Introduction to P4
Once upon a time...

- There were Smart Switches and Routers....
- Everybody was happy...Especially the vendors...
- And then came Open Flow...
In the beginning ...

• Open Flow was simple
• Included single rule table
  • Priority, pattern, actions, counters, timeouts
• Matching on any of 12 fields, e.g.,
  • MAC addresses
  • IP addresses
  • Transport protocol
  • Transport port numbers
“Classic” Open Flow

SDN Control Plane

Installing and querying rules

Target Switch
Over the past 8 years...

- Proliferation of header fields

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th># Headers</th>
</tr>
</thead>
<tbody>
<tr>
<td>OF 1.0</td>
<td>Dec 2009</td>
<td>12</td>
</tr>
<tr>
<td>OF 1.1</td>
<td>Feb 2011</td>
<td>15</td>
</tr>
<tr>
<td>OF 1.2</td>
<td>Dec 2011</td>
<td>36</td>
</tr>
<tr>
<td>OF 1.3</td>
<td>Jun 2012</td>
<td>40</td>
</tr>
<tr>
<td>OF 1.4</td>
<td>Oct 2013</td>
<td>41</td>
</tr>
<tr>
<td>OF 1.5</td>
<td>Dec 2014</td>
<td>44</td>
</tr>
</tbody>
</table>

- Still not enough... VxLAN, NVGRE, INT....
Open Flow is a nice idea...

But..

We can’t repeatedly extend the Open Flow standard
The “Dream” Open Flow

Configuring:
Parser, tables, and control flow

Populating:
Installing and querying rules

Compiler

Parser & Table Configuration → Rule Translator

SDN Control Plane

Target Switch

Classic Open Flow
P4 Language

Programming Protocol-Independent Packet Processing
P4 Goals

Reconfigurability – It should be able to re-define the packet parsing and processing in the field

Protocol independence - The switch should not be tied to a specific packet format

Target Independence - The programmer should not care what is the target
But switches still have ASICs..?

• Yes, but...
  • New generation of custom ASICs can achieve such flexibility at terabit speeds
    • Intel FlexPipe
    • Barefoot Tofino
  • Some switches are more programmable than others:
    • FPGA (Xilinx, Altera, Corsa)
    • NPU (EZChip/Mellanox, Netronome)
    • CPU (OVS, ...)

COMPUTER SCIENCE DEPARTMENT  
TECHNION · ISRAEL INSTITUTE OF TECHNOLOGY
P4 Language

• P4 program configures forwarding behavior (abstract forwarding model)

• P4 compiler translates into a target-specific representation

• OF can still be used to install and query rules once forwarding model is defined
P4 Forwarding Model

• P4 program configures forwarding behavior (abstract forwarding model)

• P4 compiler translates into a target-specific representation

• OF can still be used to install and query rules once forwarding model is defined
P4 Forwarding Model / Run Time