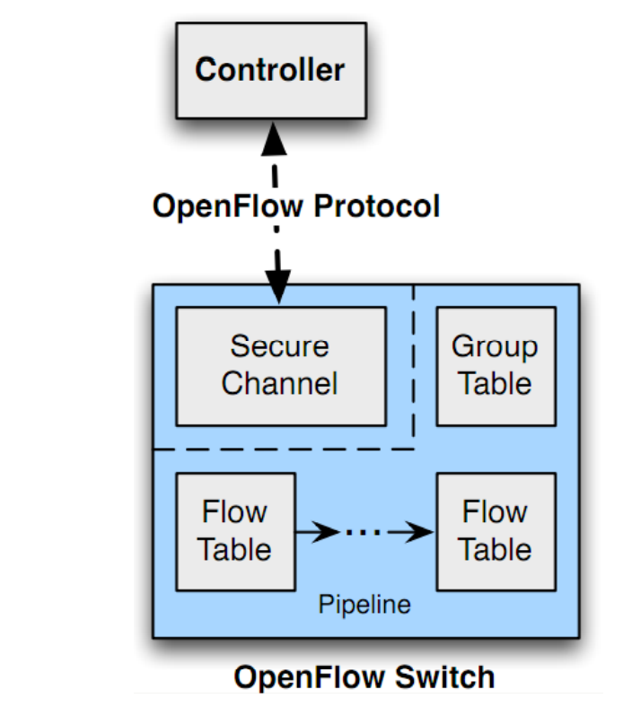
A 100 Gig OpenFlow 1.1 Switch

O. El Ferkouss, R. Matela, S. Correia, B. Boughzala, Y. Lemieux, R. Ben Ali,
M. Tatipamula, M. Lemay, O. Cherkaoui

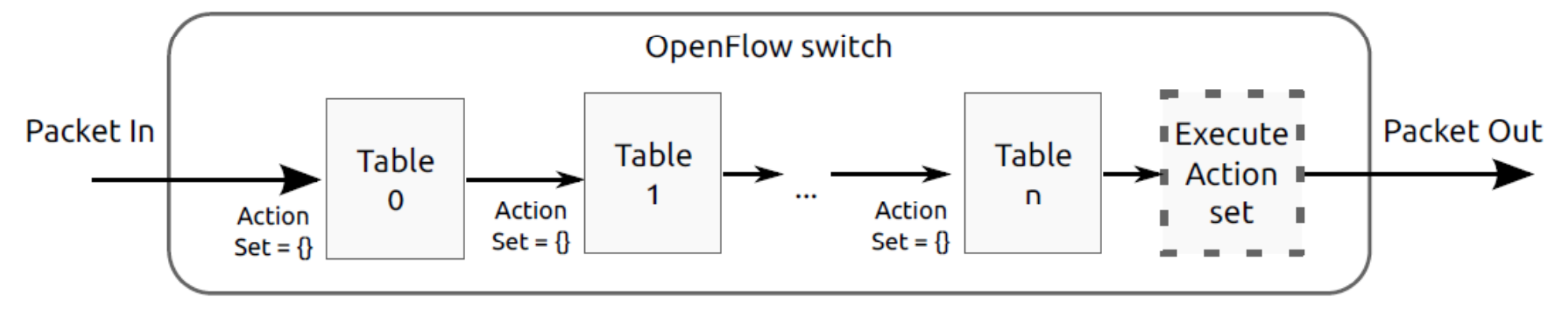
Goal of this Demo

- Show our implementation of Openflow over NP
- Programmability: Flexibility to push a new Openflow application over NP Openflow switch
- Time to change the forwarding behaviour

What is new in Openflow 1.1?

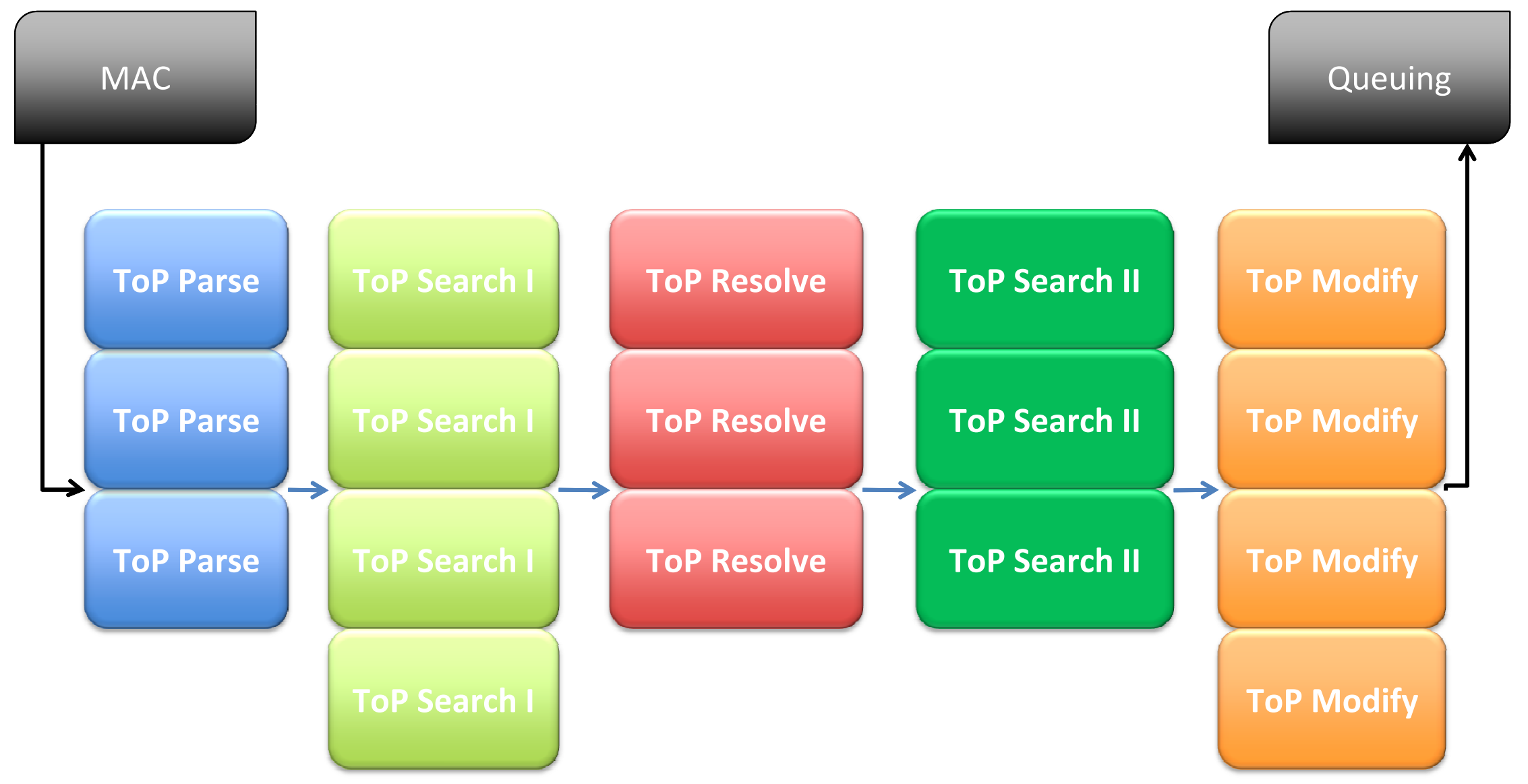


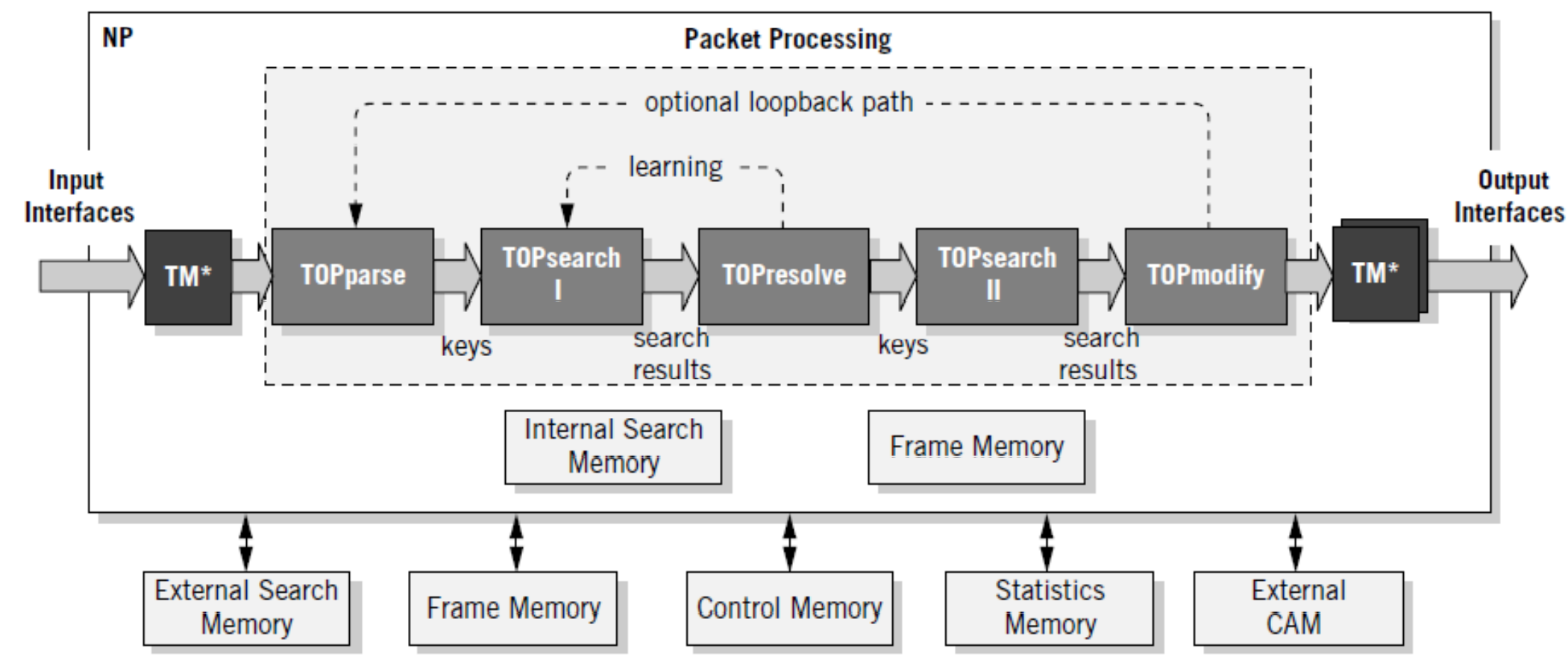
- Multiple Tables
- Tags and Tunnels
- Multi Paths
- MPLS, etc.



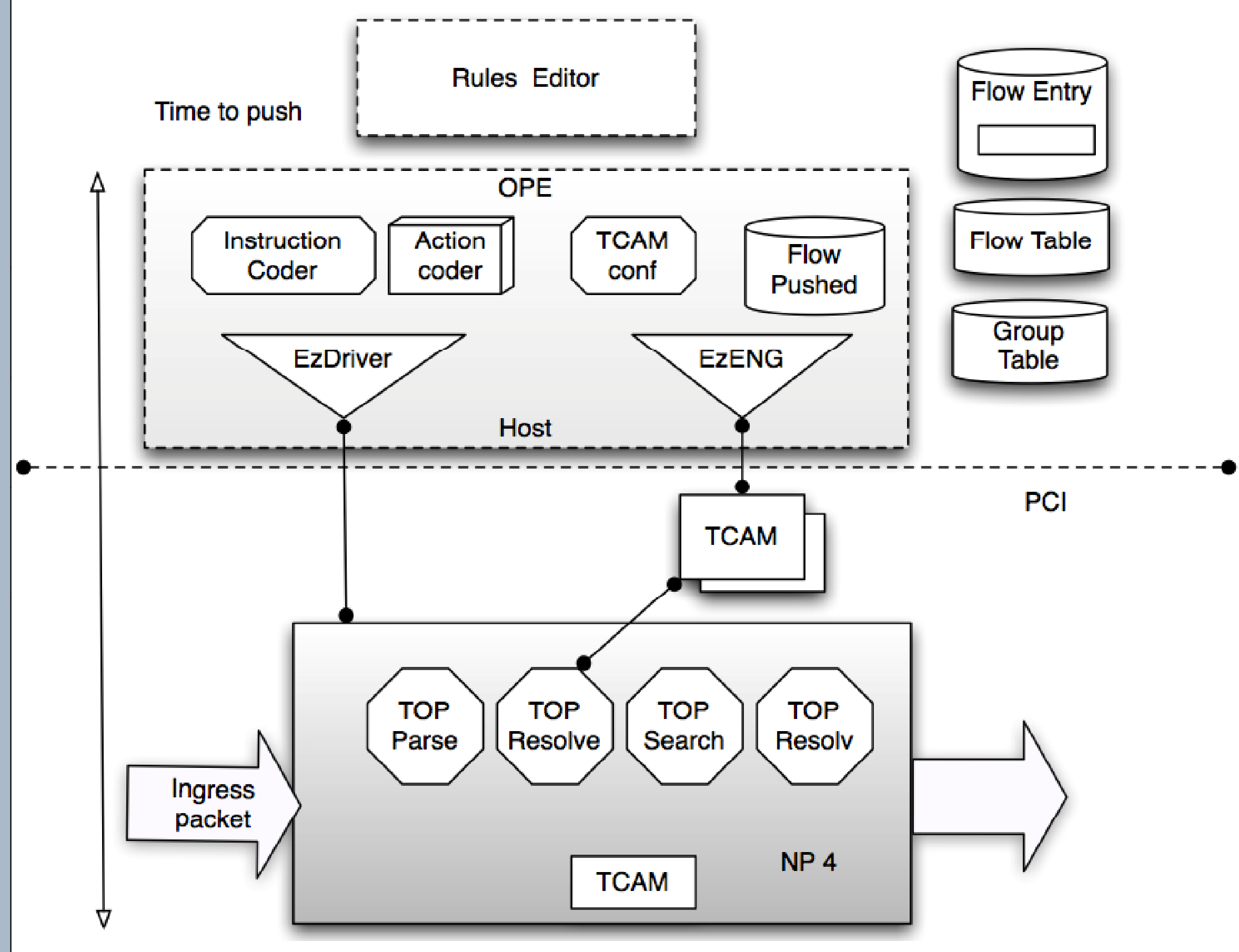
Why Ez-Chip NP4?

- •Both Flexibility and Performance
- •Flexibility : with Ez-Driver API
    - •Ez-update of the TCAM and SRAM memory structures on-the-fly
    - •Ez-update of the ToP micro code on-the-fly
- •Performance: Pipeline OF1.1 multiple tables over multiple ToPs





What is our implementation



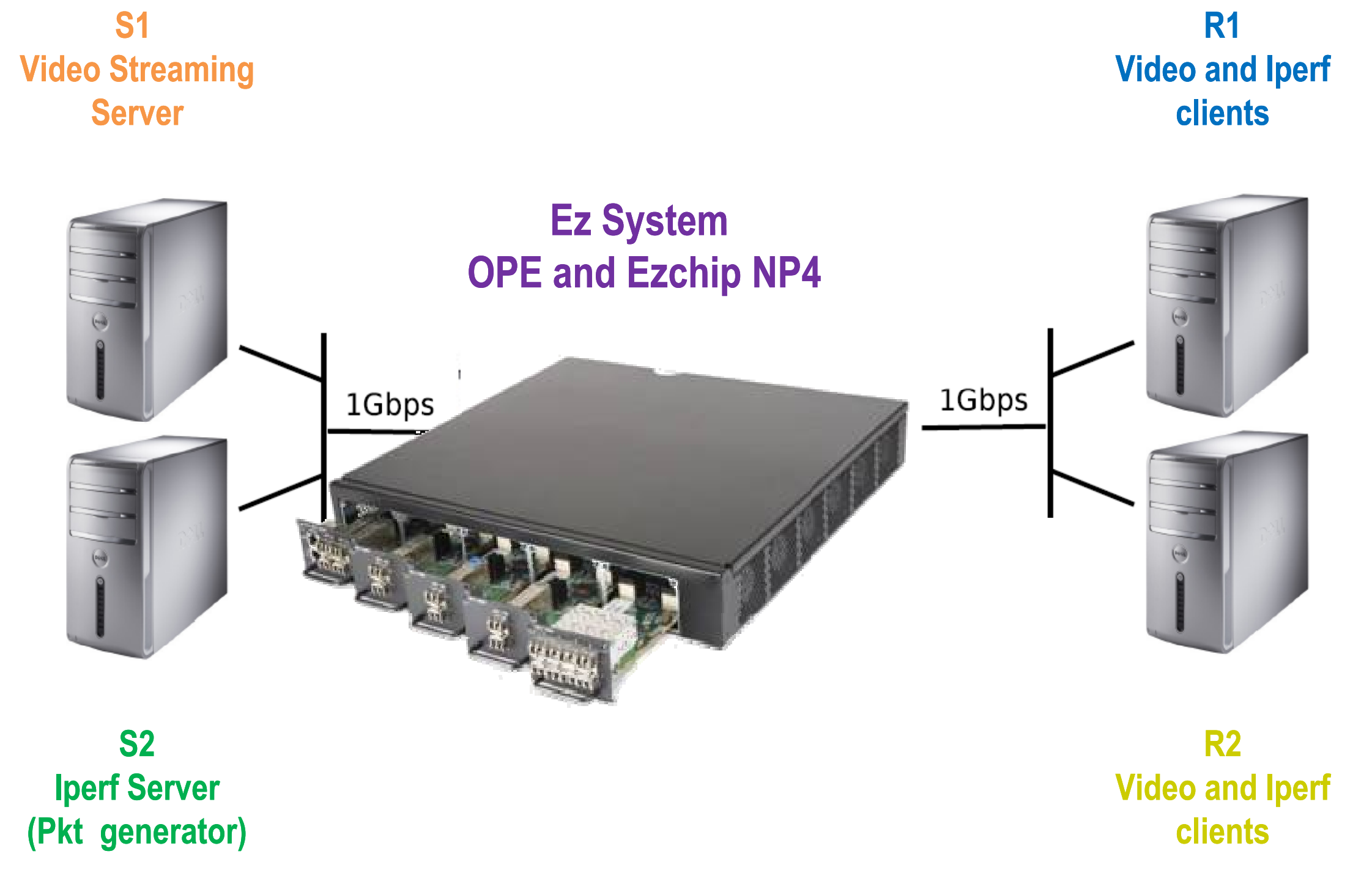
	Action	TOPModify Cycles
1	Push VLAN tag	30
2	Pop VLAN tag	30
3	Set VLAN tag	34
4	Push MPLS shim header	45
5	Pop MPLS shim header	45
6	Swap MPLS label	50
7	Decrement MPLS TTL	56
8	Decrement IPv4 TTL	65
9	Set source Ethernet address	65
10	Set destination Ethernet address	68

Table 1: List of implemented actions with their respective clock cycles

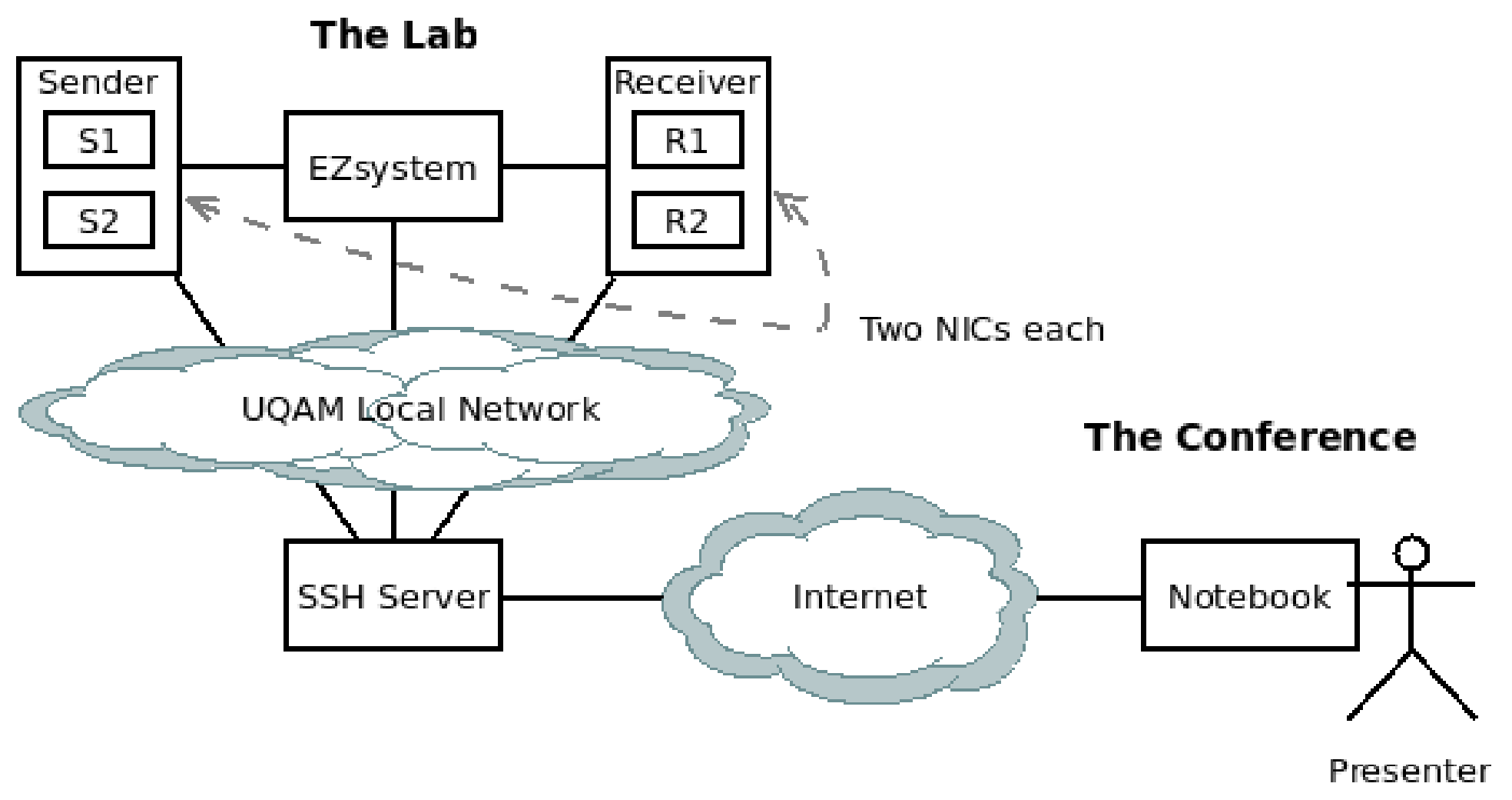
TOP	Cycles
TOP*Parse*	50
TOP$Search_1$	1 lookup in TCAM + 1 lookup in a hash table (key size: 3 bytes)
TOP*Resolve*	25
TOP$Search_2$	1 lookup in a hash table (key size: 4 bytes)
TOP*Modify*	it depends on the actions (see Table 1)

Table 2: Common cycles per TOP

Test Bed Overview

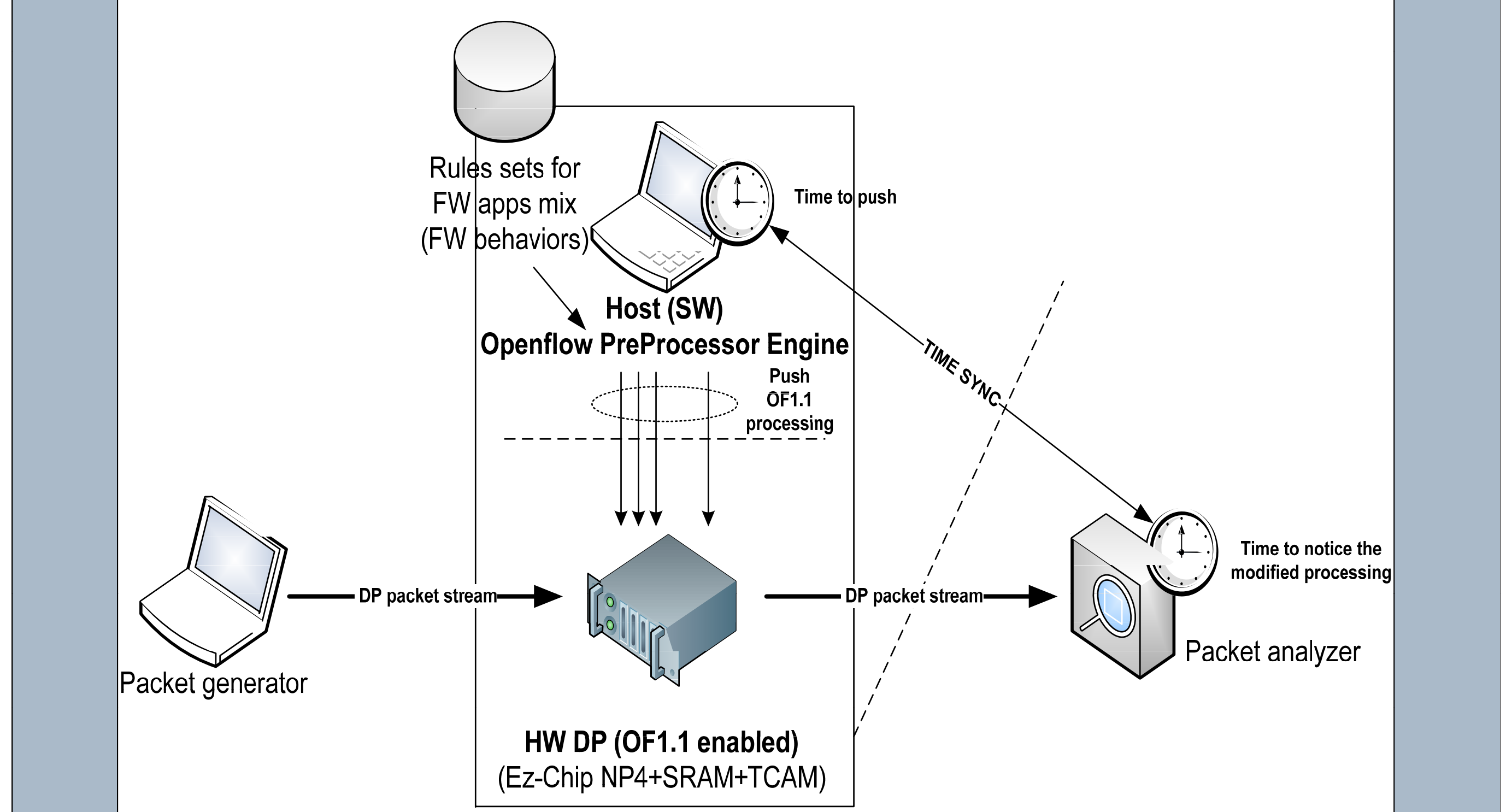


Test Bed Remote Access

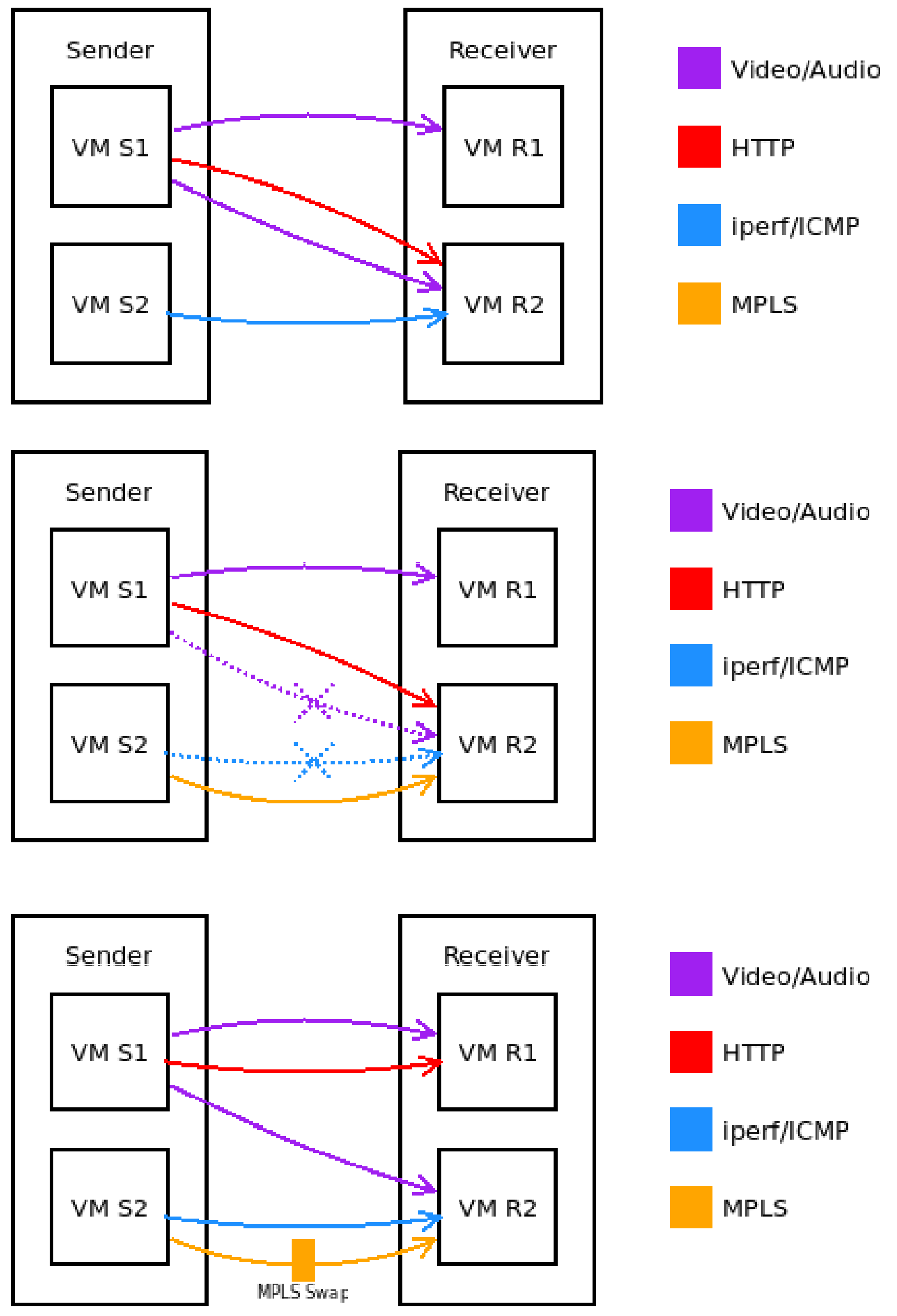


Memory Type	Update Time (1 entry)
TCAM	450us
Hash Tables	850us
Total	2.2ms

HW FW behavior update time measurement



Demo Scenario



Contact information

Prof. Omar Cherkaoui
Omar.cherkaoui@uqam.ca

Racha Ben Ali, Ph.D.
racha.ben.ali@ericsson.com