Some OpenFlow Details
Multiple Table Example: Port Based VLAN Tagging

- Most NICs don’t handle VLAN tags
  - Packets are sent untagged
- VLAN tags are inserted at the first hop switch
  - Based on the source port
- Implementing with one table results in a combinatorial explosion
  - With one table: $N_{port} \times N_{MAC}$
  - With two tables: $N_{port} + N_{MAC}$

Example: 48x1G Down, 4x10G Up

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Rules</th>
</tr>
</thead>
</table>
| Table 0:    | 7       *       *       *       0       *       *       *       *       *       *       Push VLAN ID 42; Send to Table 1
| Table 1:    | *       *       M1       *       42       *       *       *       *       *       Packet out Port 2
|             | *       *       M2       *       42       *       *       *       *       *       Packet out Port 4
Group Table Detail

Group Table: A table supporting entries with *action buckets*, which can be executed all or in part depending on the group type

Group Table Entry:

- Group identifier: 32 bit integer identifying the group on the OpenFlow switch
- Group type: a type identifier defining the group semantics
- Counters: for tracking group statistics
- Action Buckets: an ordered list of action buckets, where each bucket contains a set of actions and associated parameters that are always executed as an action set.
Action Bucket Semantics

- **Indirect**
  - Execute the one bucket in this group
  - Only a single bucket
  - Allows multiple flow entries to point to a single group
  - Example use: IP next hop forwarding

- **All**
  - Execute all buckets in the group
  - Example use: broadcast or multicast

- **Select**
  - Execute one bucket in the group
  - Bucket selected through a switch-specific algorithm
  - Example use: Equal Cost MultiPath (ECMP) forwarding

- **Fast Failover**
  - Execute the first live bucket in the group
  - Allows switch to change forwarding without contacting the controller
  - Switch must implement a liveness mechanism
  - Example use: fast failover
Meter Table Details

Meter Table: a collection of per flow meters used for rate limiting that can be combined with Queues to implement QoS strategies.

Meter Table Entry:
- Meter identifier: 32 bit unsigned integer identifying this meter
- Meter bands: unordered list of bands, where each band specifies a rate and a processing method
- Counters: keep track of the number of packets processed by this meter

<table>
<thead>
<tr>
<th>Meter Identifier</th>
<th>Meter Bands</th>
<th>Counters</th>
</tr>
</thead>
</table>

Band Entry:
- Band type: defines how the packet is processed
- Rate: lowest rate at which the band can apply
- Counters: keep track of the number of packets processed by this band
- Type specific arguments: optional arguments depending on type

<table>
<thead>
<tr>
<th>Band Type</th>
<th>Rate</th>
<th>Counters</th>
<th>Type specific arguments</th>
</tr>
</thead>
</table>