AIR Modifications/Additions for P4: part 2

David George

NETRONOME

Brief overview of previous changes

- Conditional expressions in control flow
- Parser
  - Setting metadata
  - Value + mask transitions
  - Variable control flow entry (defunct)
- Field lists + field list calcs
- Global and per construct source info

Summary of latest changes

- New constructs
  - registers
  - digests
  - external_actions
- Stacked headers fleshed out
- Plethora of new actions

Registers

- Equivalent to P4 Registers
- Three types: global, static and direct
- Declared as arrays (not direct)
  - Indexed in actions; index can be const, action data value or field value
- Field declaration similar to headers (no variable length)
- Example declaration:

```
type : register
class : static
table : table_name
instance_count : 16
fields
   - value_32 : 32
   - value_64 : 64
```

Registers (continued)

- P4 counters are declared as registers with 'counts' entry
- Both packet and byte counts are supported
- 'counts' is an array of count type + field
- 'counts' entries really for host discovery
  - Actions are explicit: count() and count_bytes()
- Continued example:

```
instance_count : 16
fields
    - value_32 : 32
    - value_64 : 64
counts:
    packets : value_32
    bytes : value_64
```


NETRONOME

Digests

- Dataplane driven events
- Includes name, numeric identifier and field list
  - P4 only declares the digest in the action: digest(1, my_field_list)
  - No naming support in P4
- Field list contains header + metadata fields
- Example declaration:

```
my_digest:
    type : digest
    identifier : 1
    field_list: my_field_list
```

External actions

- Used for calls out of PIF dataplane (think P4 black boxes)
- Declared in same way as regular actions, but have no implementation
- Parameter list as with regular actions
- Example declaration:

```
my_external_action:
    type : external_action
    parameter_list:
        - val32 : 32
```

Stacked headers

- Already represented with 'max_depth' field within headers
- In parser no reference to stack offset:
  - “extracts vlan” not “extracts vlan[latest]
- In expressions and matching referenced by header[*offset*] where offset is either a const or *latest*
  - e.g valid(vlan[0]) and vlan[latest].vid == 12
  - e.g match_on:
    vlan[1] : valid
    vlan[1].vid : exact

NETRONOME

Actions

- generate_digest()
- count(), count_bytes()
- pop(), push()
- bit_or(), bit_xor(), bit_and()
- truncate(), drop()
- copy_header()

Not implemented yet

- Packet duplication: cloning/reinjection
- Parser exceptions
- Parser value sets
- Meters
- ...