



P4 in Open vSwitch with OfP4

Ben Pfaff
Debnil Sur
Leonid Ryzhyk
Mihai Budiu

Existing P4 Software Switches

BMv2

- + Accurate simulation
- Low performance.

T₄P₄S

- + Fast
- Hard to install across operating systems.

PISCES

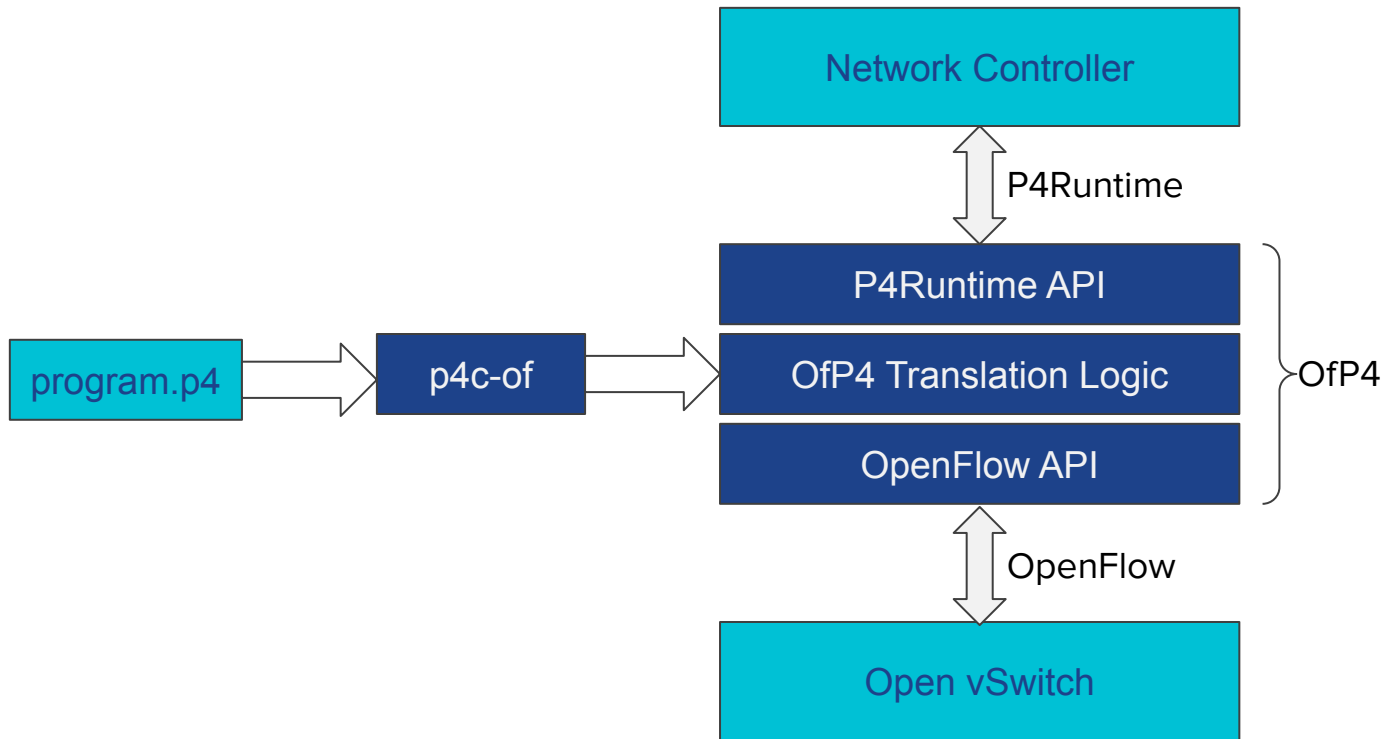
- Unmaintained
- No P4Runtime support

Others in development:

- uBPF
- DPDK
- PSA eBPF

OfP4: Software P4 with an OVS data plane

A daemon to translate between P4+P4Runtime and OpenFlow



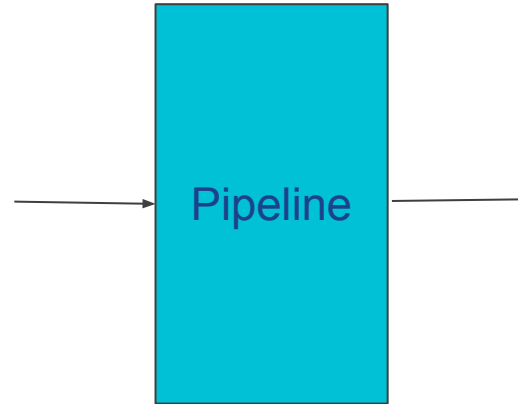
Starting from a P4 program and the controller that supports it:

- Compile P4 with p4c-of
- Connect controller to OfP4 over P4Runtime
- Connect OfP4 to Open vSwitch over OpenFlow

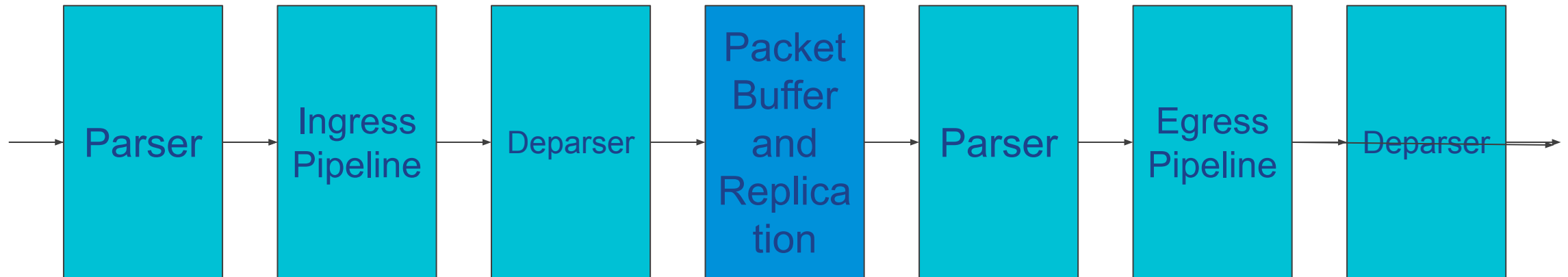
Unmodified, upstream OVS
Uses OVS extensions to OpenFlow

Architectures for P4

OpenFlow Architecture

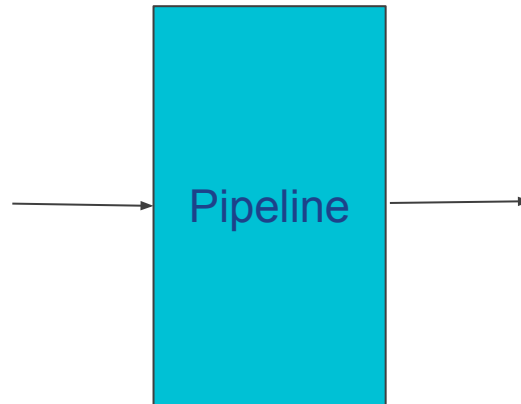


P4 Portable Switch Architecture

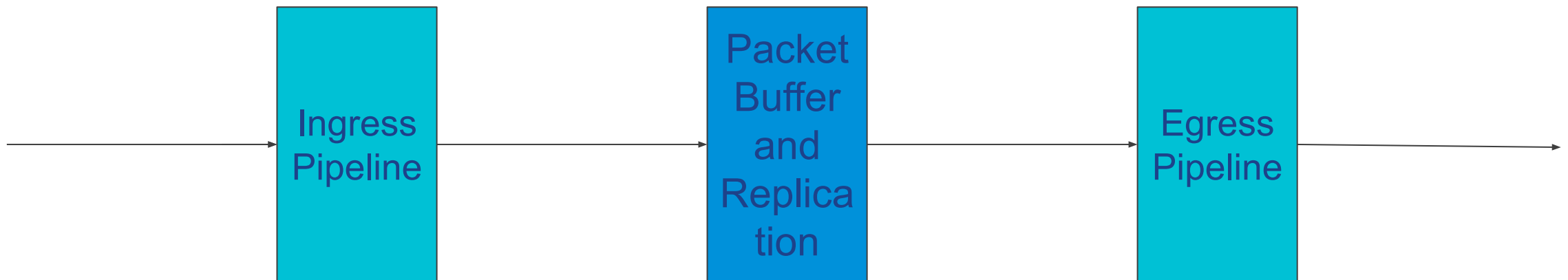


Architectures for P4

OpenFlow Architecture



OfP4 Architecture



Translating P4 Metadata to OpenFlow

P4: Flexible Metadata

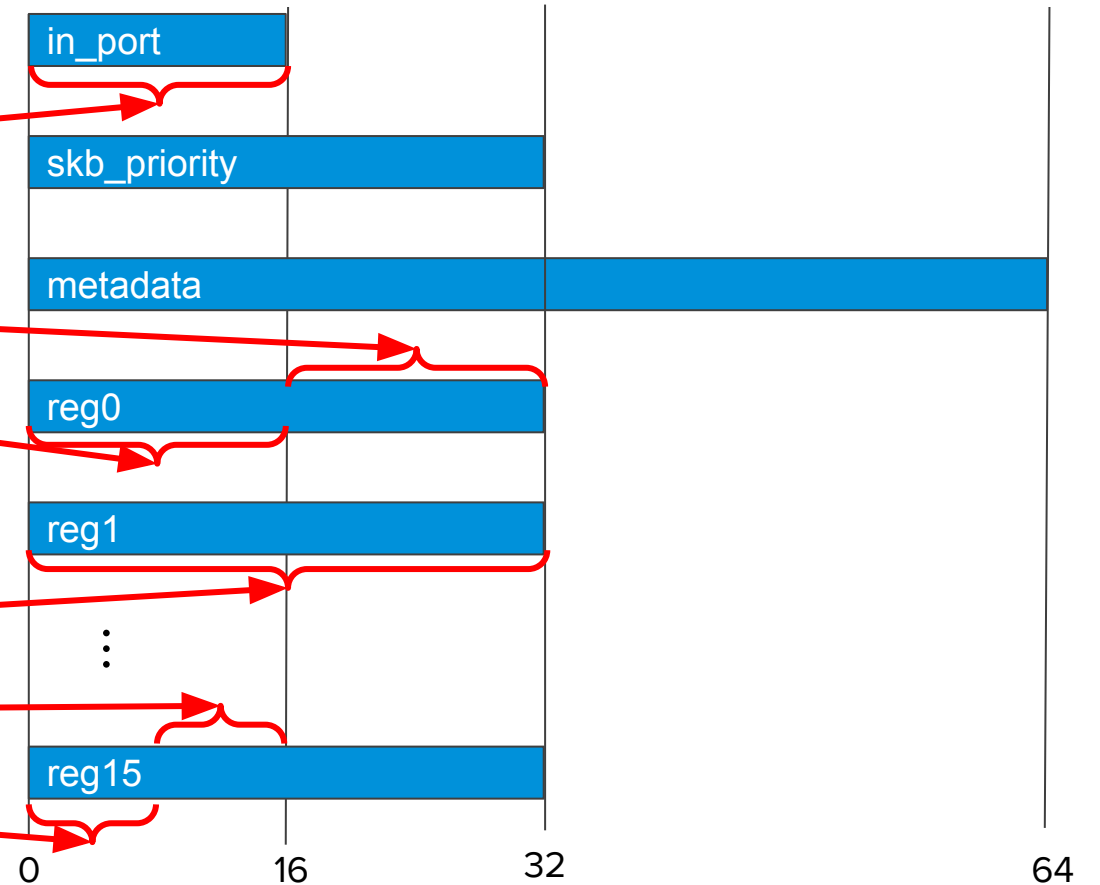
OpenFlow: Fixed Metadata

Standard Metadata

```
struct standard_metadata_t {  
    bit<16> in_port;  
    ...  
    bit<16> out_group;  
    bit<16> out_port;  
}
```

Program Metadata

```
struct metadata_t {  
    bit<32> b;  
    bit<8> c;  
    bit<8> d;  
}
```



Translating P4 Table Keys to OpenFlow

P4: Typed Table Keys

OpenFlow: Free-Form Matches

```
table InputVlan {  
  key = {  
    standard_metadata.in_port: exact;  
    hdr.vlan.isValid(): exact;  
    hdr.vlan.vid: optional;  
  }  
  actions = { Drop; SetVlan; UseTaggedVlan; }  
  default_action = Drop;  
}
```

The diagram illustrates the translation of P4 table keys to OpenFlow matches. Red arrows point from the P4 key definitions to the corresponding OpenFlow match expressions:

- standard_metadata.in_port: exact;** maps to `in_port=PORT`
- hdr.vlan.isValid(): exact;** maps to `vlan_tci=0/0x1000`
- hdr.vlan.vid: optional;** maps to `vlan_tci=0x1000/0x1000`
- hdr.vlan.vid: optional;** maps to `vlan_tci=VLAN/0xff`

Translating P4 Table Actions to OpenFlow

P4: Typed Actions

```
table InputVlan {  
    key = {  
        standard_metadata.ingress_port: exact;  
        hdr.vlan.isValid(): exact;  
        hdr.vlan.vid: optional;  
    }  
    actions = { Drop; SetVlan; UseTaggedVlan; }  
    default_action = Drop;  
}  
action Drop() {  
    mark_to_drop(standard_metadata);  
    exit;  
}  
action SetVlan(bit<16> vid) { meta.vlan = vid; }  
action UseTaggedVlan() { meta.vlan = hdr.vlan.vid; }
```

OpenFlow: Free-Form Actions

actions=load(0->reg3), resubmit(,31)

actions=load(vid->reg7[0..11]), resubmit(,3)

actions=move(vlan_tci[0..11]->reg7[0..11]), resubmit(,3)

Other translations

P4

OpenFlow

Default actions	→	Flow priorities
Arithmetic and logical expressions	→	Not supported
Control flow	→	Simple tables
Digests	→	OpenFlow send-to-controller action



Thank You

<https://github.com/vmware/nerpa>
ofp4 branch