Forwarding Programming in Protocol-Oblivious Instruction Set

Jingzhou Yu, Xiaozhong Wang, Jian Song, Yuanming Zheng, Haoyu Song

Huawei Technologies
Paradigm Shift: From Black Box to White Box

- SDN is about network programmability
- Network programming ultimately requires device programming
  - Policy deployment ≠ device programming
- Use network device as a server
  - Program network devices as needed by the application and then operate and control them
Open Programmable Network Device

• Open programming interface
• Diversified target platforms
Abstract Forwarding Element Model

• Match-Action Table (MAT) Pipeline is a good abstraction model
• Network device programming is about defining the packet parser, tables, and the actions
• Parser and actions can be programmed using a special instruction set – POF-FIS
POF Programming Model
Protocol-Independent Programming Stages

Stage 1. Configuration
- High Level Language (e.g. P4 program & libraries)
  - Front-end Compiler
    - Intermediate Representation
      - Parse graph, control flow graph, and table dependency graph
        - Proposed new ONF activity
      - Back-end Compiler
        - Switch configuration
          - Switch
            - Packet Forwarding Engine

Stage 2. Run-time
- Control Plane
  - OpenFlow
    - Run-time API
    - Wire protocol
      - Add, modify, delete flow entries, etc.
- OpenFlow Switch
  - Packet Forwarding Engine
  - Run-time API
POF-FIS in POF Framework

- Actions are programmed in instruction blocks
- Actions are downloaded to FEs through POF south-bound interface
- Actions can be dynamically or statically linked to flow entries
**POF-FIS For Forwarding Abstraction**

- Packet parsing and editing
  - Headers and fields are identified by \{offset, length\}
- Allow active and stateful data path processing
- Flexible programming with control construction

<table>
<thead>
<tr>
<th>Category</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>EDITING</strong></td>
<td>SET_FIELD, ADD_FIELD, DEL_FIELD, ALG,</td>
</tr>
<tr>
<td></td>
<td>CALCULATE_CHECKSUM, SET_FIELD, UPDATE_CHECKSUM,</td>
</tr>
<tr>
<td></td>
<td>INC_FIELD, DEC_FIELD, OR_FIELD, SRL_FIELD,</td>
</tr>
<tr>
<td></td>
<td>SLL_FIELD, AND_FIELD, XOR_FIELD, NOR_FIELD, NOT_FIELD</td>
</tr>
<tr>
<td><strong>FORWARDING</strong></td>
<td>GOTO_TABLE, COUNTER, OUTPUT, GROUP,</td>
</tr>
<tr>
<td></td>
<td>MOVE_PACKET_OFFSET, SET PACKET OFFSET</td>
</tr>
<tr>
<td><strong>ENTRY</strong></td>
<td>SET_TABLE_ENTRY, ADD_TABLE_ENTRY, DEL_TABLE_ENTRY</td>
</tr>
<tr>
<td><strong>JUMP</strong></td>
<td>BRANCH, COMPARE, JUMP</td>
</tr>
<tr>
<td><strong>FLOW</strong></td>
<td>SET_FLOW_METADATA, GET_FLOW_METADATA, ORDER_ENFORCE</td>
</tr>
</tbody>
</table>
Use Case: POF-FIS for OAM

• Insert a custom OAM field to packet header at ingress
• Edit the OAM field in the network
• Remove the OAM field at egress
• OAM usage is programmable
  – Performance
  – Routing
  – Statistics
Characteristic of POF-FIS

• Flexibility

• Independence
  – North bound interface
  – Service & Application
  – Target platform

• Completeness
  – Any protocol
  – Direct device programming at IR level
Implementation: GUI/CLI Programming Interface

```
$> ADD PROTOCOL IPvX DesMAC.48 SrcMAC.48 EthType.16 OAM
   .16 SrcAddr.64 DesAddr.64
Created new Protocol (1)
$
$> ADD TABLE 567077098 FirstEntryTable MM 100 3
Created new table (0)
$
$> ADD TABLE 567077098 IPvXProcessTable LPM 100 5
Created new table (10)
$
$> ADD ENTRY 567077098.0 0 enable 3.0889.Ffff INS=APPLY_ACTION(SFT_FIELD.4.beef.ffff);GOTO_TABLE.10.0.0
Created new flow entry(0) in table(0)
$
$> ADD ENTRY 567077098.10 0 enable 5.112233.Ffffffff INS=APPLY_ACTIONS(OUTPUT.0.0x5.0.0.0)
Created new flow entry(0) in table(10)
$`
High Level Device Programming Languages

• P4
  – Optimized for hardware-based pipeline architecture

• C/Java
  – Enough mechanism for forwarding programming
    • Struct, Class, ...

• Other languages?
Conclusions

- POF-FIS represents a self-contained forwarding abstraction
- POF-FIS is the core components of the open network device programming IR
- POF-FIS is neutral to high level programming languages and target platforms
- POF-FIS can become an important part of OpenFlow 2.0 standard
THANK YOU

www.huawei.com

Copyright©2014 Huawei Technologies Co., Ltd. All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.