Congestion Management – Congestion Isolation

Paul Congdon

Yolanda Yu

Kevin Shen

paul.congdon@tallac.com

yolanda.yu@huawei.com

kevin.shenli@huawei.com

IEEE 802.1 DCB

St John's Newfoundland

September 2017

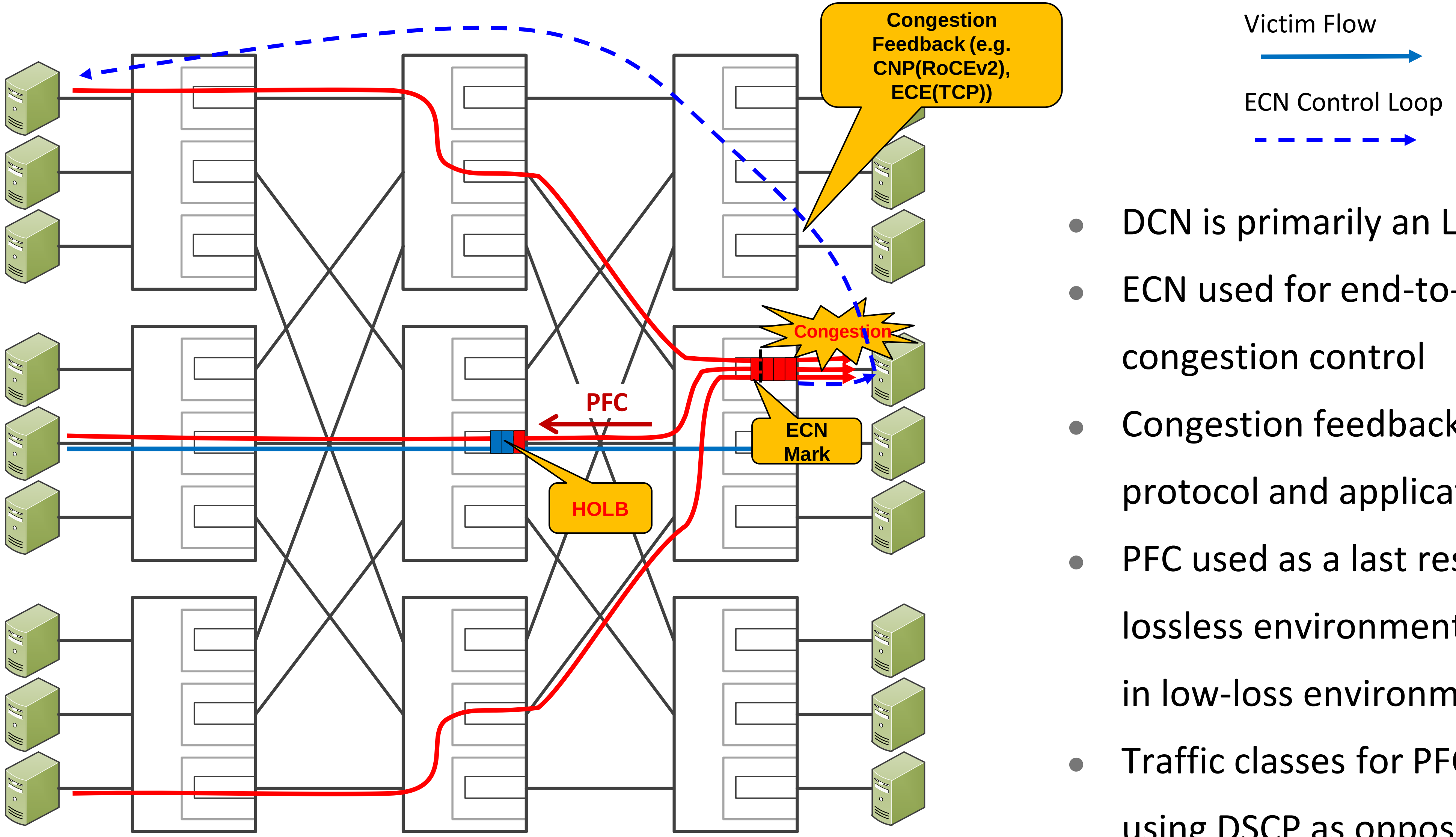
Agenda

- Low-Latency, Lossless, Large-Scale DCNs
- Challenges going forward
- Solution Goals
- Congestion Isolation Details
- Simulation Analysis
- Next Steps

The Case for Low-latency, Lossless, Large-Scale DCNs

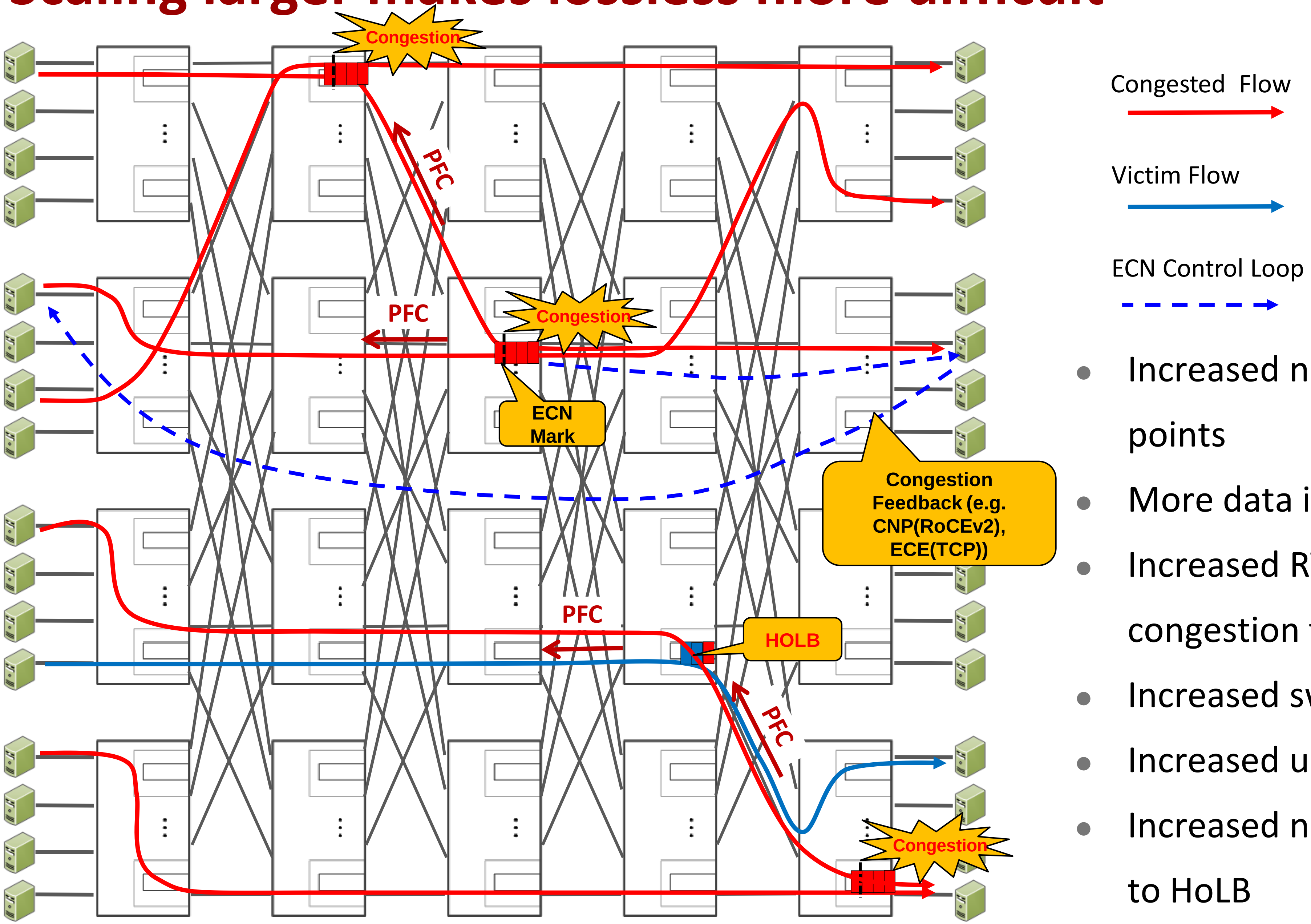
- More and more latency-sensitive applications are being deployed in data centers
  - Distributed Storage
  - AI / Deep Learning
  - Cloud HPC
  - High-Frequency Trading
- RDMA is operating at larger scales thanks to RoCEv2
  - Chuanxiong Guo, et. al., Microsoft, ”RDMA over Commodity Ethernet at Scale”, SIGCOMM 2016
  - Y Zhu, H Eran, et. al., Microsoft, Mellanox, “Congestion control for large-scale RDMA deployments”, SIGCOMM 2015
  - Radhika Mittal, et. al., UC Berkeley, Google, “TIMELY: RTT-based Congestion Control for the Datacenter”, SIGCOMM 2015
- The scale of Data Center Networks continues to grow
  - Larger, faster clusters are better than more smaller size clusters
  - Server growth continues at 25% - 30% putting pressure on cluster sizes and networking costs

Lossless DCN state-of-the-art



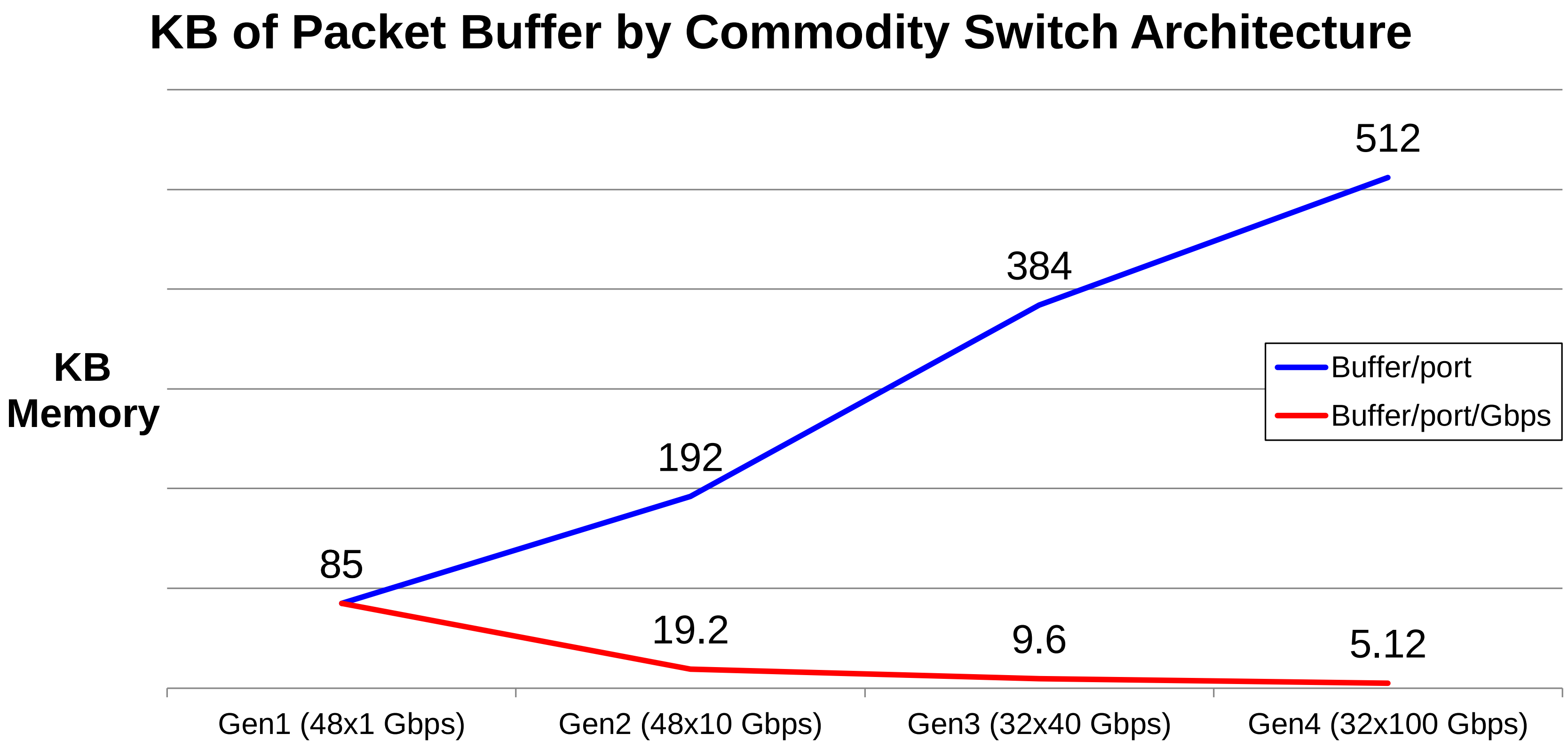
- DCN is primarily an L3 network
- ECN used for end-to-end congestion control
- Congestion feedback can be protocol and application specific
- PFC used as a last resort to ensure lossless environment, or not at all in low-loss environments.
- Traffic classes for PFC are mapped using DSCP as opposed to VLAN tags

Scaling larger makes lossless more difficult



- Increased number of congestion points
- More data in-flight
- Increased RTT and delay for congestion feedback
- Increased switch buffer requirements
- Increased use of PFC
- Increased number of victim flows due to HoLB

Switch buffer growth is not keeping up



Commodity Shallow Buffer Switches in DCNs are desirable:

- Low Latency
- Low Cost

However, packet loss can create performance issues:

- Source: Broadcom, “White Paper: Buffer Requirements for Datacenter Network Switches”, DNFAMILY-WP1101, August 25, 2015

Source: “Congestion Control for High-speed Extremely Shallow-buffered Datacenter Networks”. In Proceedings of APNet’17, Hong Kong, China, August 03-04, 2017, https://doi.org/10.1145/3106989.3107003

Concerns about over-using PFC

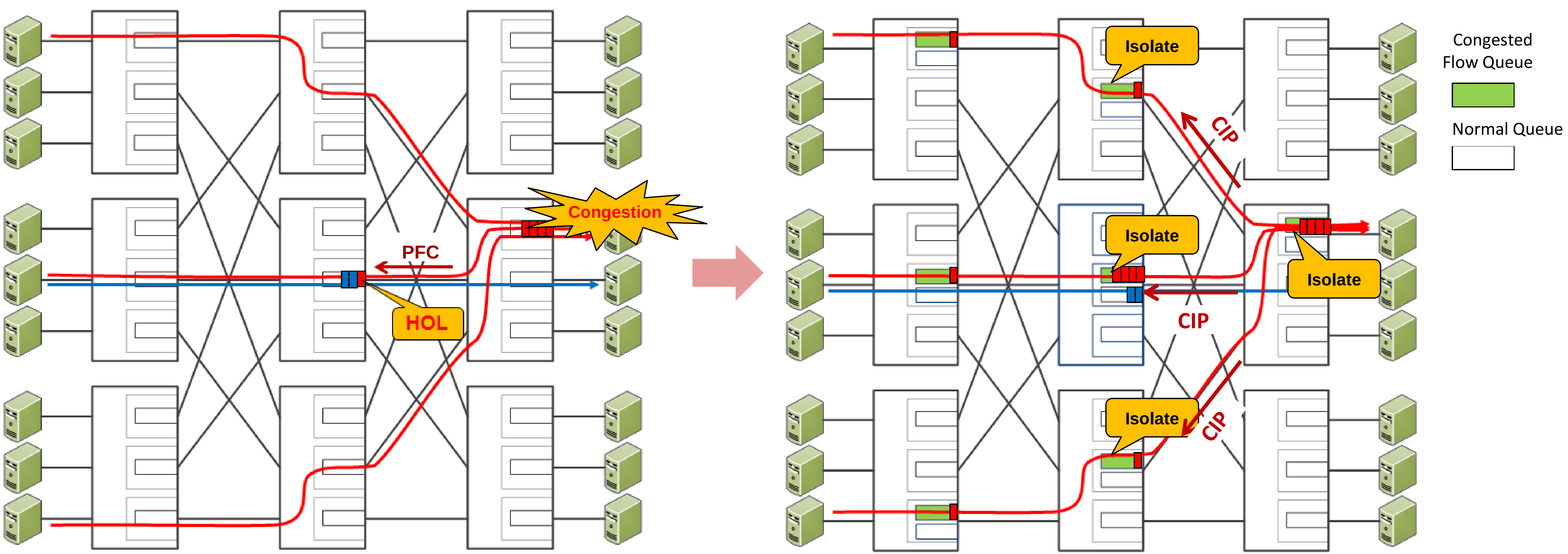
- HoL blocking
- Congestion spreading
- Buffer Bloat, increasing latency
- Increased jitter reducing throughput
- Deadlocks

Goals

- Support larger, faster data centers (Low-Latency, High-Throughput)
- Support lossless transfers
- Improve performance of TCP and UDP based flows
- Reduce pressure on switch buffer growth
- Reduce the frequency of relying on PFC for a lossless environment

- Eliminate or significantly reduce HOLB caused by over-use of PFC

Isolate the congestion to mitigate HOLB



Congestion Isolation

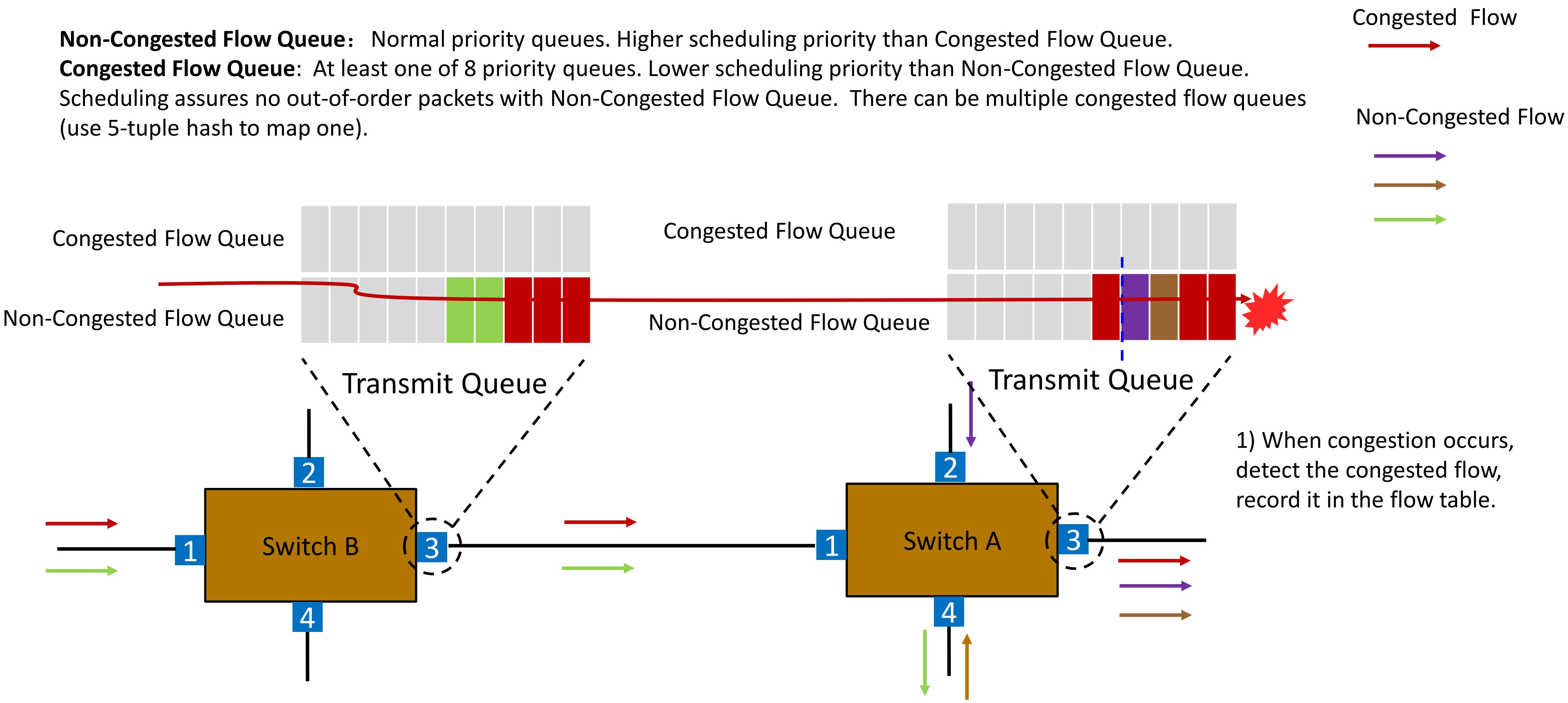
Definition: An approach to isolate flows causing congestion and signal upstream to isolate the same flows to avoid head-of-line blocking.

The approach involves:

1. Identifying the flows creating congestion (e.g. perhaps already done for QCN and/or ECN)
2. Using implementation specific approaches to dynamically adjust the traffic class of offending flows without packet re-ordering (e.g. DVL – Dynamic Virtual Lanes)
3. Signaling upstream indications via a Congestion Isolation Packet (CIP)

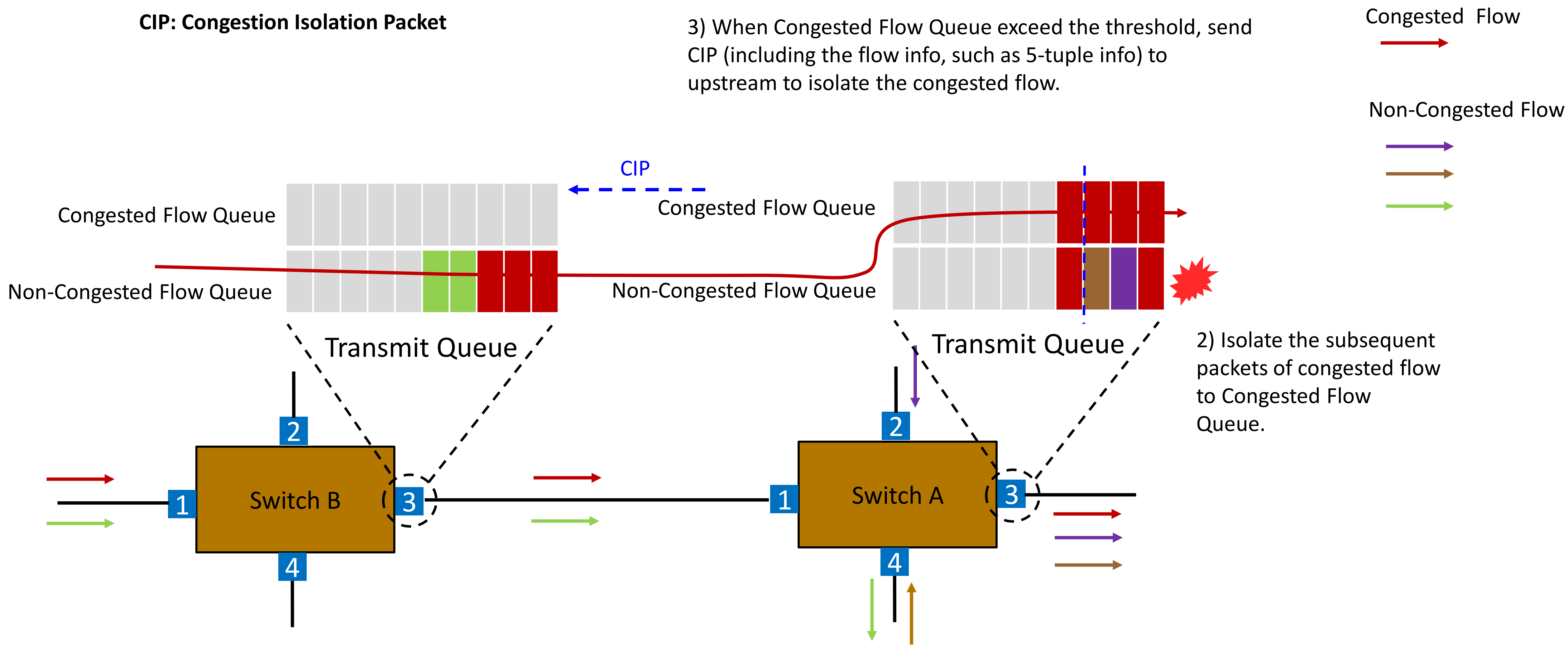
Congestion Isolation with Dynamic Virtual Lanes

Non-Congested Flow Queue： Normal priority queues. Higher scheduling priority than Congested Flow Queue.
Congested Flow Queue: At least one of 8 priority queues. Lower scheduling priority than Non-Congested Flow Queue. Scheduling assures no out-of-order packets with Non-Congested Flow Queue. There can be multiple congested flow queues (use 5-tuple hash to map one).

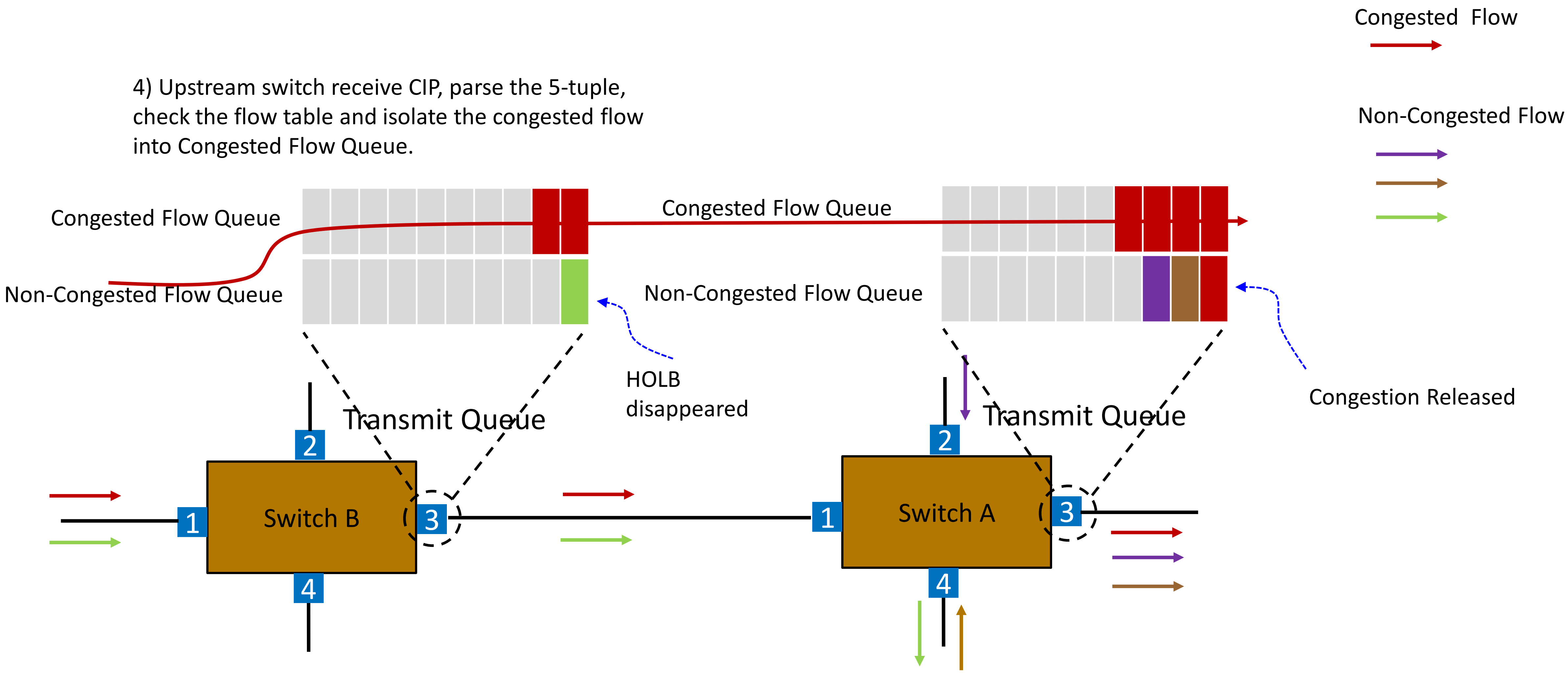


1) When congestion occurs, detect the congested flow, record it in the flow table.

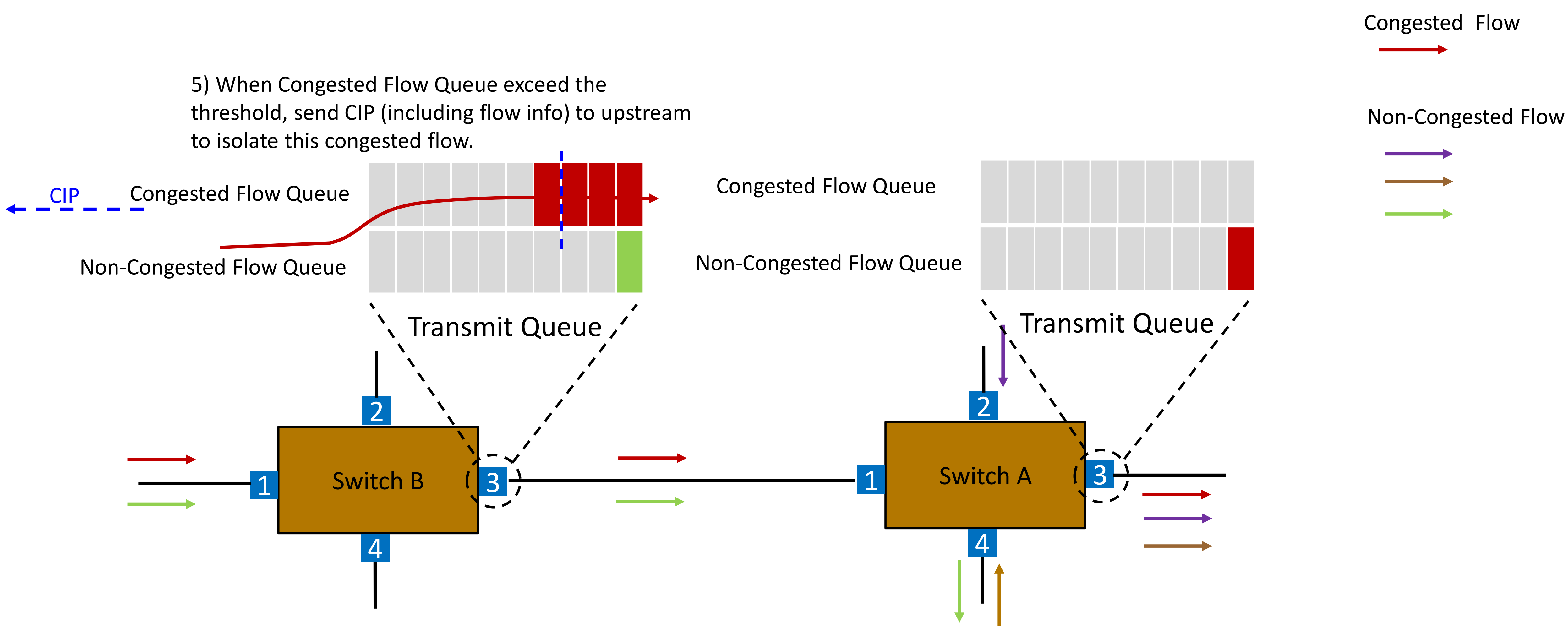
Congestion Isolation with Dynamic Virtual Lanes



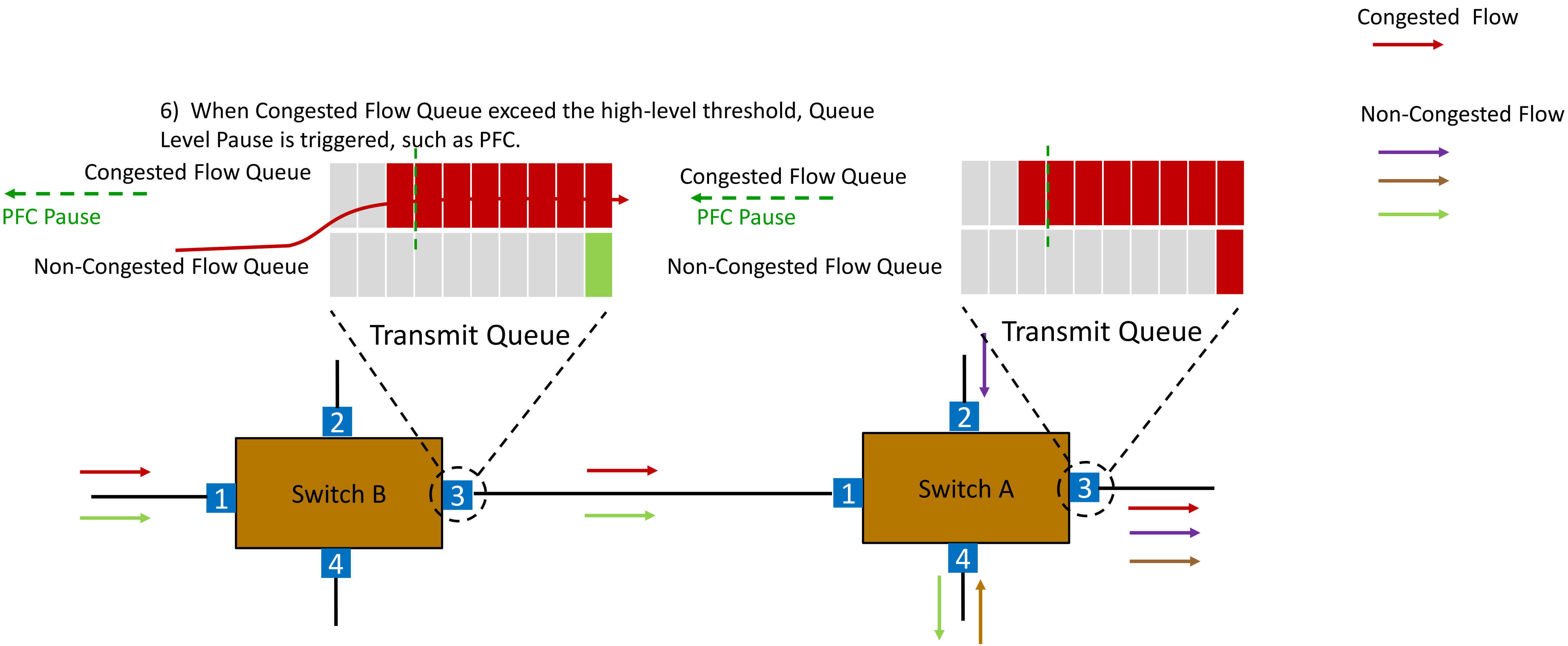
Congestion Isolation with Dynamic Virtual Lanes



Congestion Isolation with Dynamic Virtual Lanes



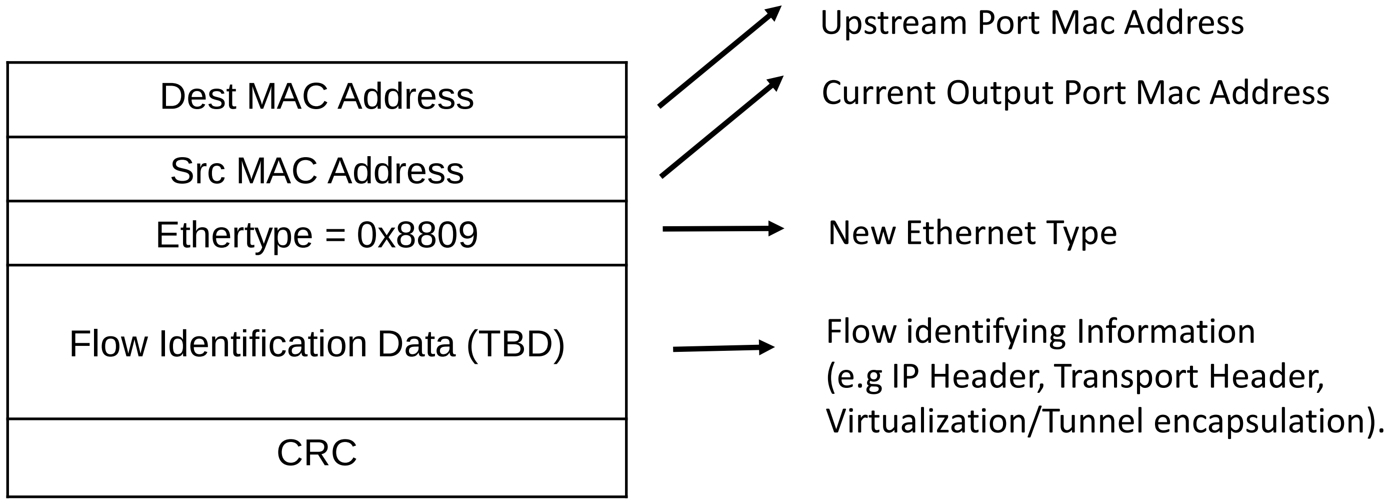
Congestion Isolation with Dynamic Virtual Lanes



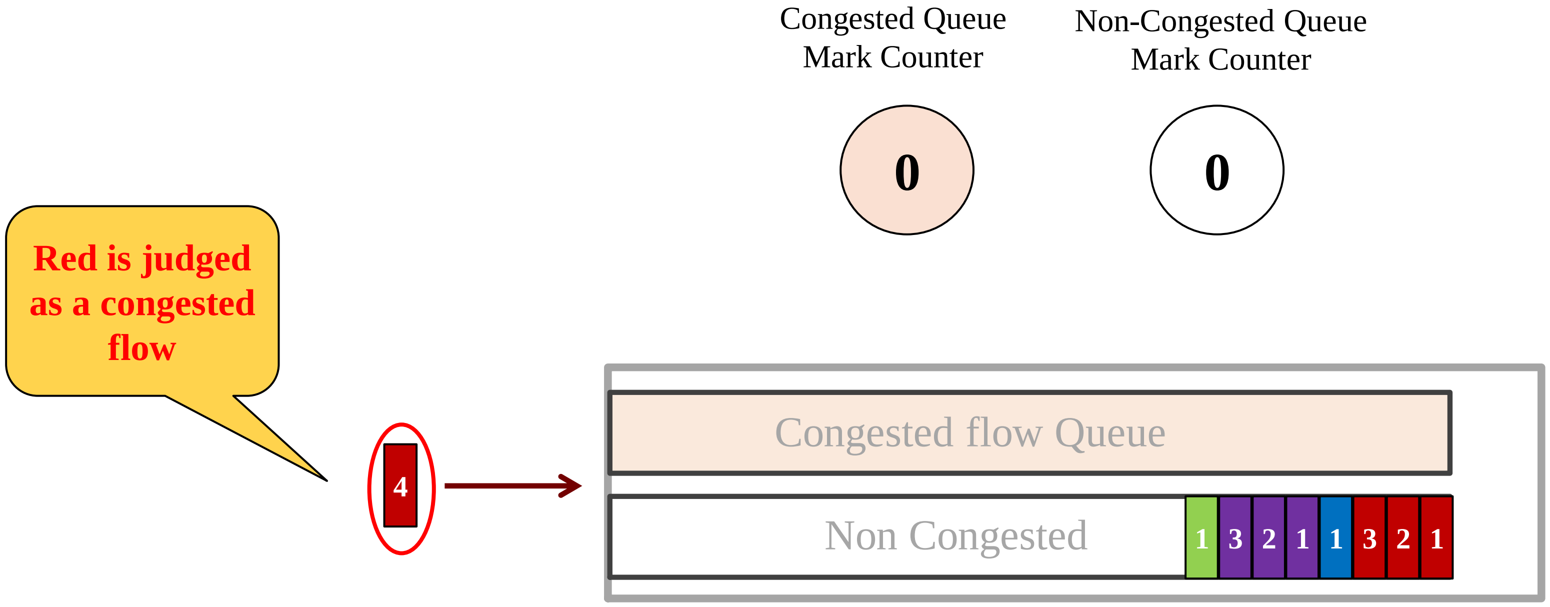
Congestion Isolation Packet

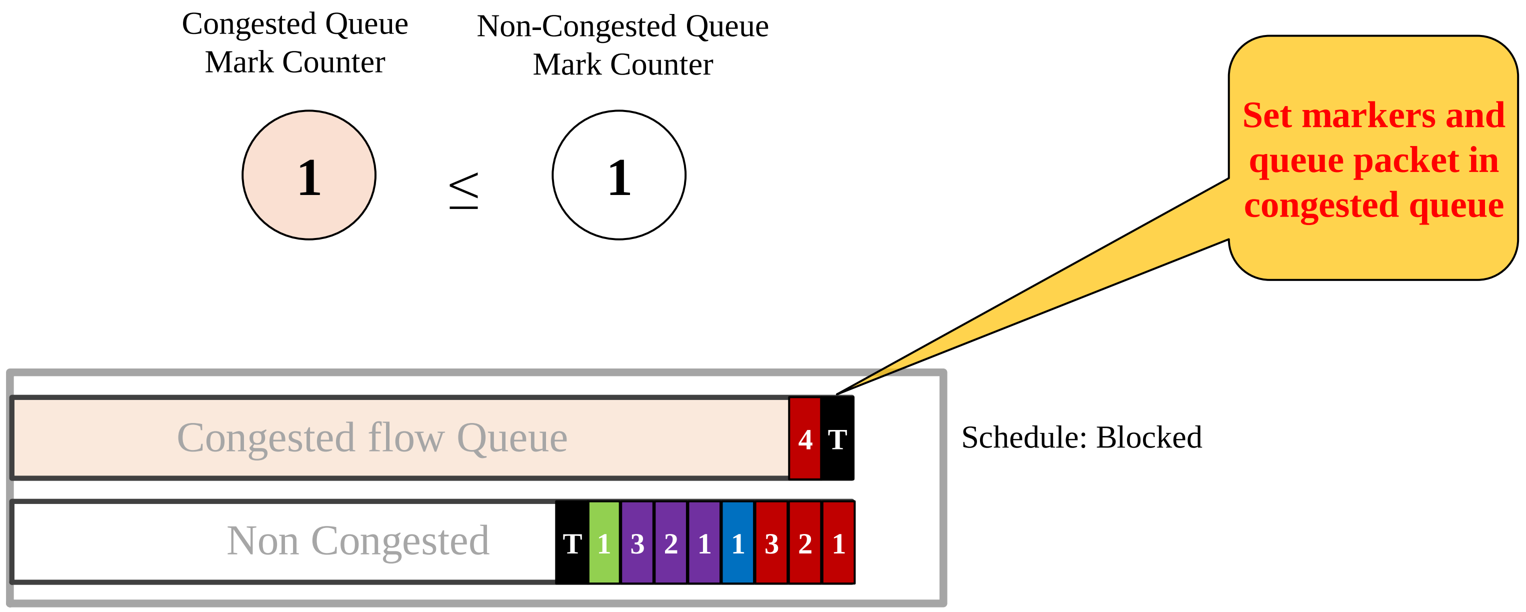
- Objectives/Requirements:
  - Provide upstream neighbor with an indication that a flow has been isolate
  - Provide upstream neighbor with flow identification information
  - No adverse effects of single packet loss
  - Low overhead

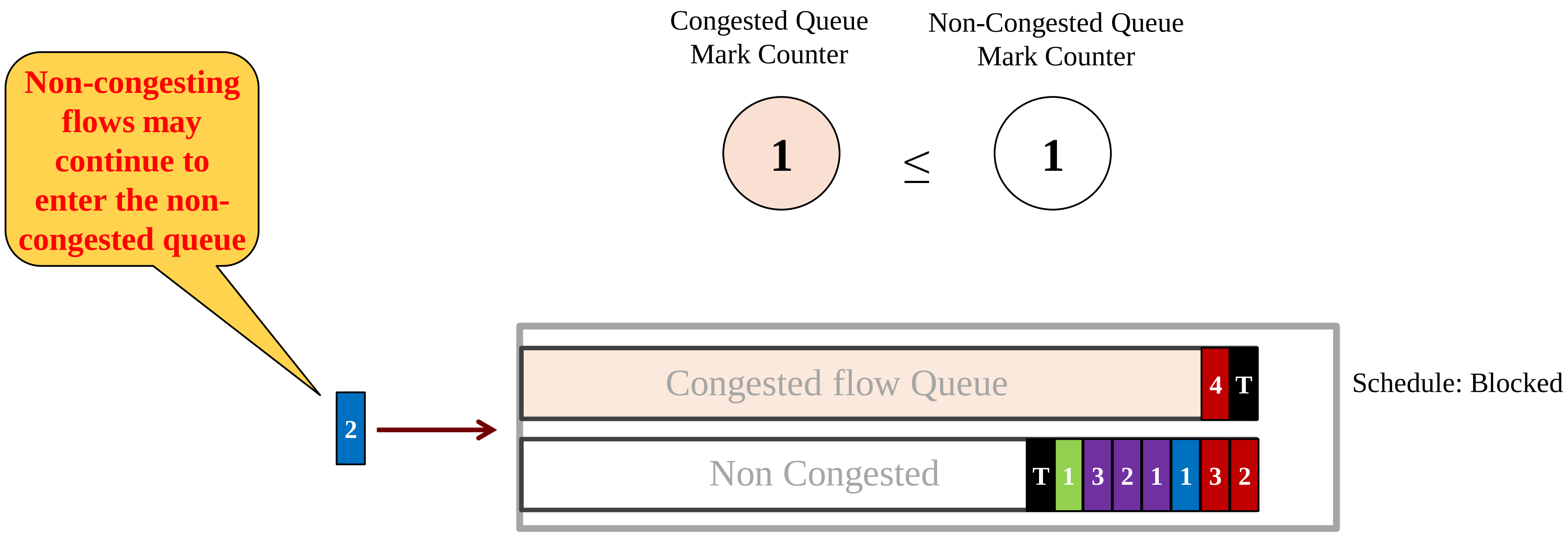
Format of Congestion Isolation Packet



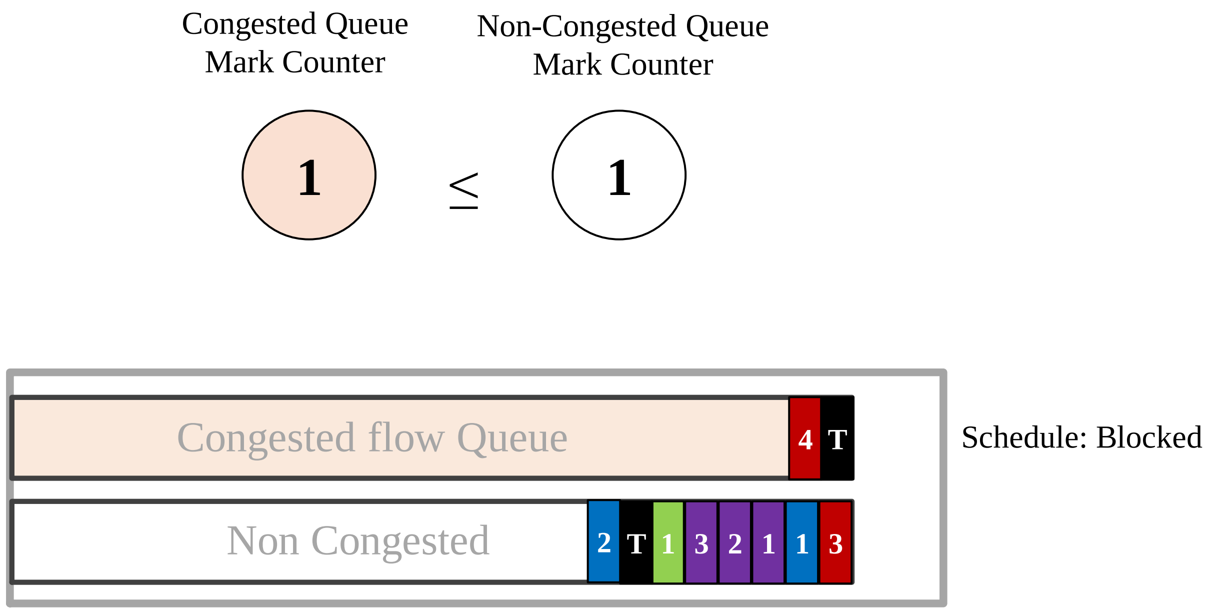
Handling the potential out-of-order problem

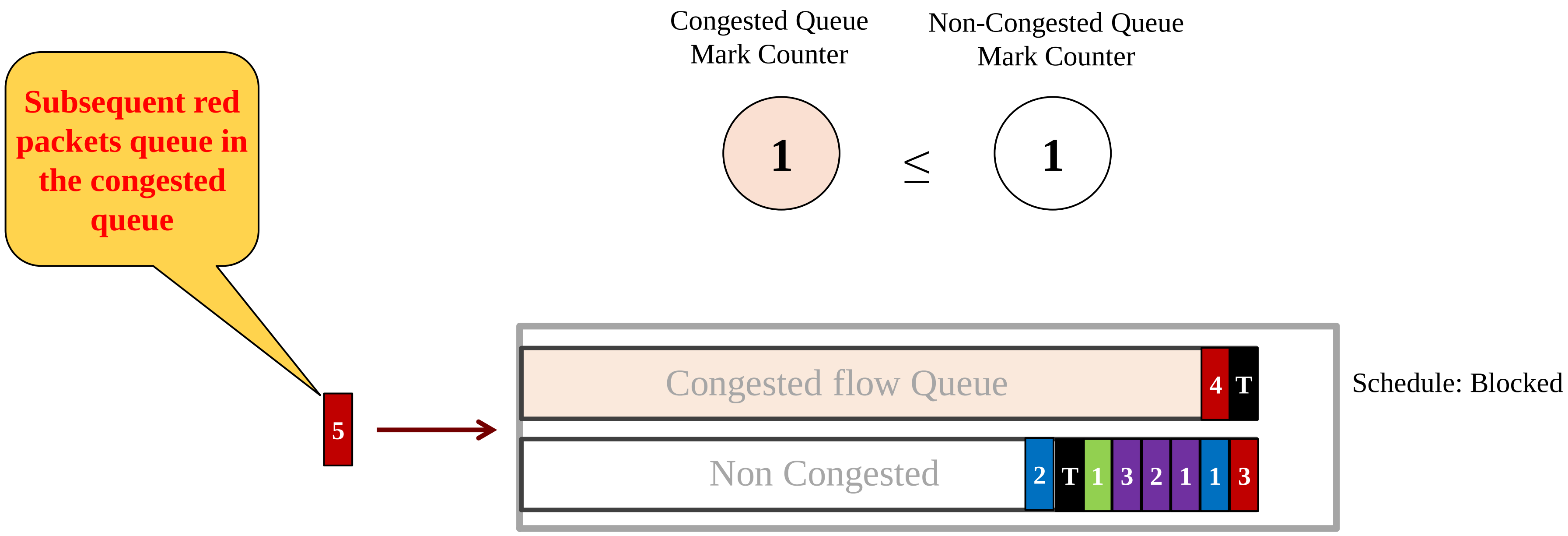


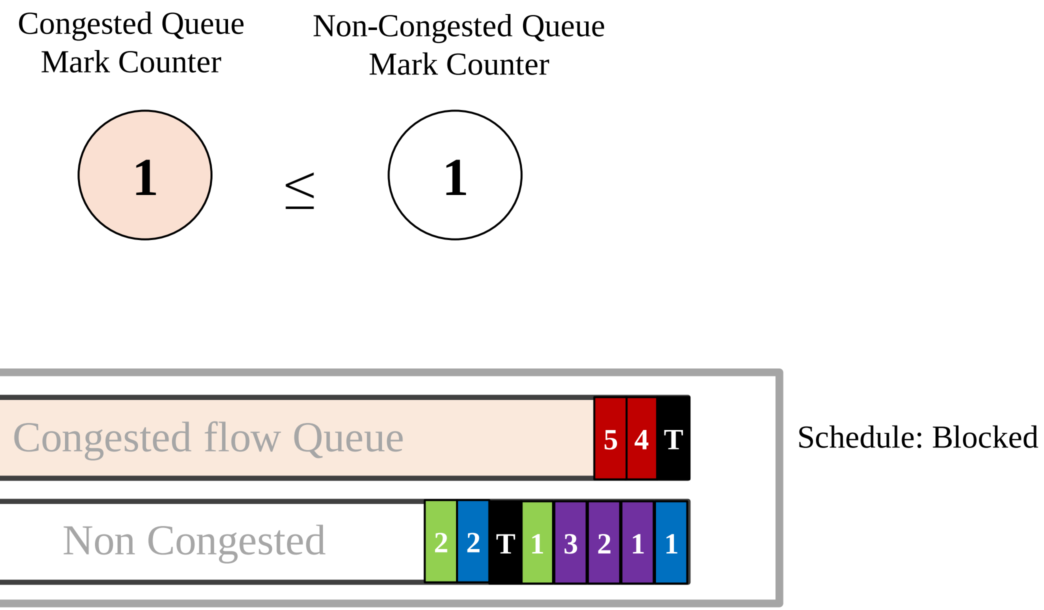

Congested Queue
Mark Counter
1
≤
Non-Congested Queue
Mark Counter
1
Set markers and queue packet in congested queue
Congested flow Queue
4
T
Non Congested
T
1
3
2
1
1
3
2
1
Schedule: Blocked

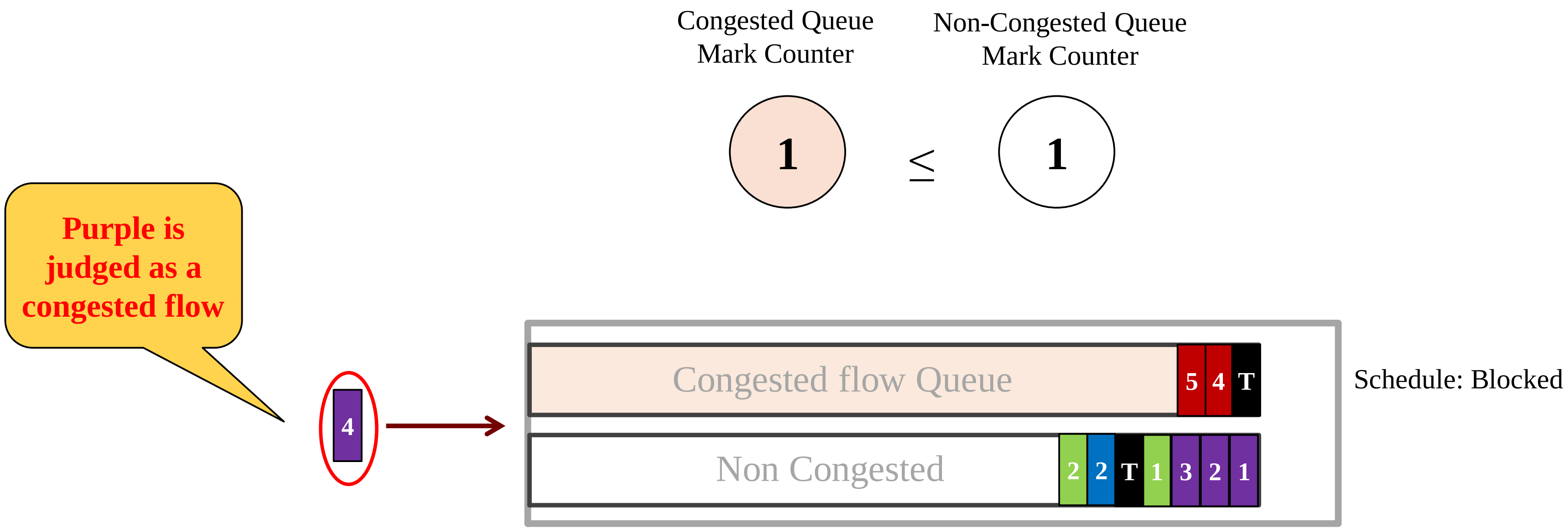


Non-congesting flows may continue to enter the non-congested queue
Congested Queue Mark Counter
Non-Congested Queue Mark Counter
1
≤
1
2
Congested flow Queue
4
T
Non Congested
T
1
3
2
1
1
3
2
Schedule: Blocked

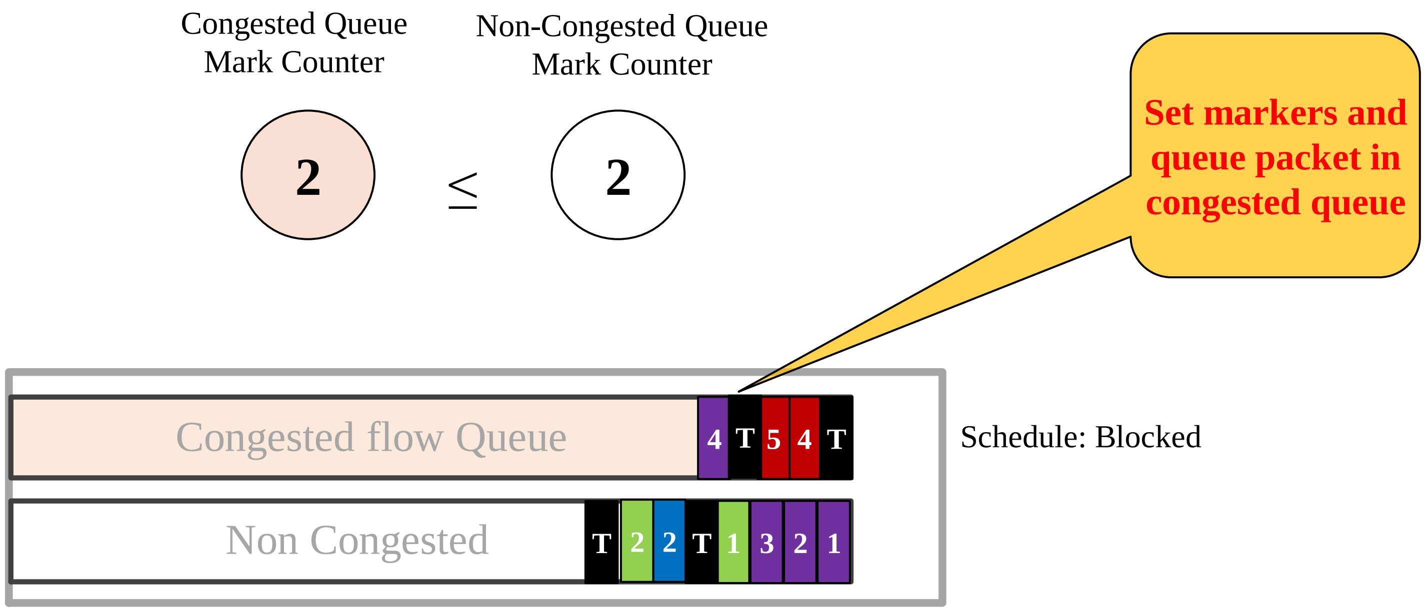

Congested Queue
Mark Counter
Non-Congested Queue
Mark Counter
1
≤
1
Congested flow Queue
4
T
Non Congested
2
T
1
3
2
1
1
3
Schedule: Blocked


Subsequent red packets queue in the congested queue
Congested Queue Mark Counter
Non-Congested Queue Mark Counter
1
≤
1
5
Congested flow Queue
4
T
Non Congested
2
T
1
3
2
1
1
3
Schedule: Blocked

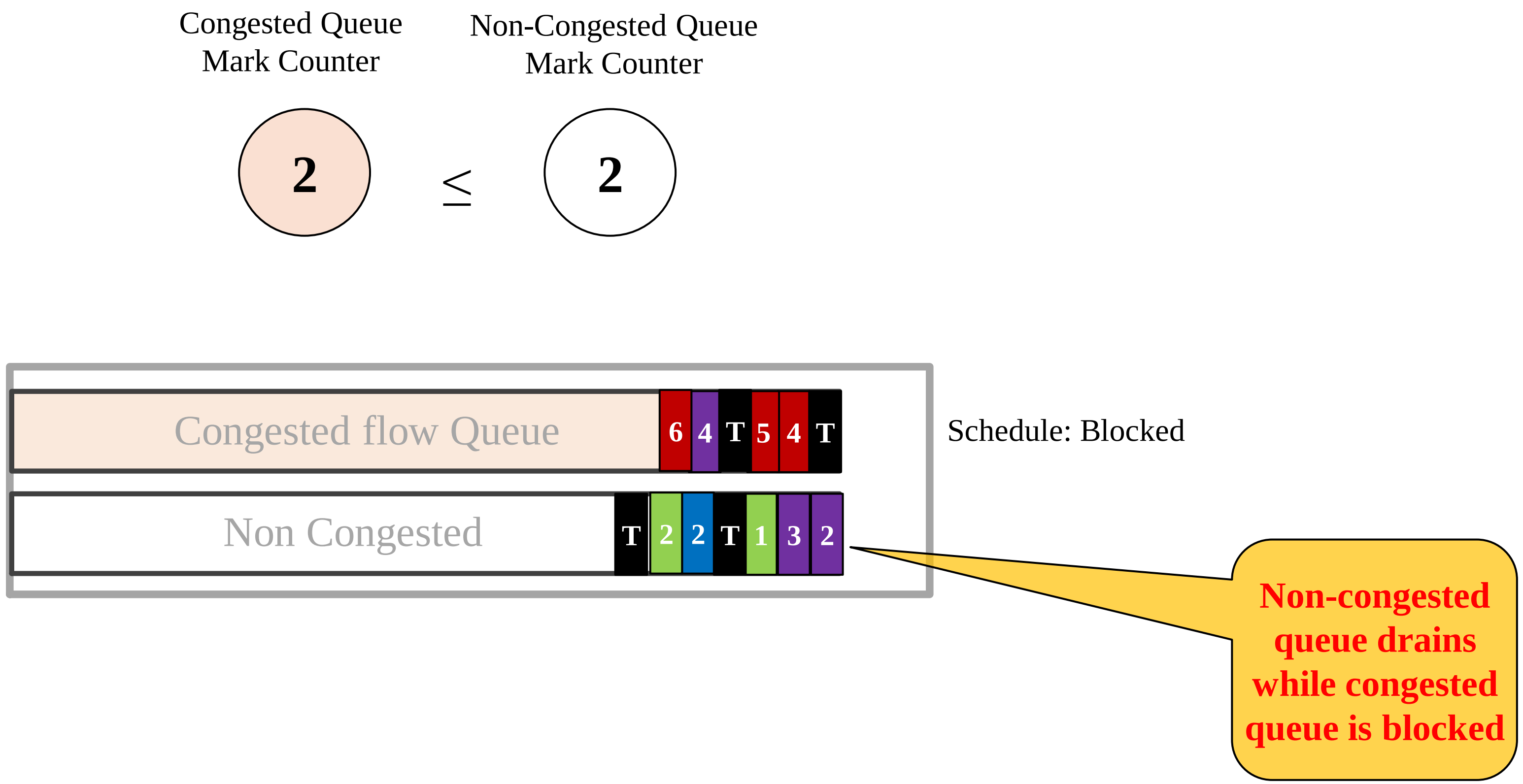

Congested Queue
Mark Counter
Non-Congested Queue
Mark Counter
1
≤
1
Congested flow Queue
5
4
T
Non Congested
2
2
T
1
3
2
1
1
Schedule: Blocked

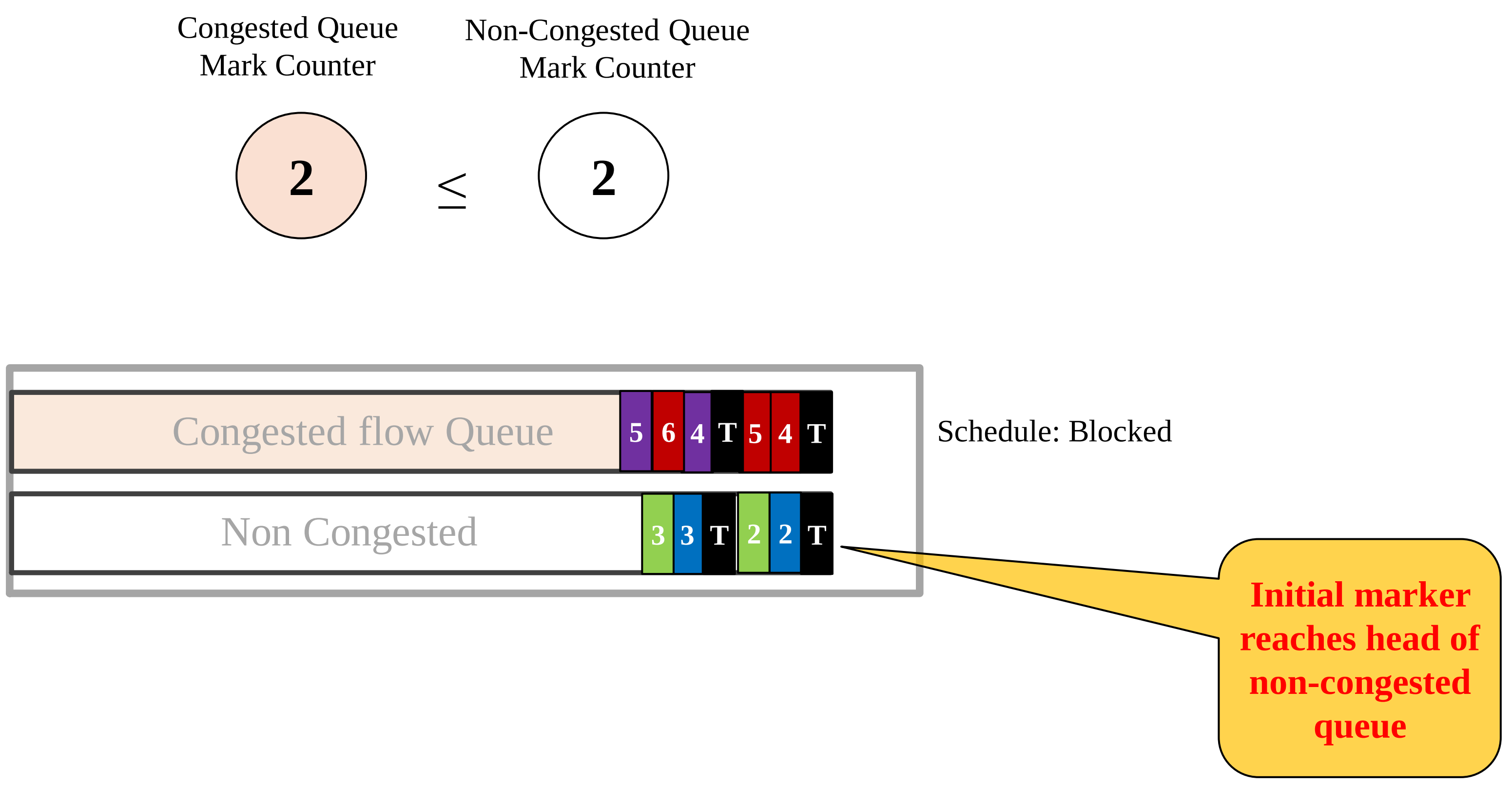


Congested Queue
Mark Counter
Non-Congested Queue
Mark Counter
1
≤
1
Purple is judged as a congested flow
4
Congested flow Queue
5
4
T
Schedule: Blocked
Non Congested
2
2
T
1
3
2
1

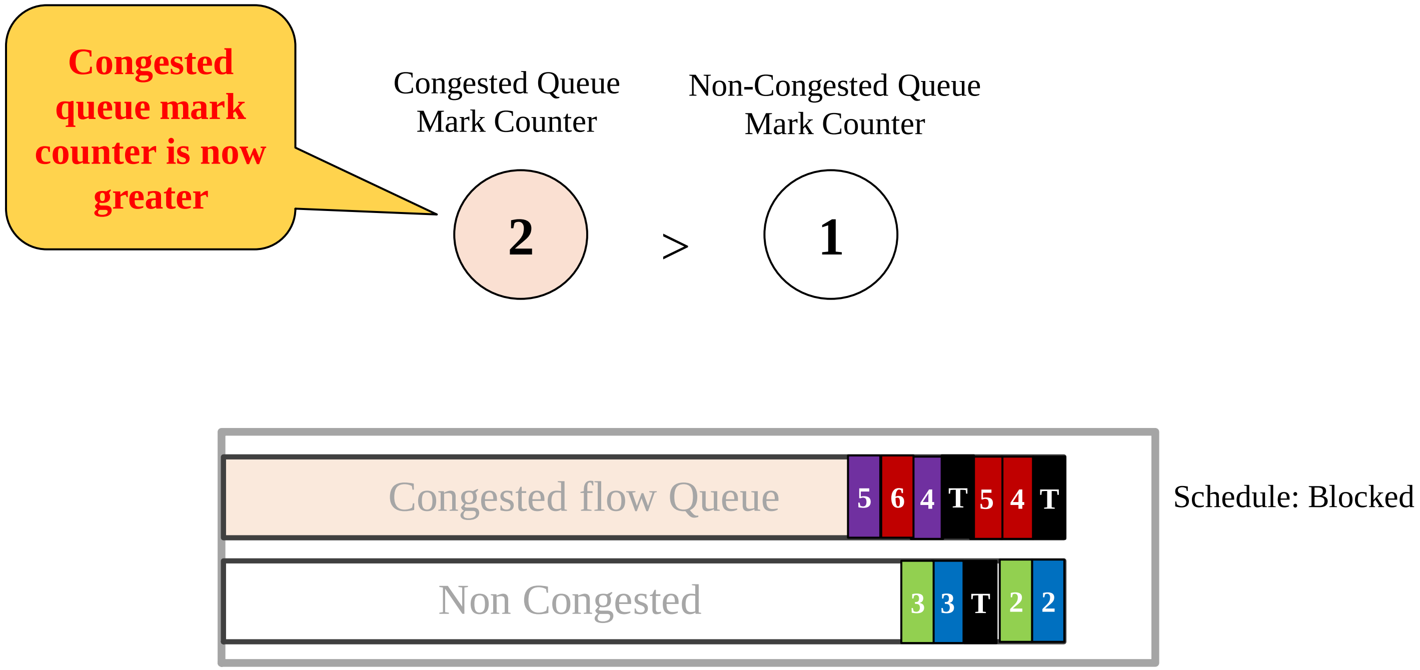


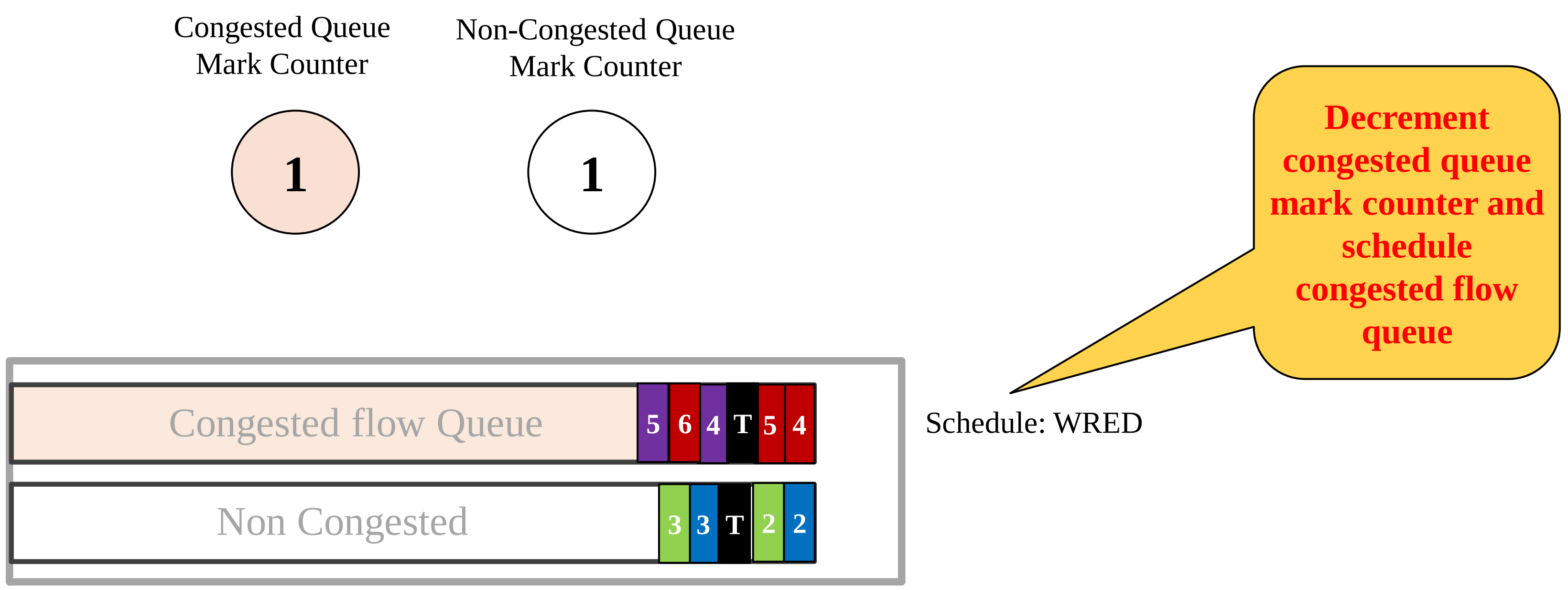
Congested Queue Mark Counter
Non-Congested Queue Mark Counter
2
≤
2
Set markers and queue packet in congested queue
Congested flow Queue
4 T 5 4 T
Non Congested
T 2 2 T 1 3 2 1
Schedule: Blocked


Congested Queue Mark Counter
Non-Congested Queue Mark Counter
2
≤
2
Congested flow Queue
6 4 T 5 4 T
Non Congested
T 2 2 T 1 3 2
Schedule: Blocked
Non-congested queue drains while congested queue is blocked

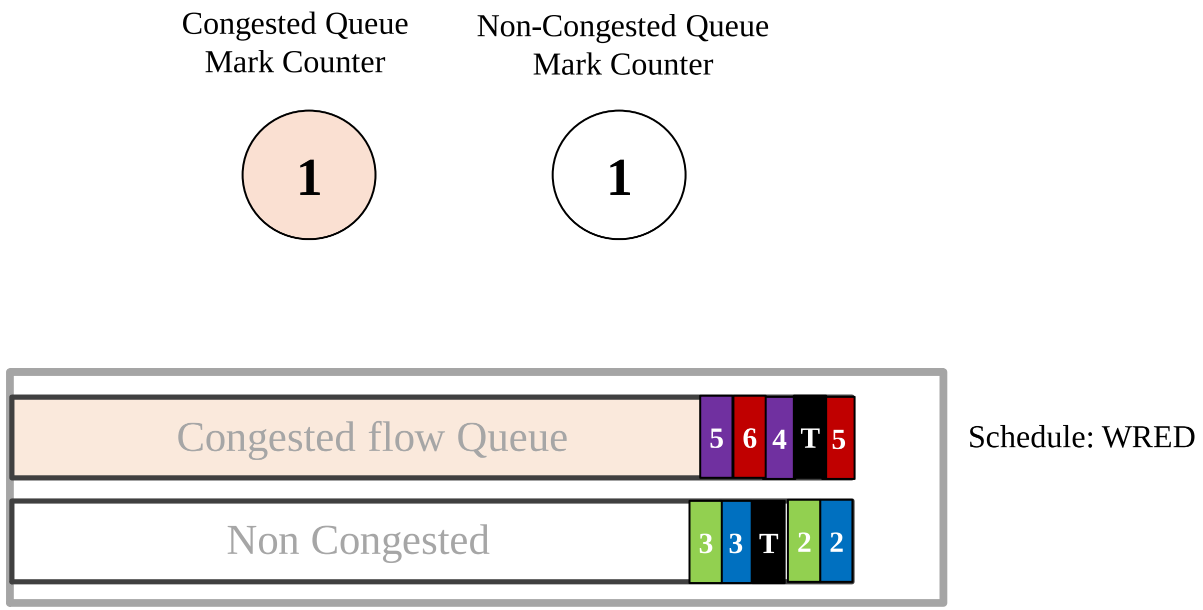


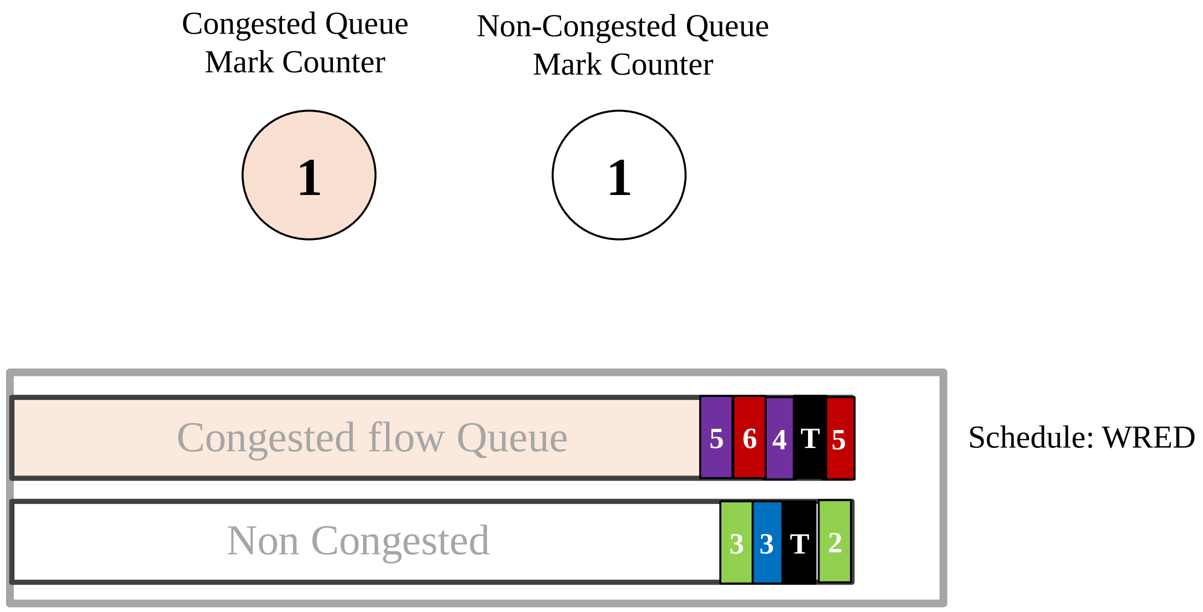
Congested Queue
Mark Counter
2
≤
Non-Congested Queue
Mark Counter
2
Congested flow Queue
5 6 4 T 5 4 T
Non Congested
3 3 T 2 2 T
Schedule: Blocked
Initial marker reaches head of non-congested queue

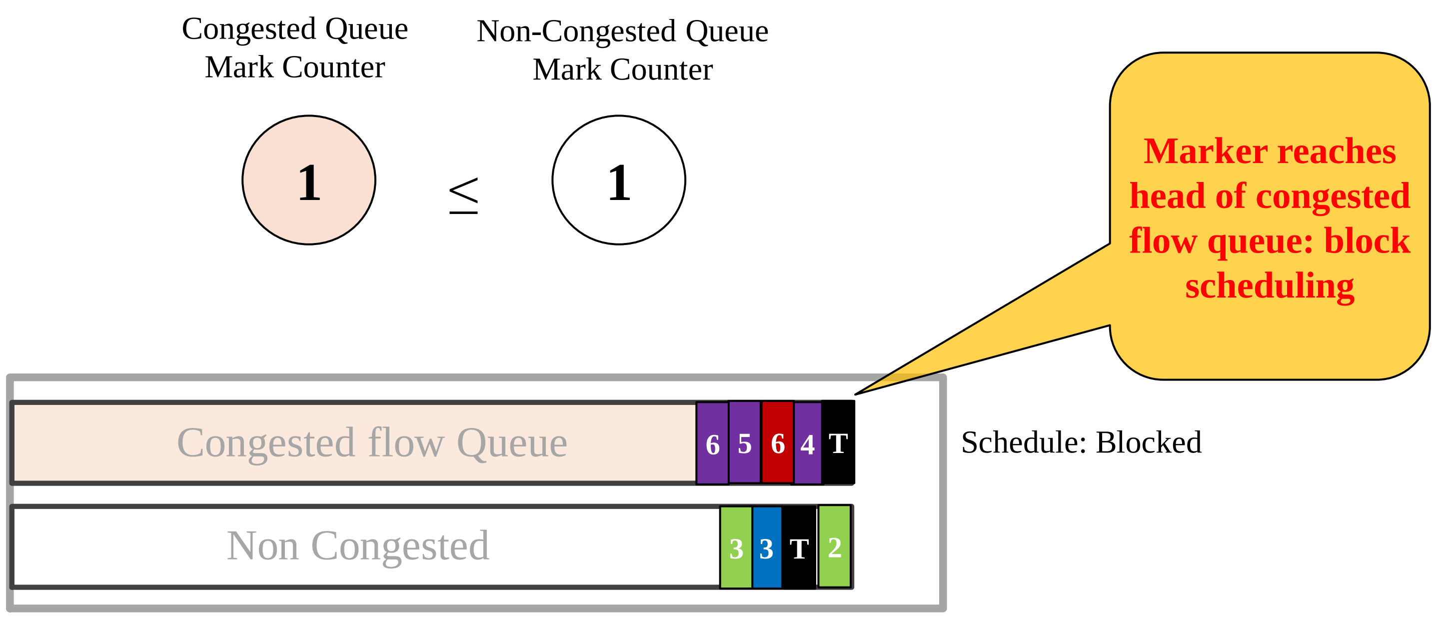

Congested queue mark counter is now greater
Congested Queue Mark Counter
2
>
Non-Congested Queue Mark Counter
1
Congested flow Queue
5 6 4 T 5 4 T
Non Congested
3 3 T 2 2
Schedule: Blocked

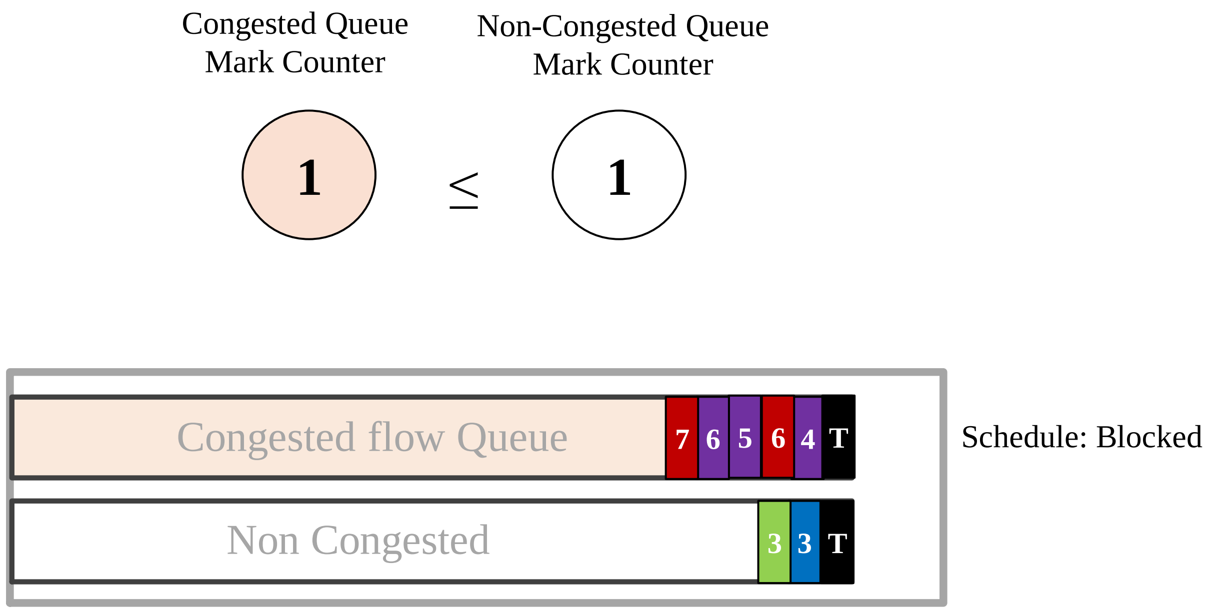


Congested Queue
Mark Counter
1
Non-Congested Queue
Mark Counter
1
Decrement congested queue mark counter and schedule congested flow queue
Congested flow Queue
5 6 4 T 5 4
Non Congested
3 3 T 2 2
Schedule: WRED

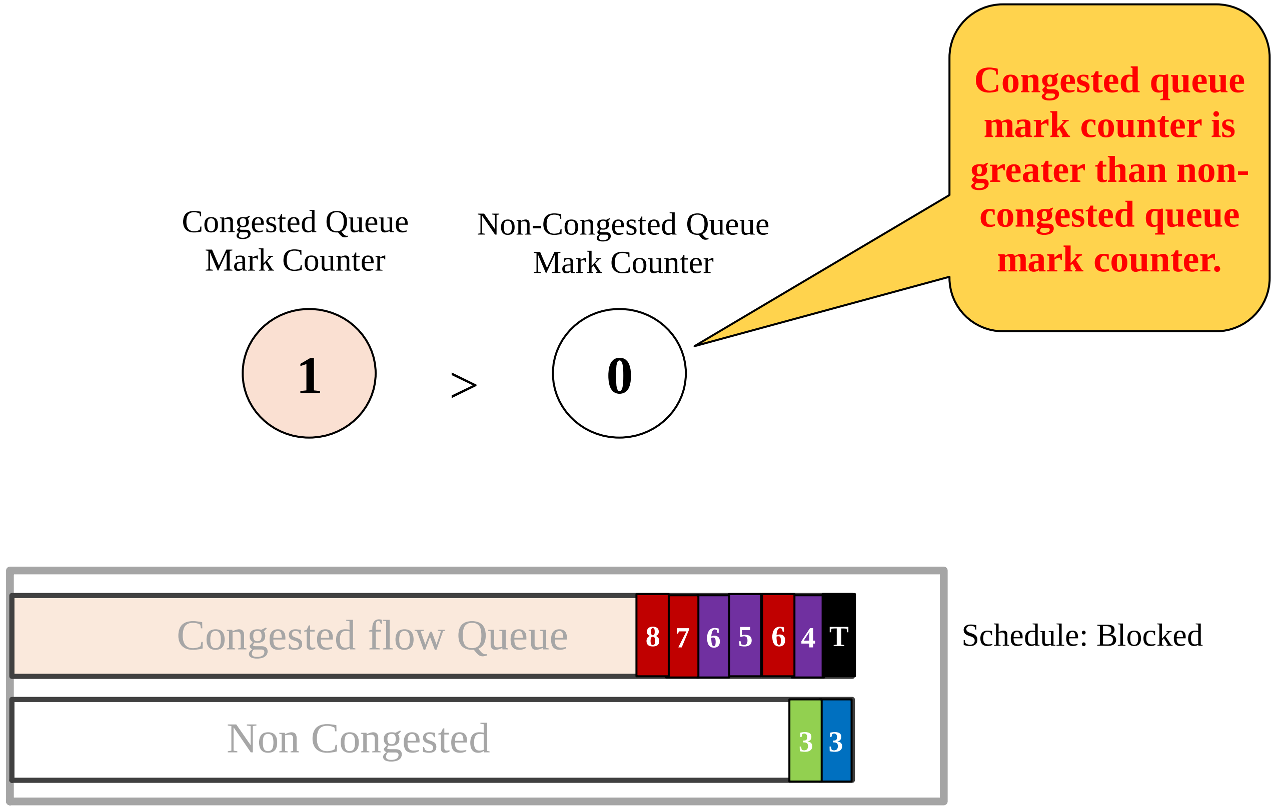

Congested Queue
Mark Counter
1
Non-Congested Queue
Mark Counter
1
Congested flow Queue
5
6
4
T
5
Non Congested
3
3
T
2
2
Schedule: WRED

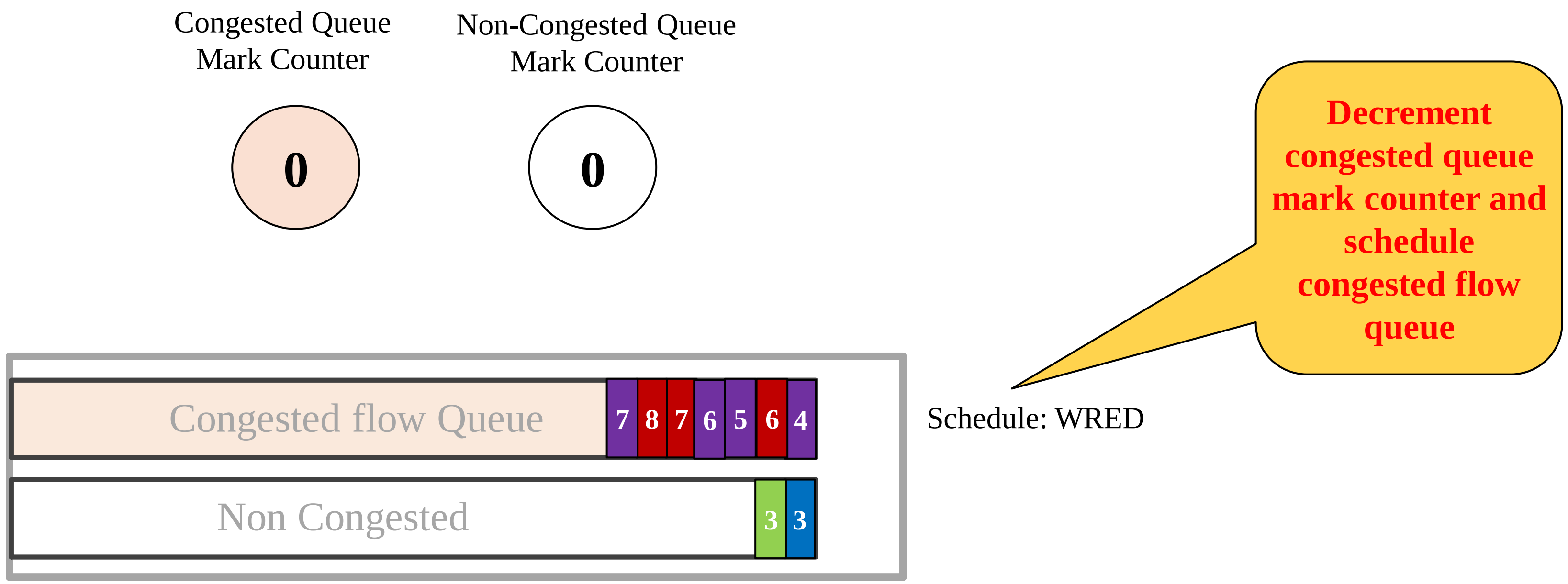

Congested Queue
Mark Counter
1
Non-Congested Queue
Mark Counter
1
Congested flow Queue
5
6
4
T
5
Non Congested
3
3
T
2
Schedule: WRED

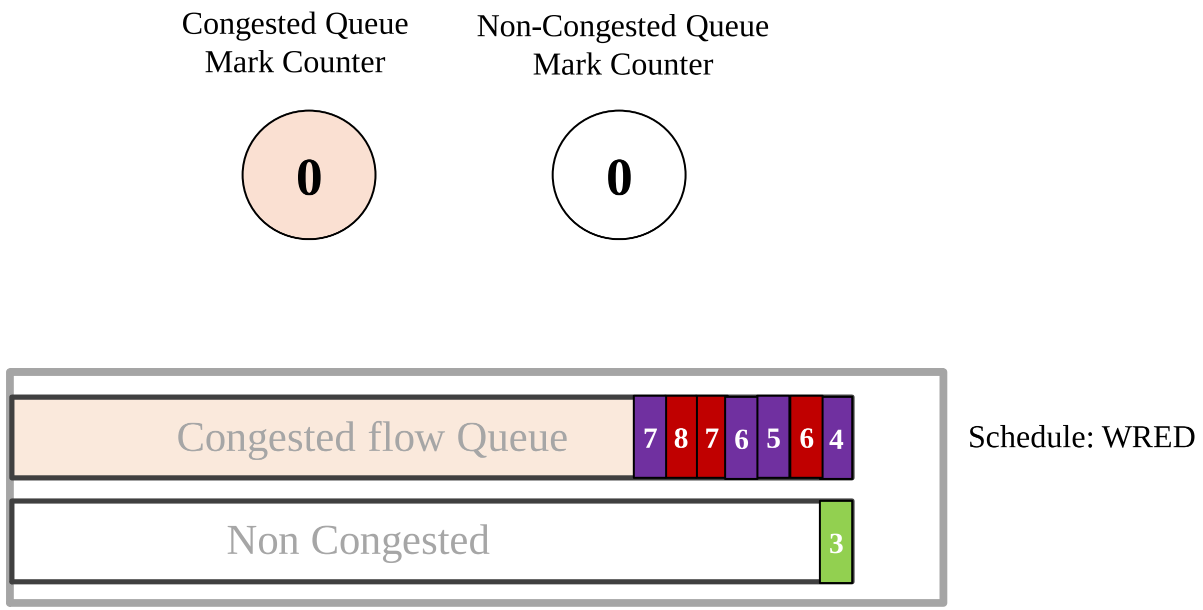

Congested Queue Mark Counter
1
≤
Non-Congested Queue Mark Counter
1
Marker reaches head of congested flow queue: block scheduling
Congested flow Queue
6 5 6 4 T
Non Congested
3 3 T 2
Schedule: Blocked



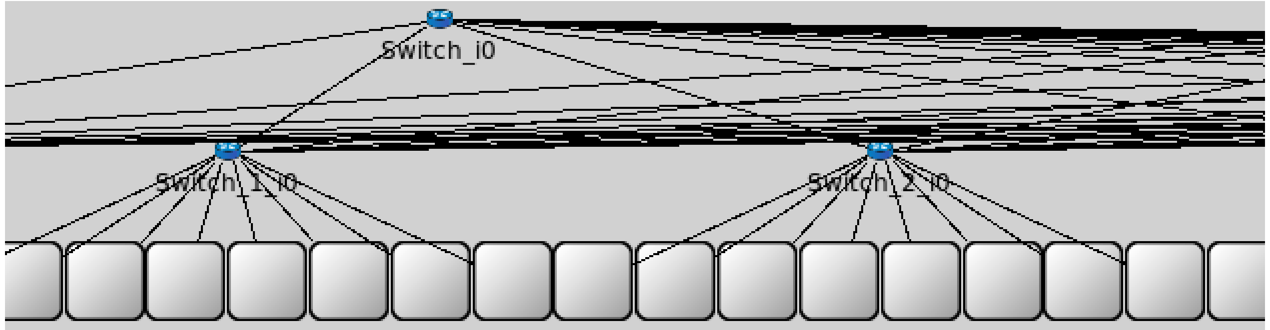
Congested Queue
Mark Counter
1
≤
Non-Congested Queue
Mark Counter
1
Congested flow Queue
7
6
5
6
4
T
Non Congested
3
3
T
Schedule: Blocked

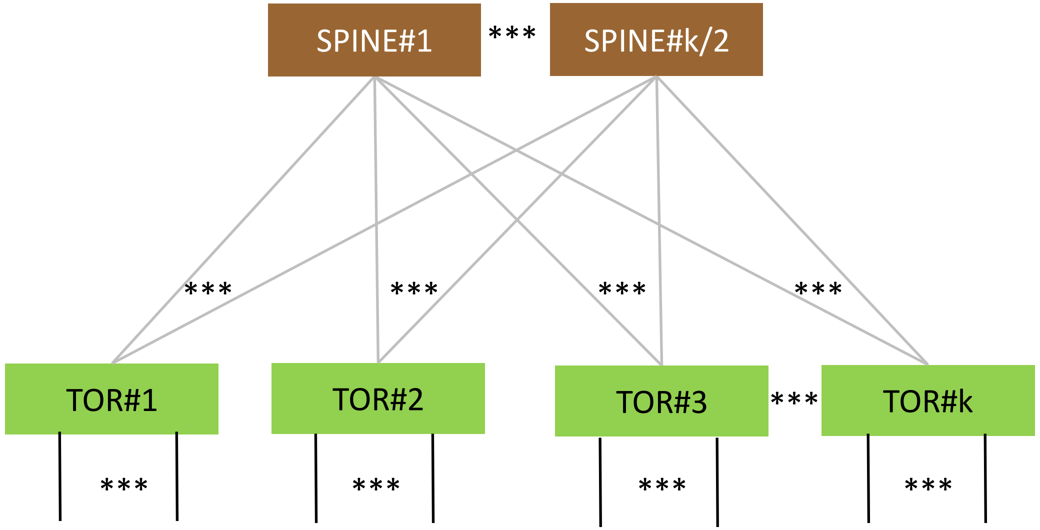

Congested queue mark counter is greater than non-congested queue mark counter.
Congested Queue Mark Counter
Non-Congested Queue Mark Counter
1
>
0
Congested flow Queue
8 7 6 5 6 4 T
Non Congested
3 3
Schedule: Blocked


Congested Queue Mark Counter
Non-Congested Queue Mark Counter
0
0
Decrement congested queue mark counter and schedule congested flow queue
Congested flow Queue
7 8 7 6 5 6 4
Non Congested
3 3
Schedule: WRED


Congested Queue
Mark Counter
0
Non-Congested Queue
Mark Counter
0
Congested flow Queue
7
8
7
6
5
6
4
Non Congested
3
Schedule: WRED

Simulation Set-up



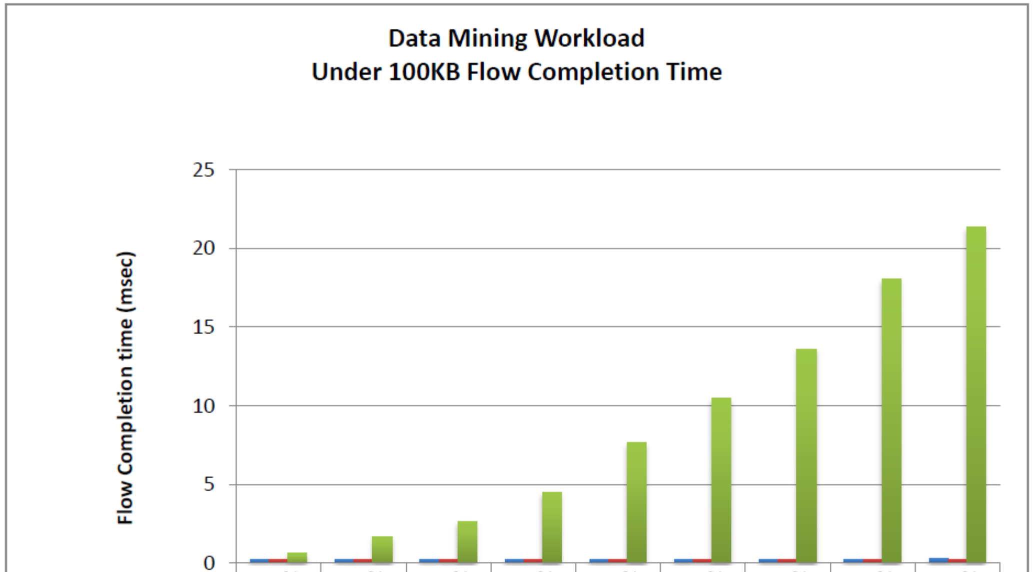


- OMNET++ Platform
- 2 Tier CLOS： 100G interface with 200ns of link latency 200ns(about 40m)
- Scale： 128 ~ 1152 servers, 24 ~ 72 switches
- Traffic Patterns:
  - Several regional all to all with some persistent incast
  - Flow size distribution is from 5 different real data center applications:
    - Enterprise IT, WebServer, Hadoop, Data Mining, Cache-Follower
- Compared Solutions:
  - PFC+ECN with CI: Congestion Isolation is implemented along with PFC+ECN
  - PFC+ECN without CI: Just PFC+ECN
  - All solutions include small flow prioritization mechanism

Traffic Pattern 1： Data Mining

Mice: Under 100-KB Flow Completion Times

The most pronounced difference in Flow Completion Times between the Cisco and Arista switches occurs when we focus on the smallest flow sizes, under 100 Kilobytes, as shown in the following graph.

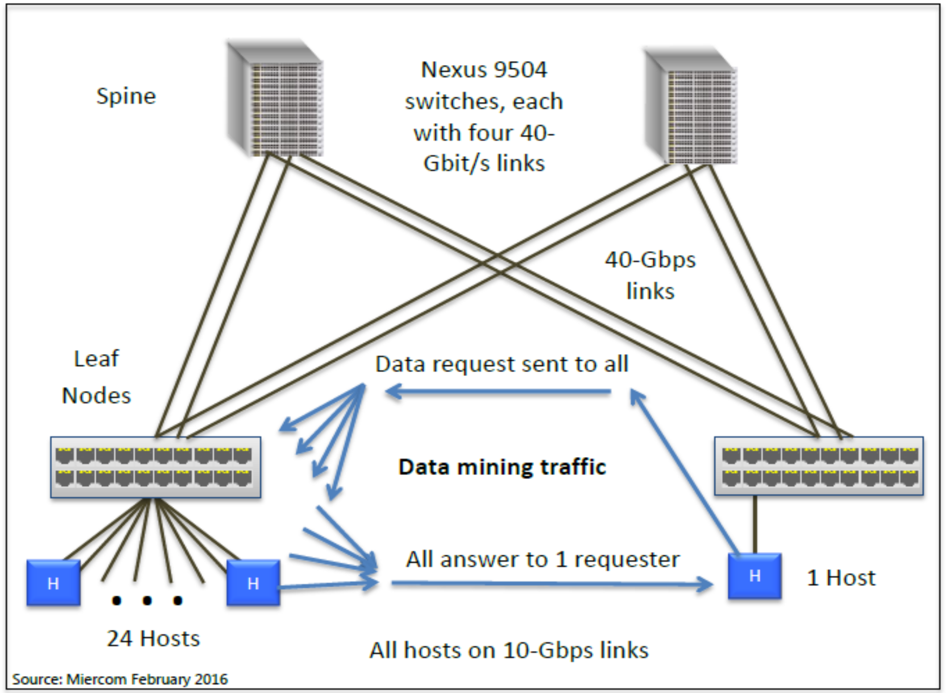


	20%	30%	40%	50%	60%	70%	80%	90%	95%
Nexus 92160YC-X	0.24	0.24	0.24	0.24	0.24	0.24	0.25	0.25	0.30
Nexus9272Q	0.24	0.24	0.24	0.24	0.24	0.24	0.25	0.25	0.25
7280SE-72	0.62	1.68	2.61	4.52	7.68	10.50	13.62	18.08	21.37

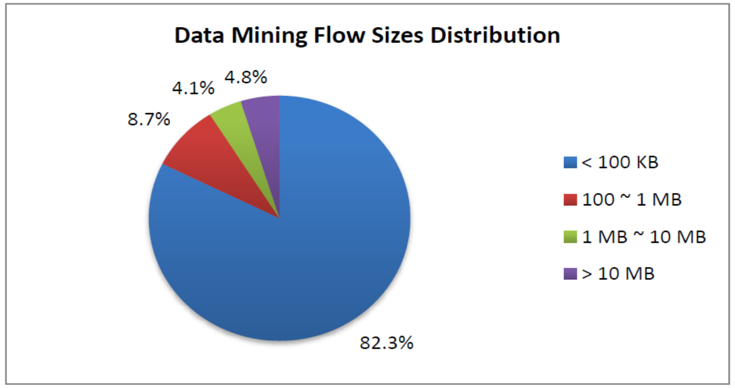
Source: Miercom February 2016 Traffic Load (% line rate)

(Source： Miercom/Cisco, 2016, Speeding Applications in Data Center Networks)

- The mainstream definition of the Mice: Flows under 100KB
- In Data Mining workload, the proportion of the Mice is about 82%



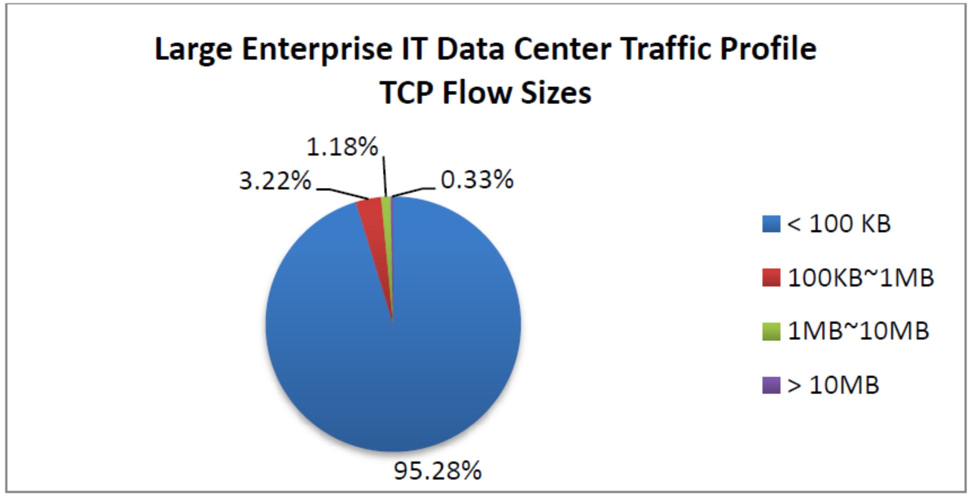
The traffic applied in the testing was modelled to reflect real-world data-mining applications. Elephant flows comprised just about 5 percent of the number of flows, but 95 percent of the total data volume. The following chart shows the TCP flow size distribution for this data-mining application. Custom scripts were written to create random flows, which collectively followed distribution based on these percentages.

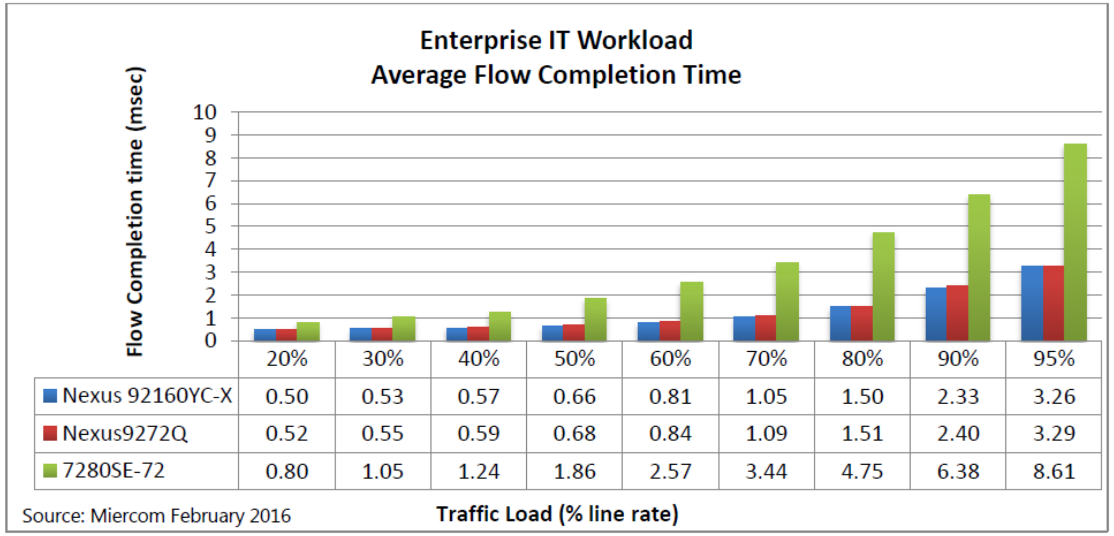


The key metric sought from this testing was Flow Completion Time, or FCT – the time it took each flow to finish. Flow completion time is used instead of typical metrics such as: average link

Traffic Pattern 2： Enterprise IT

In addition to the data mining traffic profile, we also ran the tests with a typical large enterprise IT data center traffic profile that we gathered from analysis of a real data center. The following graph shows the flow sizes distribution.





	20%	30%	40%	50%	60%	70%	80%	90%	95%
Nexus 92160YC-X	0.50	0.53	0.57	0.66	0.81	1.05	1.50	2.33	3.26
Nexus9272Q	0.52	0.55	0.59	0.68	0.84	1.09	1.51	2.40	3.29
7280SE-72	0.80	1.05	1.24	1.86	2.57	3.44	4.75	6.38	8.61

With the flow sizes and load distribution in the typical large enterprise IT data center traffic profile, the overall average flow completion time on the Cisco switches were 30%~60% better than that on the Arista switch, varying with the traffic load. The heavier the traffic load, the more advantageous the Cisco's results were in comparison to those of the Arista switch.

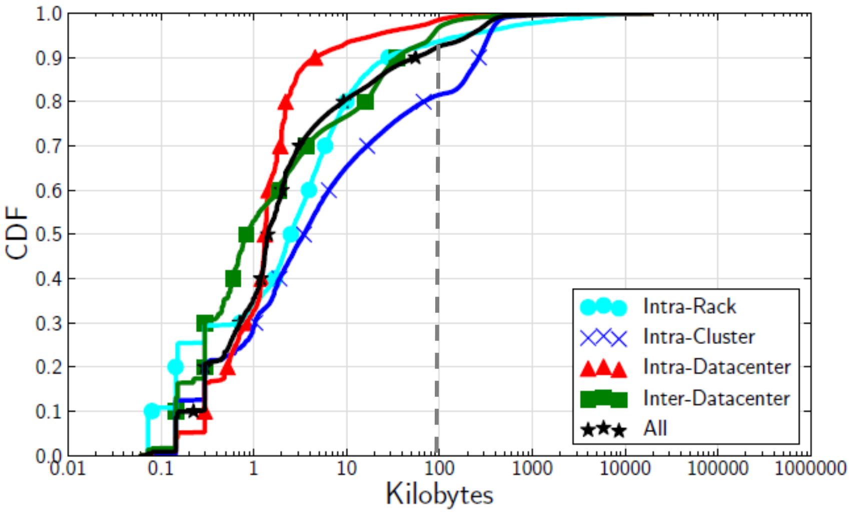
(Source： Miercom/Cisco, 2016, Speeding Applications in Data Center Networks)

- In Enterprise IT workload, the proportion of the Mice is much bigger, about 95%

- Average Flow Completion Time is a popular metric to evaluate network performance.

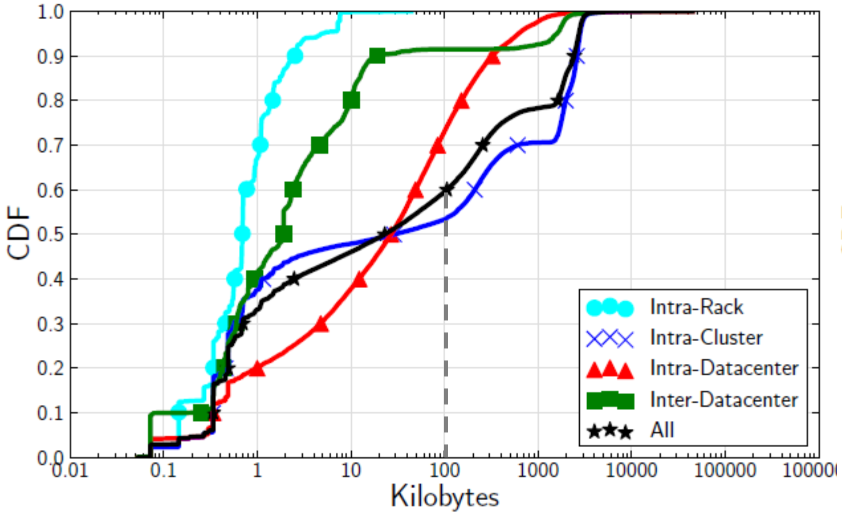
Traffic Pattern 3~5

Traffic Pattern 3



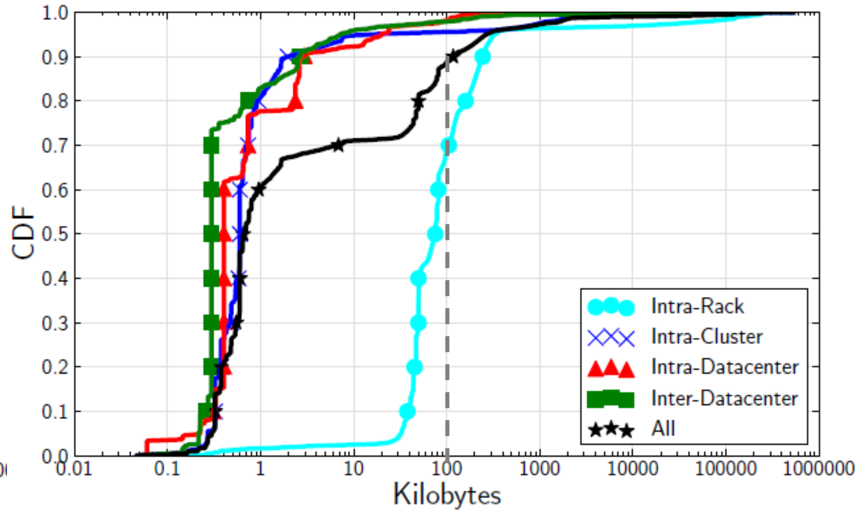
(a) Web servers

Traffic Pattern 4



(b) Cache follower

Traffic Pattern 5

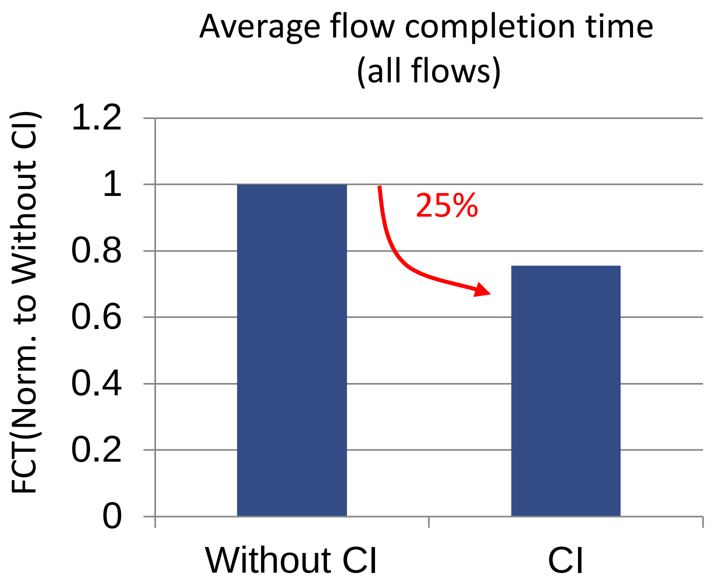


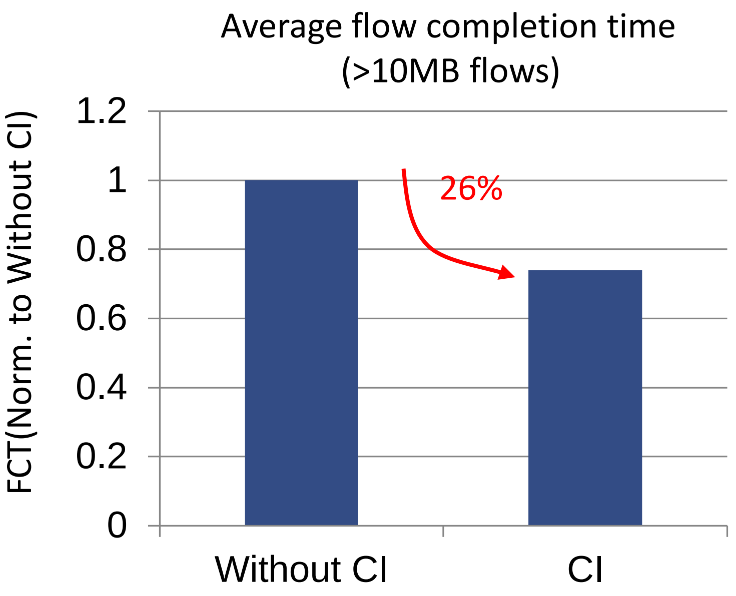
(c) Hadoop

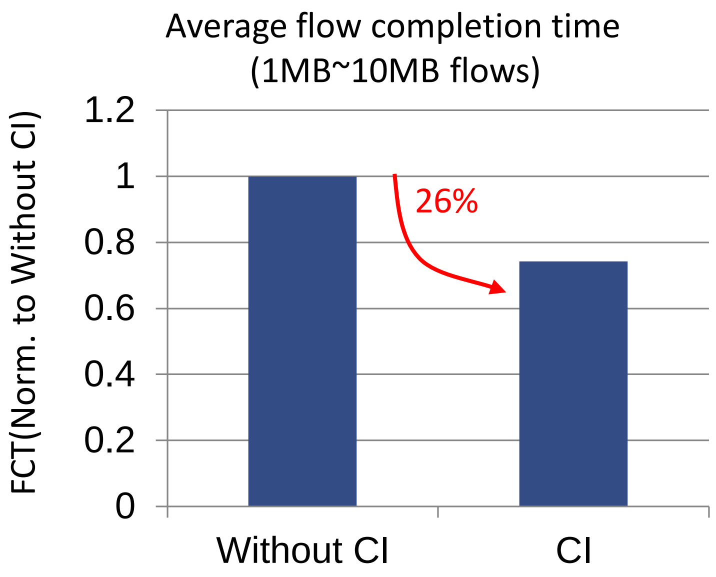
(Source： Facebook, 2015, Inside the Social Network's (Datacenter) Network)

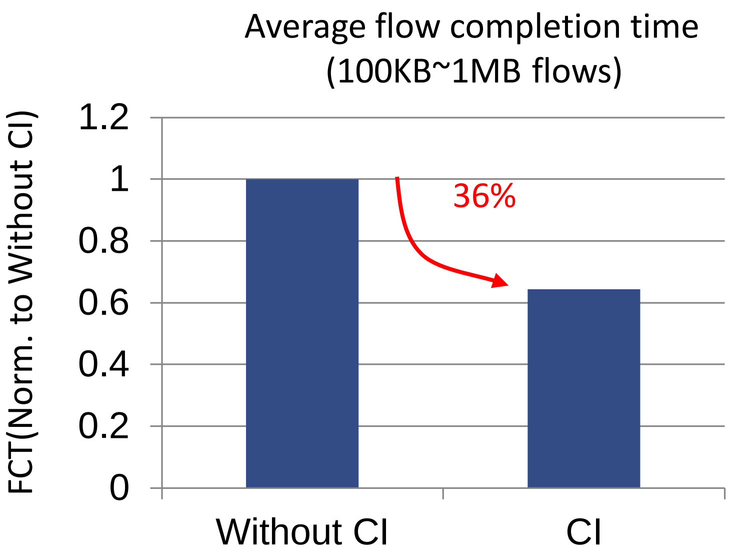
- In Web Server workload, the proportion of the Mice is about 92%
- In Cache Follower workload, the proportion of the Mice is about 60%
- In Hadoop workload, the proportion of the Mice is about 90%

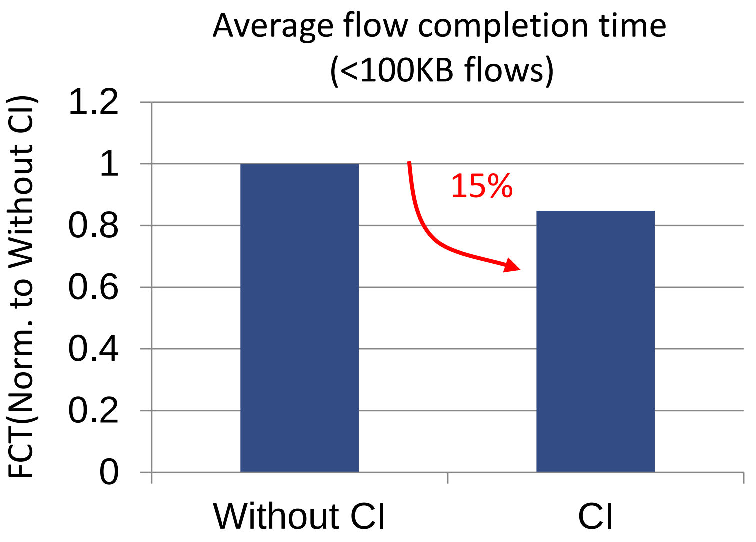
PFC+ECN with CI VS. PFC+ECN without CI





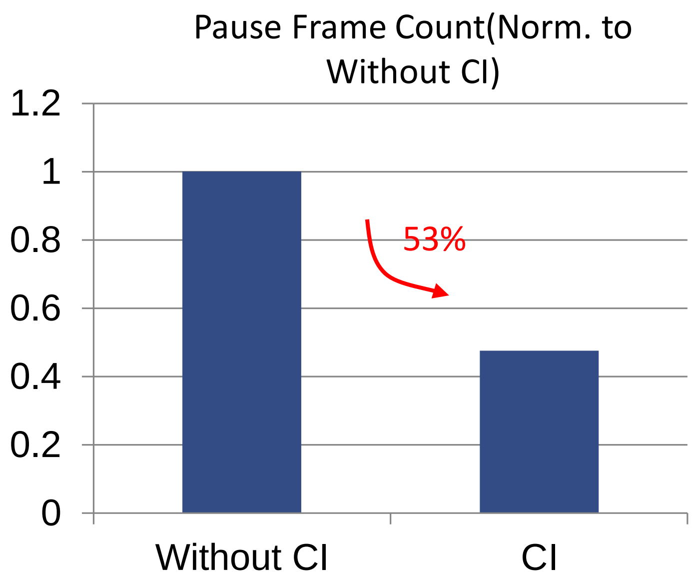


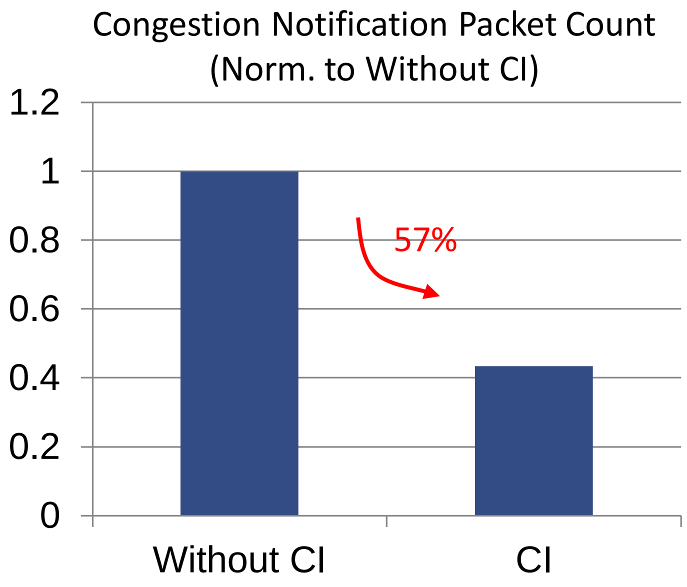


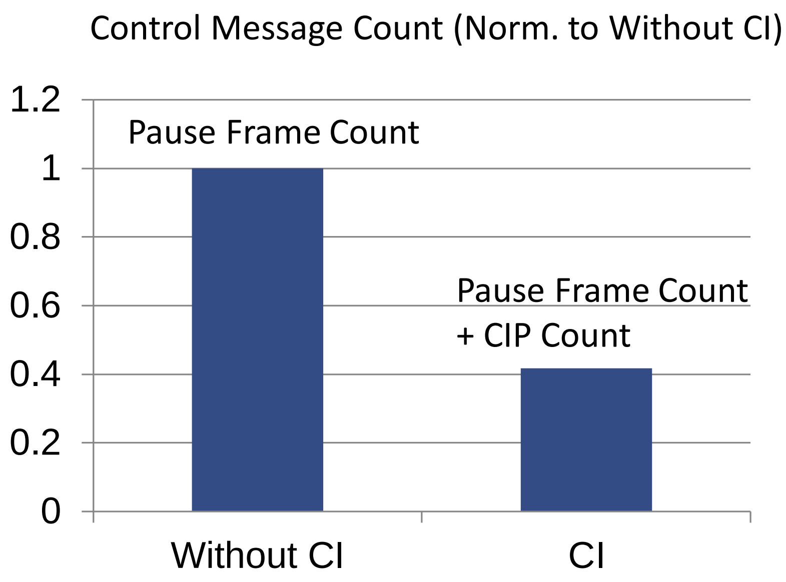


- CI reduces the count of PAUSE Frames sent to NICs of servers, so it can alleviate the HOL Blocking of the NIC, which can improve the performance of mice flows.
- In the PFC+ECN without CI, we also prioritize the mice.

Why PFC+ECN with CI outperforms PFC+ECN without CI





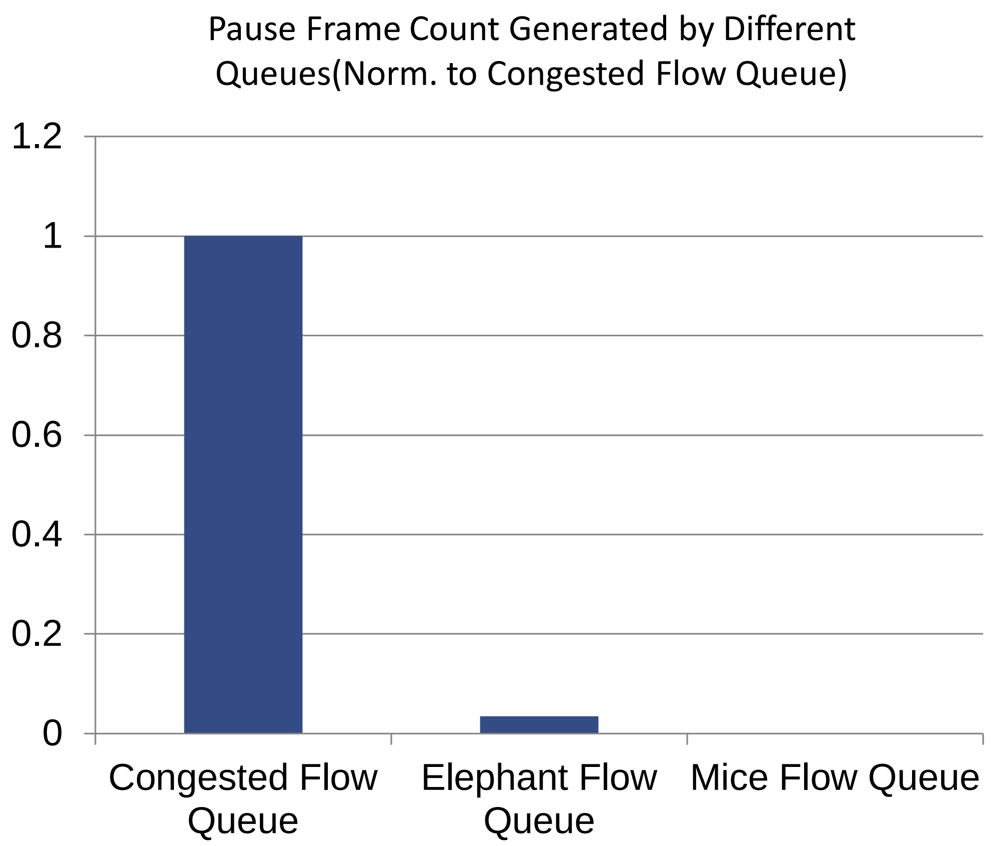


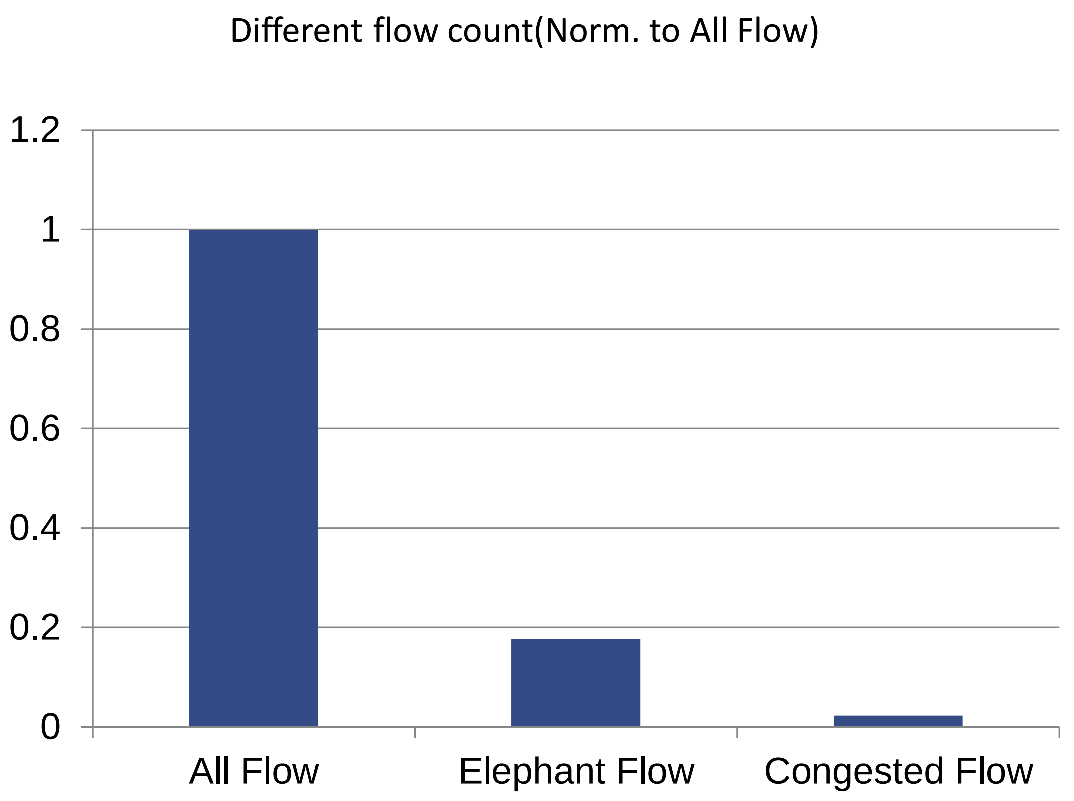
- CI reduces the pause frame count by 53%.

- CI reduces the CNP count by 57%.

- The count of new control message generated by CI is much less than the count it reduces the count of Pause frames.
- It has the same order-of-magnitude with large flow count.

Why PFC+ECN with CI outperforms PFC+ECN without CI

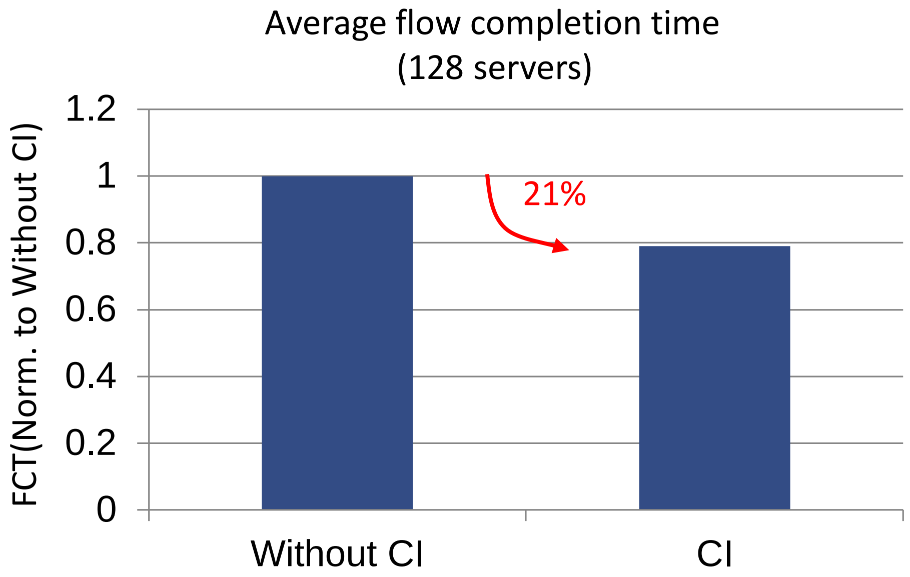


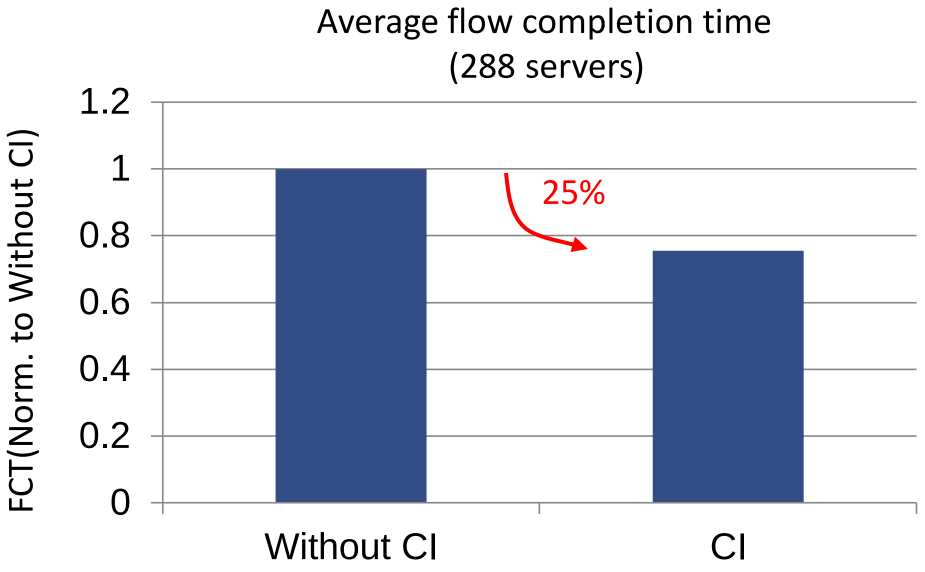


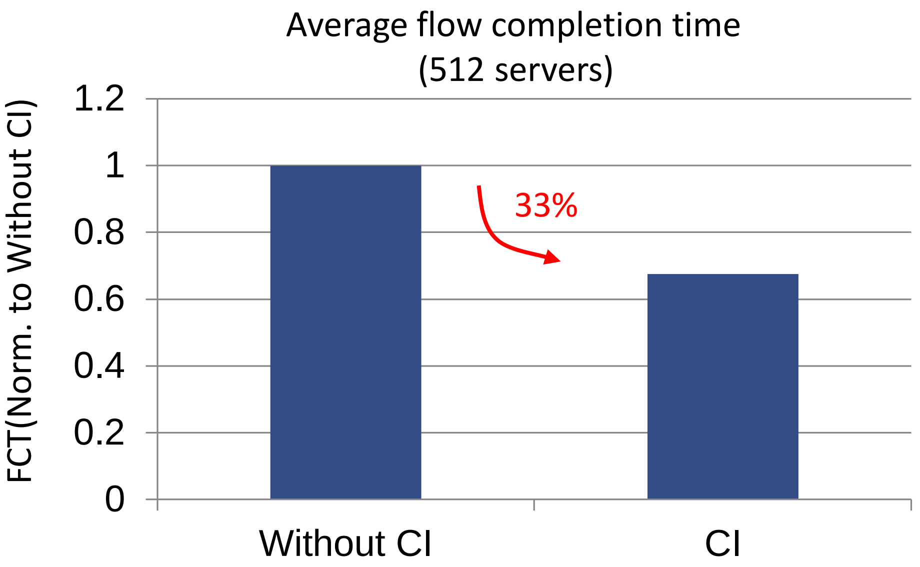
- 96.6% of the pause frames are generated by congested flow queues

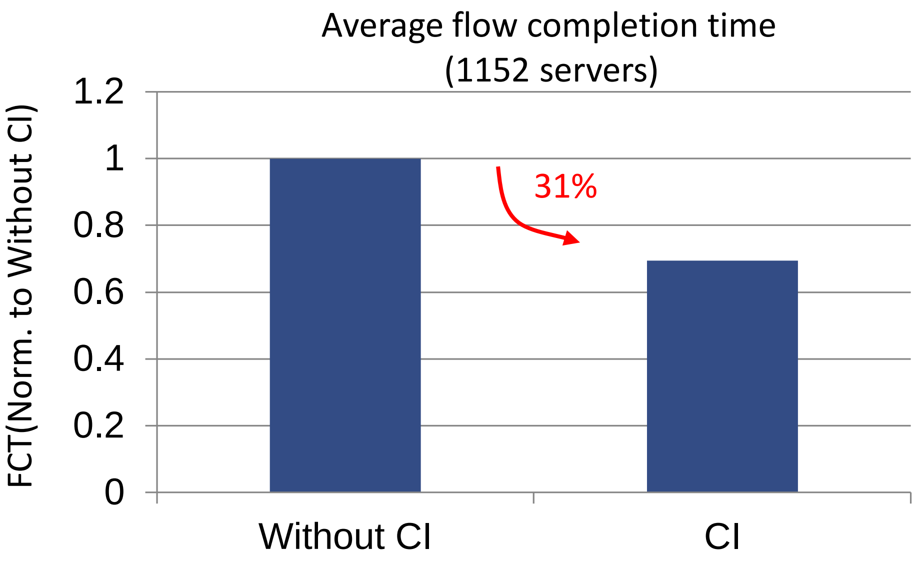
- The count of isolated flows is quite small. In our simulation with 22188 flows and 1152 server nodes. The proportion is 2% for total flows , and 12% for large flows.
- So the HOLB only occurs among the congested flows

Comparison for different scale

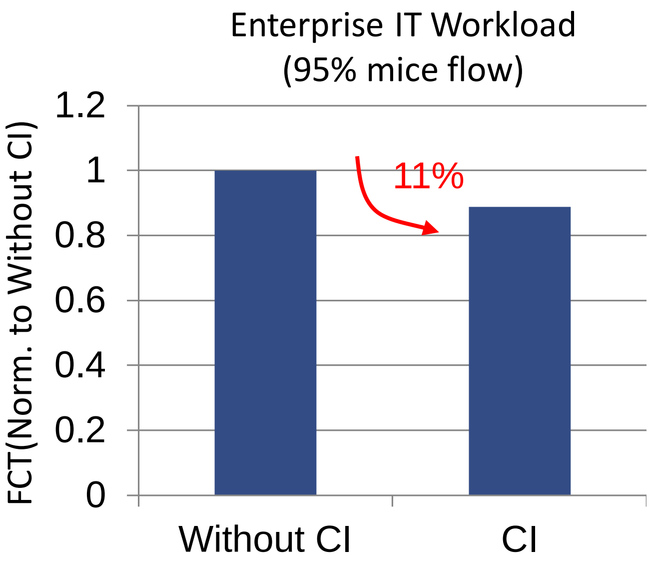


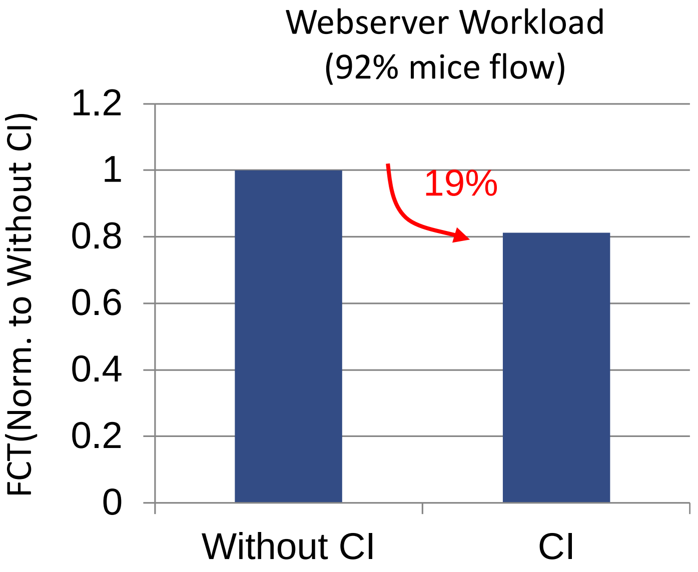


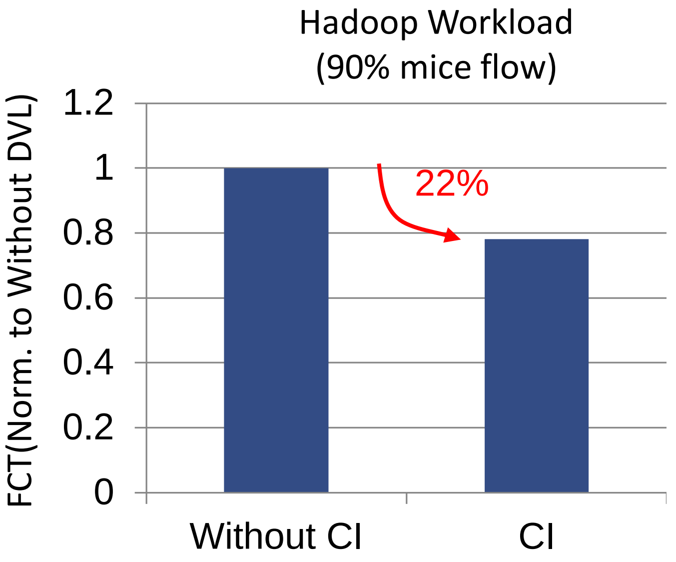


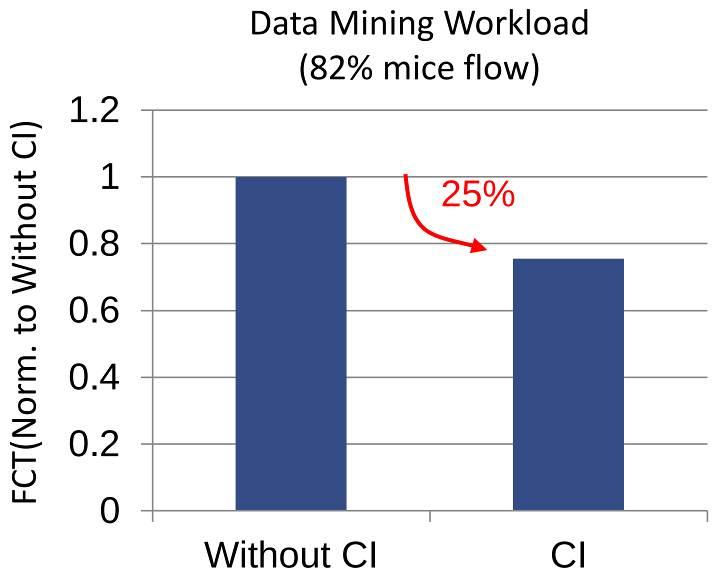


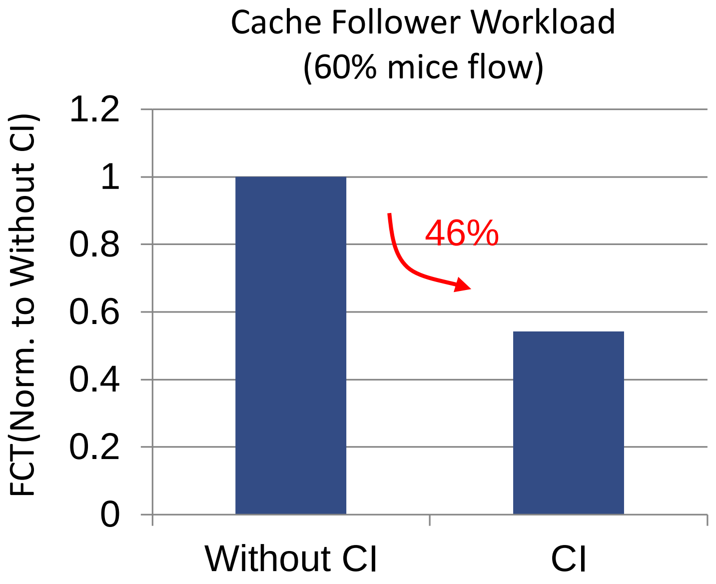
Comparison for different workload – Flow Completion Times











Summary

- Current data center design will be challenged to support the needs of large scale, low-latency, lossless networks.
- Congestion Isolation provides the following benefits:
  - Supports lossless as well as low-latency
  - Mitigates Head-of-Line blocking caused by PFC
  - Improves average flow completion times
  - Reduces or eliminates the need for PFC on non-congested flow queues
- Next Steps
  - Call for interest in creating a project
  - Respond to comments and feedback

Thank you

www.huawei.com

Options for reliable CIP transmission

- Send several (such as 3) CIPs in succession to provide redundancy.
- Send CIPs periodically. Upstream send an ACK when it has received a CIP. Downstream stop to send CIP until it has received an ACK.
- Send CIPs periodically. Upstream mark the subsequent packets when it has received a CIP. Downstream will know that the upstream has isolated successfully according to the marked packets and stop sending CIPs.

Buffer requirement of CI

- CI makes intelligent use of buffer memory.
- CI can leverage the existing low priority queue to isolate congested flows, so it won’t increase the switch buffer size.
- Theoretically, CI can reduce the switch buffer requirement because it alleviates the HOLB and decreases arising of queue building up (need to research further).