# No Silver Bullet: Extending SDN to the Data Plane

Anirudh Sivaraman, Keith Winstein, Suvinay Subramanian, and Hari Balakrishnan

MIT CSAIL & Wireless@MIT

http://web.mit.edu/anirudh/www/sdn-data-plane.html

# The Data Plane is continuously evolving.

- Progression of queue-management/scheduling algorithms:
  - WFQ, SFQ, RED, BLUE, ECN, XCP, RCP, CoDel, pFabric
- Each scheme wins in its own evaluation.
- Tacit belief in knobless queue management/scheduling

# Yet, there is no silver bullet.

- Different applications care about different things.
- Applications use different transport protocols.
- Diversity in network conditions.
- “One size fits all” is overly constraining.

# Quantifying "No Silver Bullet": Network configurations

Network Configurations:

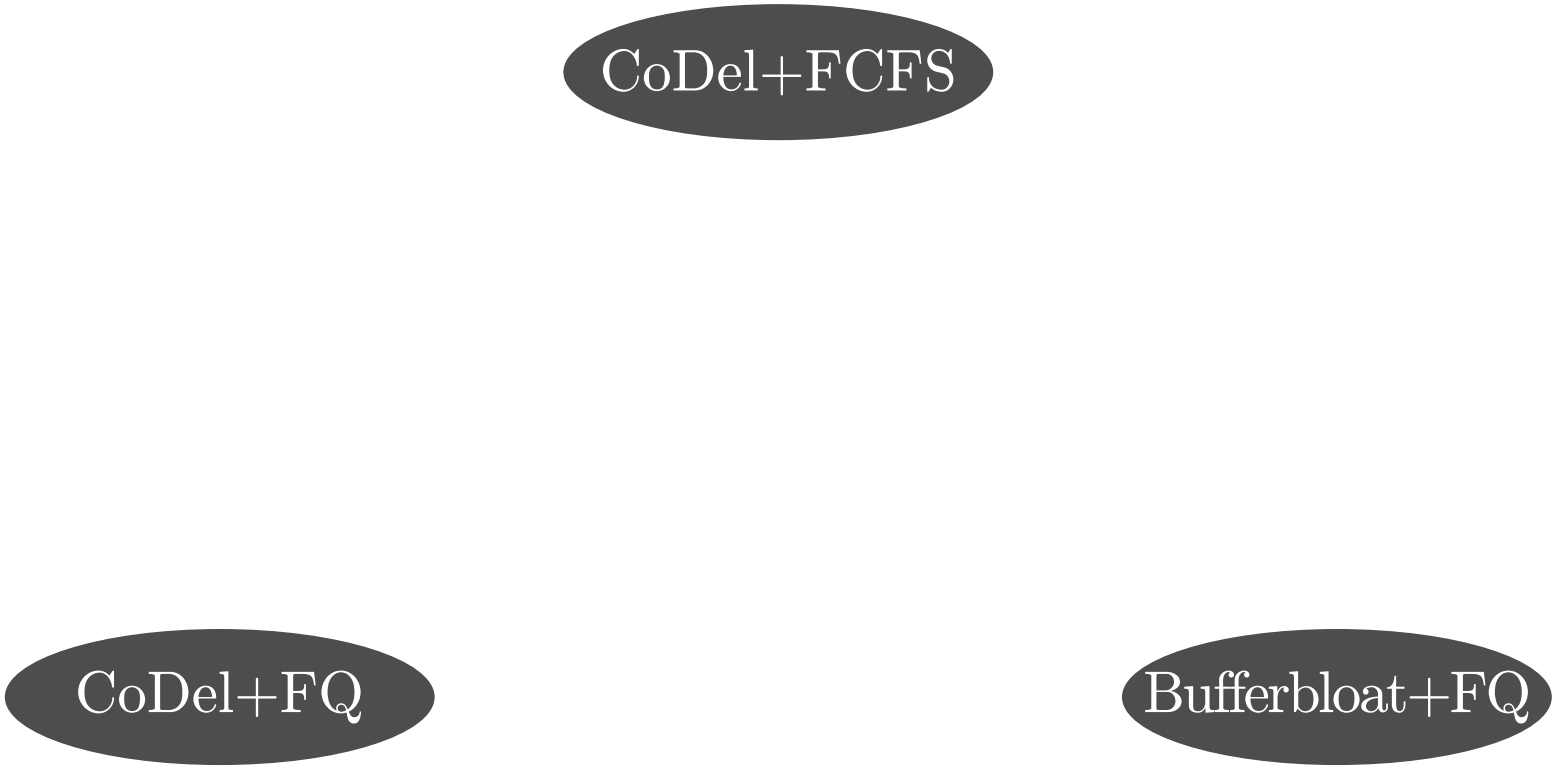
| **Network Configuration** | **Description** |
|---|---|
| **CoDel+FCFS** | CoDel running on a single shared first-come, first-served queue. |
| **CoDel+FQ** | A separate queue for each flow with an independent instance of CoDel running on each queue. Queues are serviced using fair queueing. |
| **Bufferbloat+FQ** | A separate queue for each flow with a deep buffer that doesn't drop any packets. Queues are serviced using fair queueing. |

# Quantifying "No Silver Bullet": Simulation Workloads

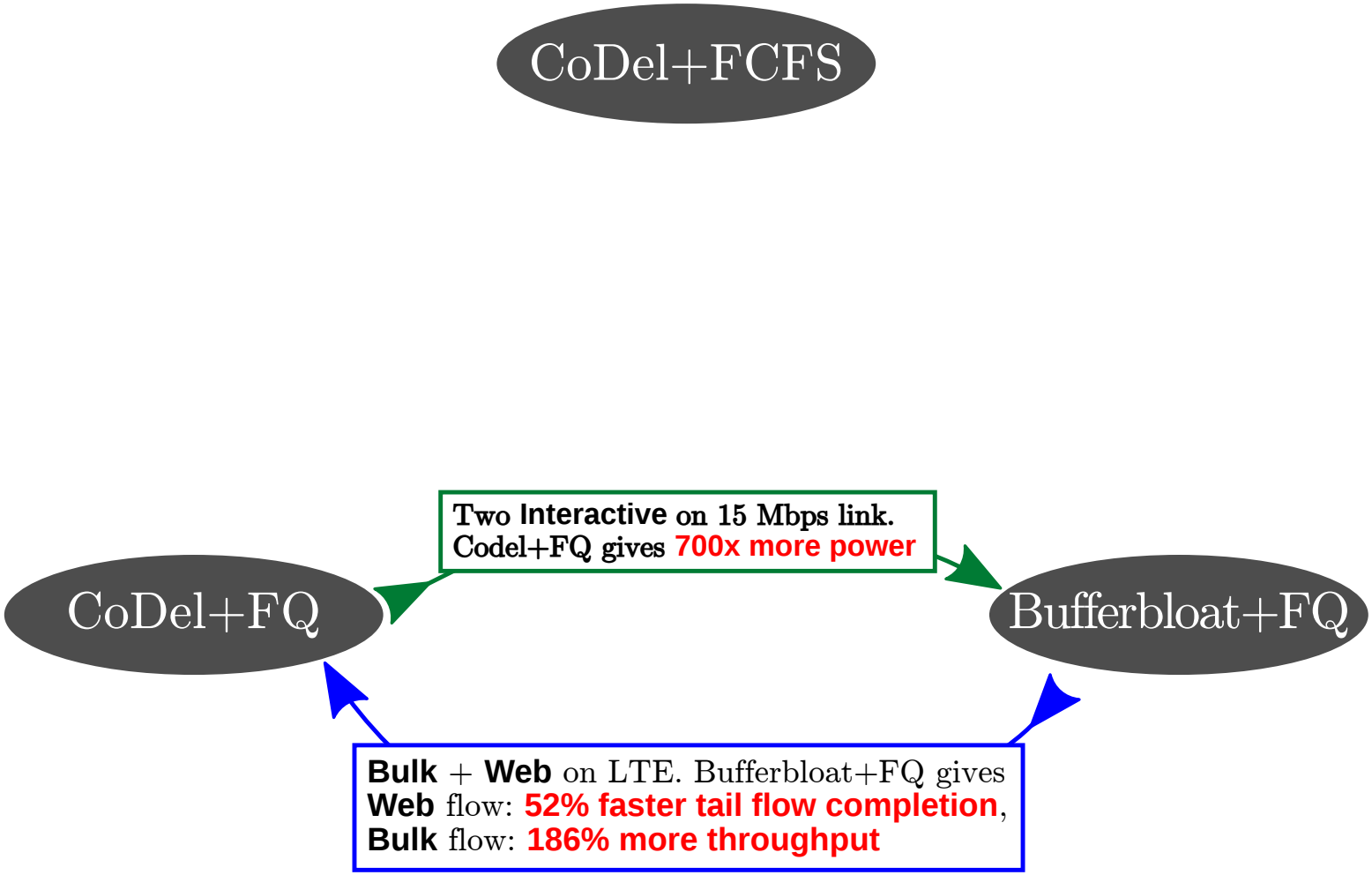
Workloads:

| Workload | Description | Objective |
|---|---|---|
| **Bulk** | Long-running TCP flow. | Maximize average throughput. |
| **Web** | Switched TCP flow that alternates between ON and OFF periods. | Minimize flow completion time at the 99.9th percentile. |
| **Interactive** | Long-running TCP flow representing a real-time interactive application. | Maximize the ratio of throughput and one-way delay, i.e., "power." |

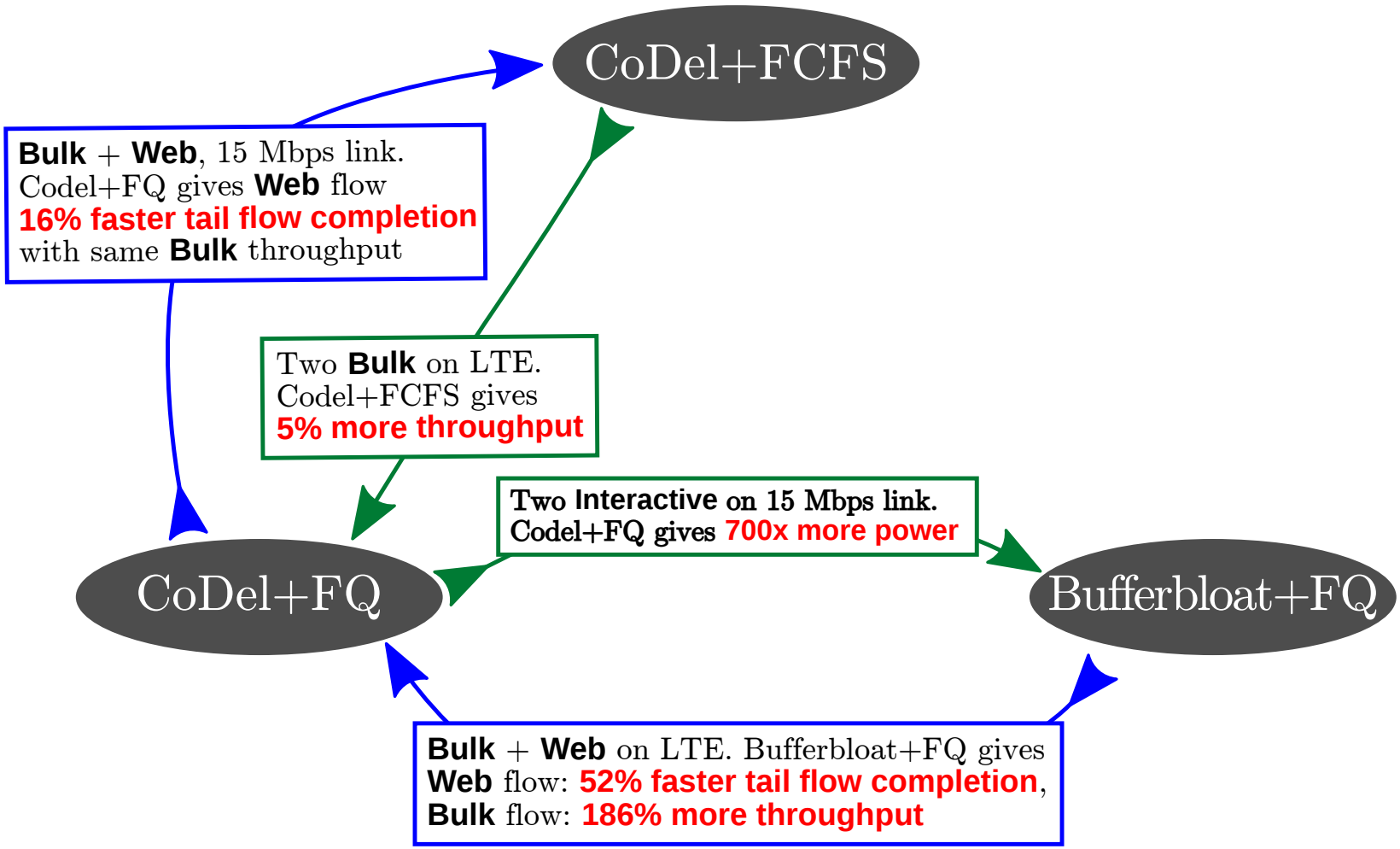
# Quantifying "No Silver Bullet"



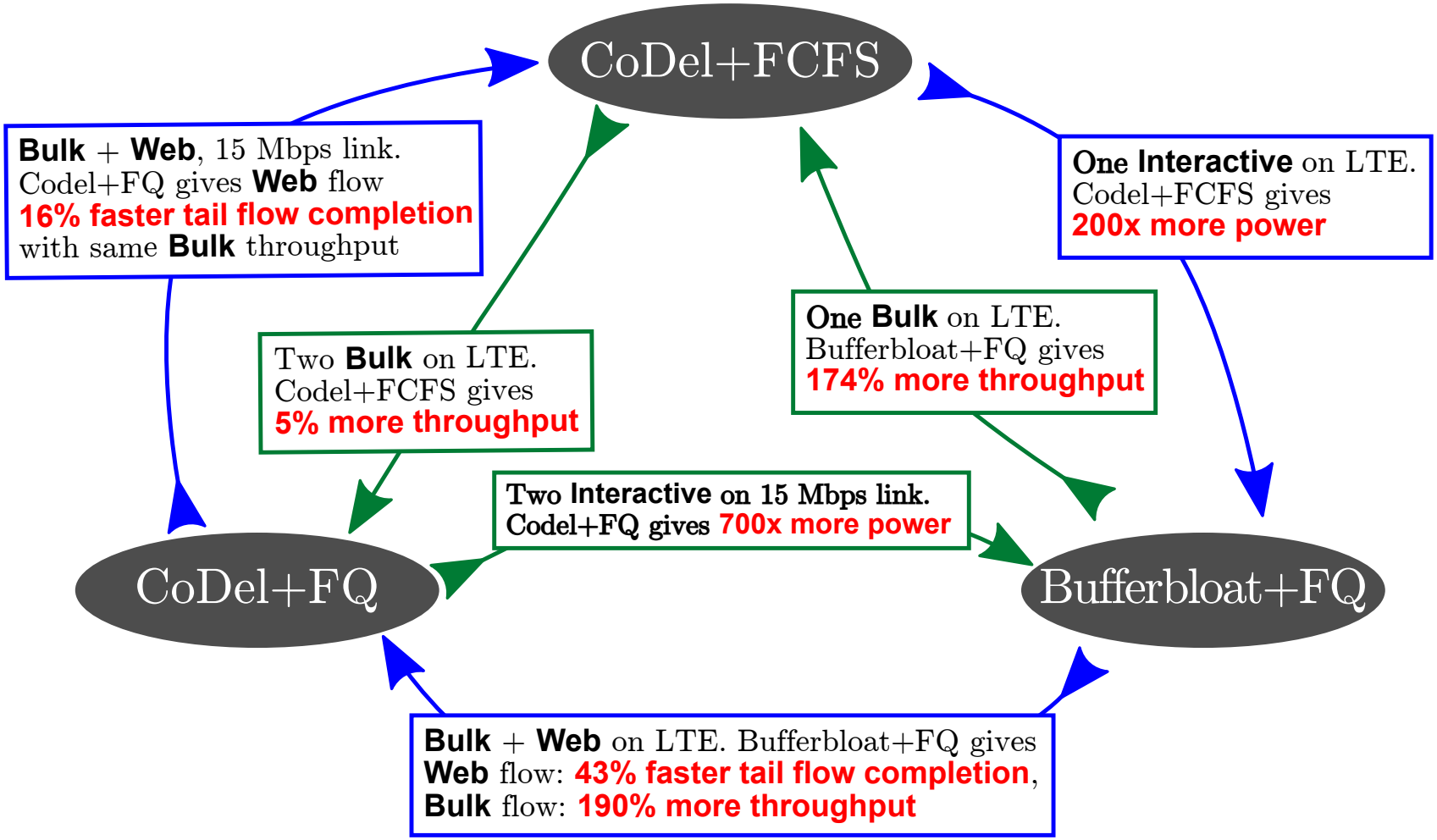
# Quantifying "No Silver Bullet"



# Quantifying "No Silver Bullet"



# Quantifying "No Silver Bullet"



No single network configuration is the best.

# Explaining cyclic preferences

- Dropping packets significantly degrades throughput.
  - Reason: Variable-rate links have an inherent delay-throughput tradeoff, unlike static links.
- FCFS is preferable to Fair Queuing in some cases.
  - Reason: When equally aggressive flows compete, they don't need protection from each other.
- Fair Queuing is required in some cases.
  - Reason: When competing flows are not equally aggressive, they need isolation from each other.

# The Solution

Flexible Switch Data Planes

- No universal queue-management/scheduling scheme
- Application demands continue to evolve.
- Networks supporting these applications will evolve as well.
- The Data Plane should support newly developed schemes.

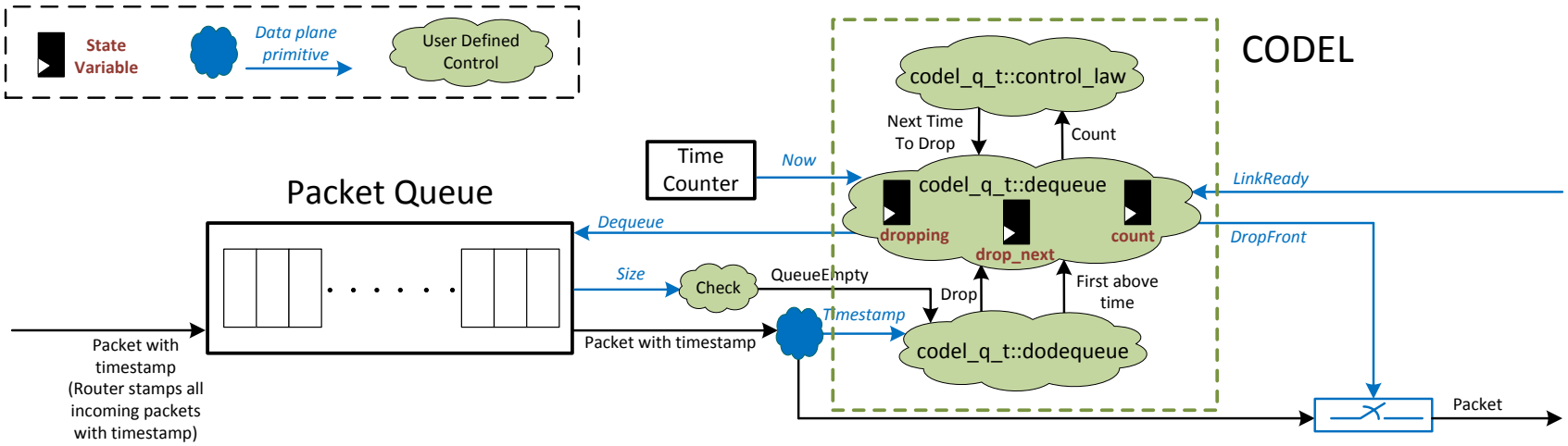
# But, do so in a controlled manner.

- Provide interfaces to the head and tail of switch queues.
- Operators specify only queue-management/scheduling logic.
- Code size limits constrain program sophistication.
- Disallow access to packet payloads.

# Re-architecting for data plane programmability.

- Hardware Primitives
  - Random number generators (RED, BLUE)
  - Binary tree of comparators (pFabric, SRPT)
  - EWMA estimators (AVQ, CSFQ)
- I/O interfaces
  - Drop/Mark head/tail of queue
  - Interrupts for enqueue/dequeue
  - Reading packet headers
- State variables
  - Per-flow (WFQ, DRR)
  - Per-src address (PF)
  - For fastest flows alone (AFD)
- Instruction Set
  - Expresses Control Flow
  - Implements new functionality unavailable in hardware

# Example implementation: CoDel




Synthesis numbers on Xilinx Kintex-7:

| **Resource** | **Usage** | **Fraction of FPGA** |
|---|---|---|
| Slice logic | 1,256 | 1% |
| Slice logic dist. | 1,975 | 2% |
| IO/GTX ports | 27 | 2% |
| DSP slices | 0 | 0% |
| Maximum speed | 12.9 $\times 10^6$ pkts/s | |