FOUNDRY®
NETWORKS

Steve Worrall
Systems Engineer

sworrall@foundrynet.com

THE POWER OF PERFORMANCE™

Agenda

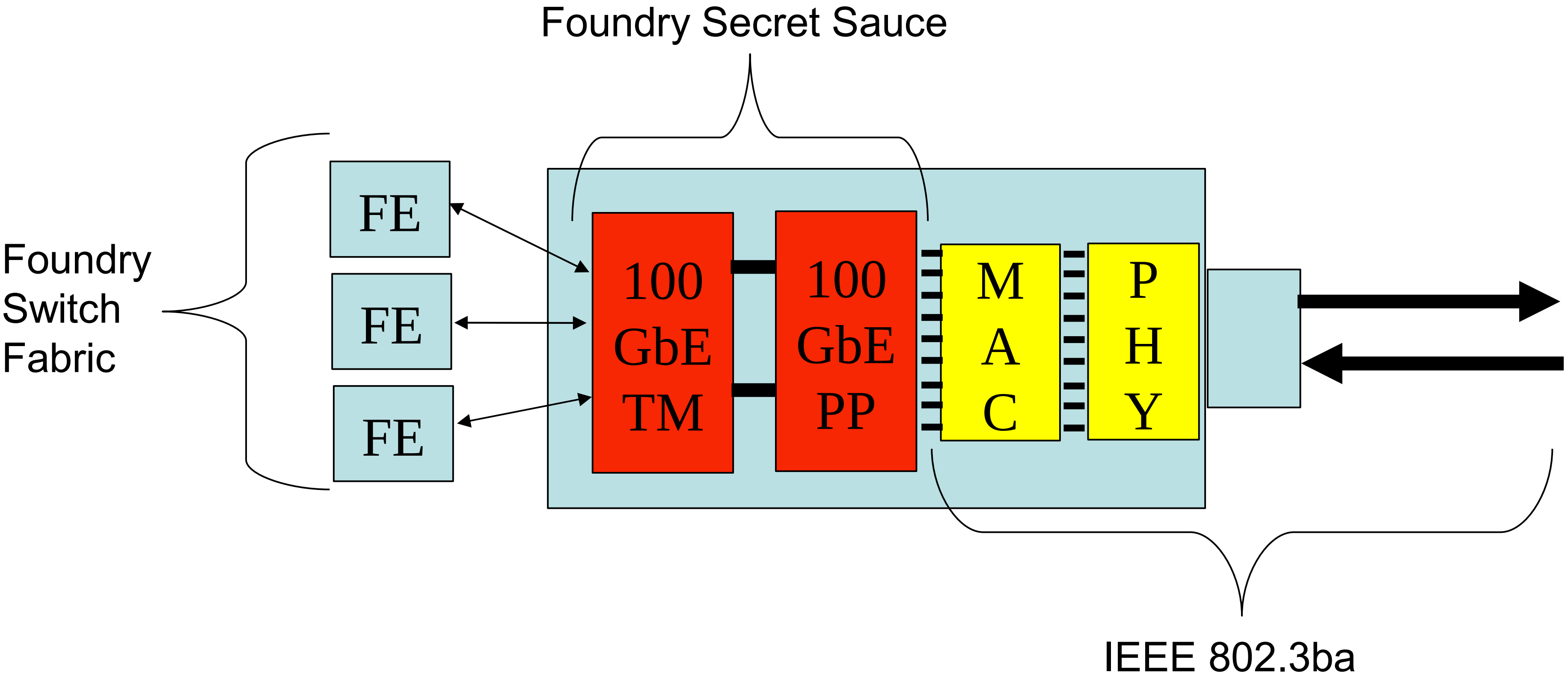
- 100GbE
- Load sharing/link aggregation
- Foundry Direct Routing

100 Gigabit Ethernet

Current Status

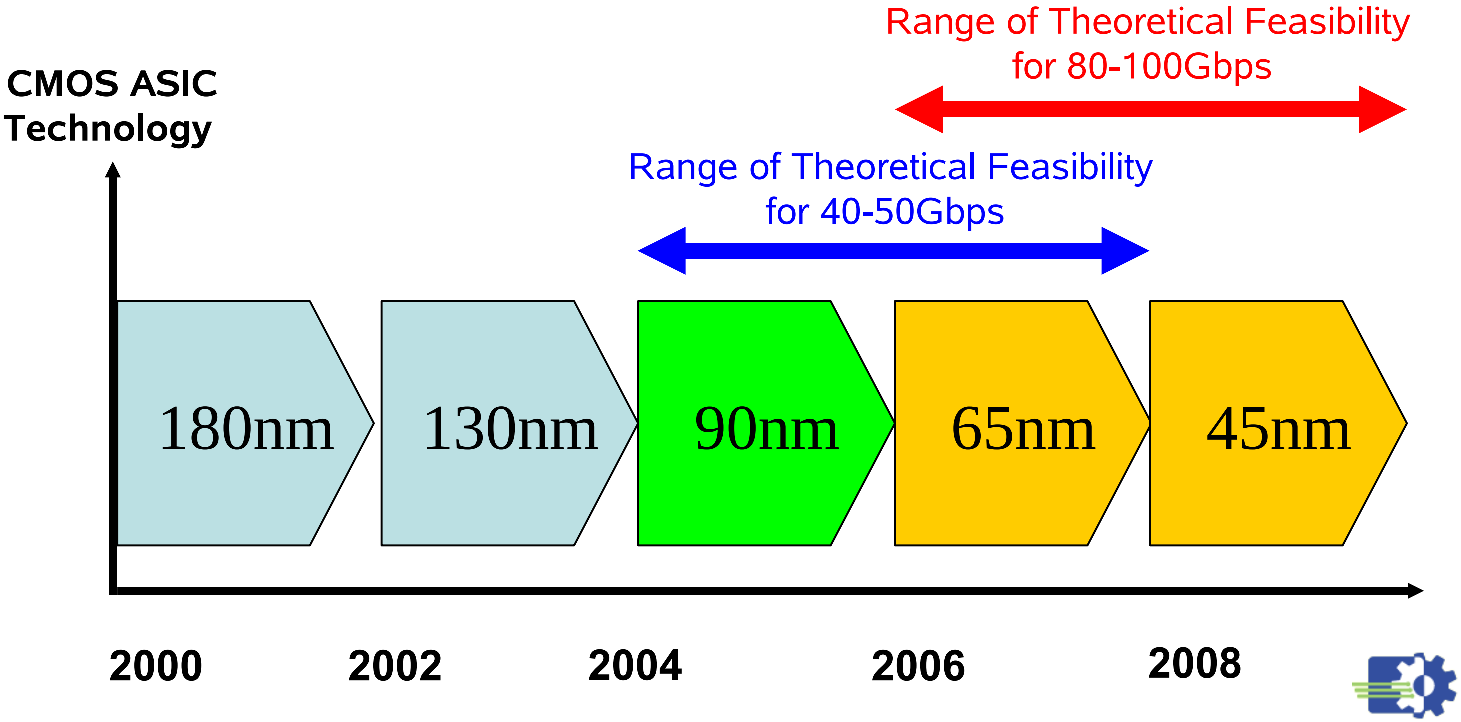
- PAR approved, 802.3ba task force set up
- 40 Gbps support for servers (max range ~10km)
- 100 Gbps support for core switch/router (max range ~40km)
- Target completion date: June 2010
- Initial implementations likely to be based on using multiple wavelengths
  - 4x 25G or 10x 10G
  - 4x 10G

How does 100GbE affect Foundry products



Other dependencies

- 45nm-65nm CMOS technology enables higher signaling speeds and reduces power consumption
- Theoretical feasibility does NOT imply immediate practicality
  - Large number of components and complex interconnects may conflict with PCB design goals
  - Technology may be very costly – above market expectations


Range of Theoretical Feasibility for 80-100Gbps
CMOS ASIC Technology
Range of Theoretical Feasibility for 40-50Gbps
180nm
130nm
90nm
65nm
45nm
2000
2002
2004
2006
2008

Foundry’s Role In Facilitating Migration to 100Gig

- Awareness building on the need for higher speed Ethernet
  - “10G and Beyond” Seminar held in May 2006 at Interop 2006 in Las Vegas, USA
  - “High Performance & High Availability Switching for IXPs”, May 2006, Euro-IX, Dublin, Ireland
- Founding Member of Ethernet Alliance
- Supported formation of High-Speed Study Group (HSSG) at the IEEE Plenary Session, July 2006
- Active participant in IEEE HSSG (802.3) and 802.1 Working Groups

Uniqueness of Foundry’s 100-Gig Approach

- BigIron RX is already delivering 96 Gbps full-duplex per full slot today with wire-speed performance on all ports!
  - In production at numerous sites **TODAY**!
- In terms of capacity per slot, what is important is the usable capacity per slot and not an artificial number internal to the system
  - Foundry’s computations on 100 Gbps readiness on 5th generation platforms assume redundancy is needed and overprovision capacity to easily accommodate internal overheads!

In the interim…

Load sharing and Link Aggregation

Load sharing and Link Aggregation

- Protocols: Determine multiple paths for ECMP
  - Routing Protocols: IGP, BGP
    - Provide path diversity
- Trunks: Offer multiple links for load-sharing
  - Link Aggregation/bundling
    - Provide link diversity
- Data Forwarding: Decision on how packets are load-shared
  - Per packet based
  - Flow based
    - Algorithm
    - Fields in the packet used for load balancing

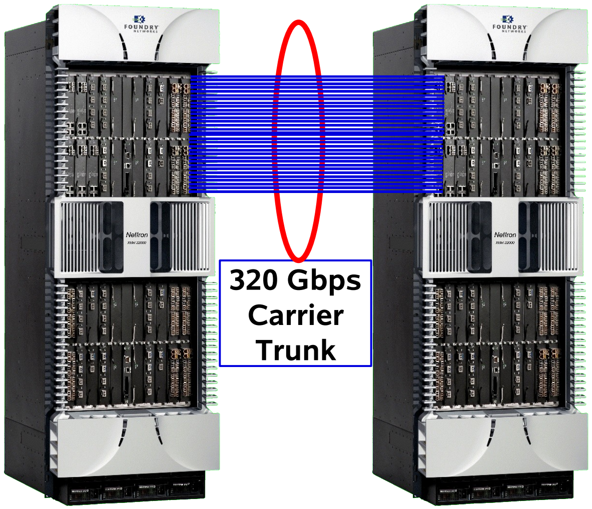
Routing Protocols ECMP

- Routing Protocols determine multiple equal cost paths to a destination
  - IGP (ISIS/OSPF) ECMP:
    - Affects paths taken by IP traffic
    - Affects paths taken by MPLS LSPs
  - BGP ECMP:
    - Affects paths taken by IP traffic
    - Affects paths taken by IP-VPN traffic

- XMR/MLX support 8-path ECMP calculation
  - Routing protocols will calculate 8 different paths per prefix
  - Each of these paths can contain trunks
  - Even distribution for any number of paths, whether 2^n (2, 4, 8) or non-2^n (3,5,6,7) number of ECMP paths
    - Through support of ECMP load balancing modulo operation

Foundry leads the way in link aggregation

- XMR/MLX support static and dynamic (LACP) trunks
- Even distribution for any number of trunks, whether 2^n (2, 4, 8, 16, 32) or non-2^n (3,5,6,7,9…) number of trunks
  - Through support of Trunks load balancing modulo operation
- XMR/MLX supports “**Carrier Trunks**”
  - 32 ports in a Trunk Group for an **unprecedented trunk capacity of 320 Gbps**

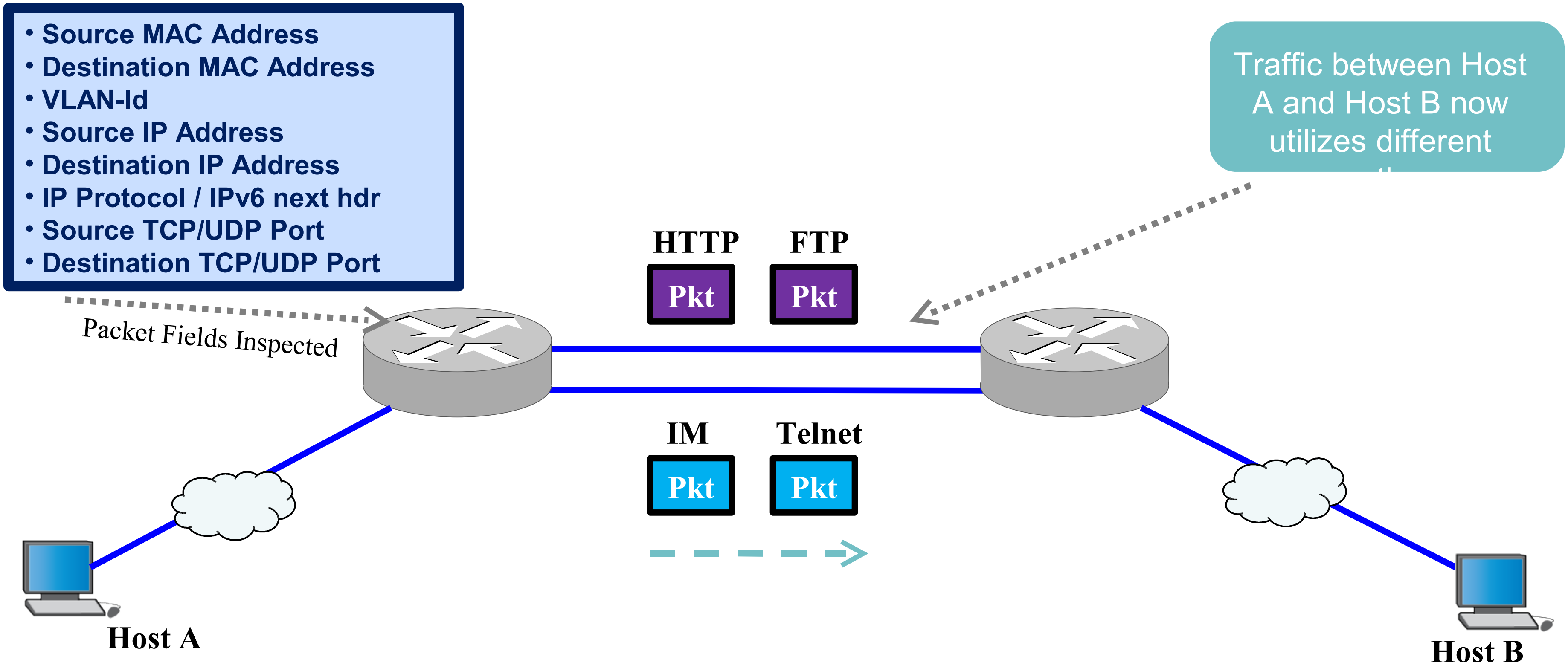


Configured number of ports/ trunk	Number of trunks/ system
32	64
16	128
8	256

Load Sharing for L3 Flows (1)

IPv4/IPv6

- XMR/MLX intelligently look at L2, L3 and L4 information for IP traffic
  - Better traffic distribution for applications between 2 hosts



- Source MAC Address
- Destination MAC Address
- VLAN-Id
- Source IP Address
- Destination IP Address
- IP Protocol / IPv6 next hdr
- Source TCP/UDP Port
- Destination TCP/UDP Port

Packet Fields Inspected

Traffic between Host A and Host B now utilizes different

Load Sharing for L3 Flows (2)

Layer 4 usage options

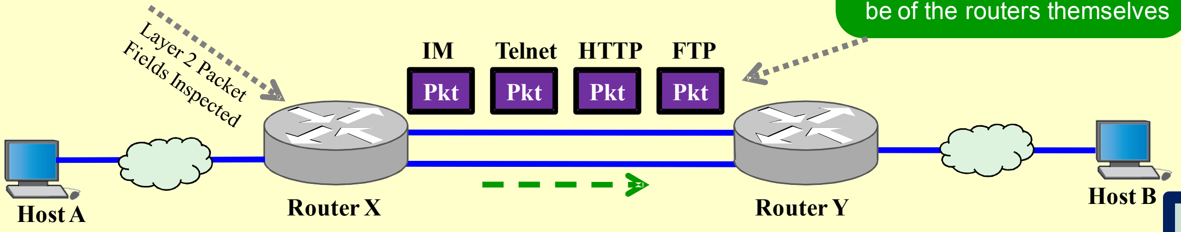
- Option to disable usage of TCP/UDP ports in hash calculations
  - If payload is fragmented in IP packets, only the first IP packet carries the L4 information which may lead to packet ordering issues
  - Recommendation:
    - Set TCP segment size lower than the IP MTU to avoid fragmentation
    - Or load balance packets using L2/L3 information only

Load Sharing for Switched Flows

- XMR/MLX can load share switched flows based on L2 header

- Source MAC Address
- Destination MAC Address
- Vlan-Id, Inner Vlan-Id
- Etype

- However, consider if the frame contains an IP packet



IP Traffic between Router X and Router Y will take the same path since MAC addresses in the packets will be of the routers themselves

- XMR/MLX determine IPv4/v6 packets in L2 flows for better distribution:
  - Load shares IPv4/v6 packets in L2 flows using “L2/L3/L4” headers
  - Load shares non-IP packets in L2 flows using “L2” header

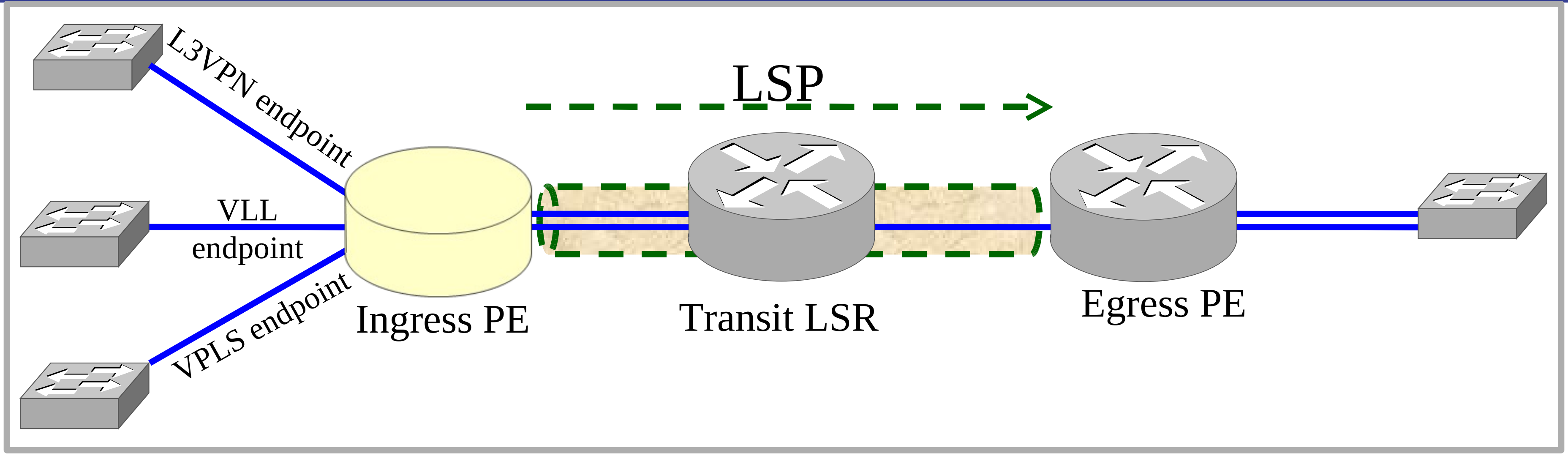
- Source MAC Address
- Destination MAC Address
- VLAN-Id
- Source IP Address
- Destination IP Address
- IP Protocol / IPv6 next hdr
- Source TCP/UDP Port
- Destination TCP/UDP Port

- XMR/MLX can determine MPLS packets in switched flows for better distribution:
  - Will load share MPLS packets in L2 flows using “L2 header, up to 3 labels”
  - Can speculate on content

- Source MAC Address
- Destination MAC Address
- VLAN-Id, Inner Vlan-Id
- Etype
- Label0
- Label1
- Label2

Load Sharing on MPLS PE

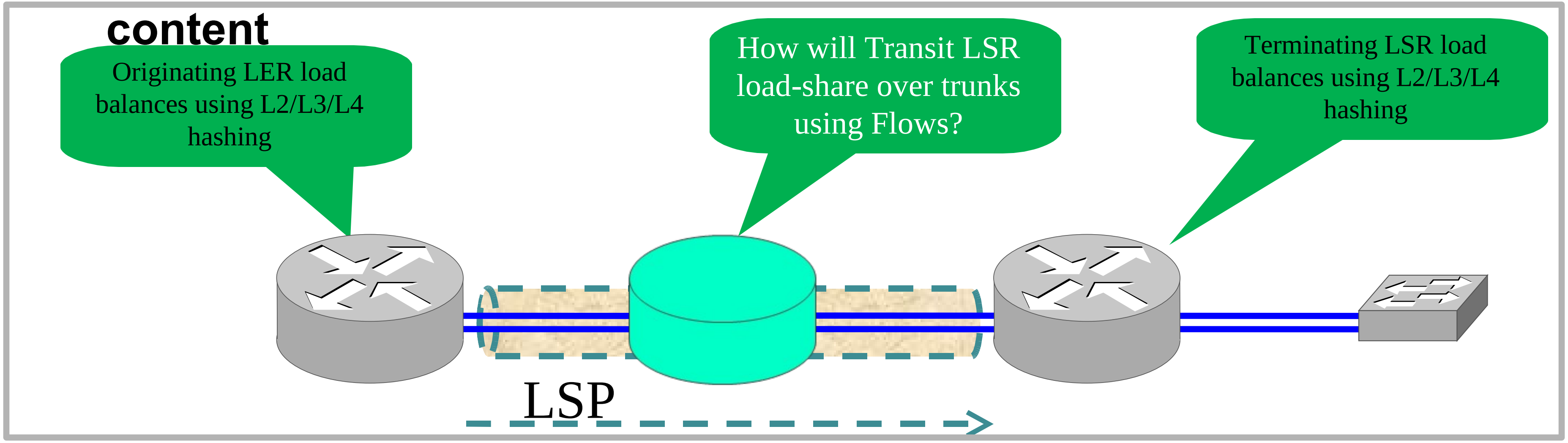
Ingress PE



- At Ingress PE (packets entering a MPLS LSP):
  - Load shares IP packets (IP/MPLS, L3VPN, IPv4/v6 in VPLS/VLL) using “L2/L3/L4” headers
    - Src Mac, Dst Mac, Vlan-Id, Src IP, Dst IP, IP Protocol / IPv6 next hdr, Src TCP/UDP Port, Dst TCP/UDP Port
  - Load shares non-IP packets (in VPLS/VLL) using “L2” headers
    - Src Mac, Dst Mac, Vlan-Id, Inner Vlan-Id, Etype

Load Sharing on MPLS LSRs
Packet Speculation (Transit LSRs, PHP LSRs)

- Transit LSRs (and PHP nodes) cannot normally hash based on packet payload since they have no information on packet content



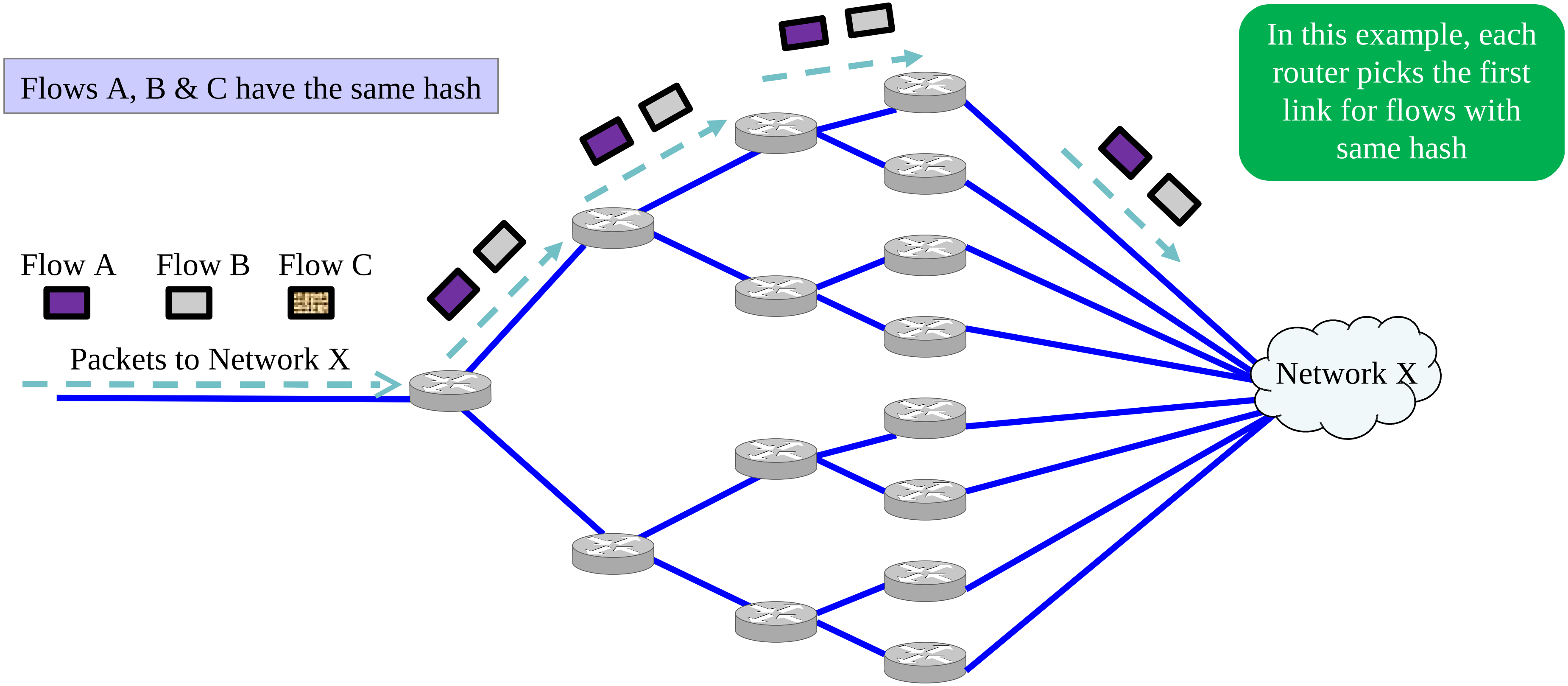
Originating LER load balances using L2/L3/L4 hashing
How will Transit LSR load-share over trunks using Flows?
Terminating LSR load balances using L2/L3/L4 hashing
LSP

- XMR/MLX as transit LSR speculates on the packet type
  - Checks first nibble after bottommost label
    - If 4/6, speculates on packet as IPv4/IPv6 (“speculate-mpls-ip” parameter)
      - Load shares using “MPLS link L2/LSP Label/VC label/Payload(L3/L4)” headers
    - Else speculates on packet as Ethernet (“speculate-mpls-enet” parameter)
      - Load shares using “MPLS link L2/LSP Label/VC label/Payload(L2/L3)” headers
    - Else Load shares using “MPLS link L2/Label1/Label2/Label3”

Problem with Hash based forwarding

Polarization Effect

- In a multi-stage network, similar routers pick the same path for flows with identical hash
  - – May lead to over-utilization of some parts of the network

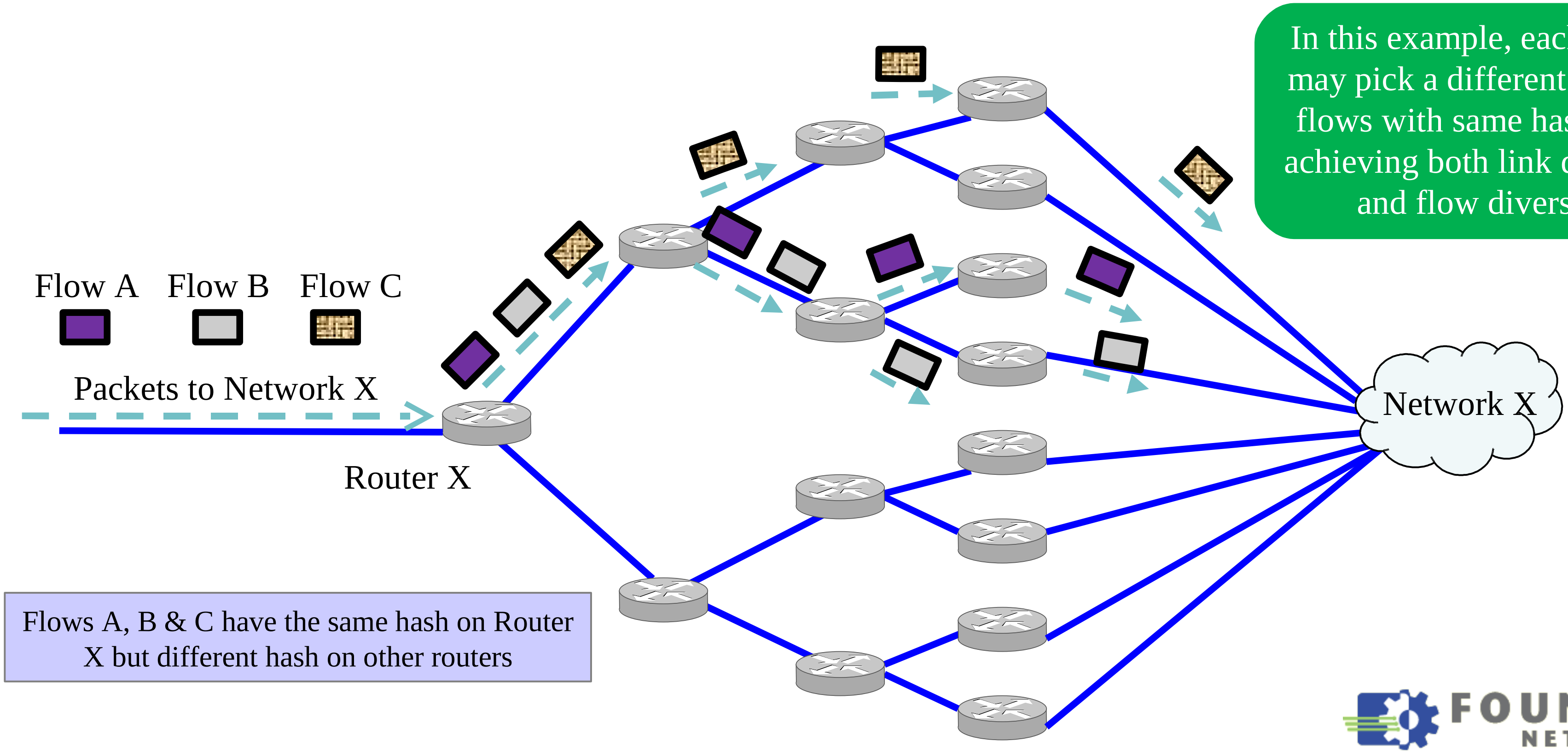


In this example, each router picks the first link for flows with same hash

Neutralizing Hash Polarization effect

Hash Diversification

- Routers in each stage of the network run a different variant of the hash algorithm (for both ECMP and trunking) and neutralize polarization effect
  - Add additional parameter into hash algorithm
  - Flows are now distributed



In this example, each router may pick a different link for flows with same hash, thus achieving both link diversity and flow diversity

Flows A, B & C have the same hash on Router X but different hash on other routers

Summary

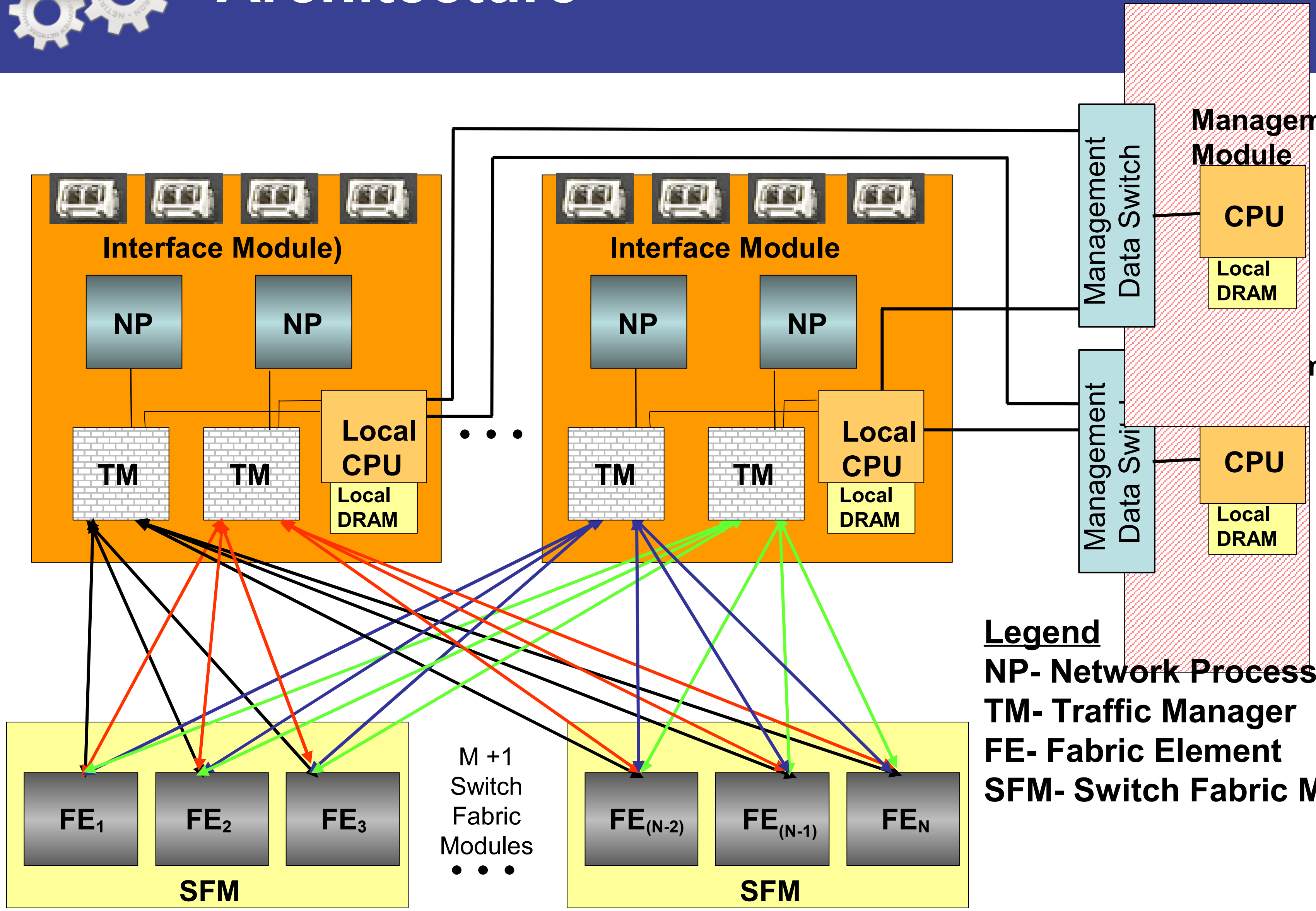
- Load-Sharing is a cost-effective technique to improve network utilization
  - Works over multiple paths and links
- Multiple methods to boost capacity at various layers
  - Can effectively increase throughput beyond the current limits of physical link capacity
- Flow/Hash based forwarding offers many advantages for efficient utilization of the increased capacity
  - XMR/MLX offer the most controls for load sharing to fit in the most demanding environments
- Not a one size fits all approach
  - Choose optimal XMR/MLX schemes based on traffic types and operator policy

Foundry Direct Routing

Forwarding Traffic

- Foundry has always supported distributed forwarding model
  - More line cards = more forwarding capacity
- Packet processor built on FPGAs
  - Wirespeed throughput
  - Flexibility to support new features
- Separate forwarding and control plane
  - Security
  - No contention for bandwidth between management and data

Architecture



Legend

NP- Network Processor

TM- Traffic Manager

FE- Fabric Element

SFM- Switch Fabric Module

Forwarding Table Management

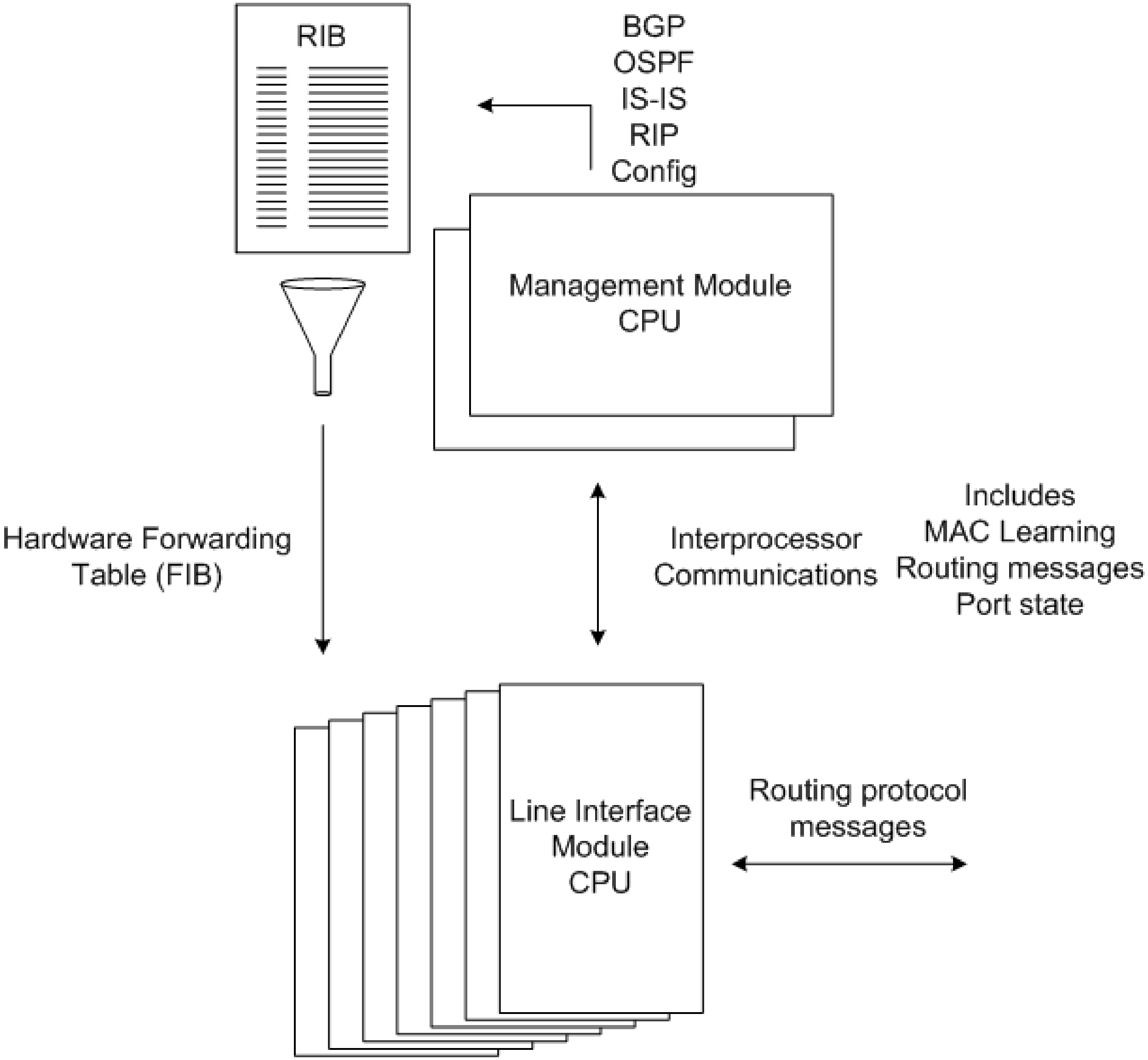


Table Memory

- Routing Information Base
  - XMR supports 10 Million IPv4 BGP paths
  - MLX supports 2 Million IPv4 BGP paths
- Best path is selected and pushed into forwarding information base
- Forwarding information base
  - Implemented in TCAM on line interface card
  - Flexible enough to enable multi-service platform
  - Different profiles allow memory to be carved up to allow the device to be tailored to the specific role

Overview of CAM profiles - XMR

- 16 profiles allow CAM partition sizes to be tuned
- Multi-service router
  - 512k IPv4, 64k IPv6 prefixes
  - 128k MAC/VPLS MAC, 128k IPv4 VPN
- IPv4 router – 1Million IPv4 prefixes
- IPv4/6 router
  - 768k IPv4, 64k IPv6 prefixes