The

Promise

of

Programmable Packet Processing

With

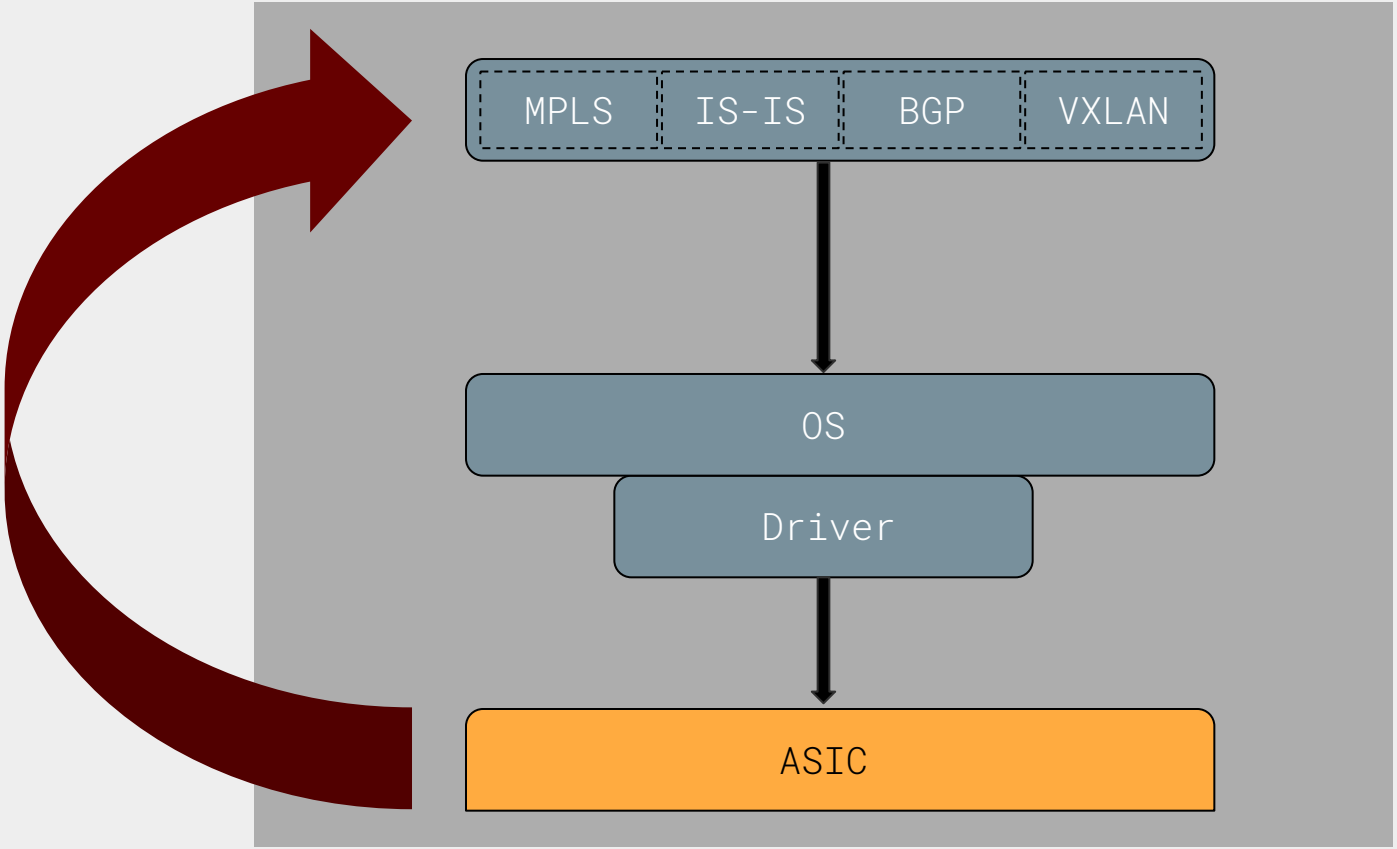
Aaron A. Glenn
internetworking curmudgeon

RIPE 76
Marseille, France
14 - 18 May 2018

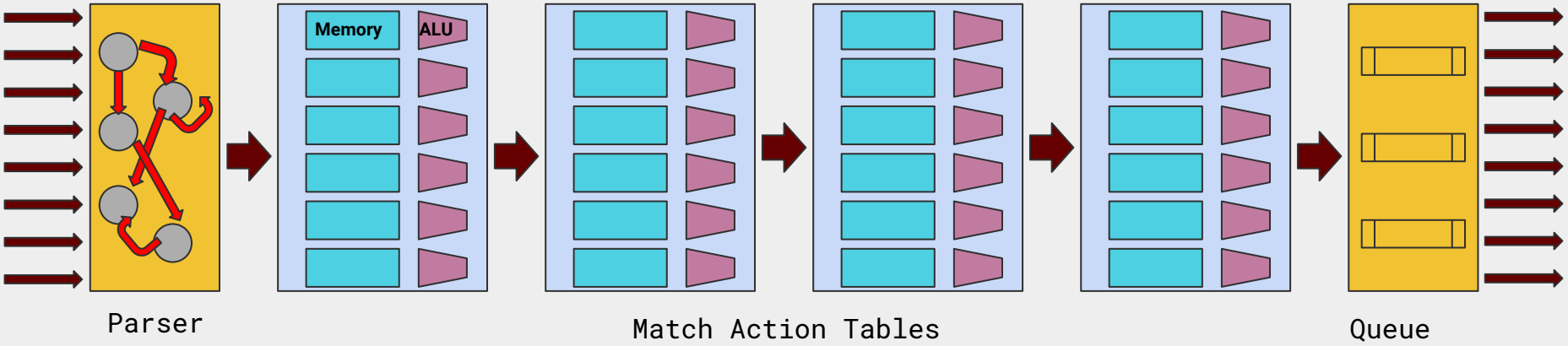
The First Router

1. Many features to make the IMPs run reliably and with minimal on-site assistance and with cross-network diagnosis, debugging, and new releases
2. Considerable facilities for network monitoring and measurement
3. No constraints put on the data hosts could exchange over the network

"Looking back at the ARPANET effort, 34 years later"
By David C. Walden



Bottom Up: Fixed Function ASICs



Generalized Packet Pipeline

Parser Program

```
parser parse_ethernet {
        extract (ethernet);
        return
switch(ethernet.ethertype) {
        0x8105 : parse_vlan_tag;
        0x0805 : parse_ipv4;
        0x8847 : parse_mpls;
        default: ingress;
        }
}
```

Header & Data Declarations

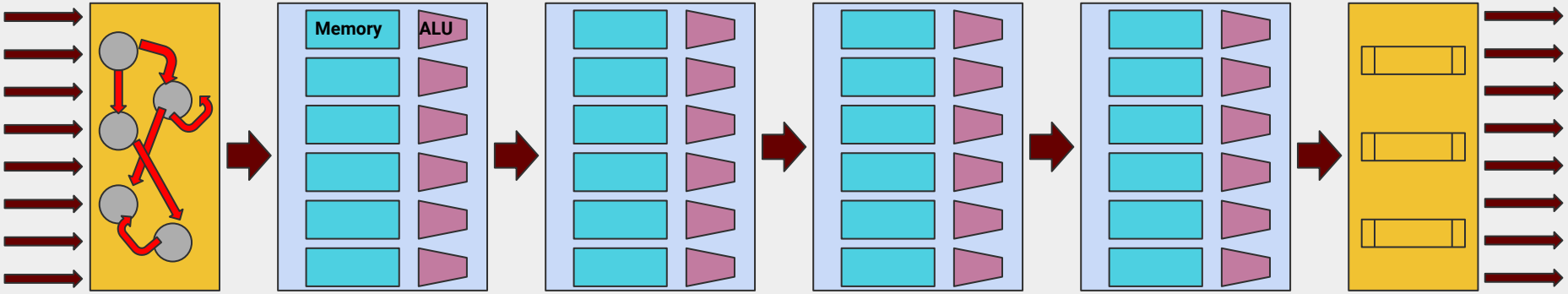
```
header_type ethernet_t      { … }
header_type l2_metadata_t   { … }

header     ethernet_t     ethernet;
extract    vlan_tag_t     vlan_tag[2];
metadata   l2_metadata_t  l2_meta;
```

Table & Control Flow

```
table port_table     { … }

control ingress {
        apply(port_table);
        if (l2_meta.vlan_tags == 0) {
                process_assign_vlan();
        }
}
```



Parser

Match Action Tables

Queue

Generalized Packet Pipeline

Programming

Protocol-Independent

Packet

Processors

Another SDN Talk?

RFC 1925 Compliance

(4) Some things in life can never be fully appreciated nor understood unless experienced firsthand. Some things in networking can never be fully understood by someone who Neither builds commercial networking equipment nor runs an operational network.


P4
Is this SDN ?
百科事典

What is $P4_{16}$?

“P4 is a language for expressing how packets are processed by the data plane of a programmable forwarding element such as a hardware or software switch, network interface card, router, or network appliance.”

https://p4.org/p4-spec/docs/P4-16-v1.0.0-spec.html#sec-overview

Protocol Independence

Devices not tied to any specific protocols.

Target Independence

Describe functionality independent of underlying hardware.

Reconfigurable

Ability to change packet processing after deployment.

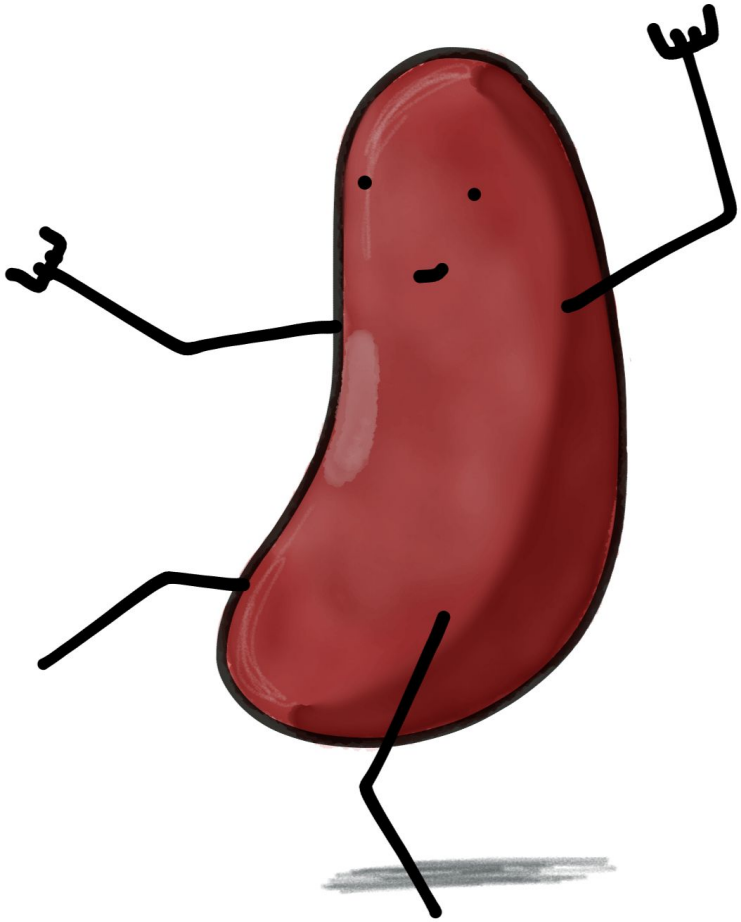
Networks Defined by Behavior

“P4 lets us define what headers a switch will recognize (or "parse"), how to match on each header, and what actions we would like the switch to perform on each header.”

Rexford, Jennifer, and Nick McKeown. “Clarifying the Differences between P4 and OpenFlow.” *P4*, 18 May 2016, p4.org/p4/clarifying-the-differences-between-p4-and-openflow.html.

What can happen when the dataplane is programmable?

- ➔ Realize new features and protocols faster
- ➔ Use hardware resources more efficiently
- ➔ Increase visibility into the network
- ➔ Greater control over the network



COOL
BEANS

@bryanMMathers

Various P4 *Compiler* Targets

eBPF

eXpress Data Path

Vector Packet Processing

Netcope VHDL FPGA

Xilinx PX FPGA

Barefoot Tofino ASIC

P4GPU (CUDA)

P4FPGA (Verilog)

PISCES (OpenVSwitch)

T4P4S (DPDK)

Netronome SmartNIC

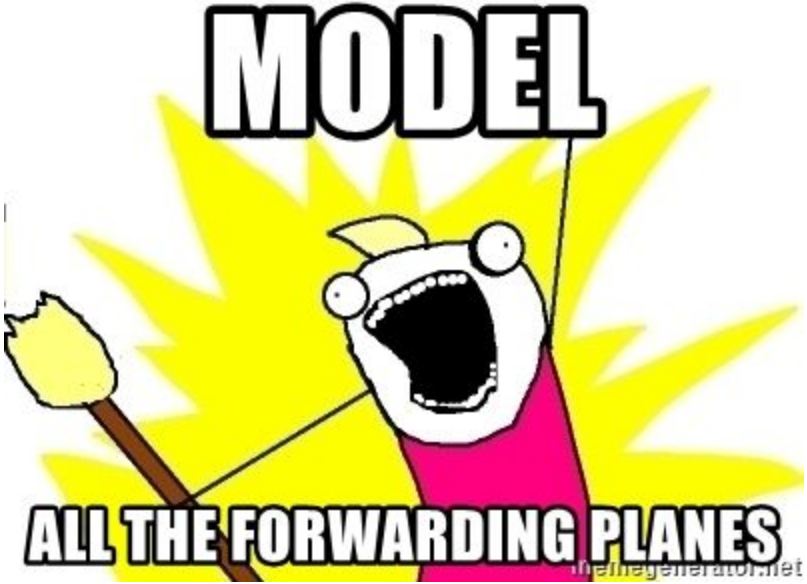
	LoC	Methods	Method Size
OVS	14,535	106	137.13
PISCES	341	40	8.53

Table 2: *Native OVS compared to equivalent baseline functionality implemented in PISCES.*

		Files Changed	Lines Changed
Connection Label:	OVS [70, 71]	36	633
	PISCES	1	5
Tunnel OAM Flag:	OVS [27, 28]	21	199
	PISCES	1	6
TCP Flags:	OVS [61]	20	370
	PISCES	1	4

Table 3: *The number of files and lines we needed to change to implement various functionality in P4, compiled with PISCES, compared to adding the same functionality to native OVS.*

Muhammad Shahbaz, Sean Choi, Ben Pfaff, Changhoon Kim, Nick Feamster, Nick McKeown, and Jennifer Rexford. *PISCES: A Programmable, Protocol-Independent Software Switch*. In ACM SIGCOMM, Florianópolis, Brazil, August 2016.

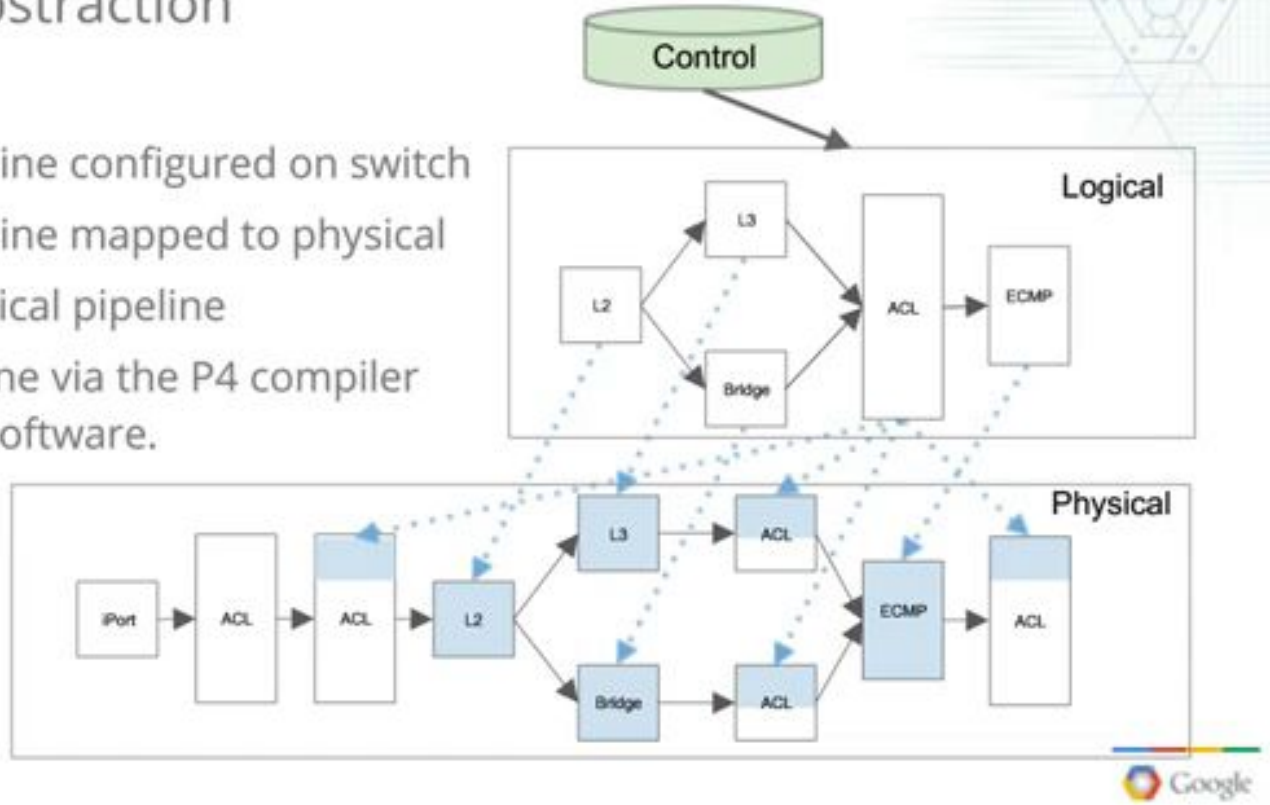

MODEL
ALL THE FORWARDING PLANES

Devices Defined by Behavior

P4 can be considered a behavioral description of a switching device which may or may not execute “P4” natively.

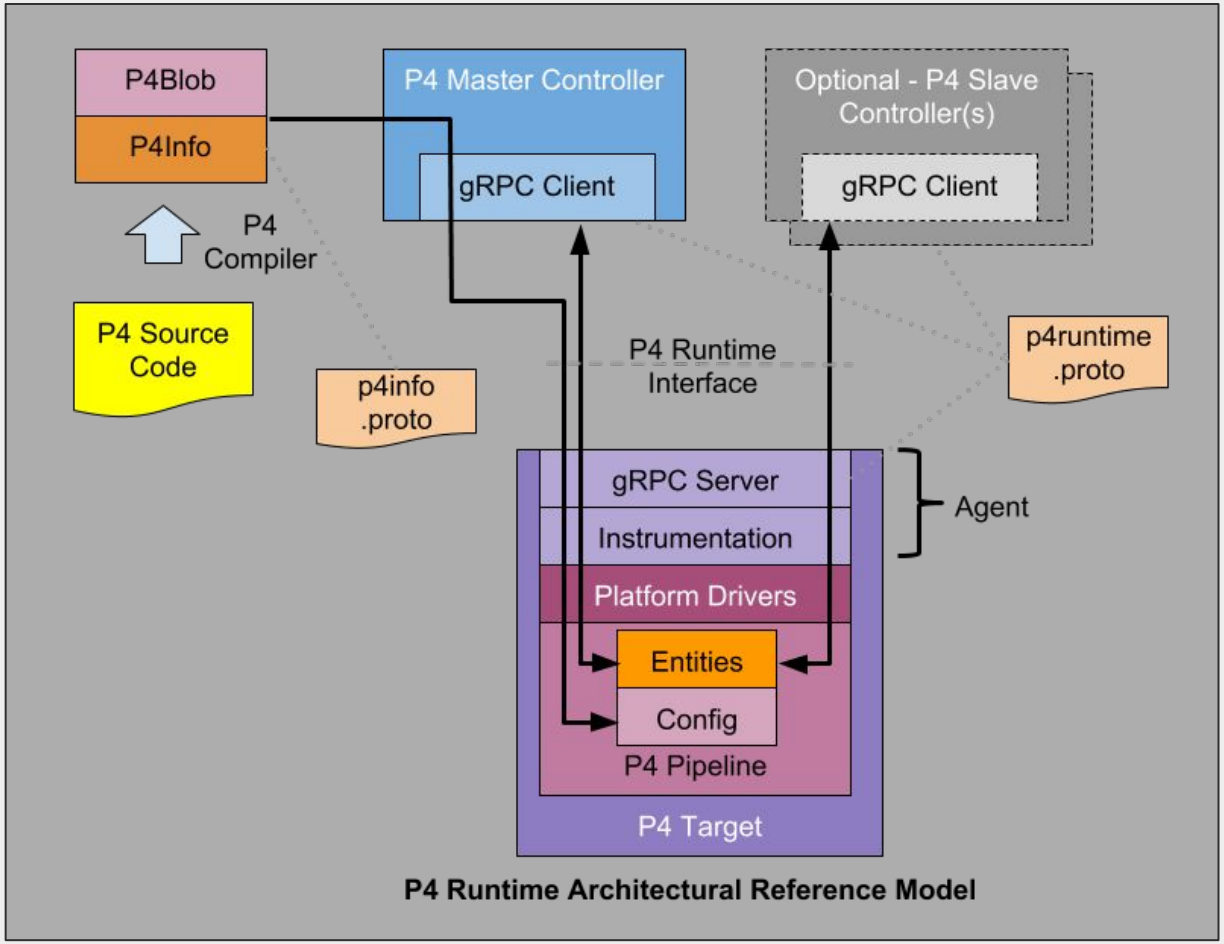
P4 Switch Abstraction

- Logical pipeline configured on switch
- Logical pipeline mapped to physical
- Program logical pipeline
- Mapping done via the P4 compiler and switch software.



Runtime Control of P4 Targets

The P4Runtime API defines the messages and semantics of the interface between the client(s) and the server.


P4Blob
P4Info
P4 Compiler
P4 Source Code
P4 Master Controller
gRPC Client
Optional - P4 Slave Controller(s)
gRPC Client
p4info .proto
P4 Runtime Interface
p4runtime .proto
gRPC Server
Instrumentation
Platform Drivers
Agent
Entities
Config
P4 Pipeline
P4 Target
P4 Runtime Architectural Reference Model

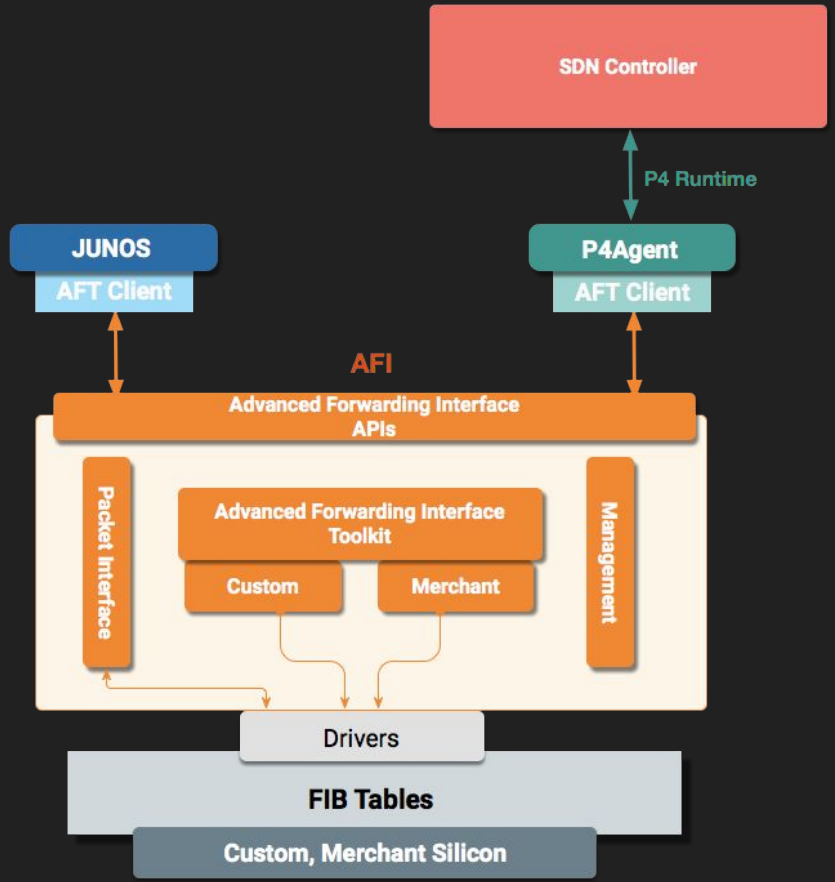
P4Runtime

It is not necessary to have a P4 source program to begin with, since the controller does not use it.

The minimum requirement is a P4Info file which can be loaded by a controller in order to render the correct P4Runtime API. As long as the target supports the operations implied by the P4Info file, the underlying implementation is moot.

“Juniper has adopted P4 as the language that describes the contract between the control plane and the data plane of switches and routers.

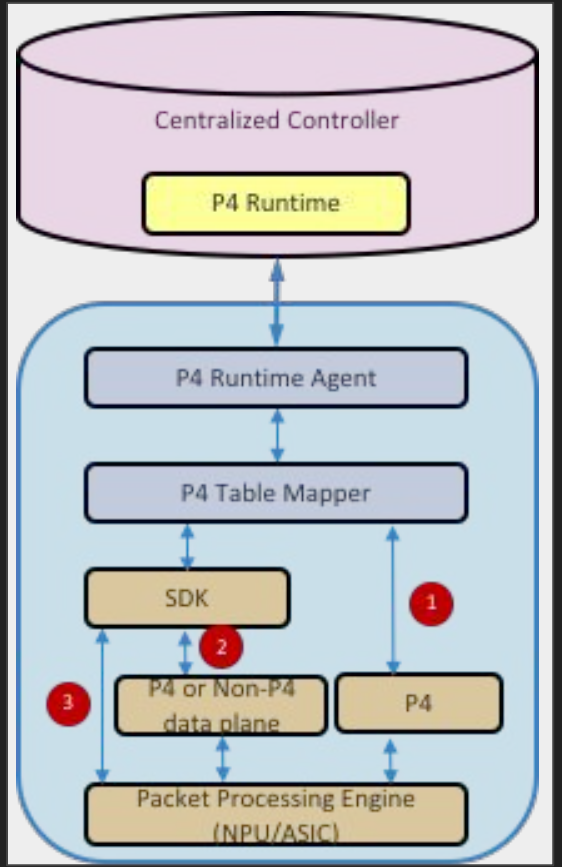
Juniper has also implemented the P4 Runtime across the portfolio as an open data plane programming API.”



from *“Juniper Advancing Disaggregation Through P4 Runtime Integration”*

“The new Open Forwarding Abstraction (OFA) API provides a logical representation of all the forwarding and telemetry capabilities of the underlying hardware.

We used our OFA API to develop a P4Runtime agent that provides seamless integration of P4Runtime-enabled applications to manage multiple platforms.”



from “*Introduction to P4 and P4Runtime*” by Praveen Bhagwatula

Correct by Construction Networks

P4 as an intermediate representation for higher level languages and abstractions

Network programming can readily benefit from verification methods more difficult to use in generic programming

Ryzhyk, Leonid, et al. "Correct by Construction Networks Using Stepwise Refinement." *NSDI*. 2017.

More P4

P4 Developer Days + SIGCOMM 2017 Tutorials + many more
https://github.com/p4lang/tutorials/

Andy Fingerhut's Variety of Potentially Useful P4 Information
https://github.com/jafingerhut/p4-guide

P4 Works (not officially affiliated with the P4 Association)
https://p4.works/

I regularly tweet about P4, to the annoyance of some, as @networkservice