



Flow APIs for hardware offloads

November 2014
OVS Fall Conference 2014

Legal Disclaimers

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: <http://www.intel.com/design/literature.htm>

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families: Go to: [Learn About Intel® Processor Numbers](#)

Intel, the Intel logo, Itanium, Intel Atom, Intel Xeon Phi, Intel AppUp, and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

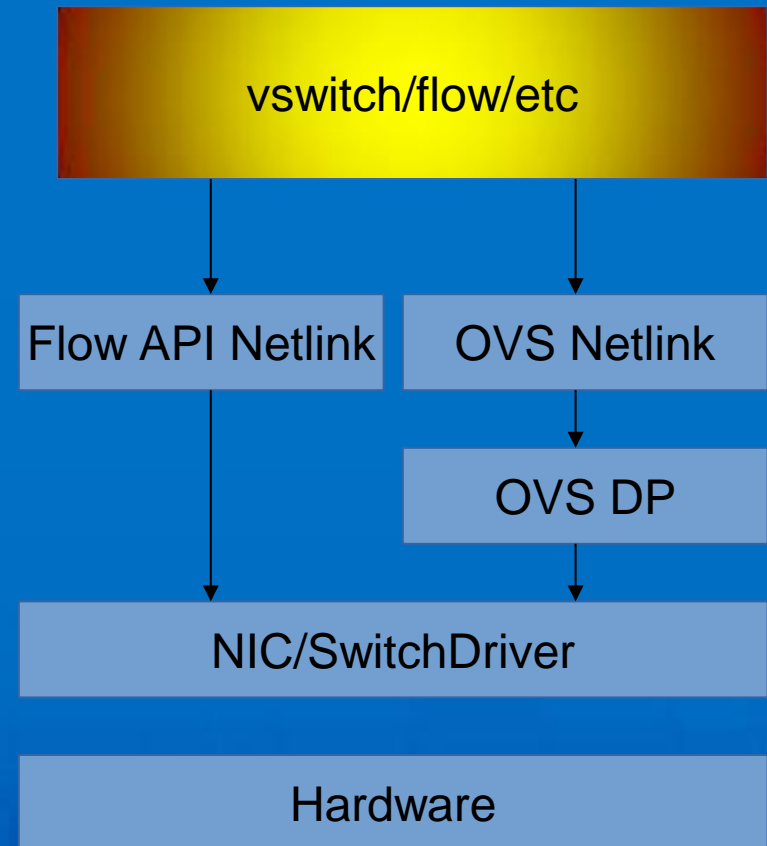
Copyright © 2013 Intel Corporation. All rights reserved

Flow API

- Why?
- Some Examples
- Flow API, Demo

Mechanics

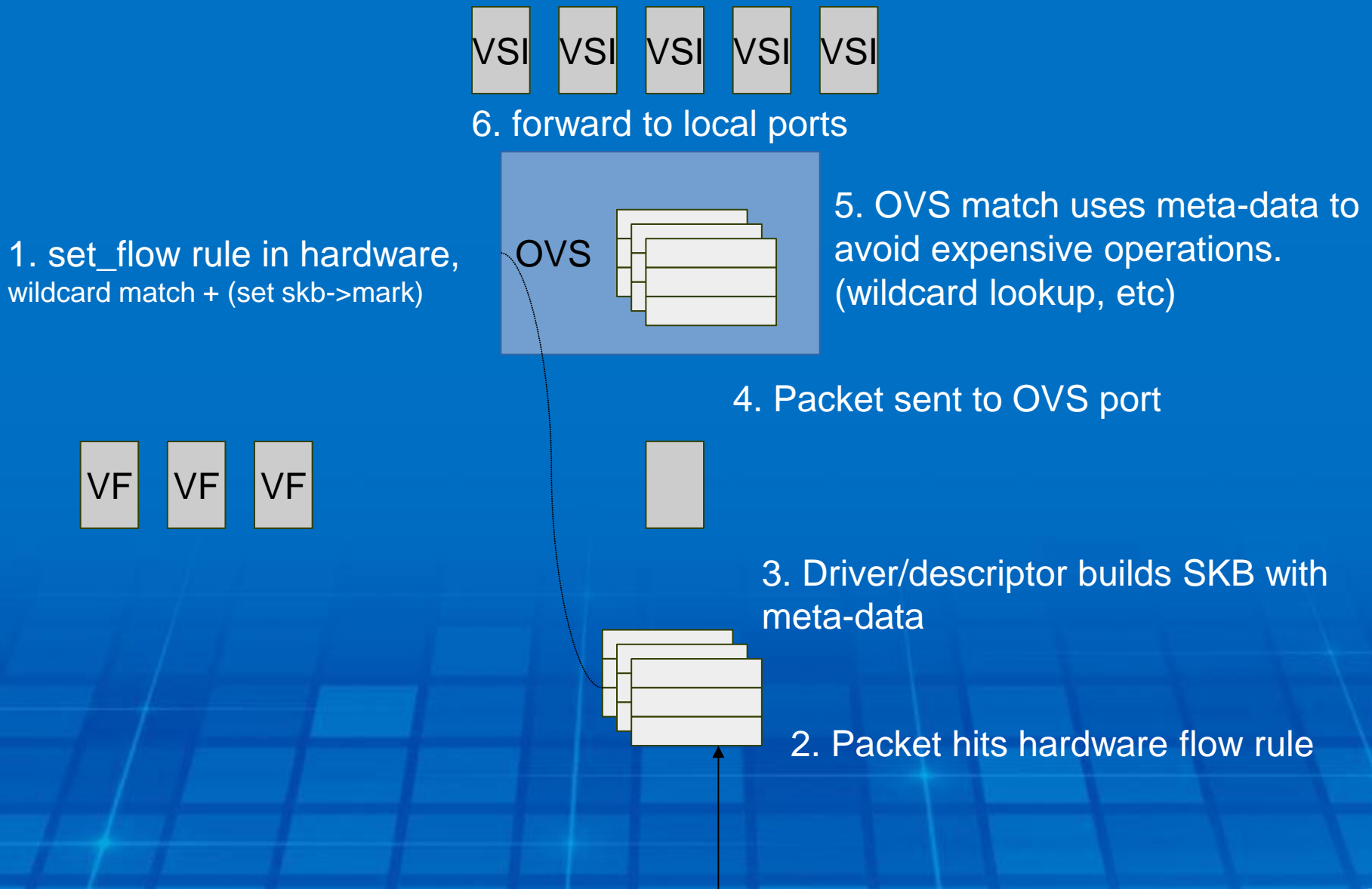
- Kernel API
 - <https://github.com/jrfastab/flow-net-next>
- Flow tool
 - <https://github.com/jrfastab/iproute2-flow-tool>
- Driver Extensions
 - Set of ndo ops for drivers to implement.
 - Endpoint agnostic could sync to user space application



Desire to expose hardware more directly

Some Example Flows for Hardware Offloads

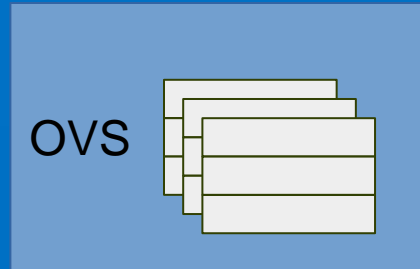
TCAM cookies for OVS (Partial Offload)



Forwarding Actions for SR-IOV (Full Offload)



5.1 forward to local ports

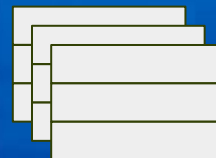


4. packet ingress OVS dataplane

3. Default VSI assigned OVS port



2. Default rule to send packet to default VSI



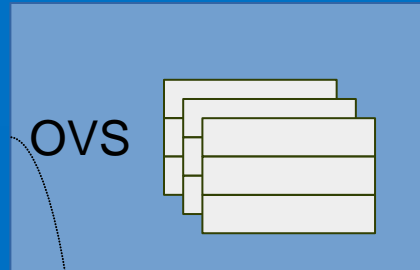
1. Packet miss hardware flow tables

Forwarding Actions for SR-IOV (Full Offload)



5.1 forward to local ports

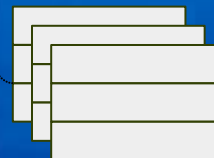
5.2(a) set_flow in hardware



5.2(b) forward to default VSI

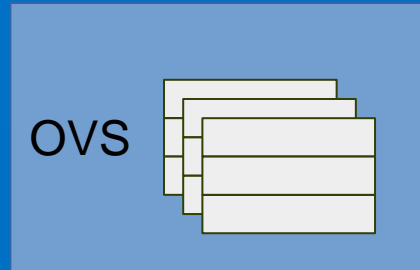


5.3 flow rule hit in flow table

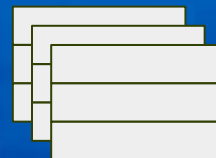


6. flow action to forward packet to VF

Forwarding Actions for SR-IOV (Full Offload)



3. packet received by VF



2. rule action forward to VF

1. Next packet hits hardware flow rule

OK but,

How do we program hardware?

OK but,

How do we program hardware?

And how do we “know” hardware capabilities

Flow API

Flow API

- Capabilities
- Table Allocation
- Flows

Flow API (capabilities)

- Headers (get_headers, get_parse_graph)
- Tables (get_tables, get_tables_graph)
- Actions (get_actions)

Demonstrate rocker switch, 10Gbe Intel NIC

Flow API (Table Allocation)

- Create (create_table)

```
#!/flow ethx create size 1024 name decap_table source 1 \  
    match ipv4.src_ip \  
    match ipv4.dst_ip \  
    action set_egress_port
```

- Destroy (destroy_table)

```
#!/flow ethx destroy decap_table
```

Demonstrate rocker switch, 10Gbe Intel NIC

Flow API (Flow Allocation)

- Set Flow (set_flow)

```
# ./flow set_flow prio 1 handle 1 table decap_table \  
  match ipv4.src_ip 10.0.0.1 \  
  match ipv4.dst_ip 10.0.0.2 \  
  action set_egress_port 10
```

- Delete Flow (del_flow)

```
#./flow del_flow prio 1 handle 1
```

- Get Flows (get_flows)

```
#./flow get_flow decap_table
```

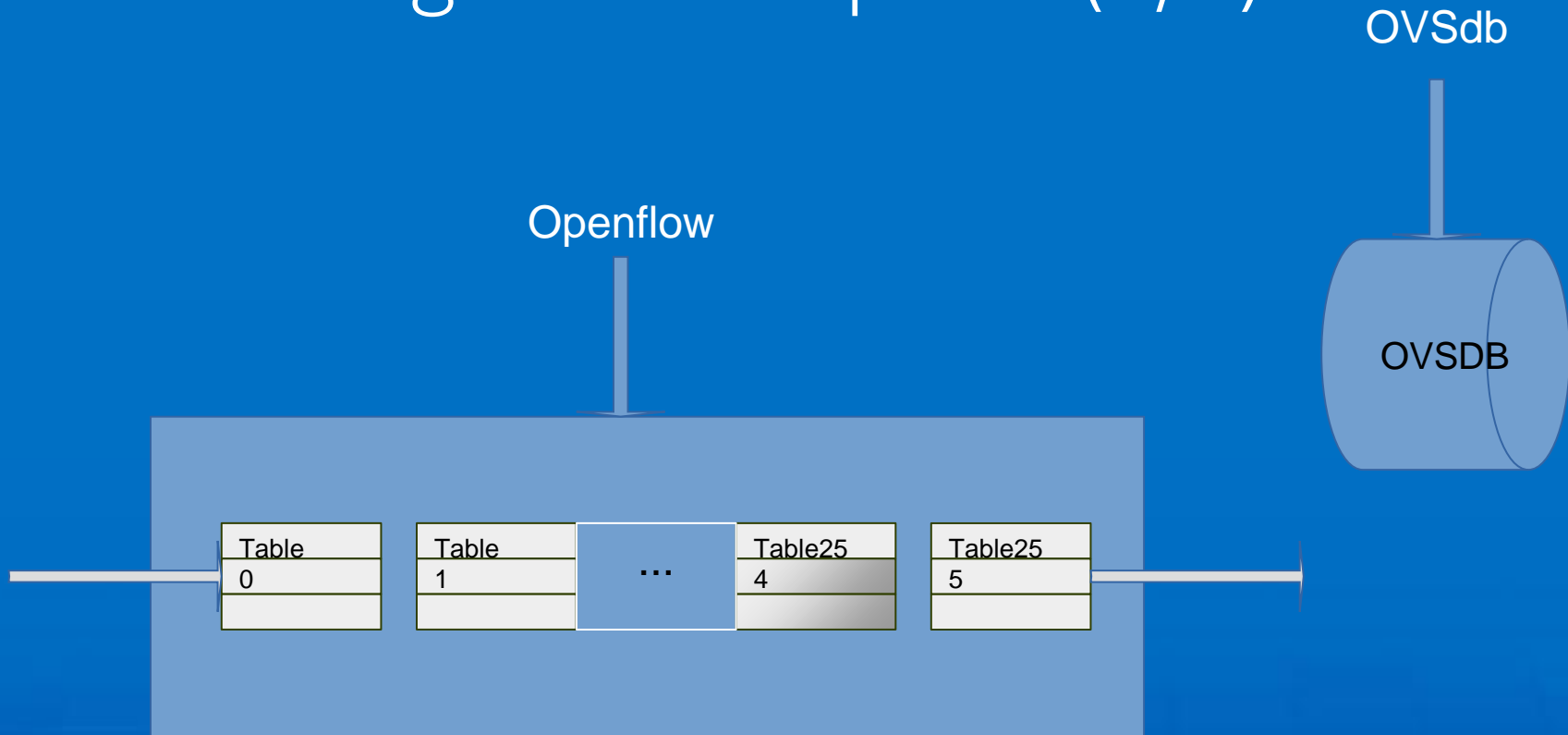
Demonstrate rocker switch, 10Gbe Intel NIC

Missing

- Asymmetric Paths (TX vs RX)
- Support masks
- Implementation for rocker switch (under development)

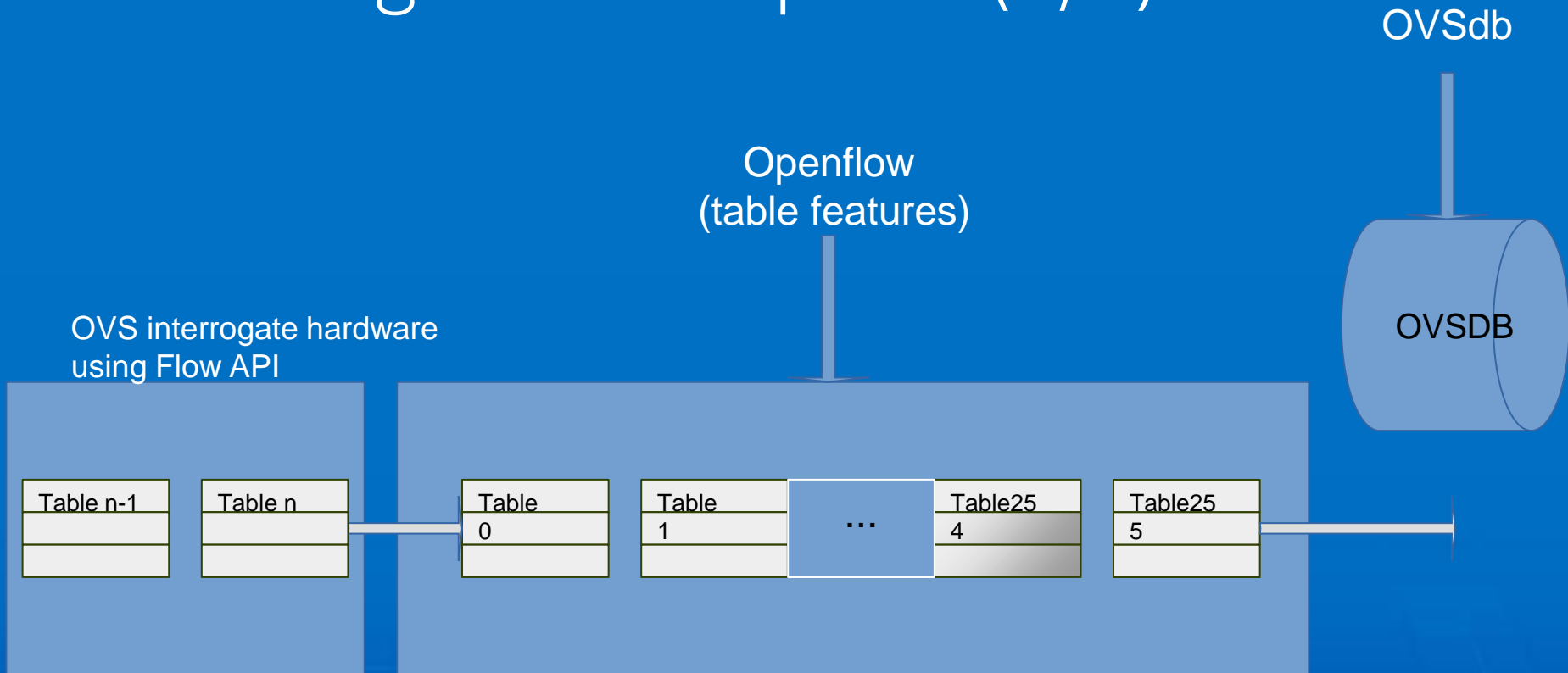
Questions

OVS Integration Proposal (1/2)



OVS (simple) view: 254 almost homogeneous tables + Table-254

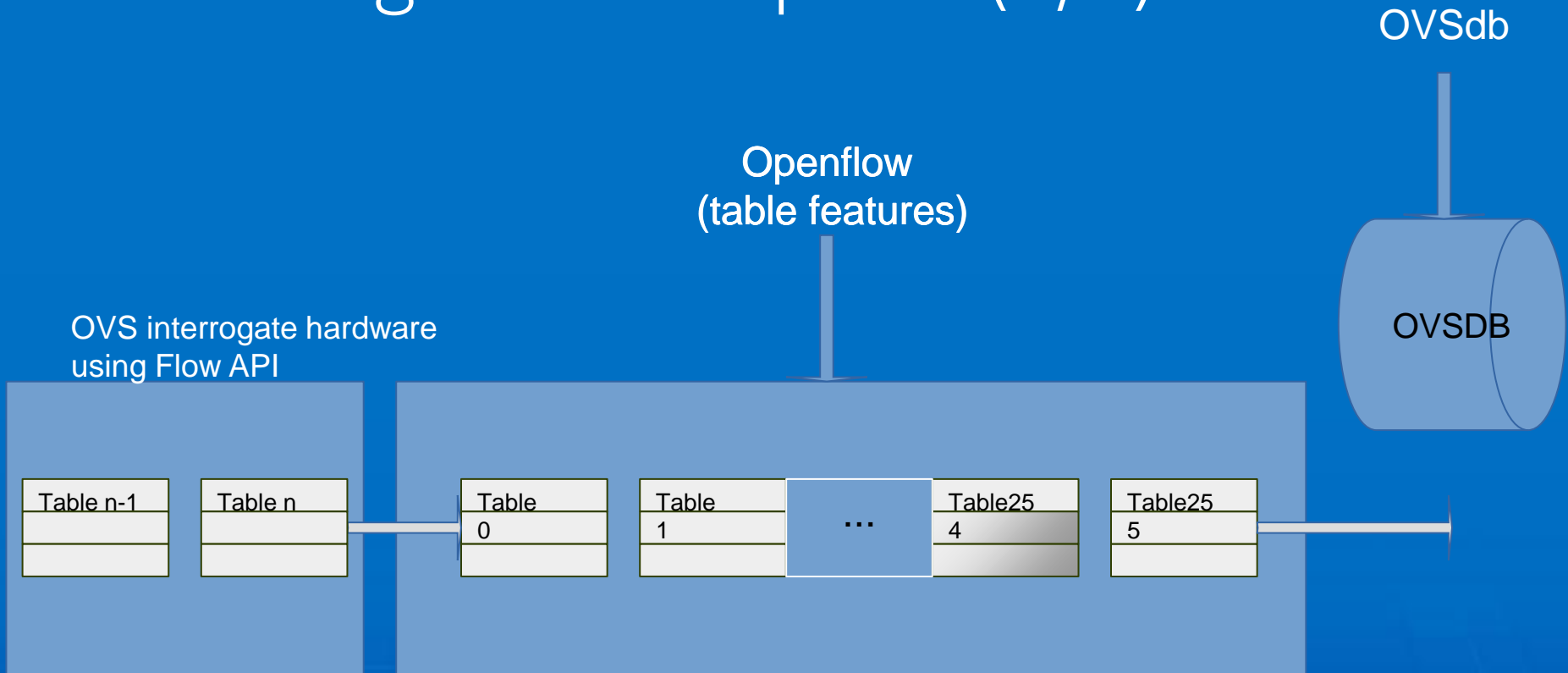
OVS Integration Proposal (2/2)



*OVS view: 254 **non**-homogeneous tables + Table-254 + n hardware tables*

Need to indicate hardware tables (likely via OVSdb) cost model

OVS Integration Proposal (3/2)



More Fun Questions: Flexible hardware support for actions/matches not in OF1.x