

# SoftFlow

Ethan J. Jackson<sup>†</sup> Melvin Walls<sup>¶†</sup> Aurojit Panda<sup>†</sup> Justin Pettit<sup>\*</sup>

Ben Pfaff<sup>\*</sup> Jarno Rajahalme<sup>\*</sup> Teemu Koponen<sup>‡</sup> Scott Shenker<sup>†\$</sup>

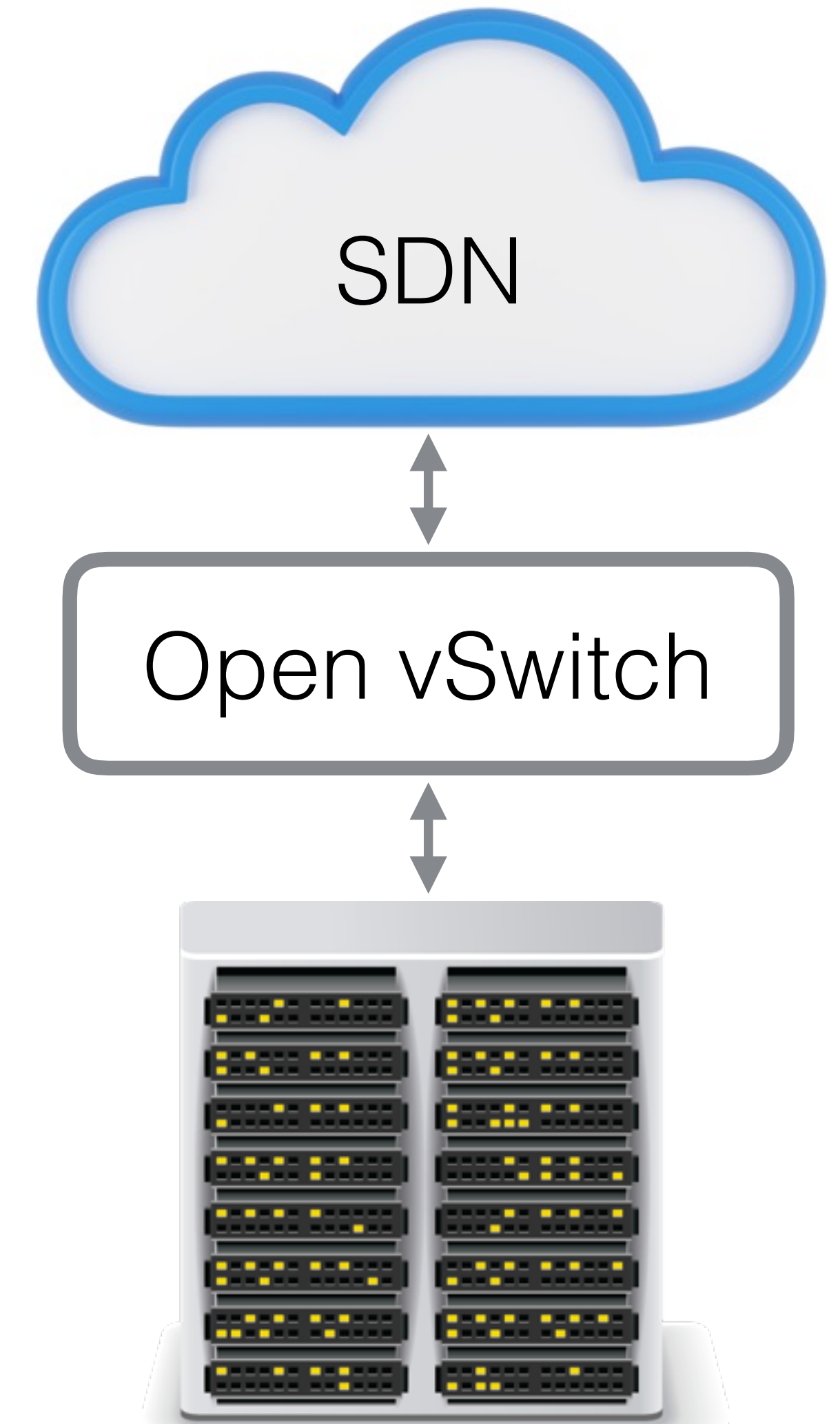
<sup>\*</sup>VMware, Inc. <sup>†</sup>UC Berkeley <sup>‡</sup>Styra, Inc. <sup>\$</sup>ICSI <sup>¶</sup>Penn State Harrisburg

# SoftFlow

- Middleboxes for Open vSwitch
- While maintaining OpenFlow programmability
- And good performance
  - Run to completion
  - Flow caching

# Open vSwitch

- Open source software switch
- Dominant software OpenFlow implementation
- Use cases
  - Hypervisor vSwitch
  - Network virtualization gateway



# OpenFlow

The Open vSwitch forwarding model.

# OpenFlow

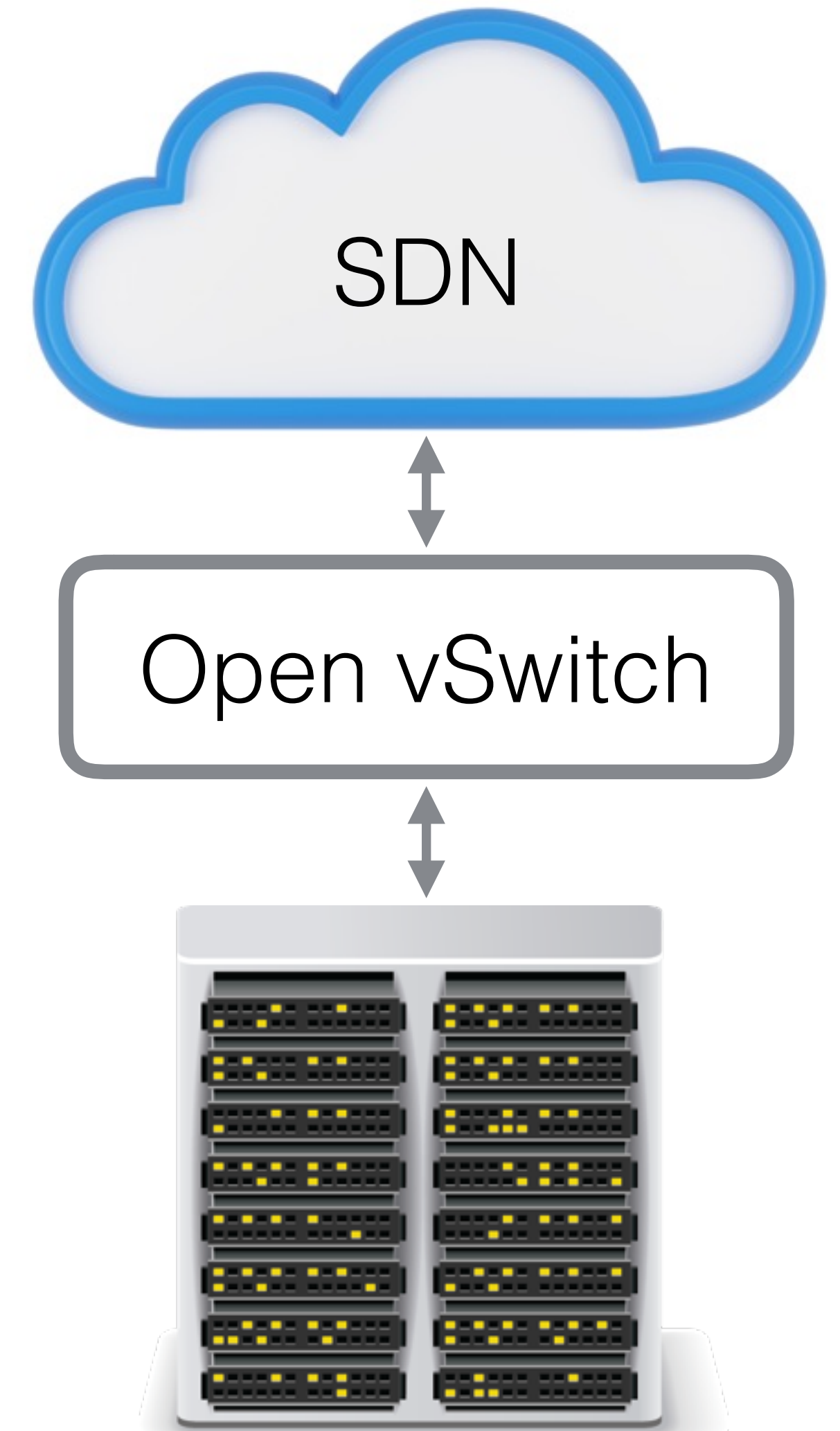
- Great for L2/L3
- Packet Classification
- Flow Caching

# OpenFlow — Limitations

- Stateful processing
- Deep packet inspection
- Header non-determinism
- ***Middleboxes***

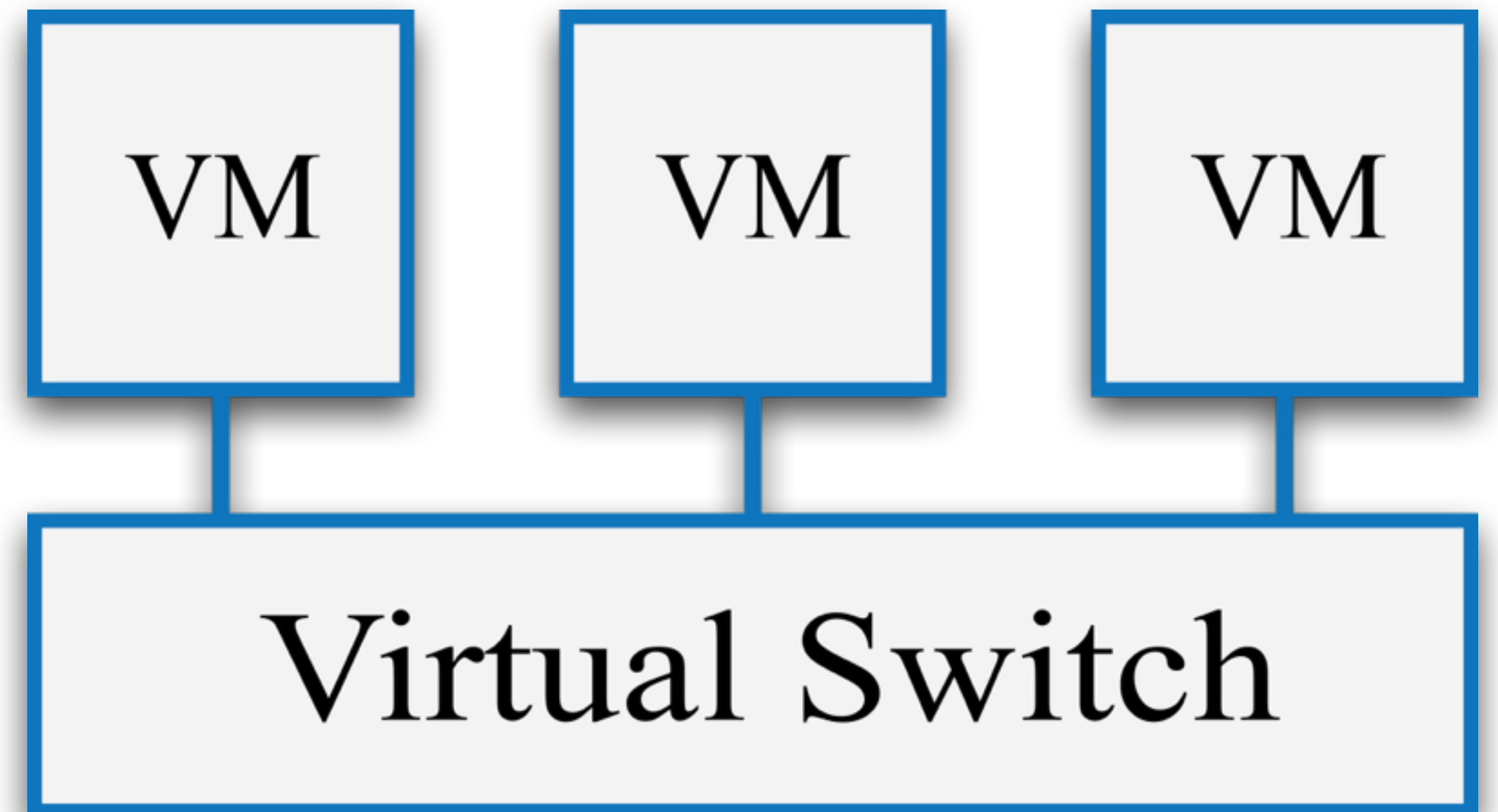
# The Problem

- Wide deployment of OVS SDN gateways
- Customers want new middlebox services
  - Firewall
  - NAT
- OpenFlow is *ill suited* for the task



# What about NFV?

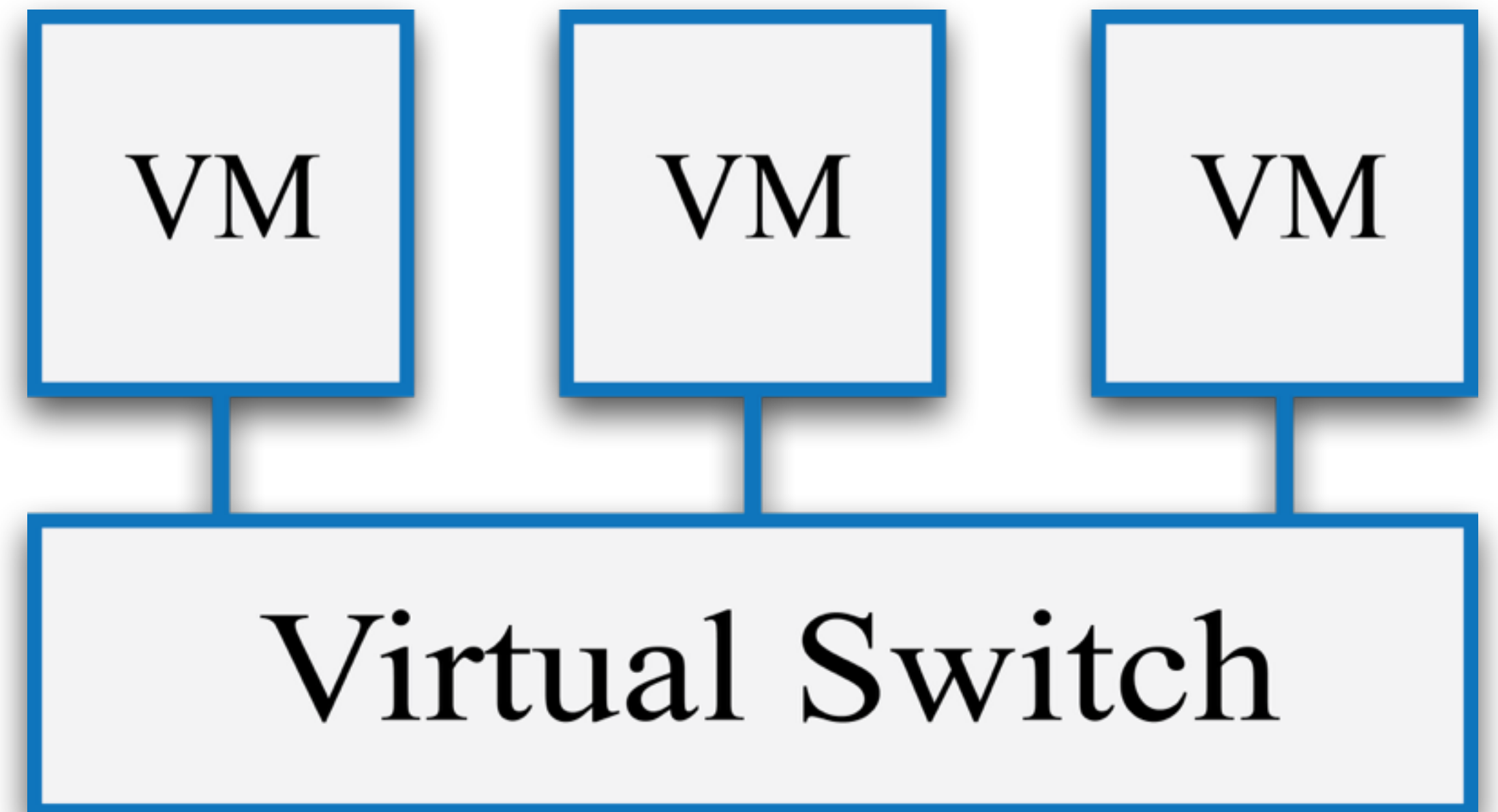
- Service chain of Middleboxes (NFs)
- Highly expressive
- Strong Isolation





# What about NFV?

- Block-box network functions
  - No cross VM flow caching
- Isolation overhead
- Abandons OpenFlow programmability



# SoftFlow

Middleboxes for Open vSwitch

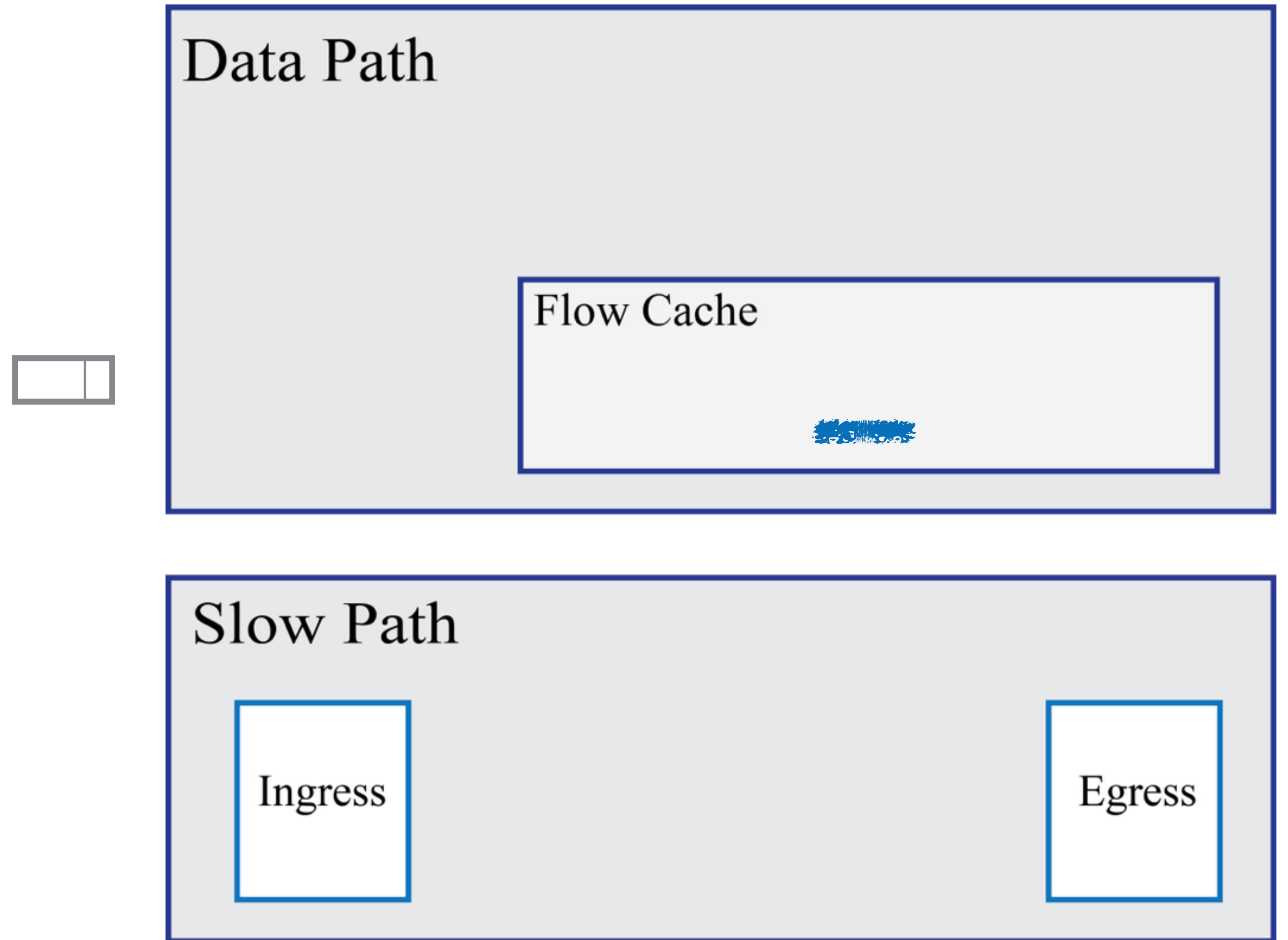
# SoftFlow

- Start with Open vSwitch
  - Use OpenFlow wherever possible
- Middlebox services use SoftFlow Actions
  - Plugin library of stateful processing elements
  - Accessible from OpenFlow

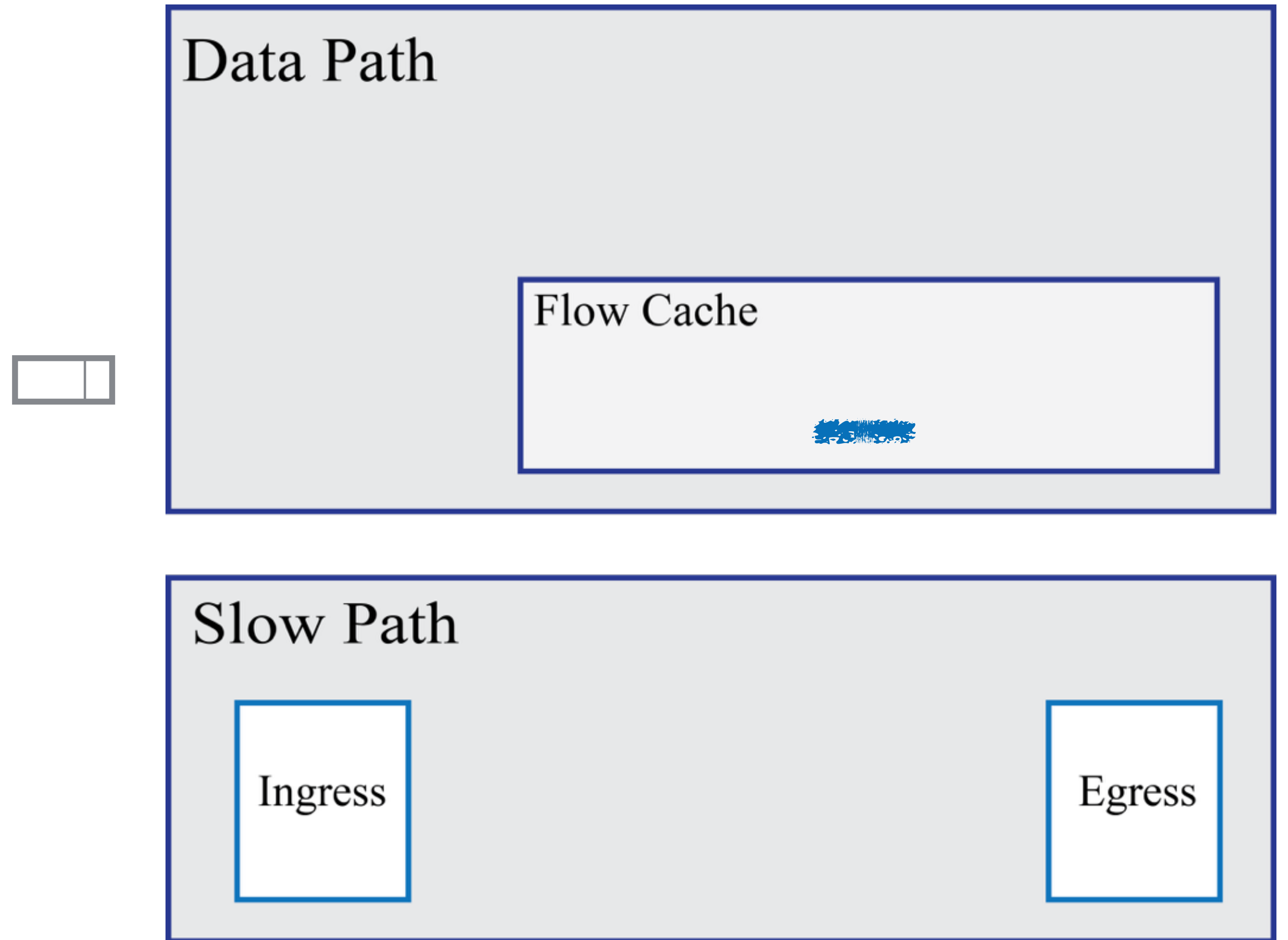
# SoftFlow Firewall

- Access Control List
  - Stateless packet classification
  - Implemented In OpenFlow
- Connection Tracking
  - Stateful processing
  - Implemented as a SoftFlow Action

# Open vSwitch

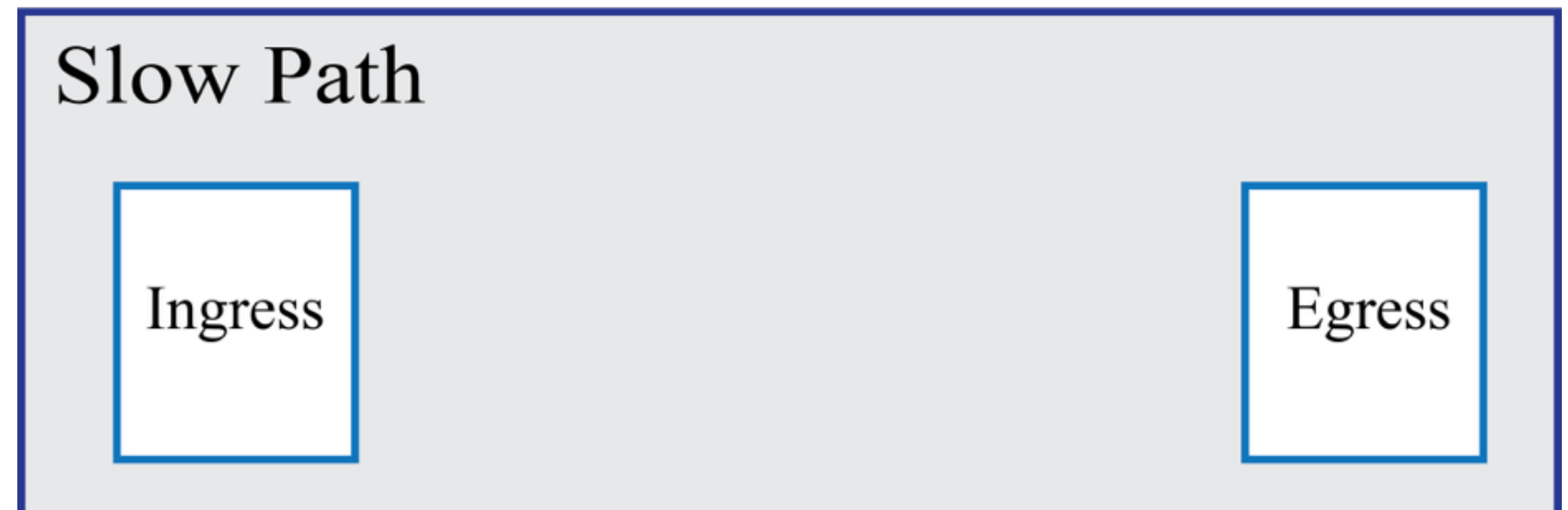
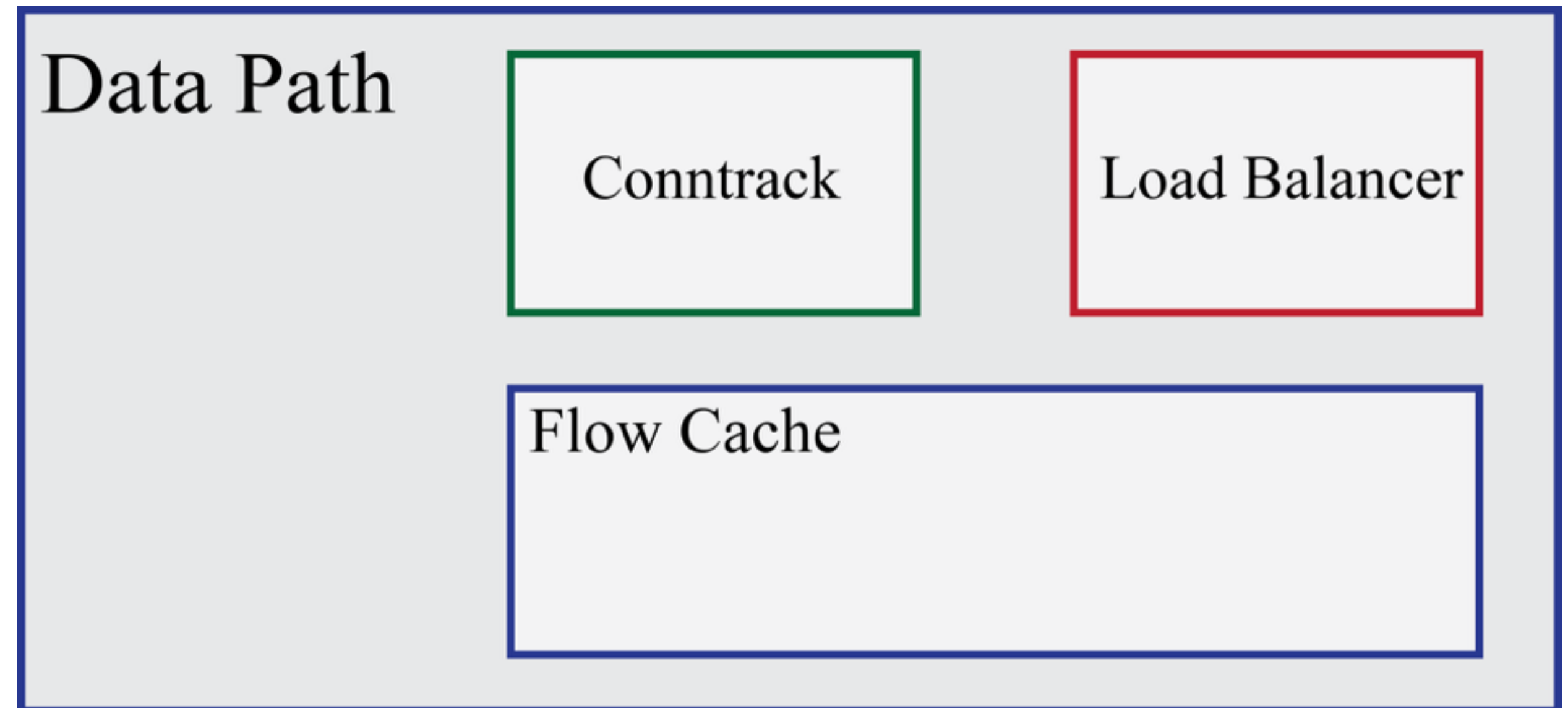


# Open vSwitch

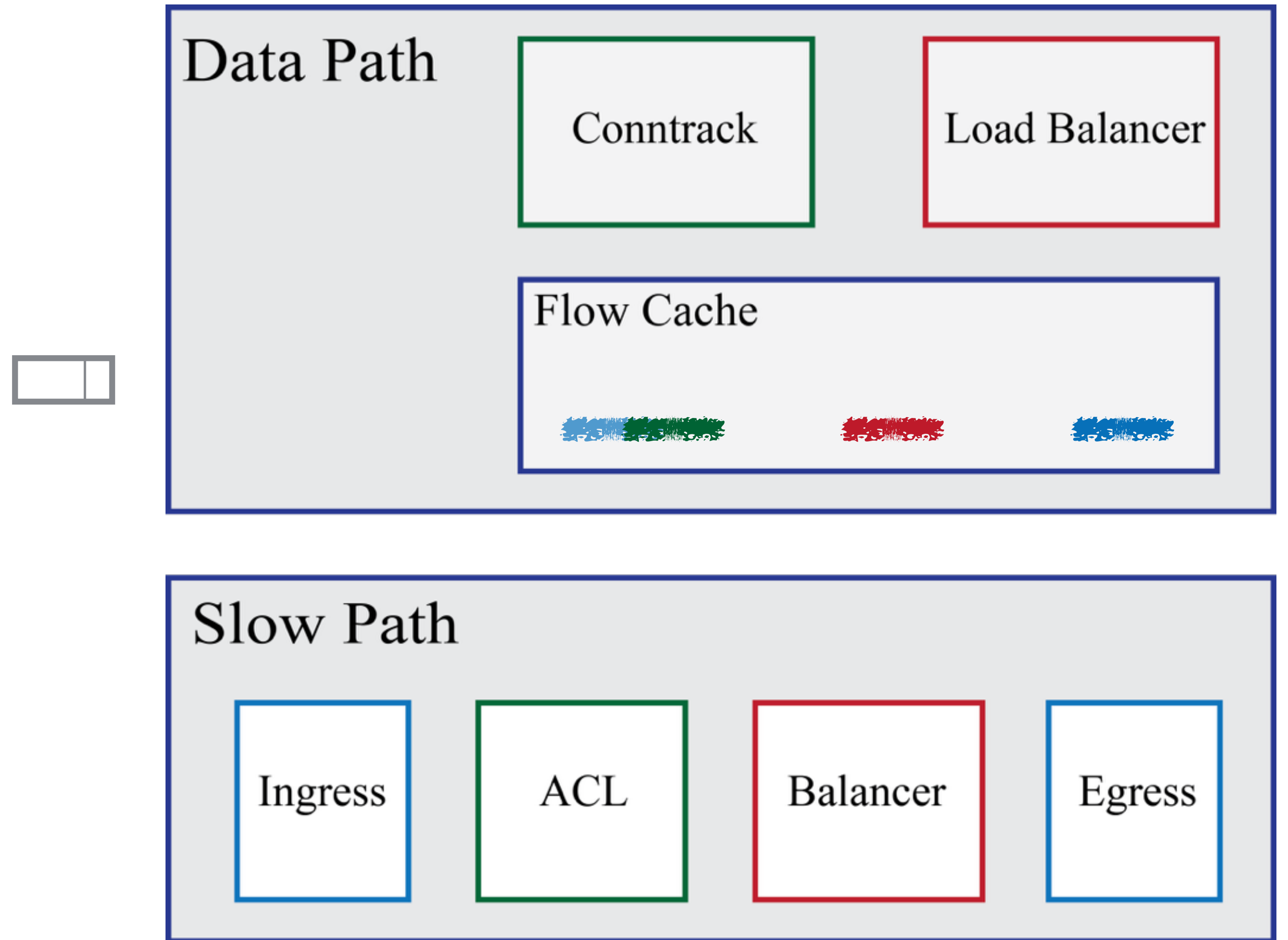


# SoftFlow

- Actions
  - Arbitrary x86 Code
  - Reside in the datapath
  - Callable from OpenFlow

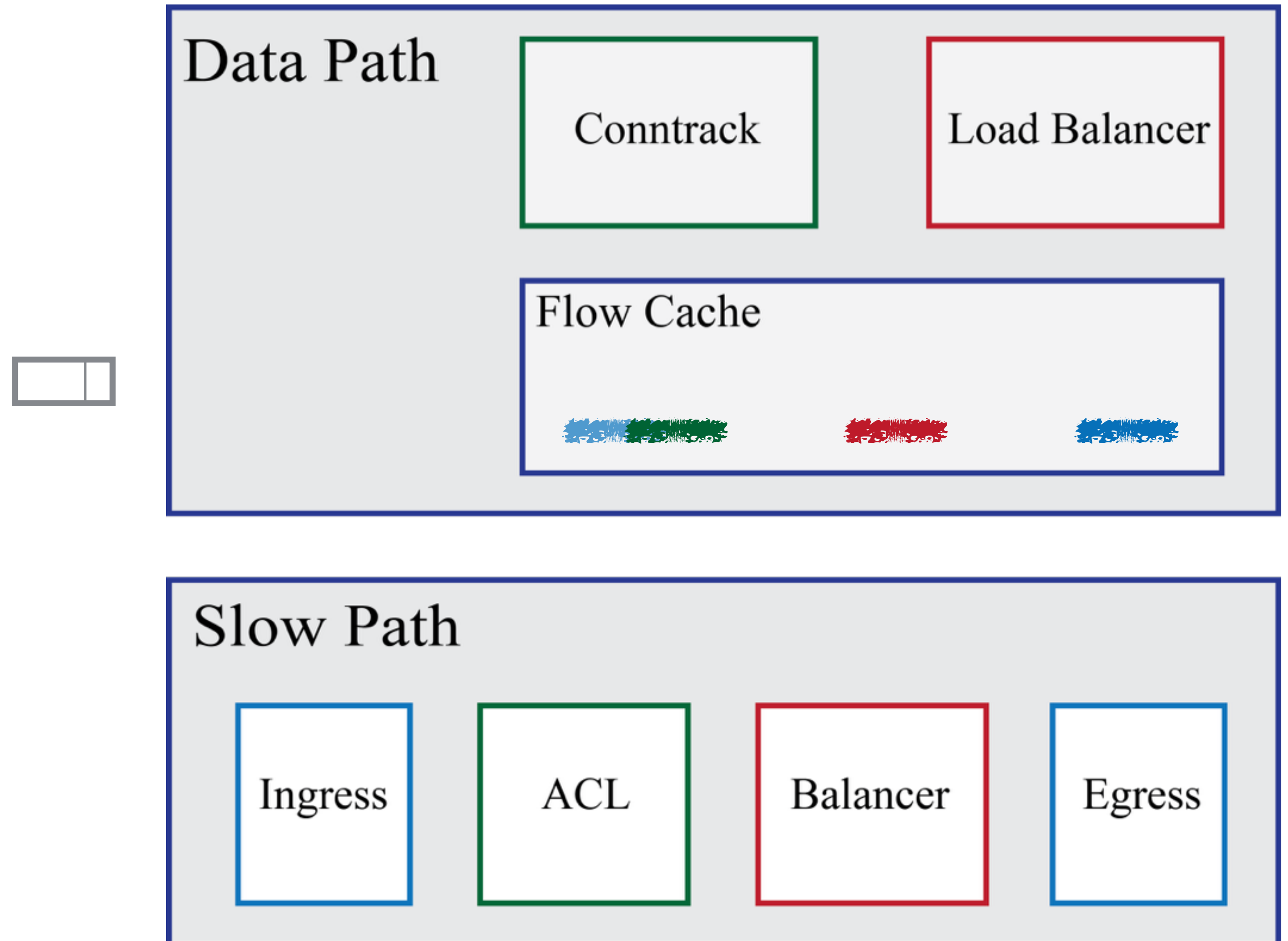


# SoftFlow — First Packet



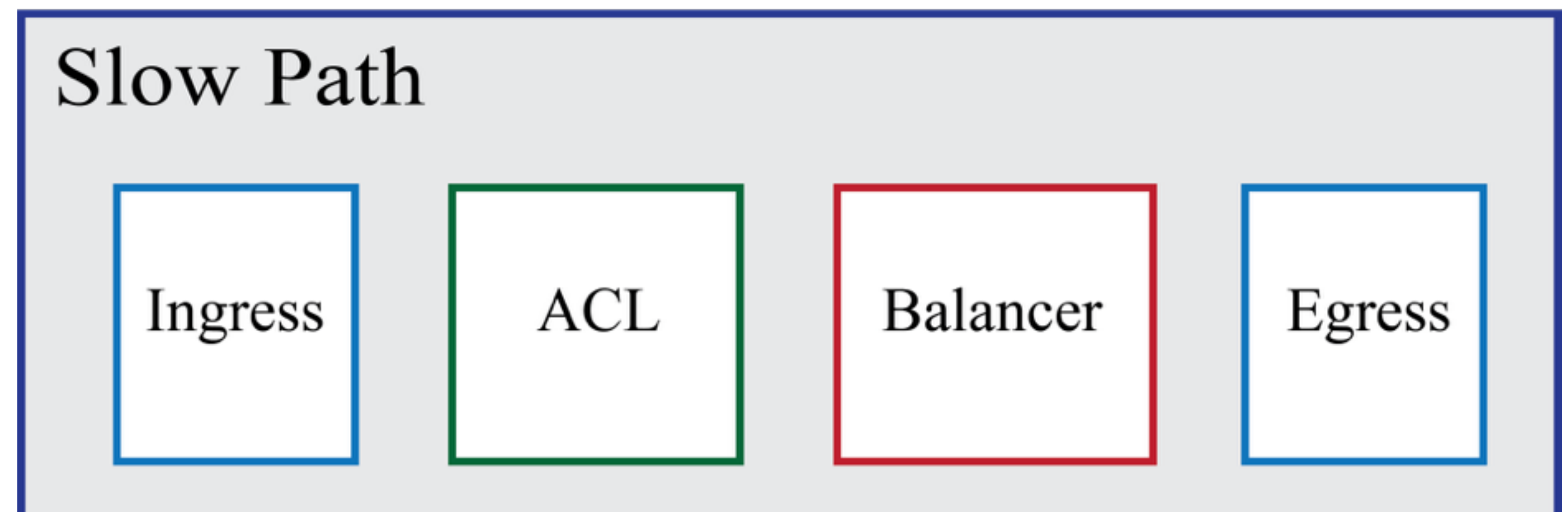
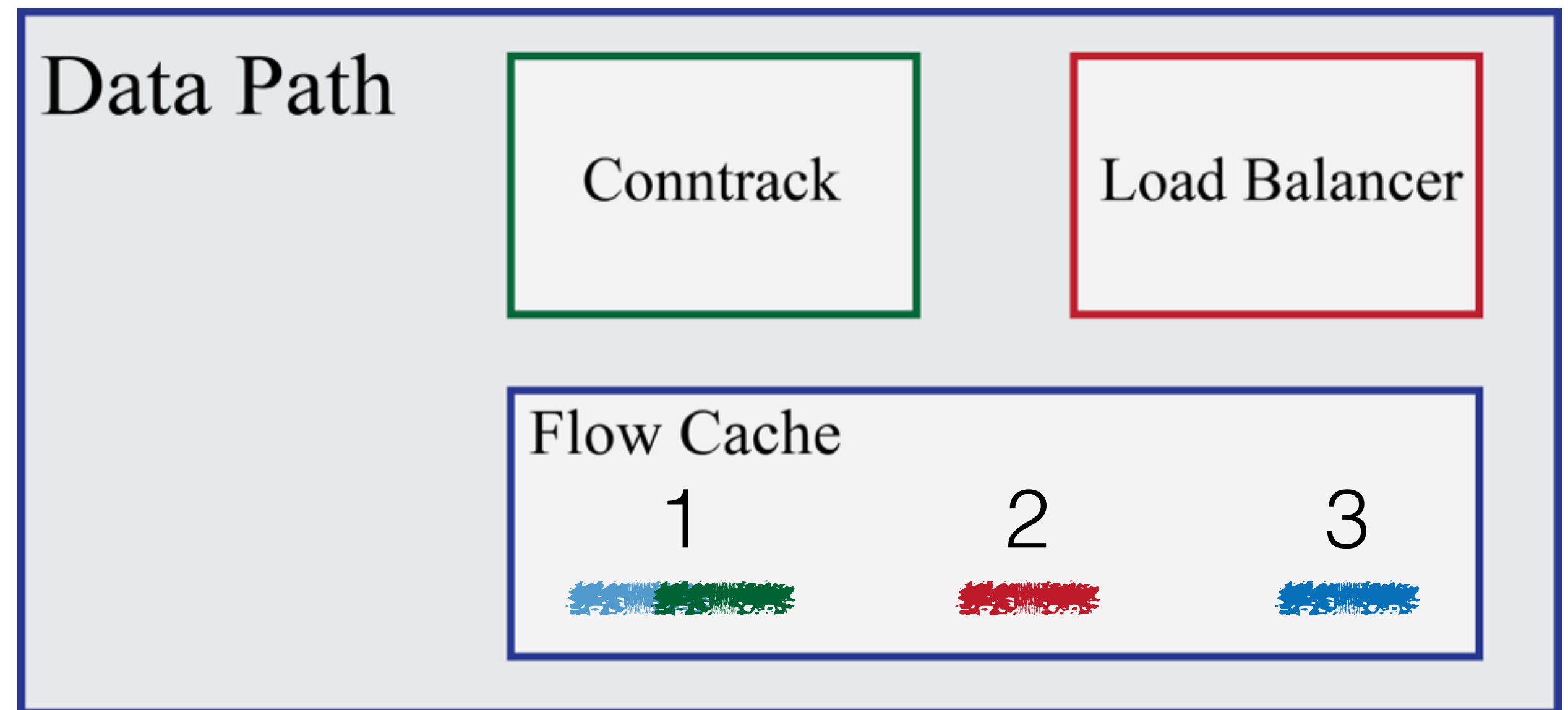


# SoftFlow — Subsequent Packets



# SoftFlow — Subsequent Packets

- Three classifications
  - Ingress
  - Post Conntrack
  - Post Load Balancer



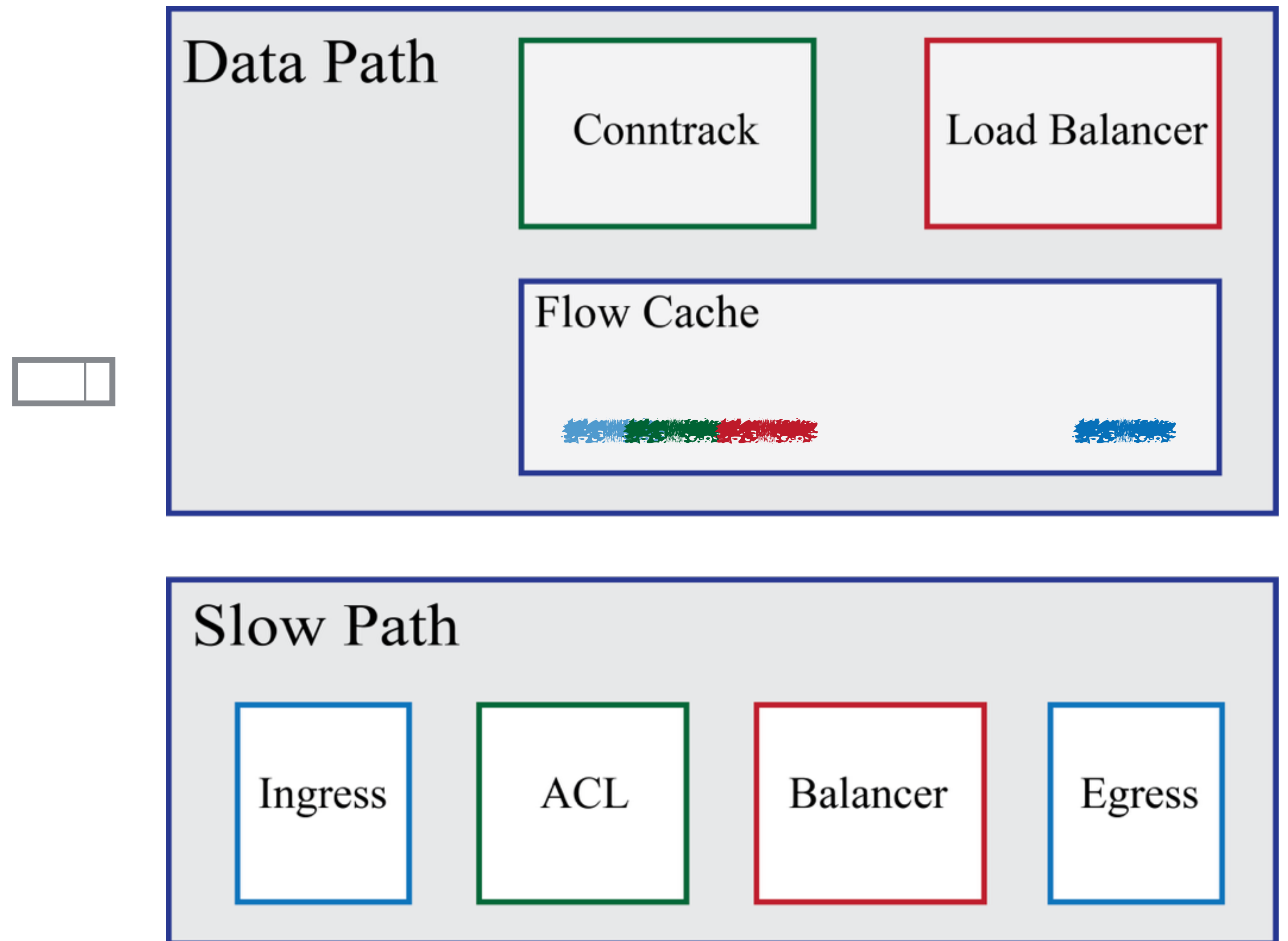
# Data Path Classifications

- Open vSwitch
  - One data path classification
- Middlebox processing is non-deterministic
  - Must re-classify after each SoftFlow action traversal

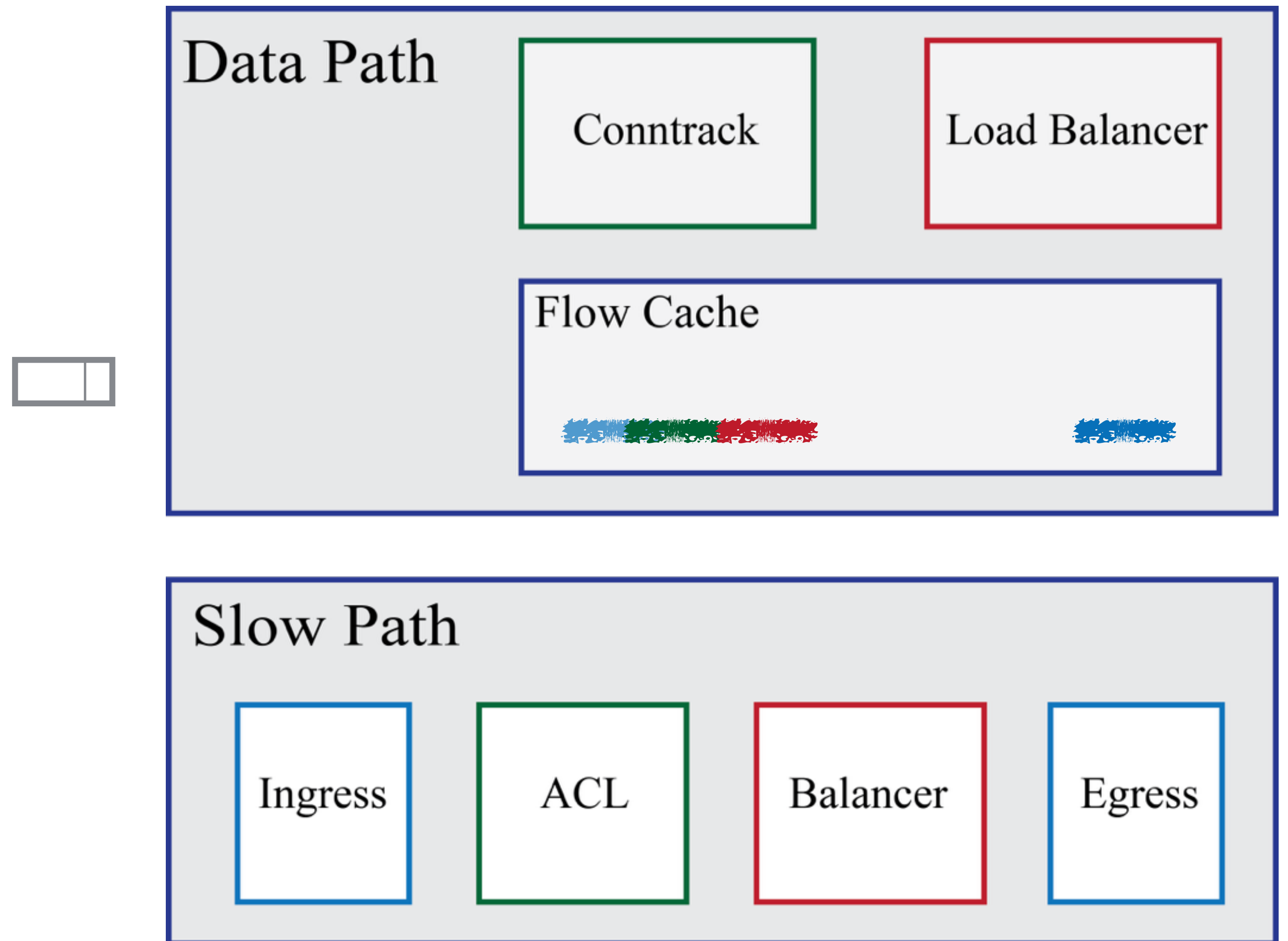
# Classification Coalescing

- Some Middleboxes are *mostly deterministic*
  - Connection tracker — almost always “allow”
- Furthermore, they know when they’re deterministic
- In these cases, skip data path classification

# Classification Coalescing

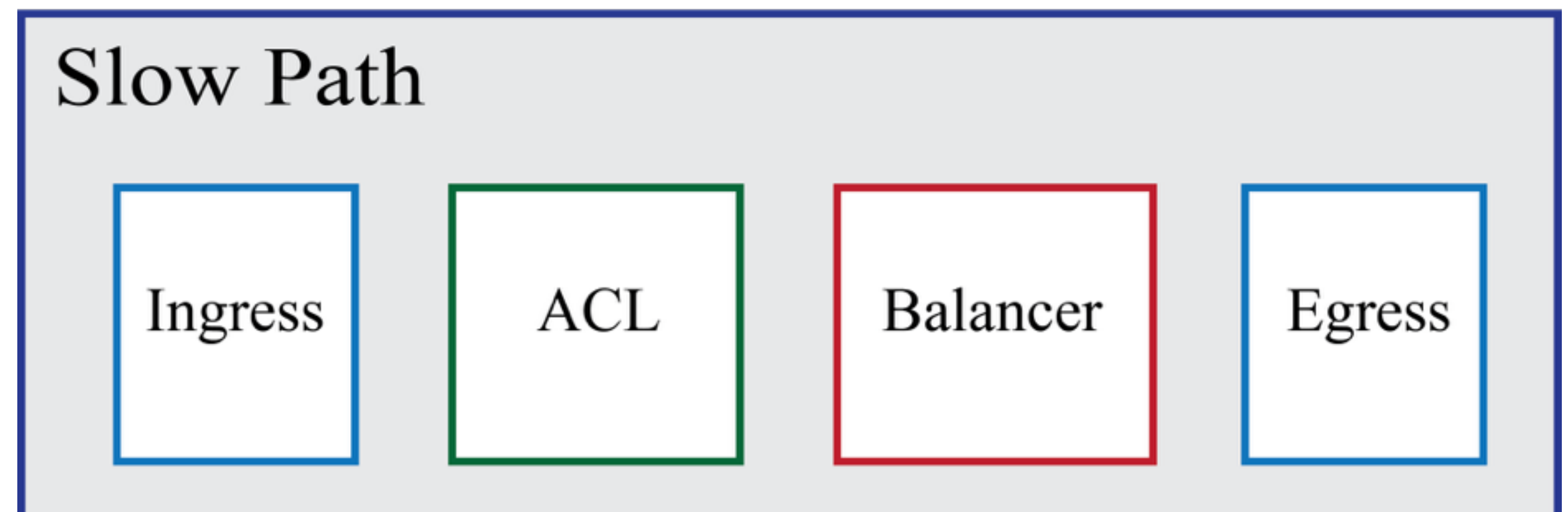
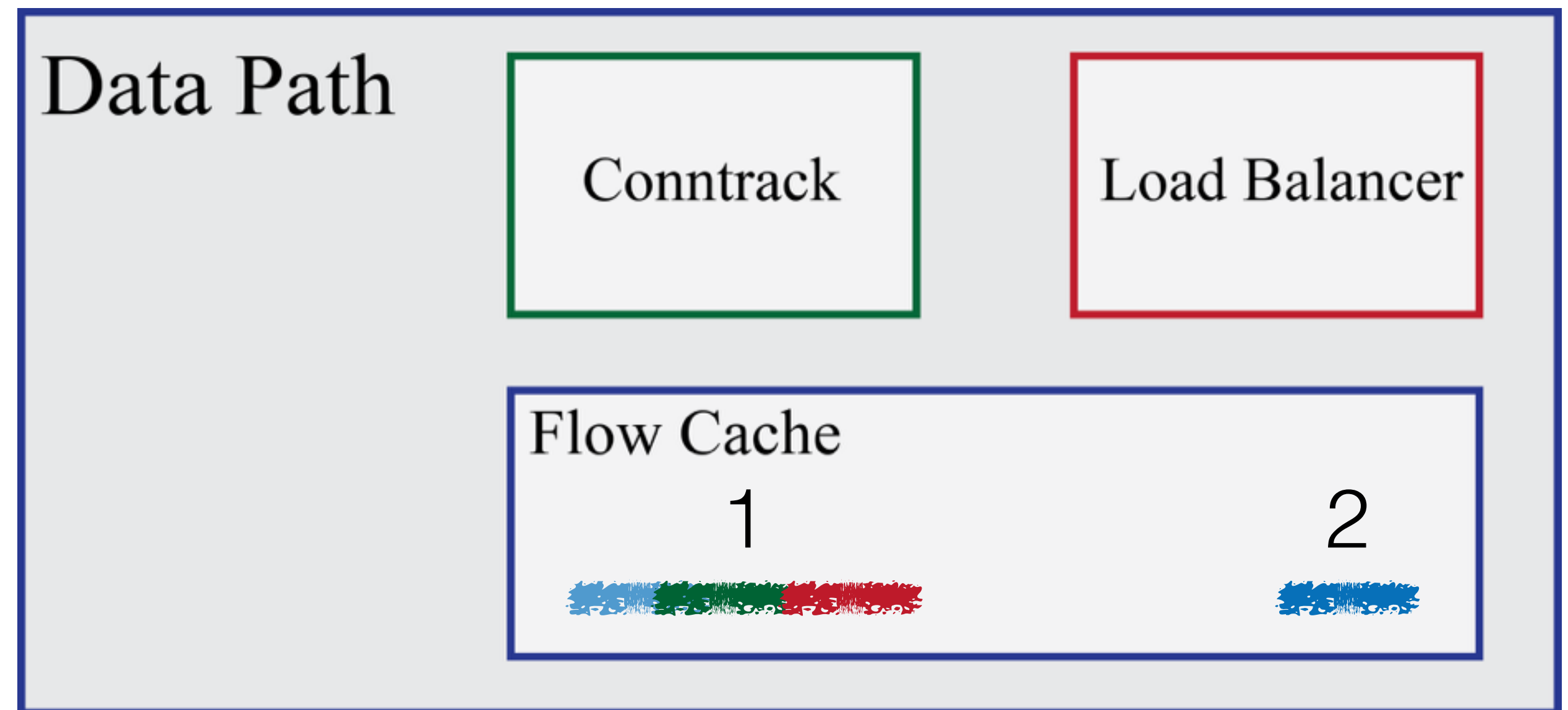


# Classification Coalescing



# Classification Coalescing

- Exploit Determinism
- Reduce Packet Classification
- Cross-NF flow caching



# Evaluation



# Evaluation

- Measure performance impact
  - Run to completion
  - Classification coalescing
- Much more in the paper

# Traces

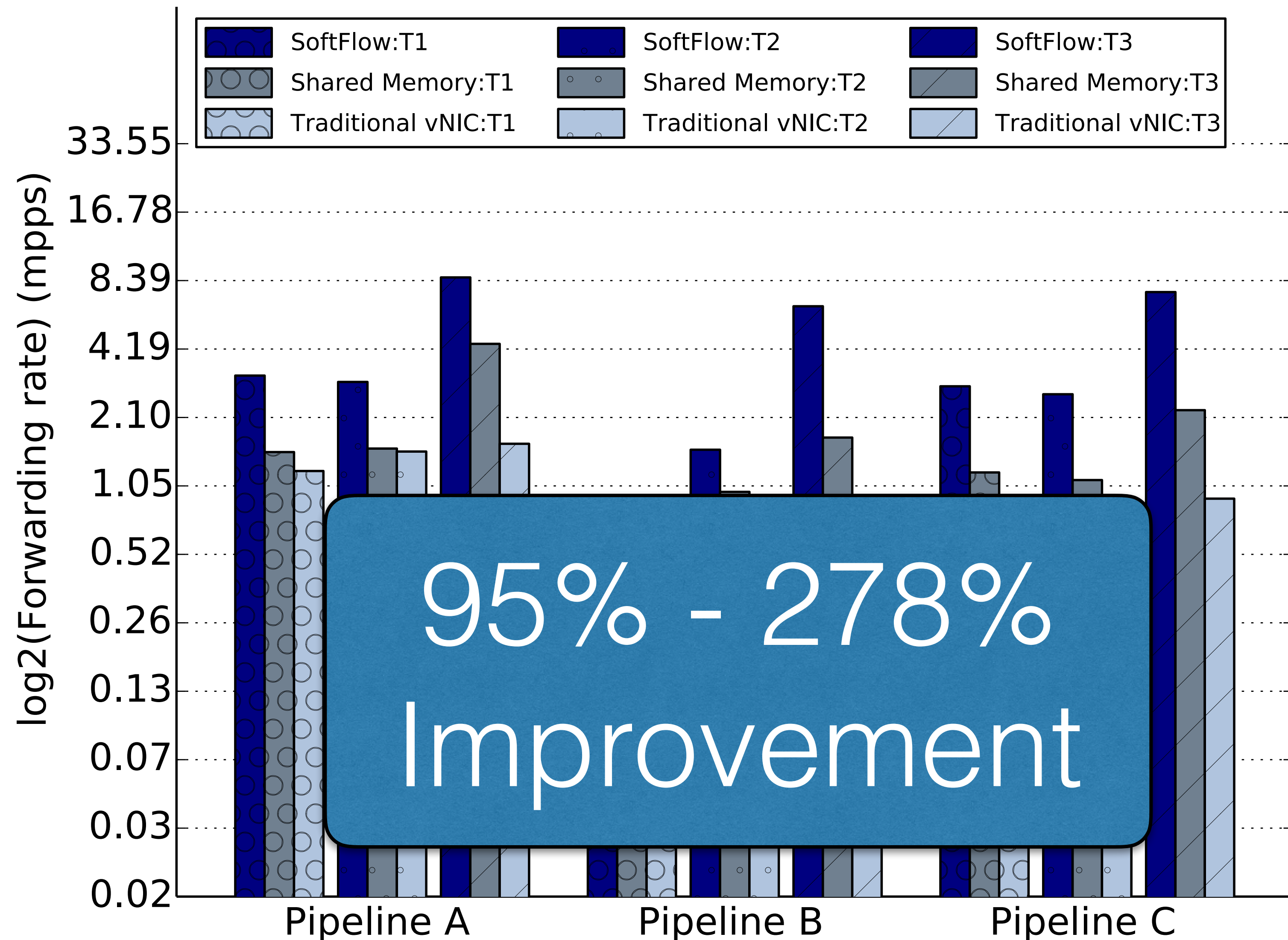
- Trace 1
  - Collected from SDN Gateway
- Trace2
  - Trace 1 with packets truncated to 64 bytes
- Trace 3
  - Synthetic trace with 32 long-lived connections

# Pipelines

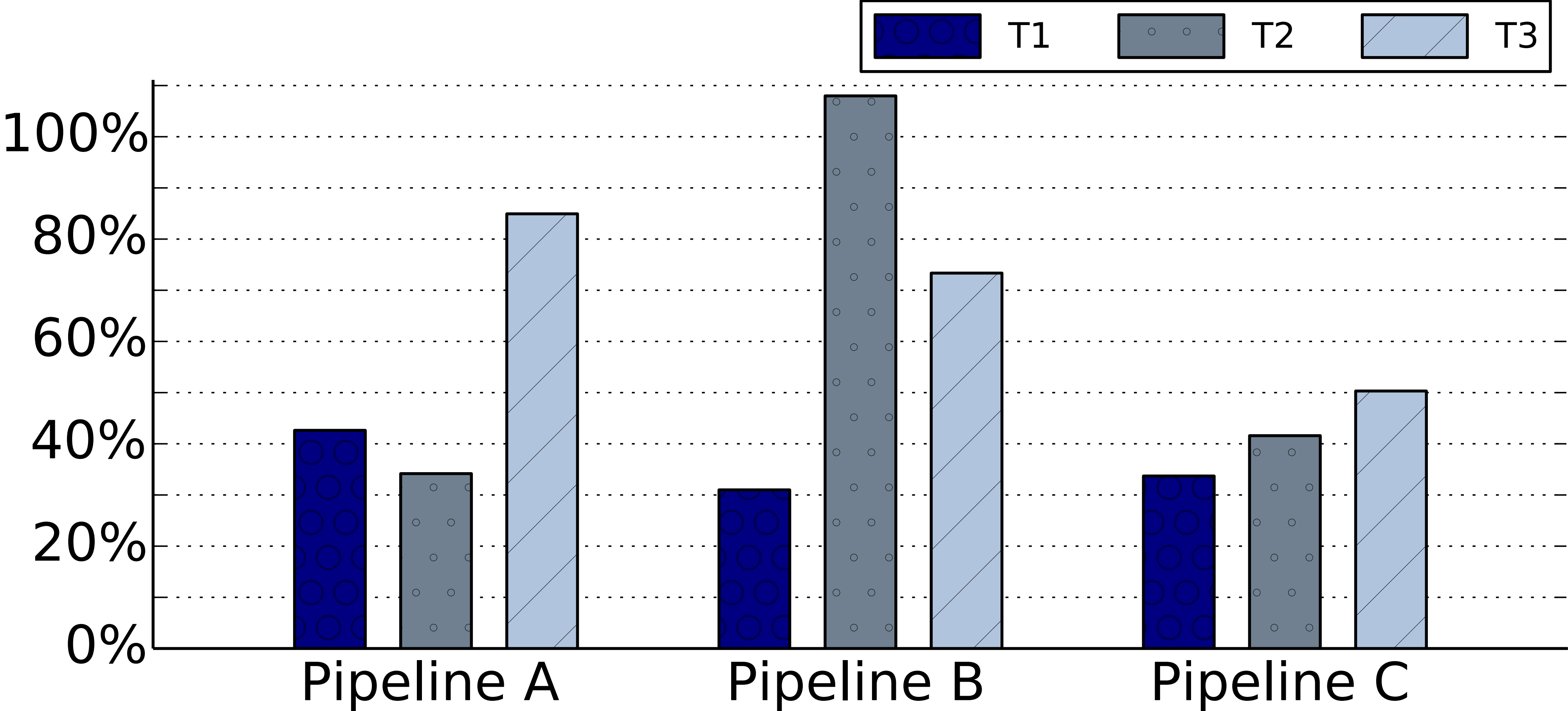
- Pipeline A
  - L2 -> L3 -> Stateful Firewall -> L2
- Pipeline B
  - L2 -> L3 -> Stateful Firewall -> AES Transcoder -> L2
- Pipeline C
  - L2 -> L3 -> Stateful Firewall -> Load Balancer -> L2

# SoftFlow vs VM NFs

- Equivalent Middleboxes Implemented as VMs
- Shared Memory
- Traditional vNIC
- Log Scale



# Coalescing Percent Improvement



# SoftFlow

- Middleboxes for Open vSwitch
- Maintain the benefits of OpenFlow
- Plugin library of flexible NFs
- Much more in the paper

Thank You