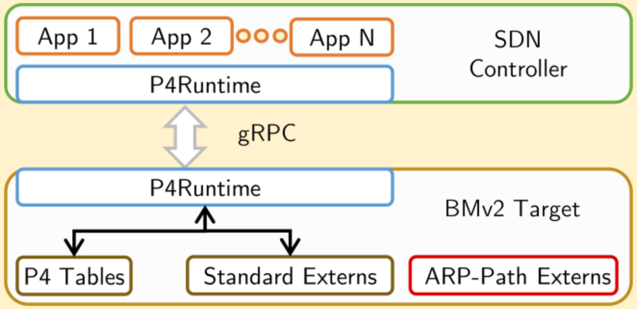
ARP-P4: A hybrid ARP-Path/P4Runtime switch

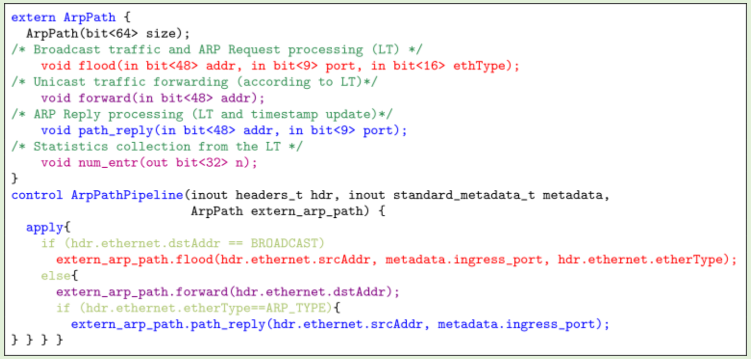
Motivations

- To explore the current capabilities of the P4 specifications to establish autonomous paths.
- To study the feasibility of hybrid SDN approaches such as ARP-Path/P4Runtime

Limitations



ARP-Path P4 Implementation



```
extern ArpPath {
  ArpPath(bit<64> size);
/* Broadcast traffic and ARP Request processing (LT) */
    void flood(in bit<48> addr, in bit<9> port, in bit<16> ethType);
/* Unicast traffic forwarding (according to LT)*/
    void forward(in bit<48> addr);
/* ARP Reply processing (LT and timestamp update)*/
    void path_reply(in bit<48> addr, in bit<9> port);
/* Statistics collection from the LT */
    void num_entr(out bit<32> n);
}
control ArpPathPipeline(inout headers_t hdr, inout standard_metadata_t metadata,
                        ArpPath extern_arp_path) {
  apply{
    if (hdr.ethernet.dstAddr == BROADCAST)
      extern_arp_path.flood(hdr.ethernet.srcAddr, metadata.ingress_port, hdr.ethernet.etherType);
    else{
      extern_arp_path.forward(hdr.ethernet.dstAddr);
      if (hdr.ethernet.etherType==ARP_TYPE){
        extern_arp_path.path_reply(hdr.ethernet.srcAddr, metadata.ingress_port);
} } } }
```

Conclusions

- ARP-P4 does not need to use any local controller.
- The ARP-P4 switch can interact with P4 Runtime and ARP-Path so ARP-P4 is a hybrid switch.
- The ARP-P4 performance is very similar than ECMP when the load has low load, while the performance on high load decreases due to BMv2 Target implementation.

Workshop P4EU September 2018

Joaquin Alvarez Horcajo - J.Alvarez@uah.es