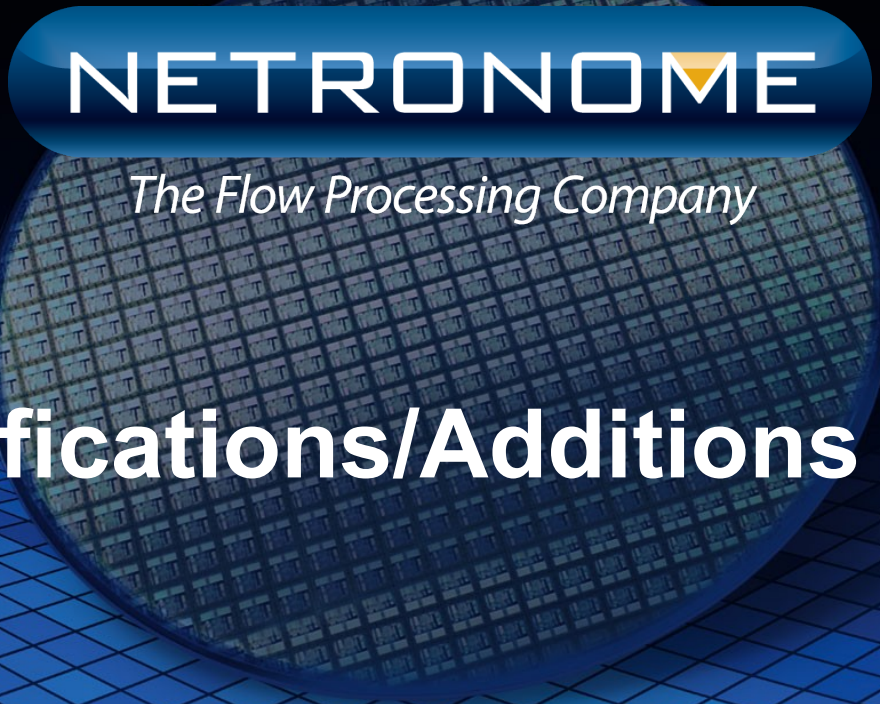

AIR Modifications/Additions for P4

David George

NETRONOME

Rationale

- Goal: re-use AIR/IRI Network Flow Processor implementation – extend it to run P4 code
- P4 HLIR to AIR already a good match
    - header definitions, table control flow, parsing
- Some P4 constructs not covered by AIR/IRI
- Other bits and bobs added for IDE integration

Missing IR constructs needed for P4 (Parser)

- Setting metadata within parse node
  - added “sets” key with array metadata refs
- Value + Mask select values
  - Tagged transition with optional mask field
    - start -> parse_ethernet [value=0xdead, mask=0xffff]
- Variable control flow entry point
  - parser can 'jump' to any node in ingress/egress flow

Missing IR constructs needed for P4 (Control flow)

- Conditional (i.e. not table) transition based on field values/header valid
  - _condition_0:
    - type : conditional
    - format : pyexpr
    - condition : (ingress_metadata.ingress_bypass == 0x0)
  - In digraph:
    - _condition_0 -> exit_control_flow [condition = false]

Missing IR constructs needed for P4 (General)

- Field lists
  - lag_hash_fields :
    - type : field_list
    - fields :
    - - l2_metadata.lkp_mac_sa ...
- Field list calculation
  - lag_hash :
    - type : field_list_calculation
    - algorithm : crc16
    - output_width : 8
    - inputs :
    - - lag_hash_fields

Bits and bobs

- A source info section:
  - Date, revision strings, source files, output file name…
    - source_info :
      - type : source_info
      - date : 2015/06/30 09:03:35
      - source_files : ...
- Source Filename + Lineno per object
  - allows tracing AIR output to original P4 source

Not addressed

- Parser value sets
- Parser exceptions
- Tag stacks
- Digests (Events)
- Meters
- Counters
- …