Taurus: A Data Plane Architecture for Per-Packet ML

Tushar Swamy

Alexander Rucker, Muhammad Shahbaz, Ishan Gaur, and Kunle Olukotun

Stanford University

Datacenter networks are becoming harder to manage…

> *“Our current generation — Jupiter fabrics — can deliver more than 1 Petabit/sec of total bisection bandwidth”*

— A Look Inside Google’s Data Center Networks[1]

Networks require complex management with high performance

Automate decision-making with machine learning (ML)

- Making decisions based on data → ***machine learning***

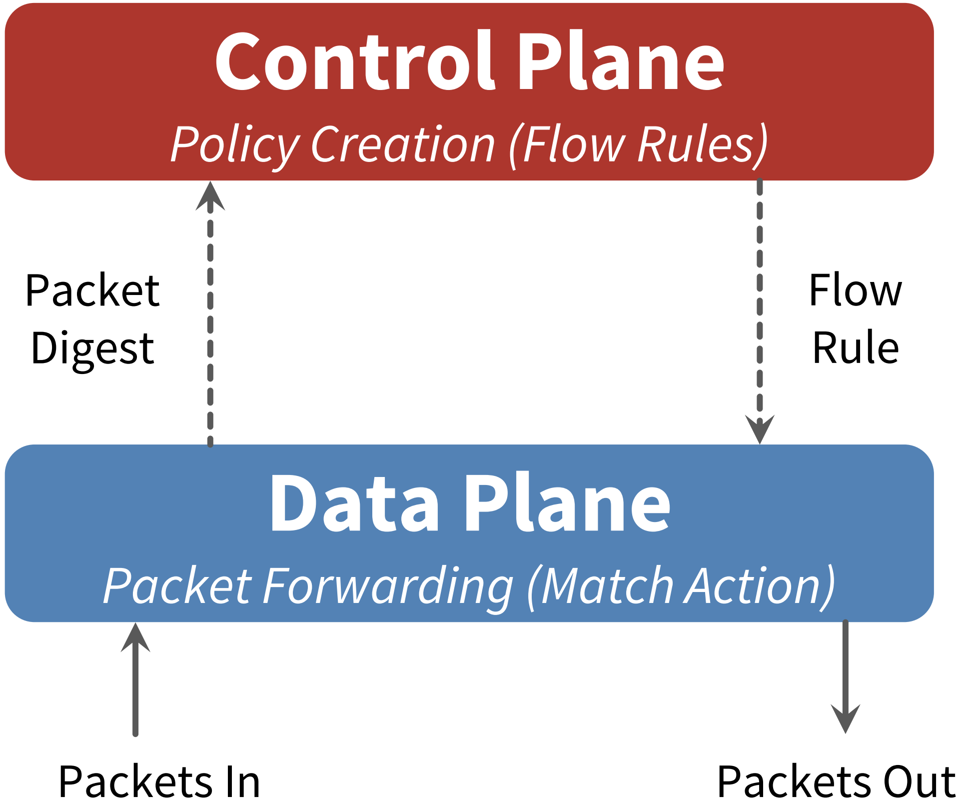
- Machine learning can:

  - ***Approximate*** network functions based on data
  - ***Customize*** network functions based on data

- Currently, we use by hand-written heuristics in the network…

Where in the network should ML happen?

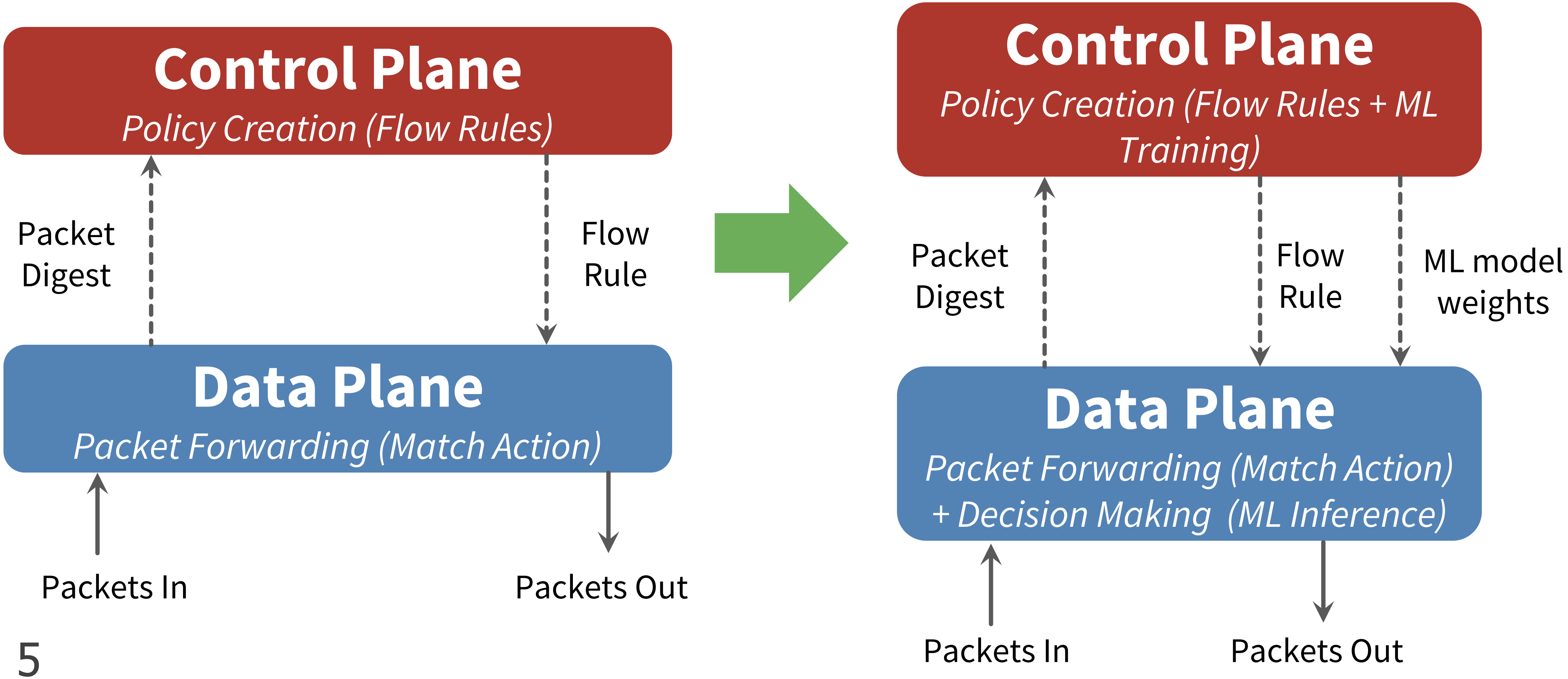
Software Defined Network



A Taurus network introduces ML for management

Software Defined Network

Software Defined Network with Taurus

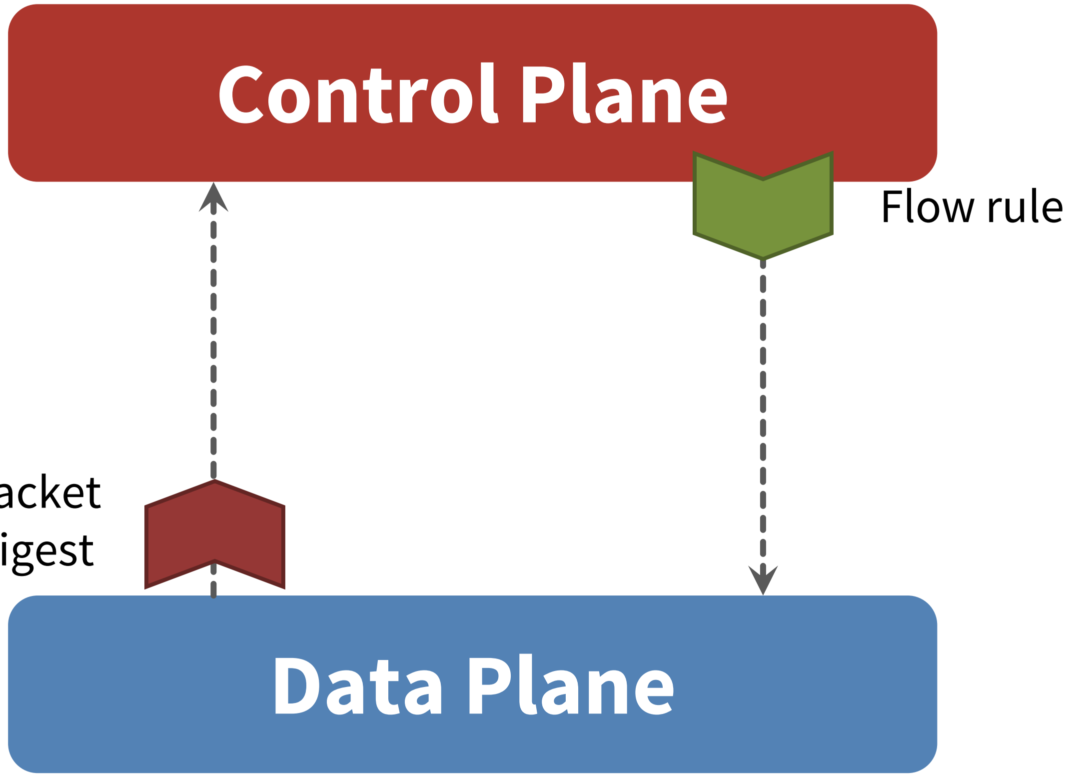


ML inference should happen *per-packet* in the *data plane*

Example: Anomaly Detection

Processing time: [illegible]

Packets missed: [illegible]



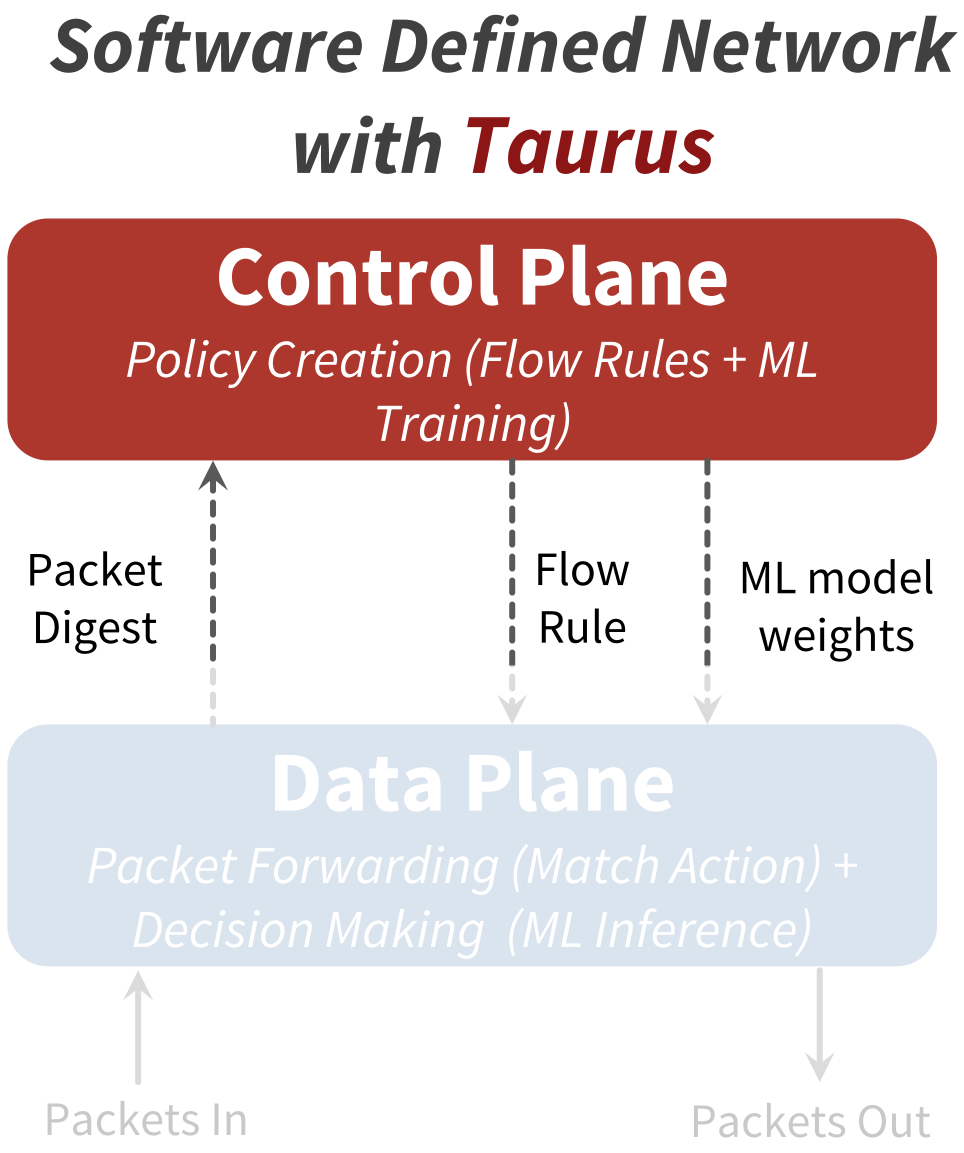
1.5 M Packets missed during flow rule installation time

Robustness and performance of the network are determined by:

- *Quality of reaction*
- *Speed of reaction*

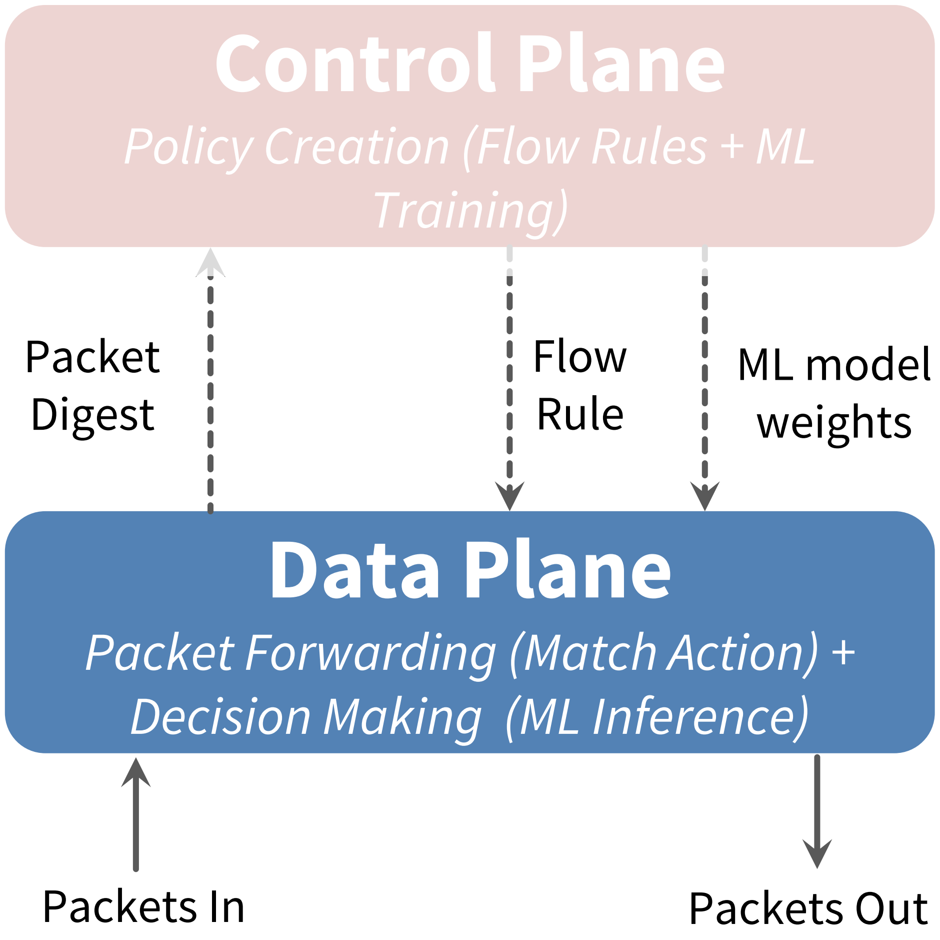
ML training happens in the control plane

ML Training is off critical path



ML Inference happens in the data plane

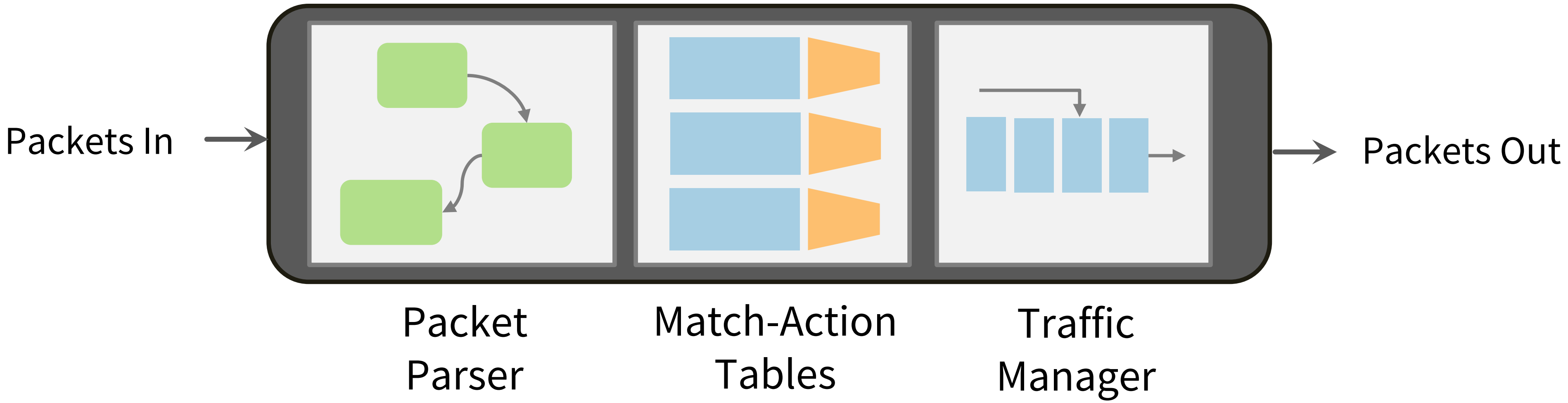
Software Defined Network with Taurus



ML Inference is on critical path

Taurus **is an architecture for per-packet ML inference in the data plane**

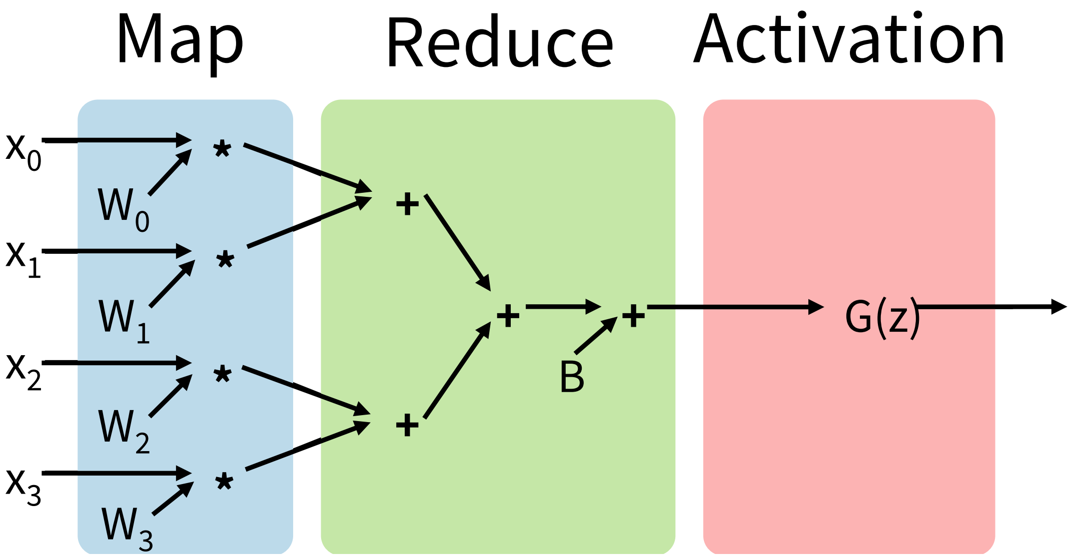
What do programmable switches look like?



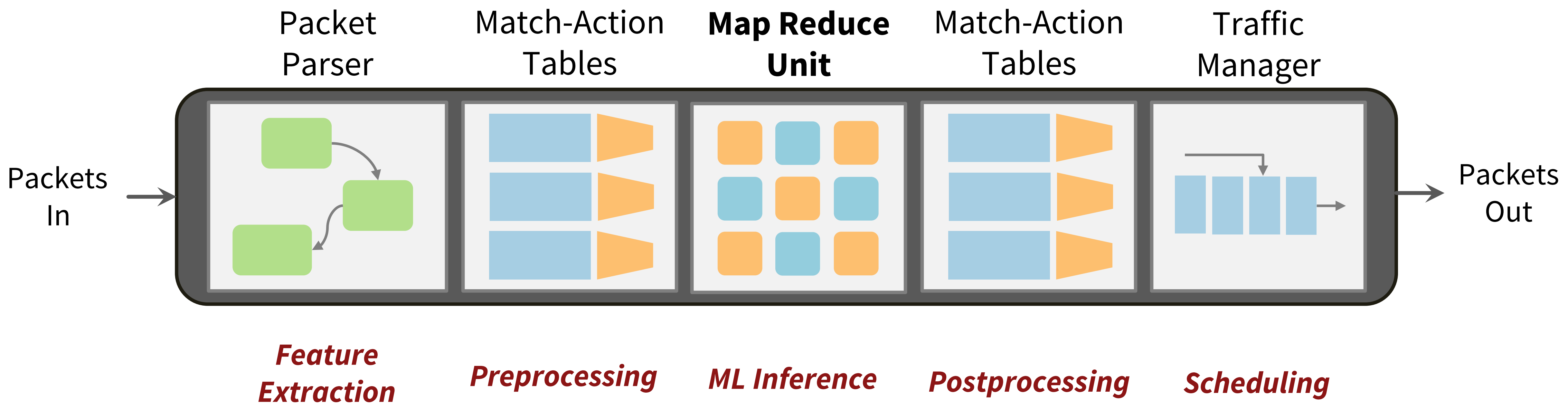
A Protocol Independent Switch Architecture (PISA)

What abstraction should we use?

- ***Map-reduce*** can support linear algebra operations common in ML
  - Neural networks, SVMs, etc.
- ***SIMD Parallelism*** enables performance with minimal logic
- ***Unrolling*** patterns allows for flexibility
  - More unrolling → better performance
  - Less unrolling → less resource usage

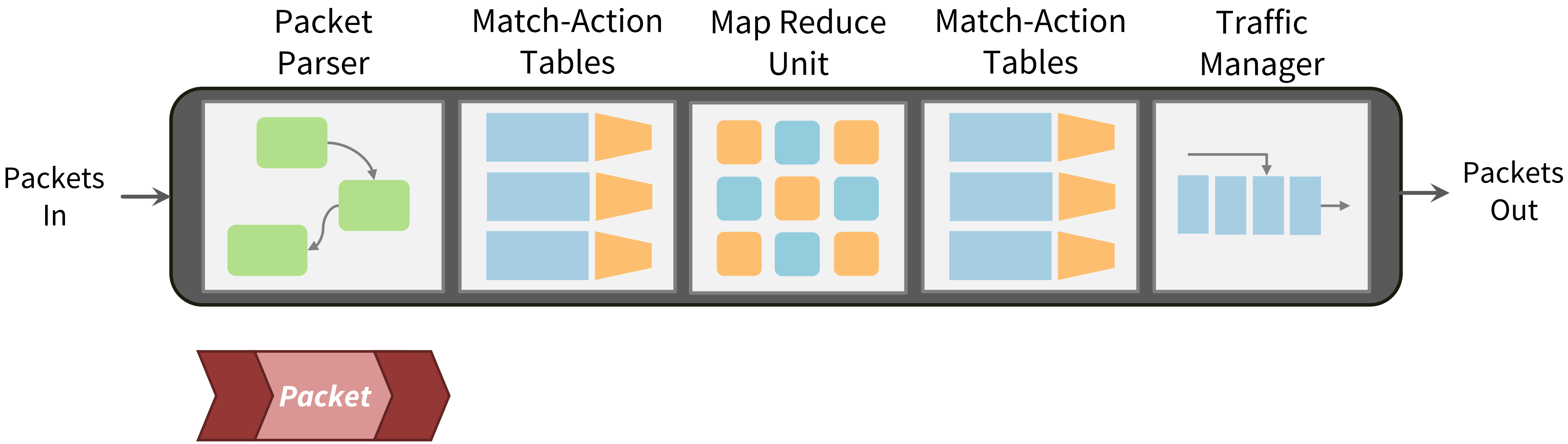


The Taurus pipeline with a Map Reduce Unit



- ***Map Reduce Unit*** must:
  - be reconfigurable
  - meet line rate (with a fixed clock)
  - incur minimal area and power overhead

Example Application: Anomaly Detection



Read local features (e.g., IP address)

Retrieve out of network events (e.g., failed logins per IP)

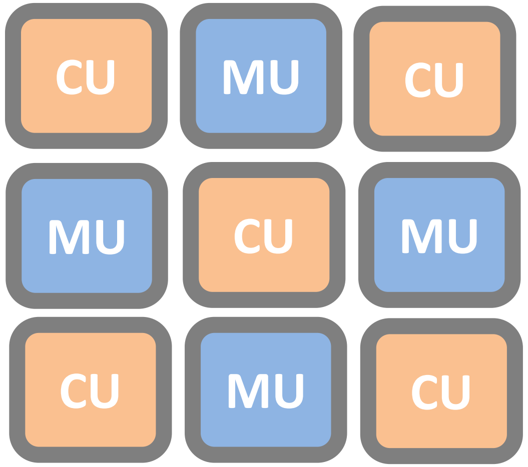
Apply learned anomaly detection

Select a port or action (e.g., drop if score == 1)

Send packet to destination

Evaluation of a Taurus ASIC

- Our evaluation platform is based on ***Plasticine***
- We program our map-reduce applications in the ***Spatial HDL***



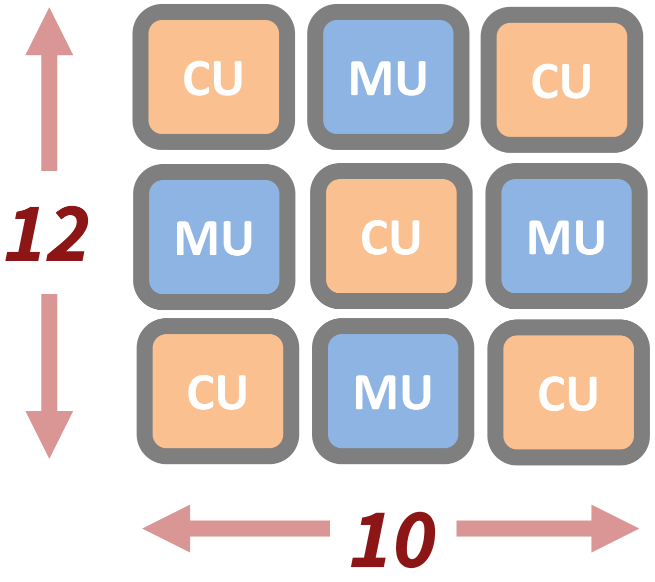
Compute unit

Memory unit

More architectural details in full paper!

Evaluation of a Taurus ASIC

- Our evaluation platform is based on ***Plasticine***
- We program our map-reduce applications in the ***Spatial HDL***



	Area	
Hardware	mm^2	+%
12x10 MR Grid	4.8 x 4	3.8
Prog. Switch	500	---

**Overheads are calculated relative to state of the art programmable switches*

Evaluation of an Anomaly Detection (AD) benchmark

- ***AD SVM: 8 support vectors***
- ***AD DNN: 4 layers - 12x6x3x2 neurons***

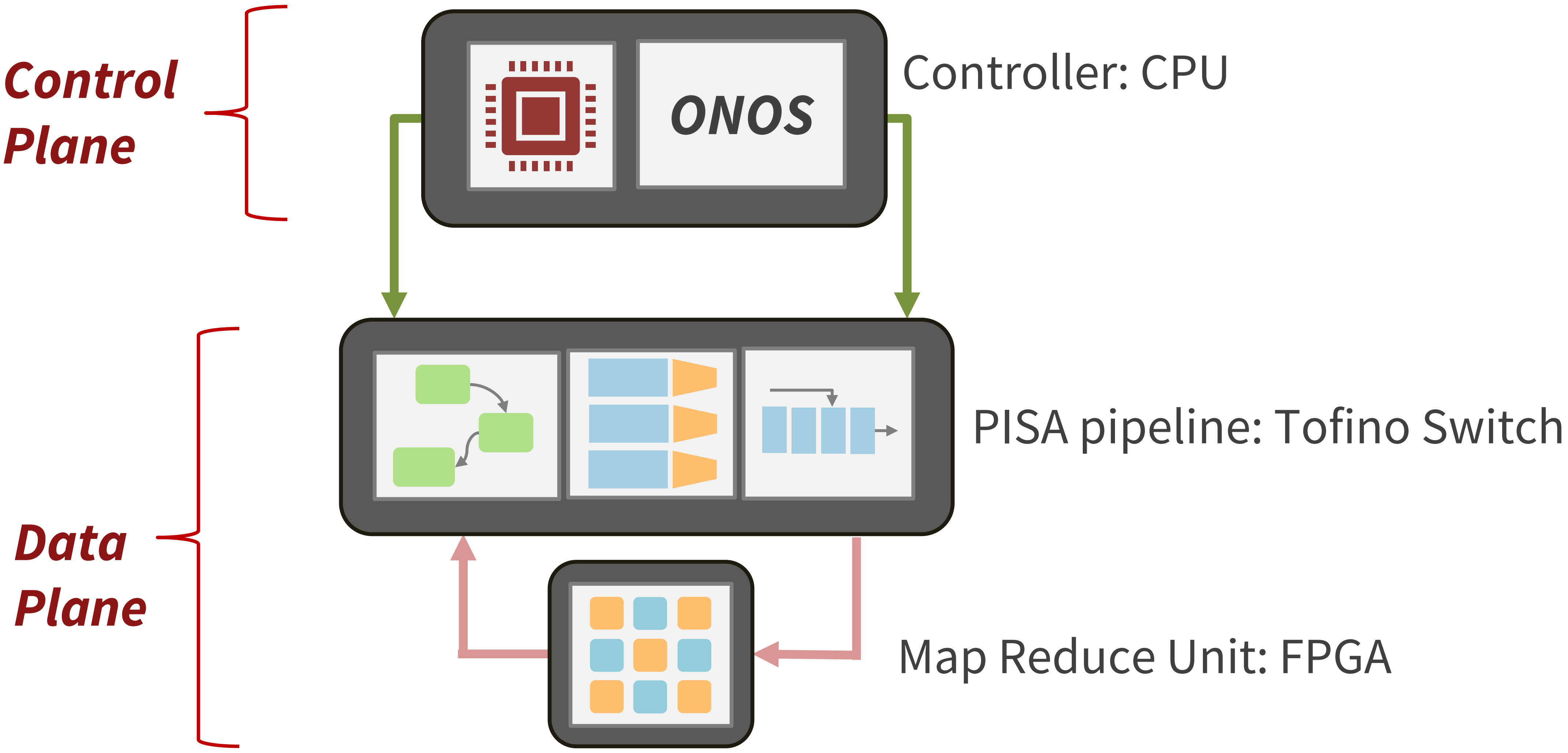
Overhead of Map Reduce Unit

			Area	Power
Model	TP (GPkt/s)	Lat (ns)	+%	+%
SVM	1	83	0.5	0.6
DNN	1	221	0.8	1.0

**Overheads are calculated relative to state of the art programmable switches*

More apps in full paper!

We provide an open-source, FPGA-based testbed



Questions?

Tushar Swamy

tswamy@stanford.edu

Try it out!

https://gitlab.com/dataplane-ai/taurus