trace

Define trigger. Enable mode required

```
debug platform hardware fed active capture trigger
[ipv4 | ipv6 <src><dst> [l3 protocol | icmp | igmp | sctp | tcp | tos | udp<src_port><dst_port>]]
[layer2 [ethertype | src_mac | dst_mac]]
[if-id <if_id>ingress | egress ] [interface <ifname>ingress | egress] [vlan <vlan-id>ingress |
egress]
```

Start the capture

```
debug platform hardware fed active capture start
```

Verify status.

```
show platform hardware fed active capture status
Asic: 0 Status: Running

<..packet arrives..>

show platform hardware fed active capture status
Asic: 0 Status: completed
```

Waiting for the first matching packet

Capture triggered & completed

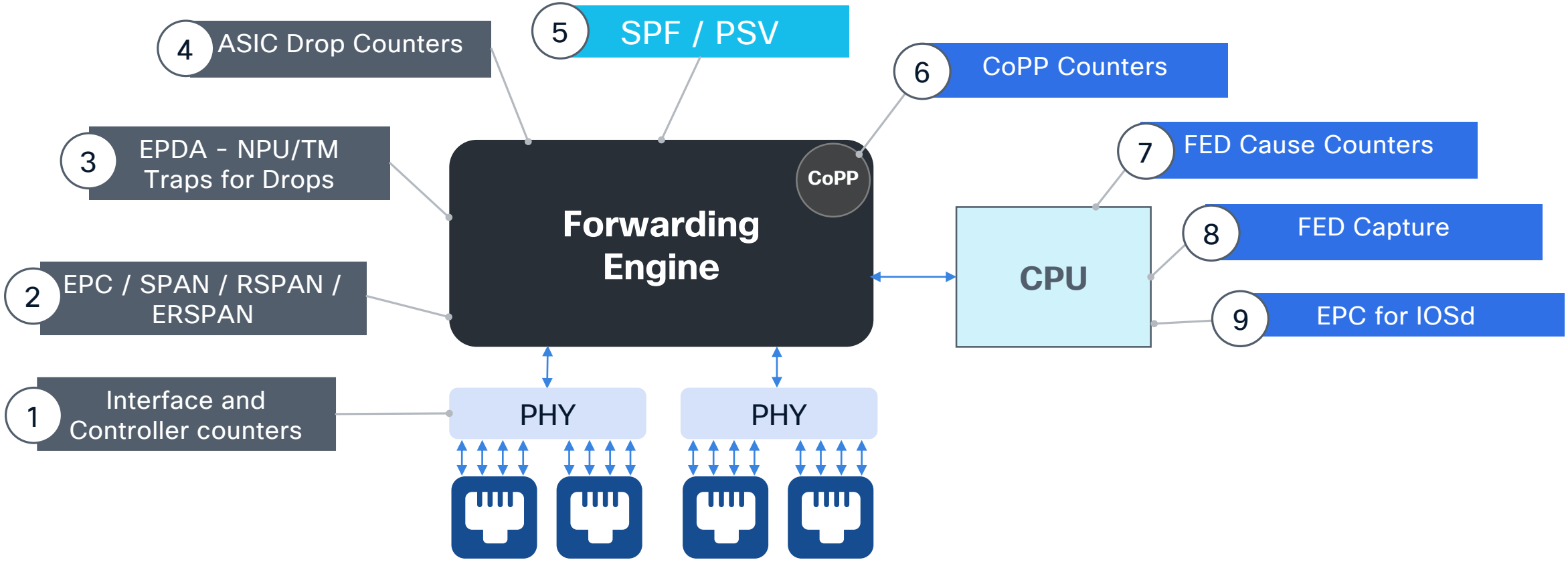
Show captured packets information

```
show platform hardware fed active capture
[summary] | [packet]
[psv [ingressFc | egressFc]] |
[detailed [ingressFc | egressFc]]
```

DEMO

Conclusion

End to End Verification



Hardware Verification | **Software Verification**

What we understand, we control. What we explore, we master.

Vernor Vinge

Complete your session evaluations

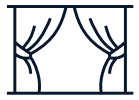
Complete a minimum of 4 session surveys and the Overall Event Survey to be entered in a drawing to win 1 of 5 full conference passes to Cisco Live 2026.

Earn 100 points per survey completed and compete on the Cisco Live Challenge leaderboard.

Level up and earn exclusive prizes!

Complete your surveys in the Cisco Live mobile app.

Continue your education



Visit the Cisco Showcase for related demos

Book your one-on-one Meet the Engineer meeting

Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs

Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

Q&A

Thank you

cisco Live !


CISCO