

The Power of the Express Architecture

Brian Petersen

ASIC Architect, Distinguished Engineer

LEGAL DISCLAIMER

This statement of direction sets forth Juniper Networks' current intention and is subject to change at any time without notice. No purchases are contingent upon Juniper Networks delivering any feature or functionality depicted in this presentation.

This presentation contains proprietary roadmap information and should not be discussed or shared without a signed non-disclosure agreement (NDA).

Goals

- Insights into:
 - The nature of pipelines
 - How to add flexibility to a pipeline
 - Headline features of Express
 - Where Express may be deployed



Trio: MX

ExpressionPTX, QFX10K





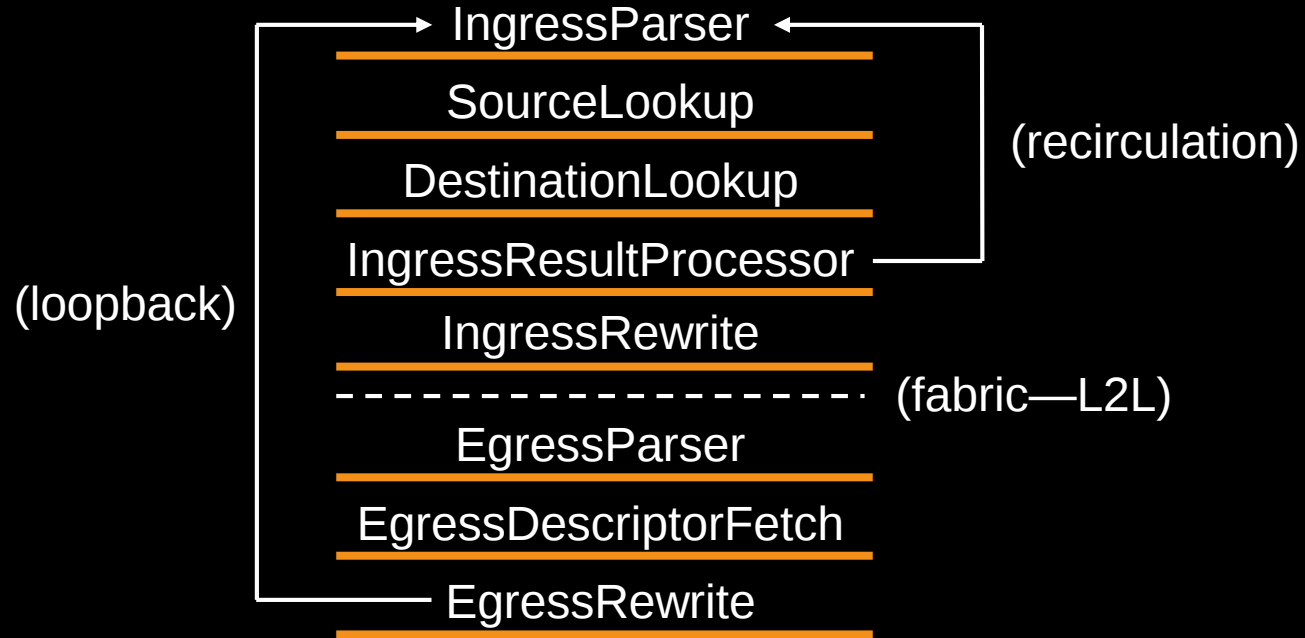






The Express Pipeline

The Express Pipeline



The Express Pipeline

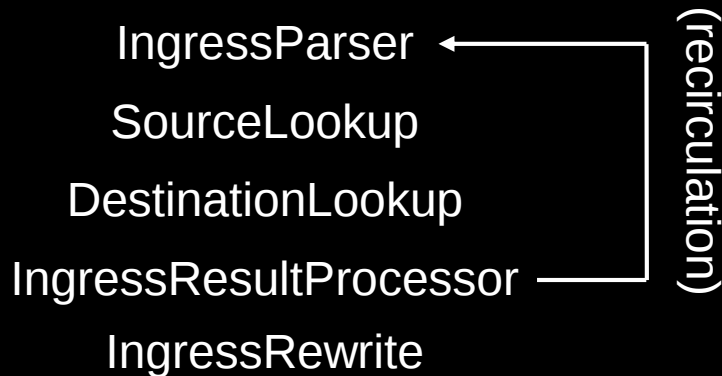




Packet Recirculation

What is Recirculation?

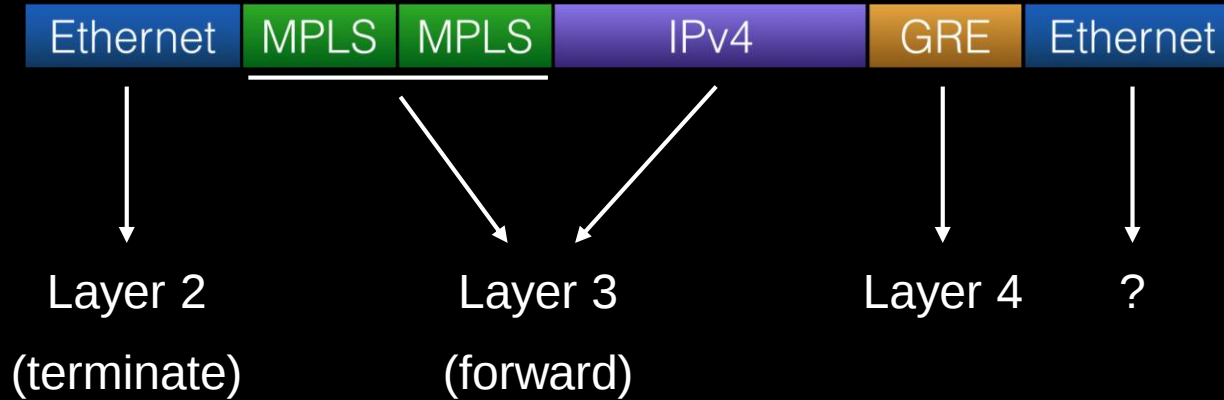
- Returns headers to start
- Processing is repeated
- Outer header is advanced
- Effectively strips outer headers
- This is tunnel termination



Why Recirculate?

- Low overhead
- Header data only
- Processing speedup
- 1 recirculation: 167 -> 230

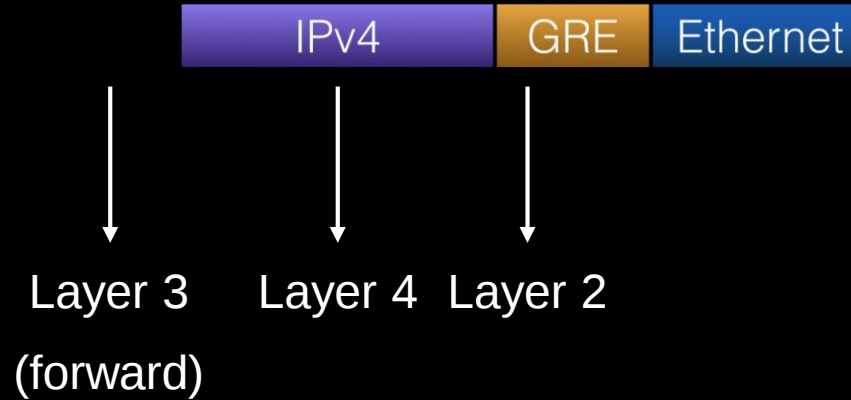
The Power of Recirculation



The Power of Recirculation



The Power of Recirculation





DestinationLookup



Lookup Instructions

- Pre-lookup and post-lookup
- Pre-lookup:
 - table ID
 - search algorithm
 - search argument format
- Post-lookup:
 - hit vs. miss actions
 - sequencing

Table ID

- Distinguishes tables
- Shared key format
- Virtualization

Search Algorithms

- Indexed read
- Exact match
- Longest-prefix match

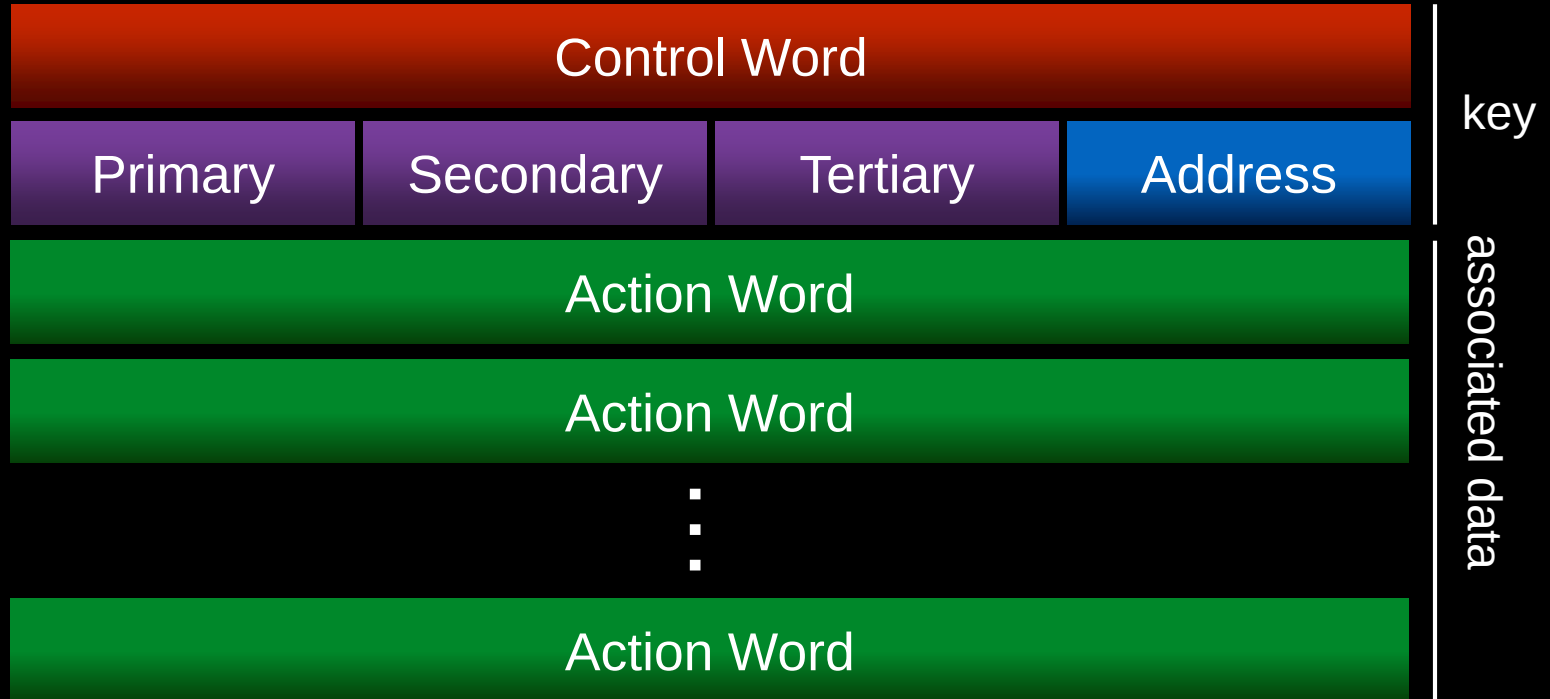
Search Argument Types

- Index selection
 - indexed reads
- Four-part value
 - exact and longest prefix match

Four-Part Search Argument Format



Forwarding Database Entry Format



The Control Word

- Entry size
- Key length
- Lookup sequencing index
- Action word structure

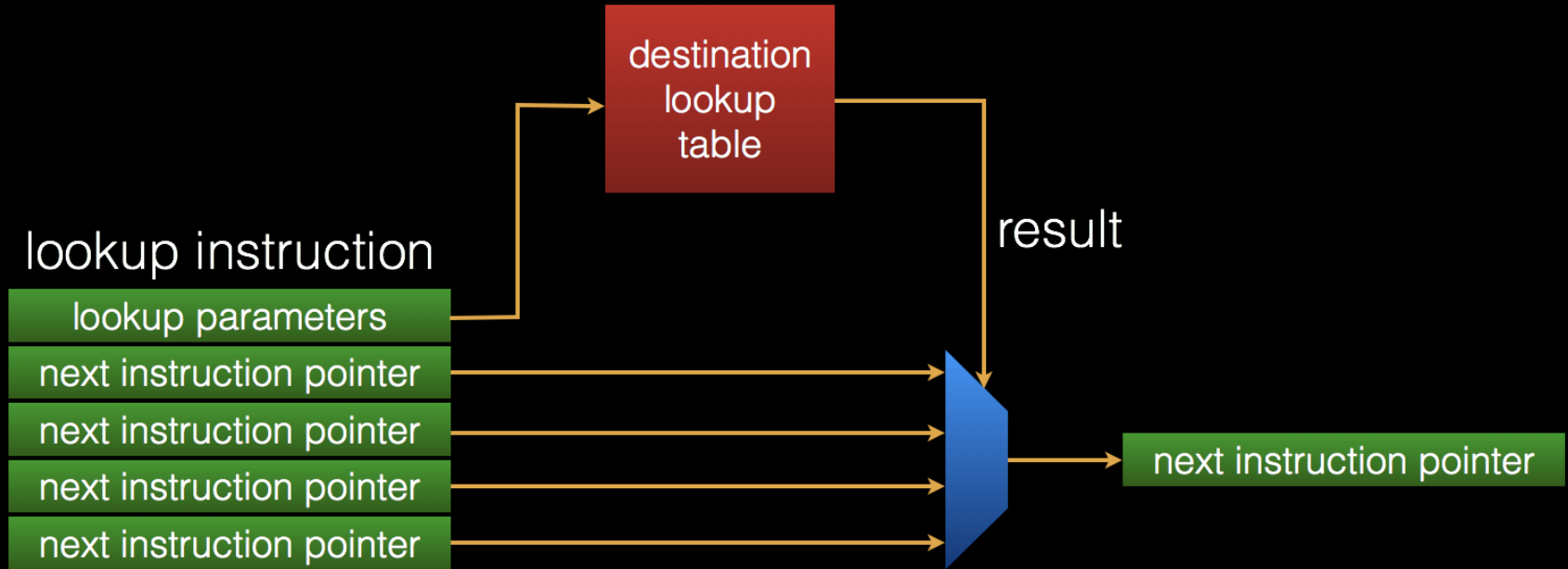
Action Words

- Next-hop instructions
- Effects
- Actions

Actions and Effects

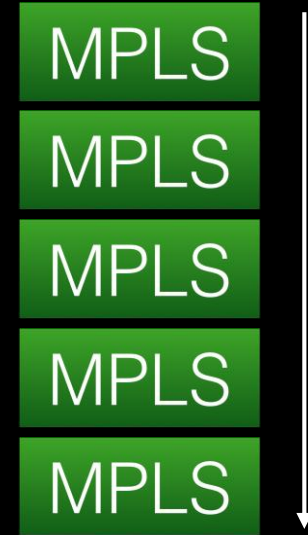
- MPLS handling
- Metadata updates
- IPv4 NAT
- RPF check
- Counter triggers

Lookup Sequencing



MPLS Label Sequencing

- Look up label
- Advance label index
- Update forwarding domain
- Update derived TTL
- Repeat



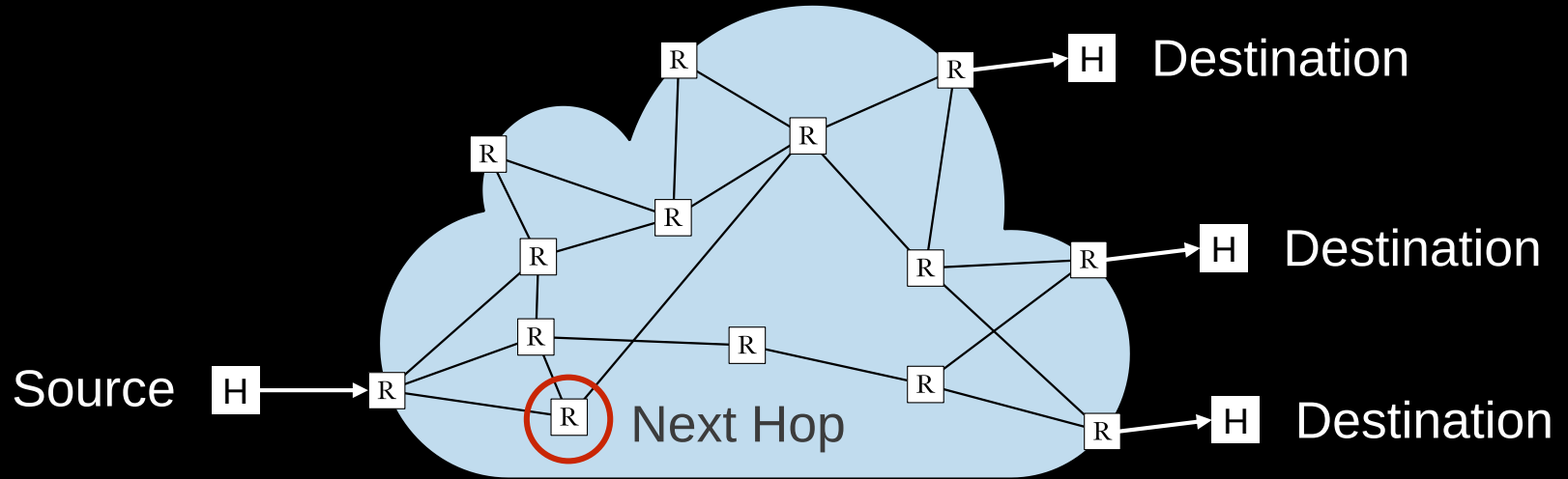
Lookup Sequencing Summarized

- MPLS label stacks
- IP Parent routes
- Routing hierarchies
- Multicast
- Reverse path forwarding check
- Same header, new parameters, subsequent lookup

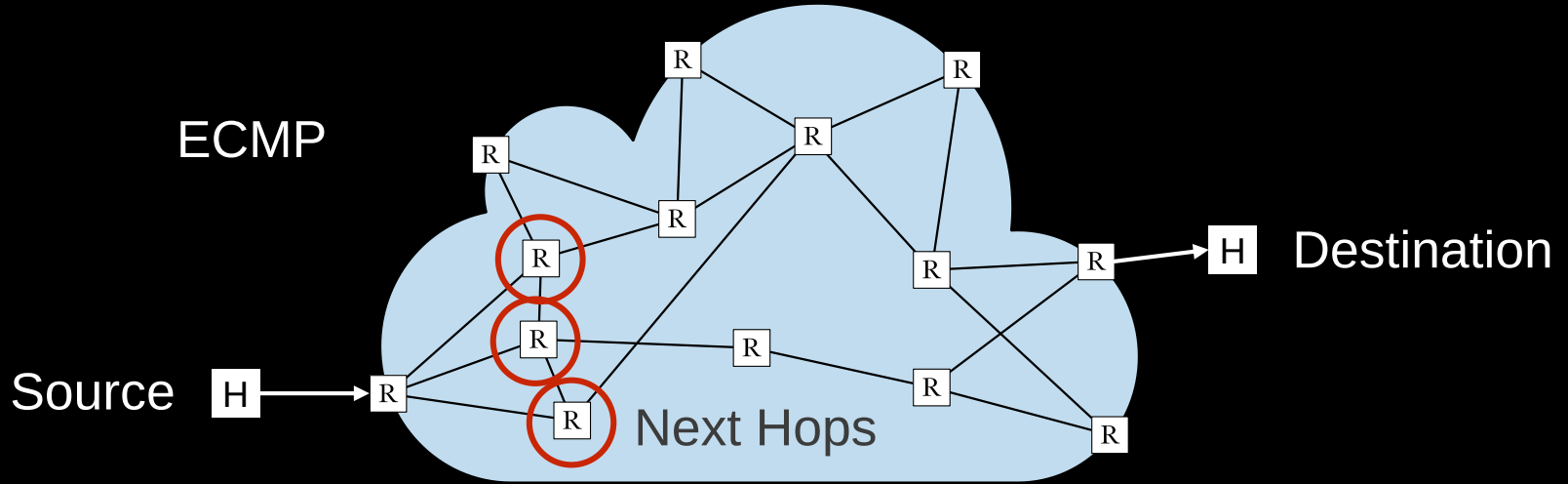
IngressResultProcessor



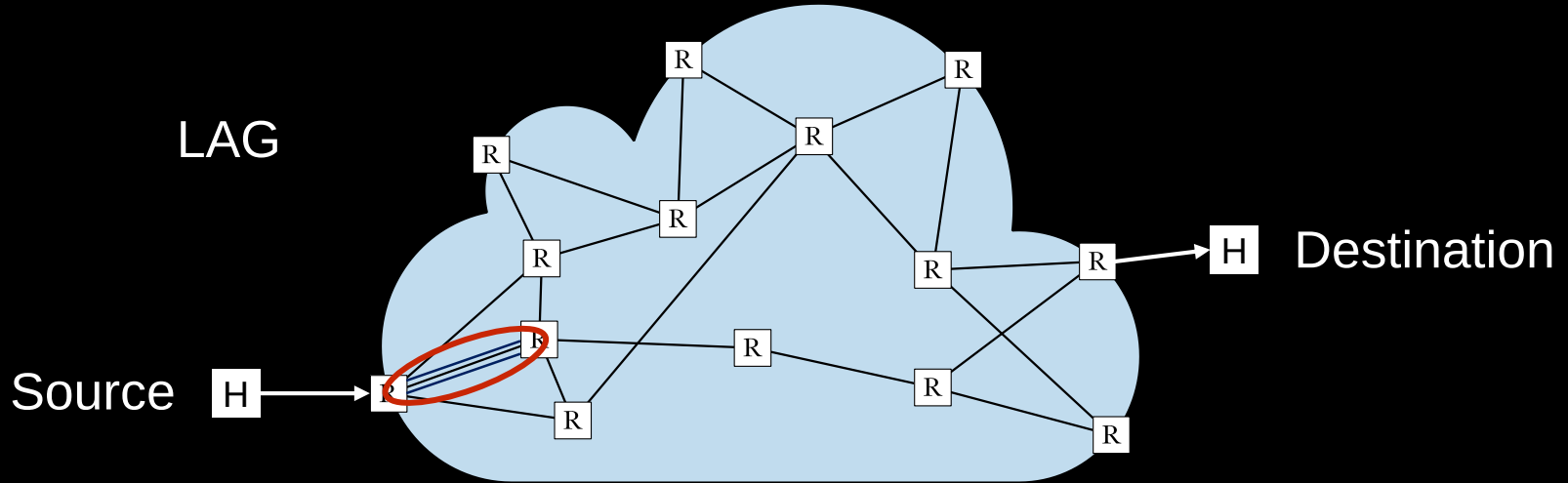
Lookups and Next-Hops



Lookups and Next-Hops

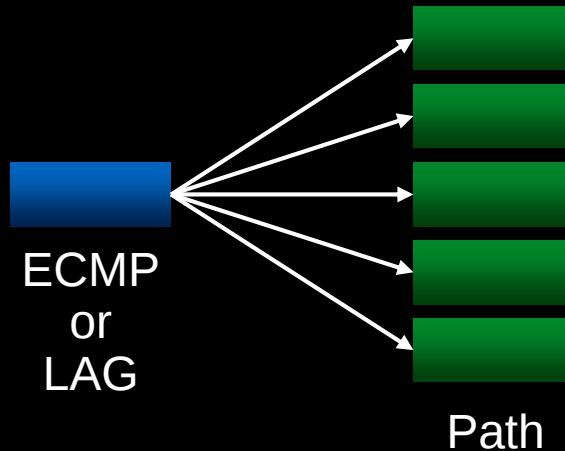


Lookups and Next-Hops



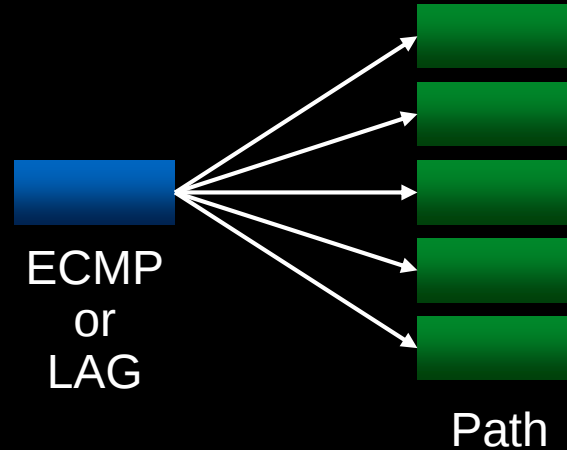
ECMP and LAG Behaviors

- Common infrastructure in IngressResultProcessor
- Multi-way branch
- Like C's **switch** statement
- One-of-n
- Uses hash value



Load Balancing Instructions

- NH_EQ_LIST
- NH_EMBED_EQ_LIST
- NH_IDX_LIST
- NH_WT_LIST
- NH_EMBED_UB_LIST
- NH_TH_LIST



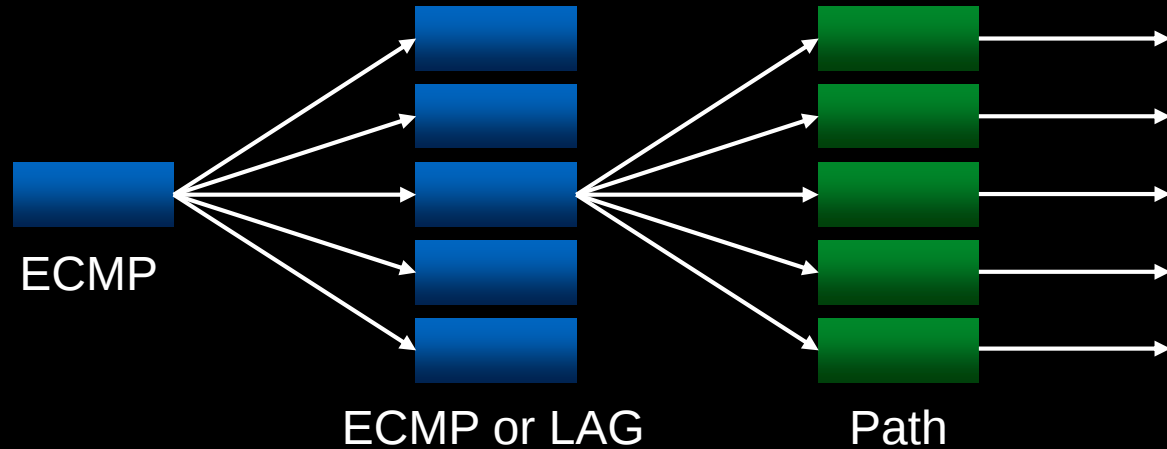
Load Balancing Scale

- NH_EQ_LIST 64
- NH_EMBED_EQ_LIST 8
- NH_IDX_LIST 256
- NH_WT_LIST 256
- NH_EMBED_UB_LIST 8
- NH_TH_LIST 256

Load Balancing Scale

- >800 ports
- Many more logical interfaces
- 256-way load balancing?

Load Balancing Scale



MPLS Label Management

- Uses large DestinationLookup tables
- Up to four labels

Metadata Updates

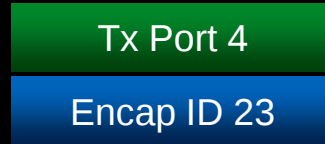
- Forwarding domains
- Traffic class
- Drop precedence
- Control packet class
- Receive contexts
- PacketType
 - simple index
 - updates several values

Transmit Port and Encapsulation ID

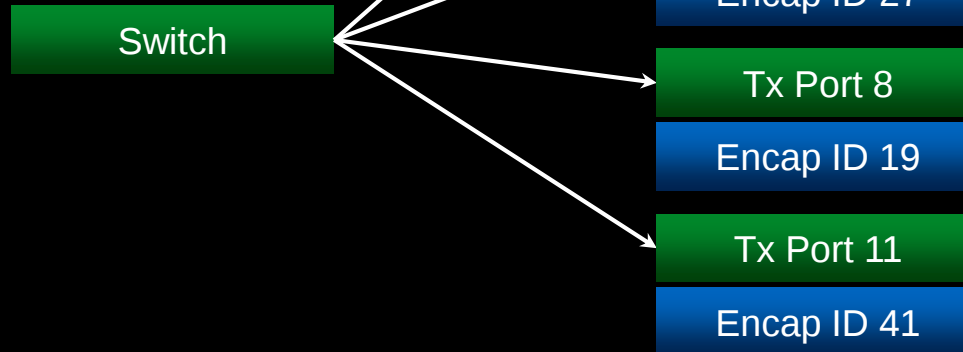
- Global port ID
 - Combined with trafficClass -> VOQ
- Encapsulation ID is abstract
 - Sent across fabric
 - Mapped to instruction pointer

Next Hop Processing Examples

Simple:

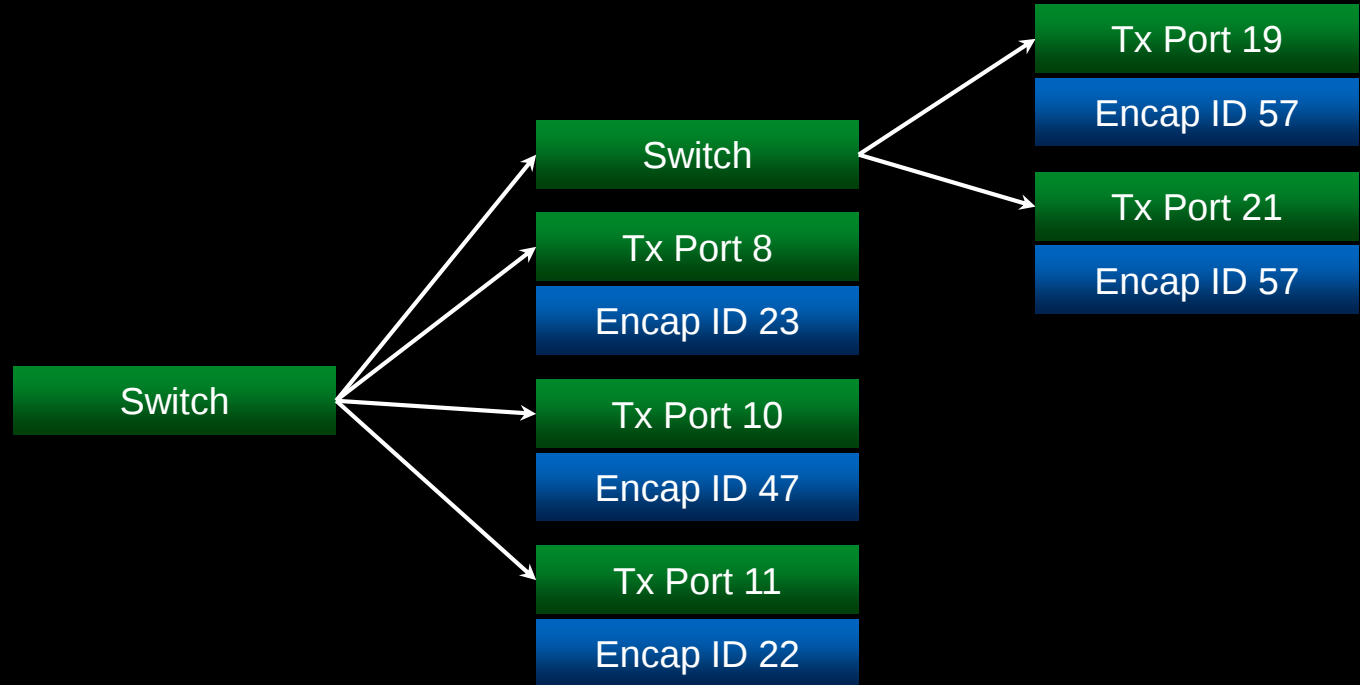


LAG:



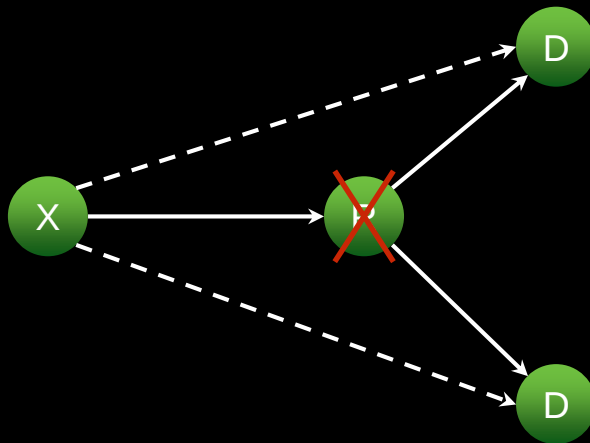
Next Hop Processing Examples

ECMP + LAG:



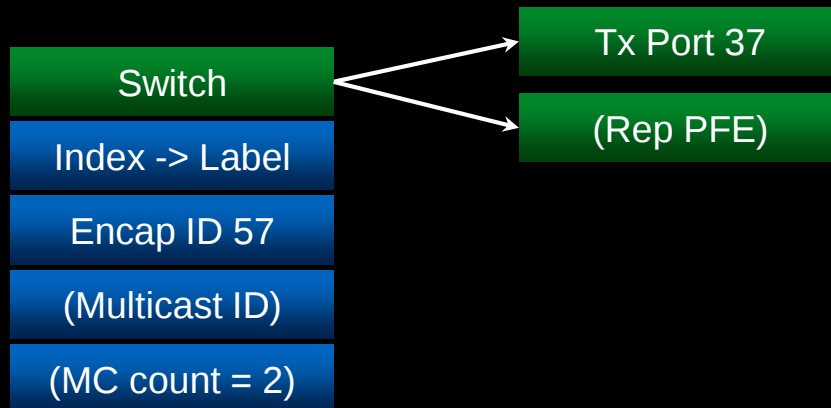
Next Hop Processing Examples

MPLS multicast with node protection:



Next Hop Processing Examples

MPLS multicast with node protection:



Other Functions

- Sample processing
- Recirculation management
- Etc.

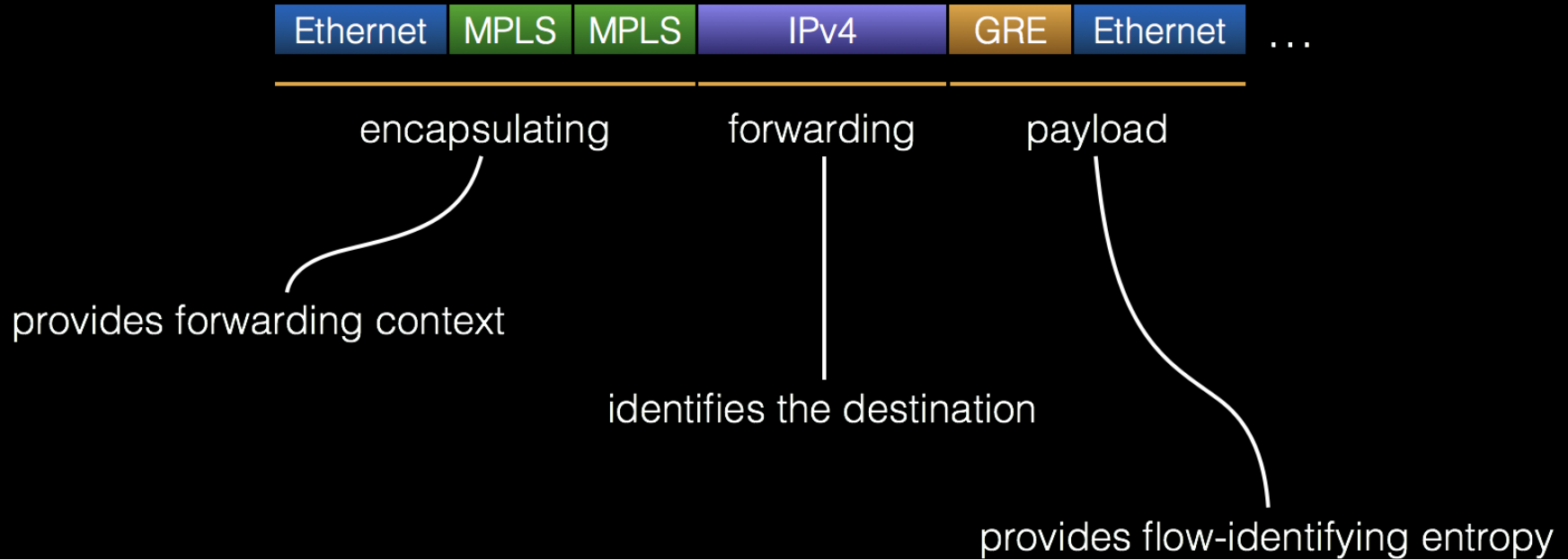
EgressDescriptorFetch



Descriptors

- Processed by EgressDescriptorFetch
- Updates forwarding header
- Builds and appends new headers
- Contains:
 - operators
 - operands
 - continuation pointer (optional)

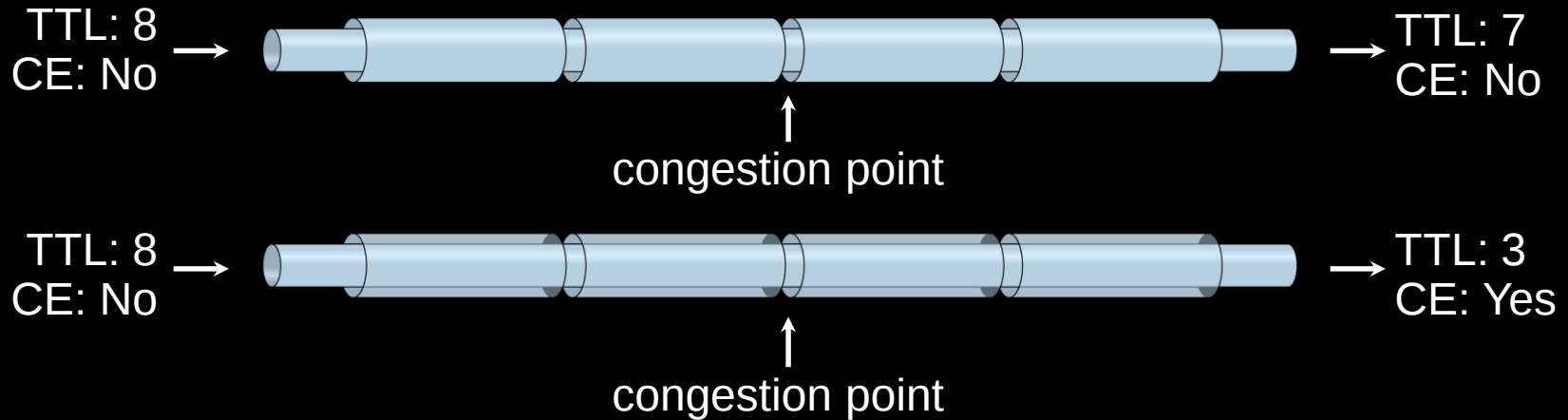
Header Regions



Forwarding Header Updates

- Swap MPLS labels
- VLAN translation
- VLAN tag pop/push
- QoS updates
- TTL decrement
- Update checksum
- QoS and TTL propagation

Opaque vs. Transparent Tunnels



New Encapsulations

- Three encapsulating types:
 - layer 2
 - MPLS
 - tunnel

Header Sequencing

- Tunnel + Layer 2 + MPLS
- Tunnel + MPLS + Layer 2
- Layer 2 + Tunnel + MPLS
- Layer 2 + MPLS + Tunnel

Other Functions

- Counters
- Policers
- IPv4 fragmentation
- More...

Recap



Recirculation

- Powerful tunnel termination
- Advances to next header
- Efficient
- Flexible
- 3:2 speed-up



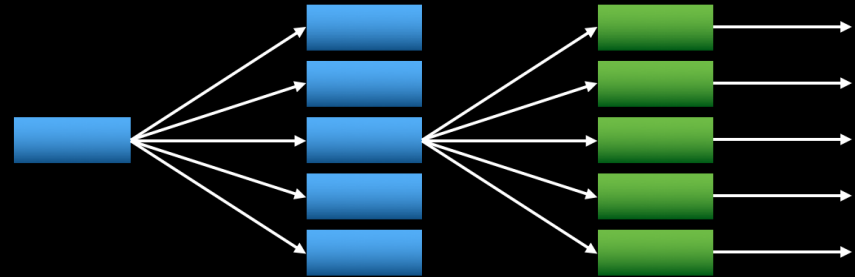
DestinationLookup

- Four-part search argument assembly
- Branching and looping
- Label sequencing
- Expansive associated data
- Destination-specific behavior
- Offloads capacity burden



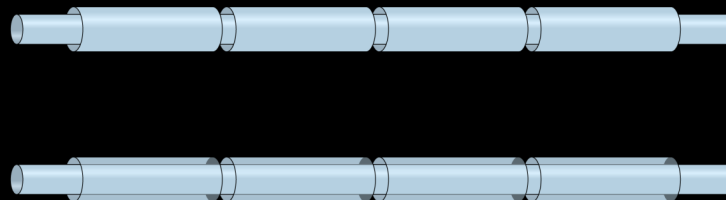
IngressResultProcessor

- Flexible load balancing
 - equal and weighted
 - fine-grained and large-scale
 - hierarchical
- Metadata updates
- Counter & policer operations



EgressDescriptorFetch

- Flexible encapsulations
- Forwarding header updates:
 - TTL, QoS, checksum, etc.
- Three sets of new headers:
 - layer 2
 - MPLS
 - tunnel
 - opaque & transparent



In Summary...

- Recirculation
- DestinationLookup
- IngressResultProcessor
- EgressDescriptorFetch

Thank You