P4 apps at ONF

fabric.p4, spgw.p4, and INT

April 19, 2018

An Operator Led Consortium


Outline

- ONF - Who we are
  - ONOS, Trellis, CORD
- P4 apps at ONF
  - fabric.p4
  - spgw.p4
- Plans for INT

ONF – An Operator Led Consortium

AT&T

China unicom中国联通

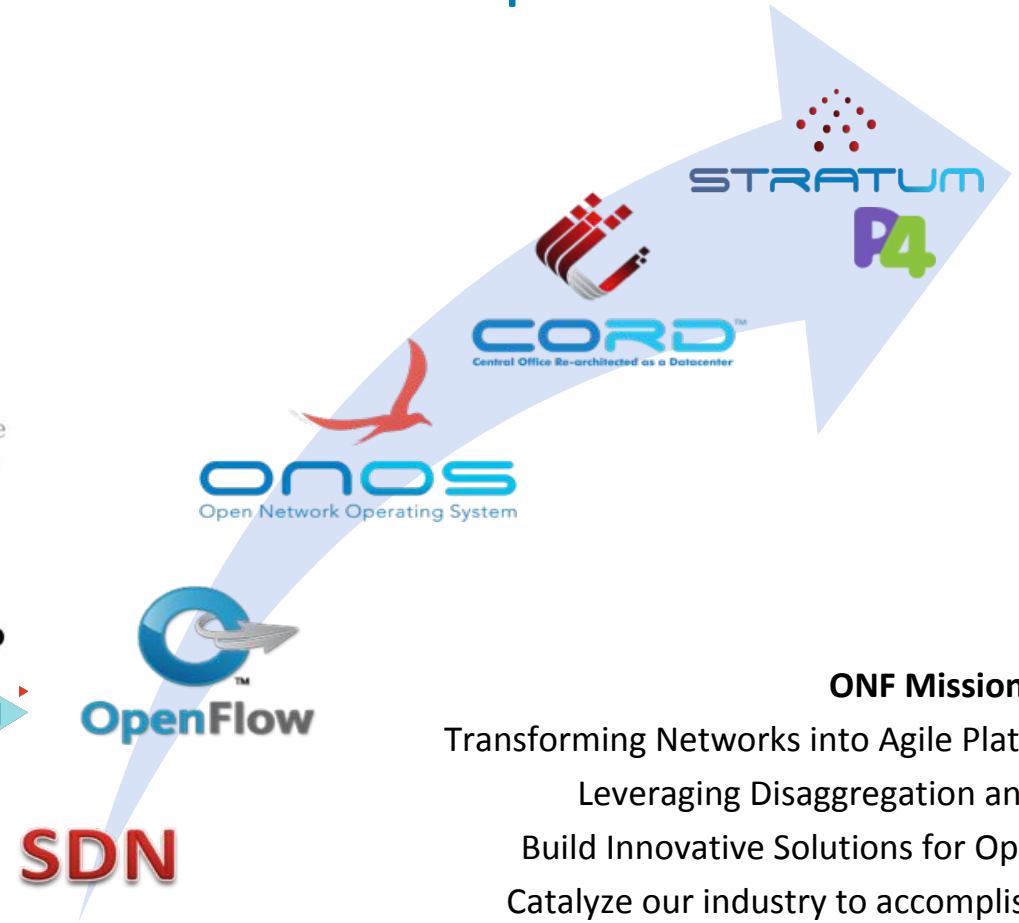
COMCAST

Deutsche Telekom

Google

NTT Group

Türk Telekom



"Nearly 40% of all end-customers will have service provided by … CORD by mid-2021"

Roz Roseboro
Heavy Reading

"70% of operators worldwide are planning to deploy CORD"

Michael Howard
IHS Markit

ONF Mission

Transforming Networks into Agile Platforms for Service Delivery

Leveraging Disaggregation and Open Source to

Build Innovative Solutions for Operator Networks and

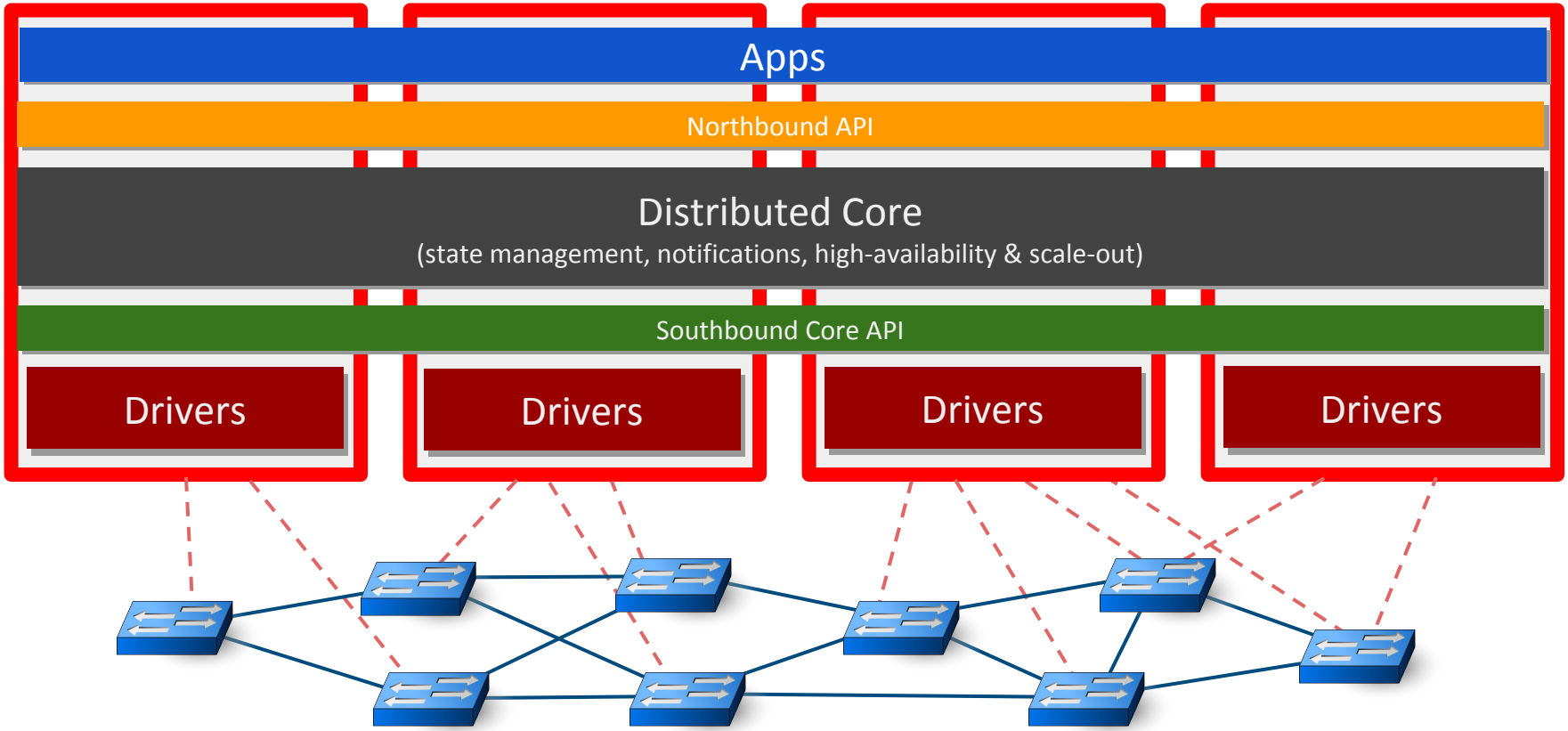
Catalyze our industry to accomplish this transformation

ONF

Open Network Operating System (ONOS)

- **Provides the control plane for a software-defined network**
- **Runs as a distributed system across many servers**
  - For scalability, high-availability, and performance
- **Open source project**
  - Created by ON.Lab and currently hosted by the Linux Foundation
  - Active community: 8,152 commits and 276 authors in last 2 years

ONOS Architecture

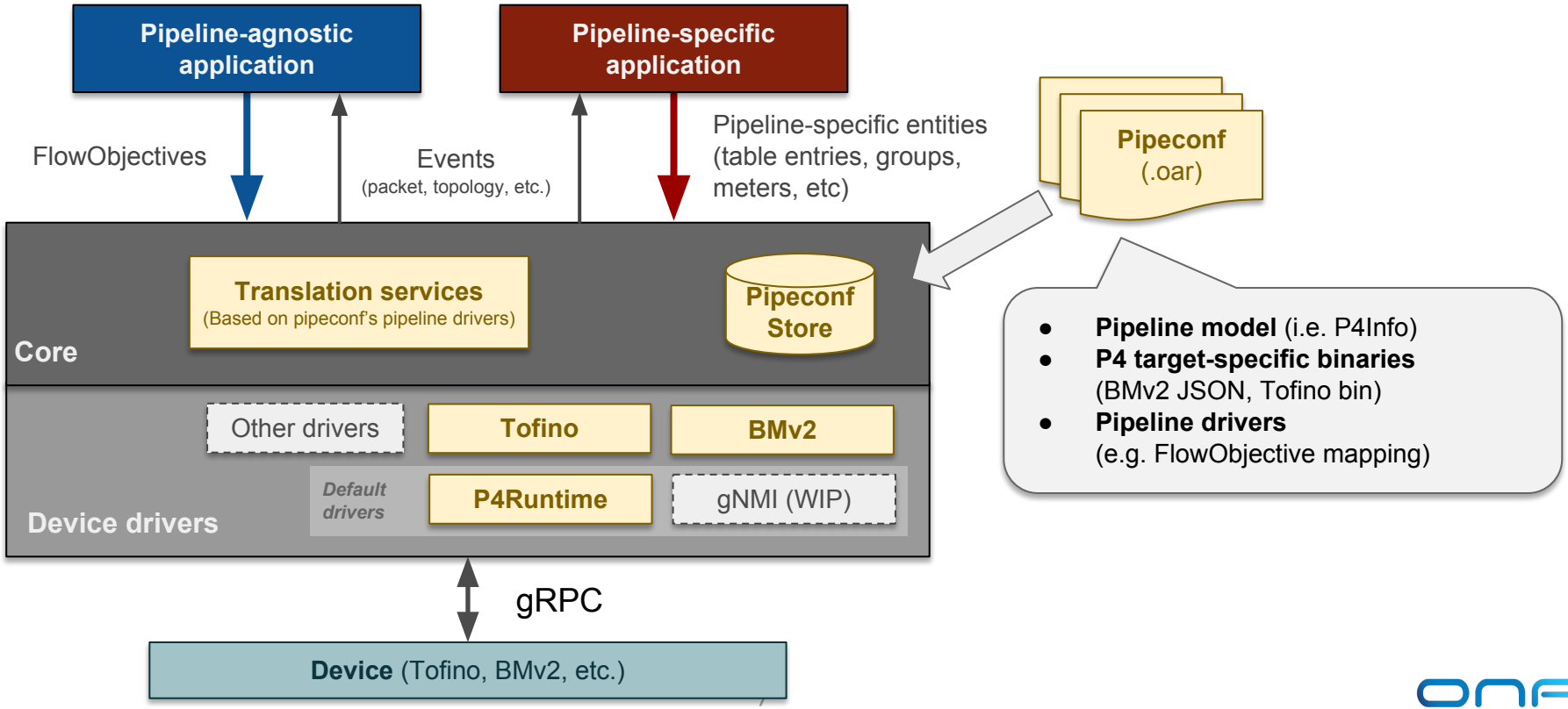


ONOS Network Programming Abstractions

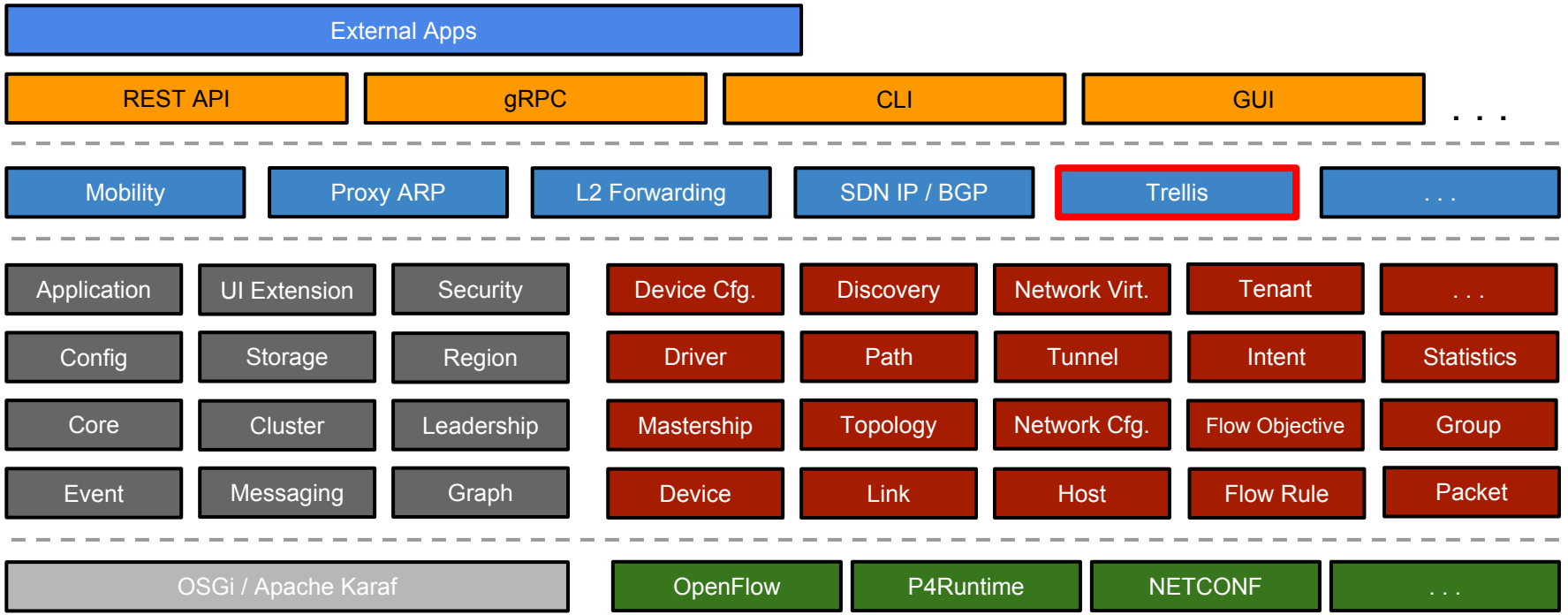
- **Control protocol independence**
  - E.g. same ONOS app can be used with OpenFlow and/or P4Runtime devices
- **Pipeline independence**
  - ONOS apps can perform flow programming using different APIs:
    - FlowRule API - table-specific entries
    - FlowObjective API - ONOS logical pipeline, mapped to device pipeline by drivers
- **Extensibility for new data plane capabilities**
  - FlowRule API support either:
    - Pre-defined standard headers and actions (e.g. OpenFlow match/actions)
    - Custom headers and actions (e.g. as defined in a P4 program)

P4 support in ONOS

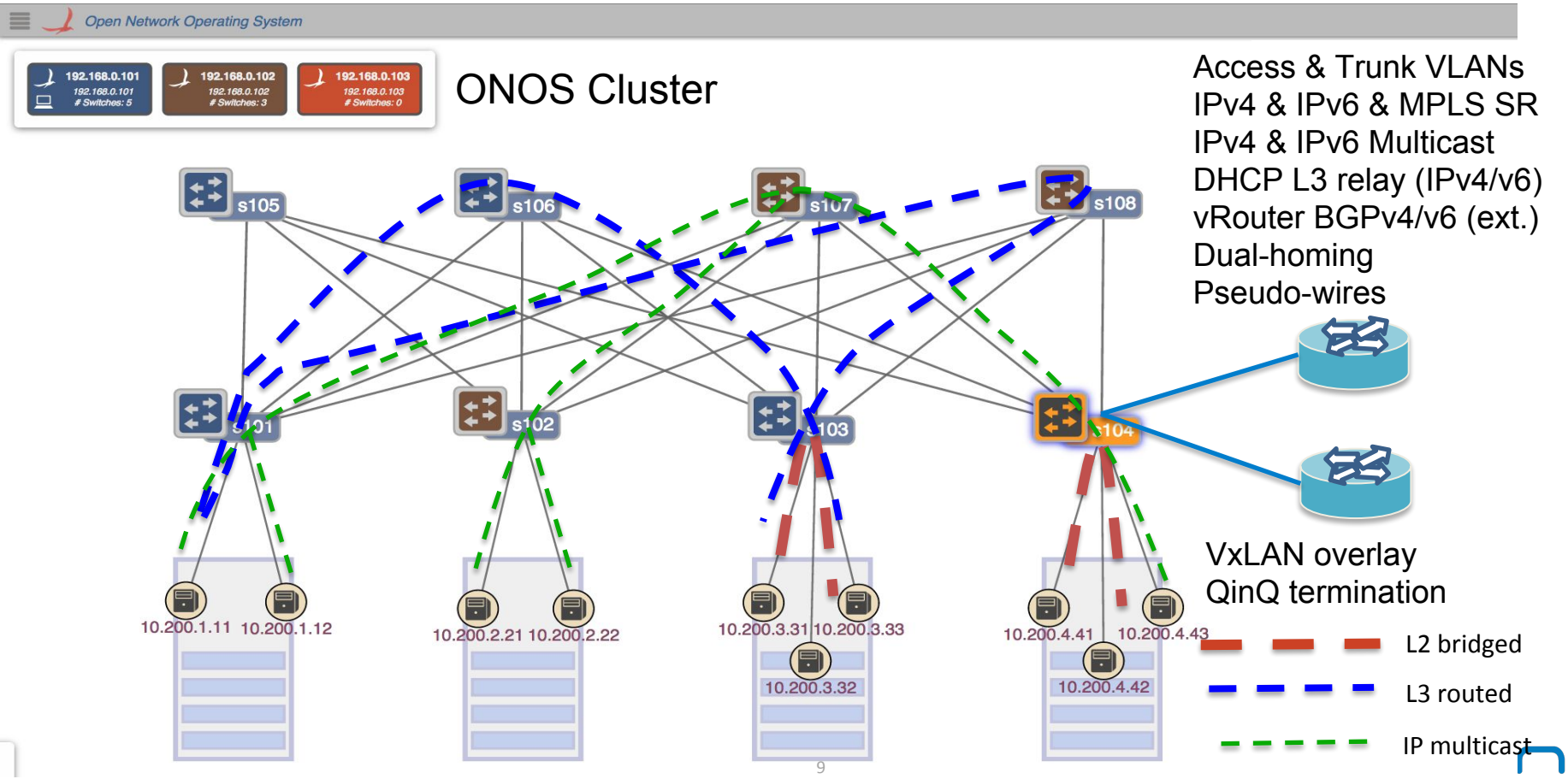
Control *any* pipeline, with *any* app



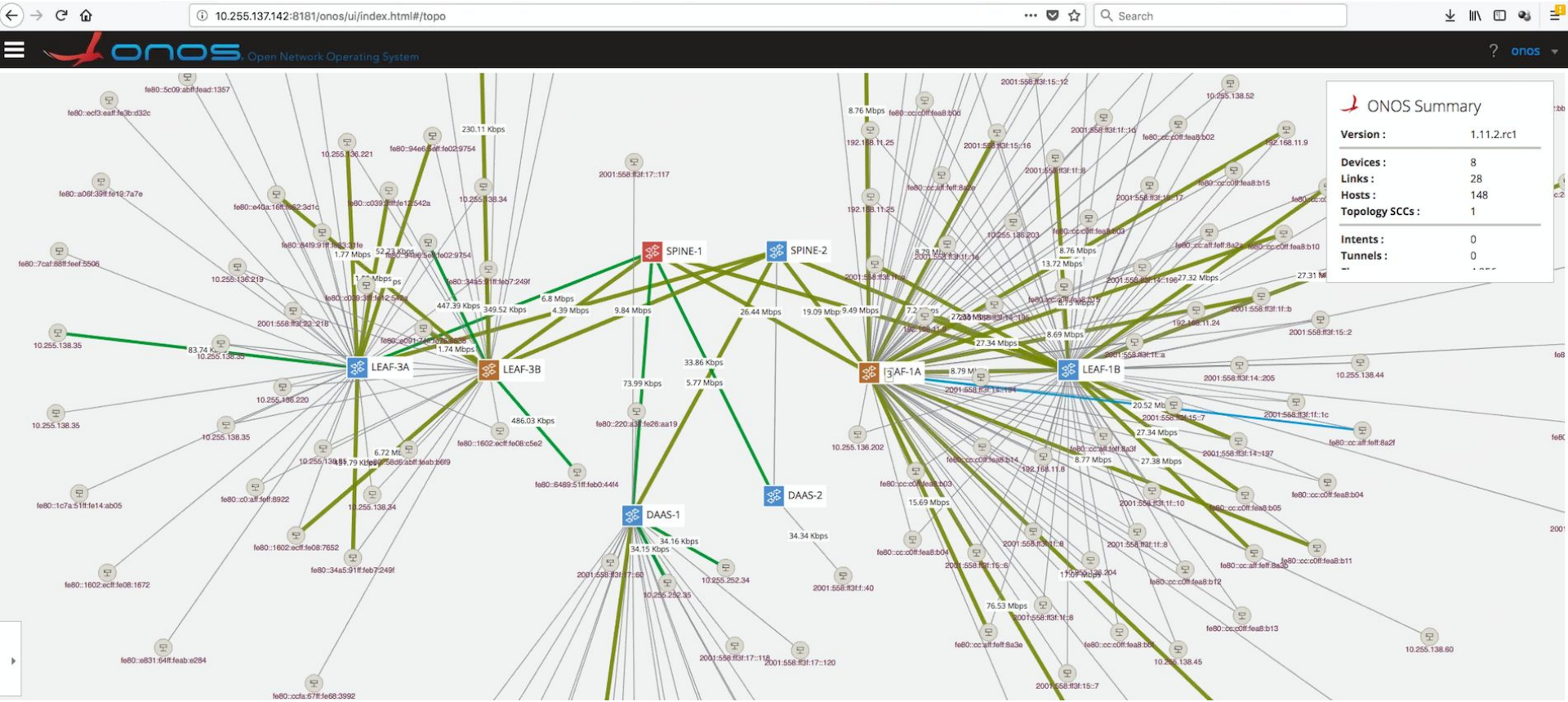
ONOS Core Subsystems



Trellis – Multi-purpose Leaf-Spine Fabric

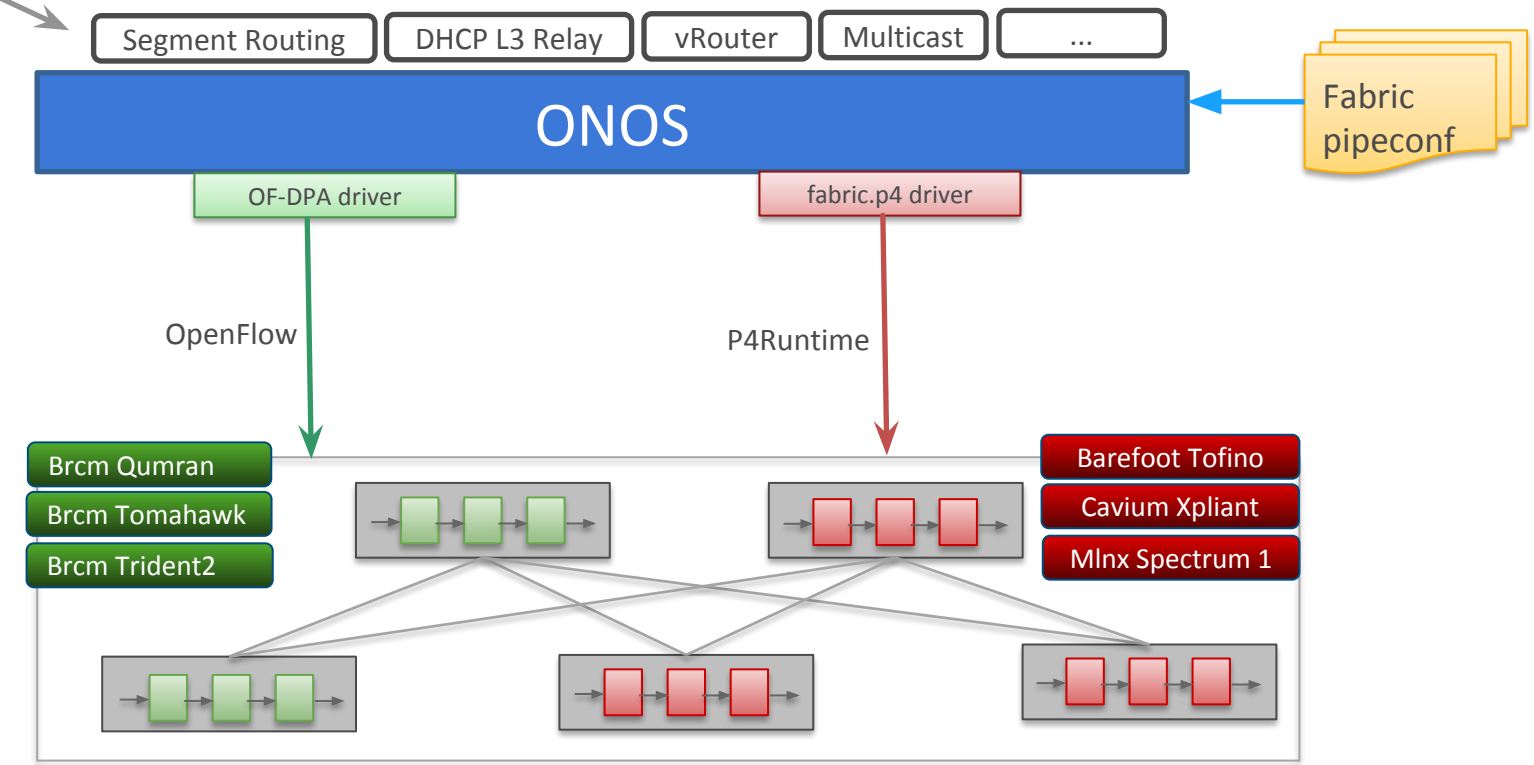


Trellis @ Comcast



Trellis & P4

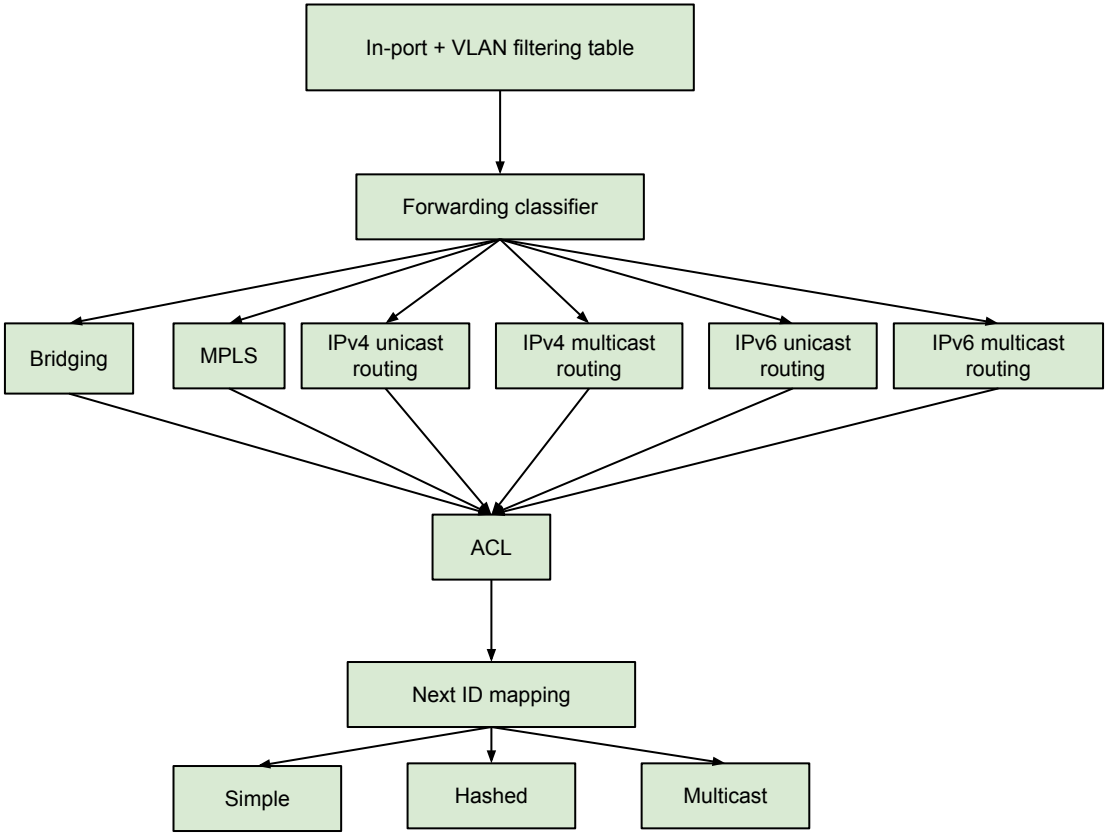
Same set of Trellis applications on ONOS



fabric.p4

- **P4 implementation of the Trellis underlay reference pipeline**
  - Inspired by Broadcom OF-DPA
  - Tailored to Trellis needs (fewer tables, easier to control)
  - Work in progress:
    - Tested support for L2 bridging, IPv4 routing, MPLS segment routing
- **Goal: bring more heterogeneity in Trellis with P4-capable silicon**
  - Both programmable and fixed-function
  - Provided control via P4Runtime
- **Open-source implementation based on P4_16**
  - https://github.com/opennetworkinglab/onos/.../fabric.p4
  - Depends only on open-source libraries (v1model.p4)
  - Can compile and test on BMv2, but needs few private tweaks to compile on HW

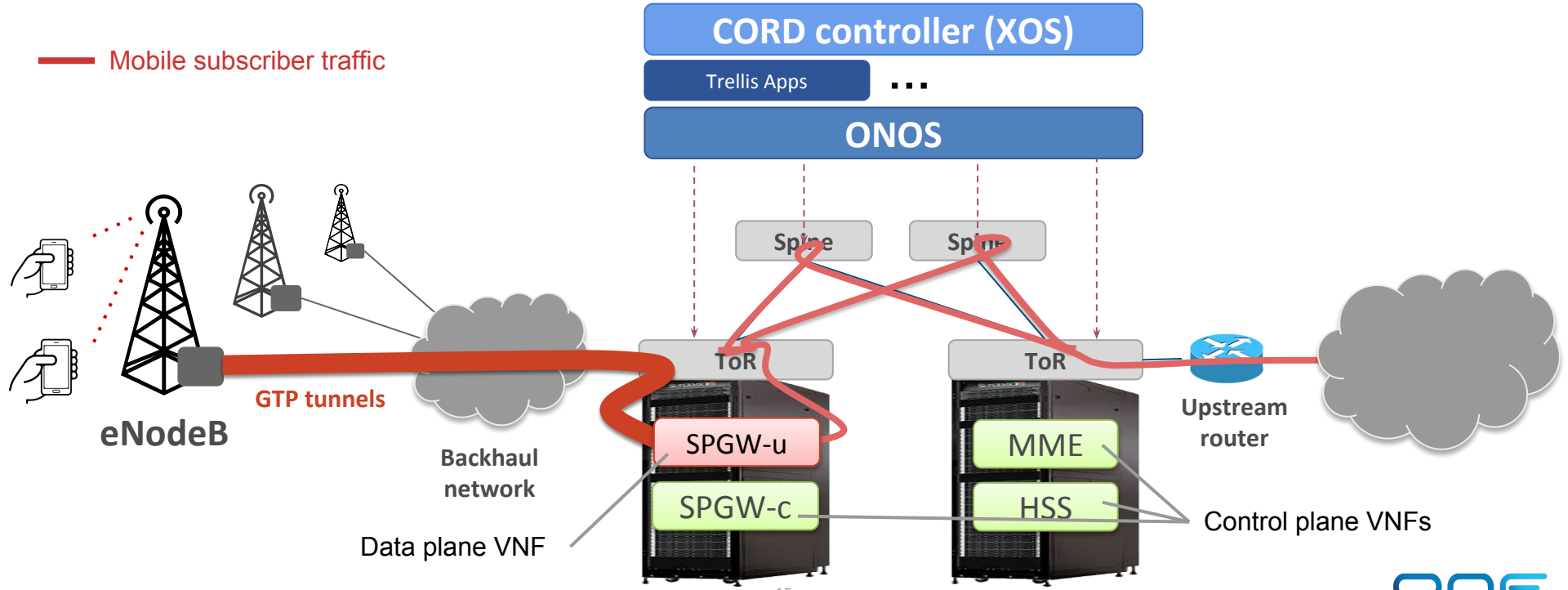
fabric.p4 pipeline



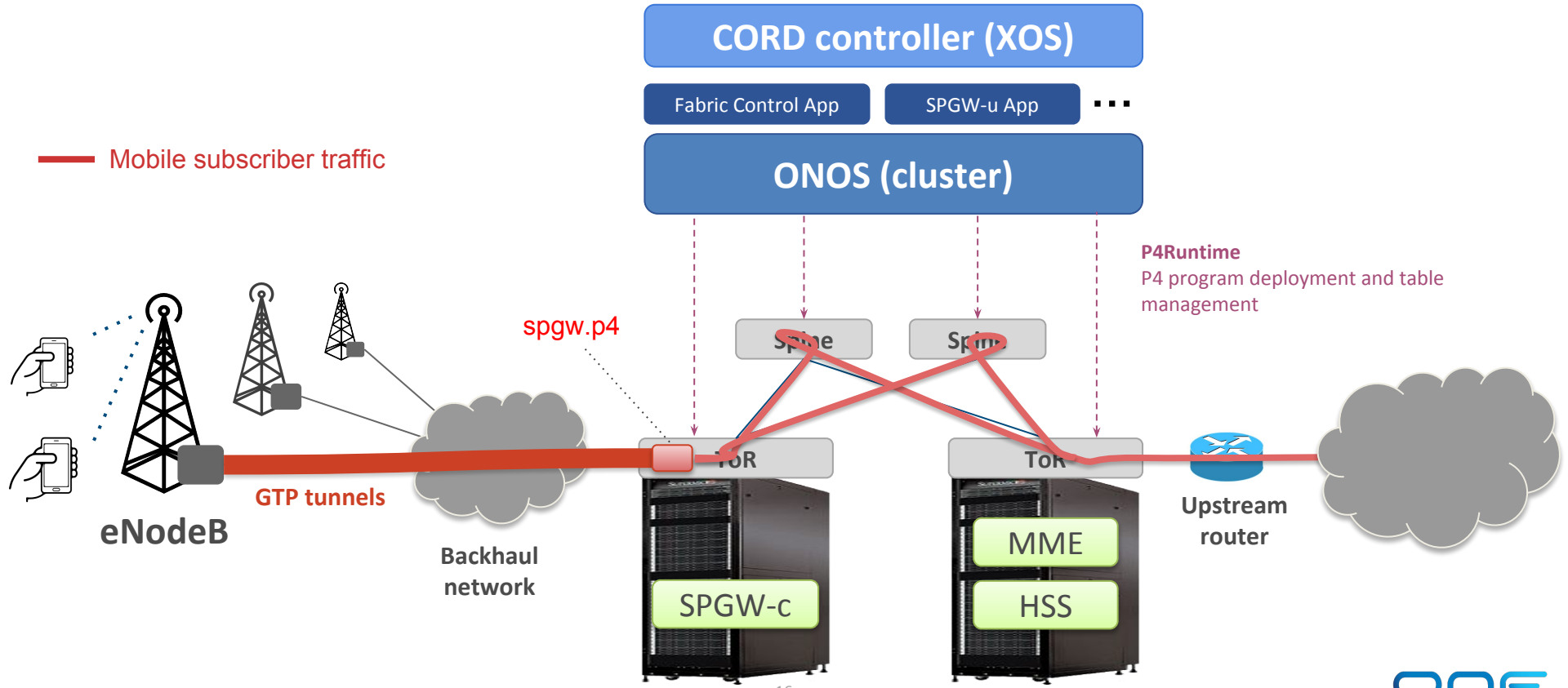
Extending fabric.p4 beyond forwarding:
Implementing the 5G SGW and PGW in M-CORD with P4

M-CORD

- **CORD**: NFV platform for the Telco central office
- **M-CORD**: CORD **M**obile profile (other profiles exist, e.g. for **R**esidential access)



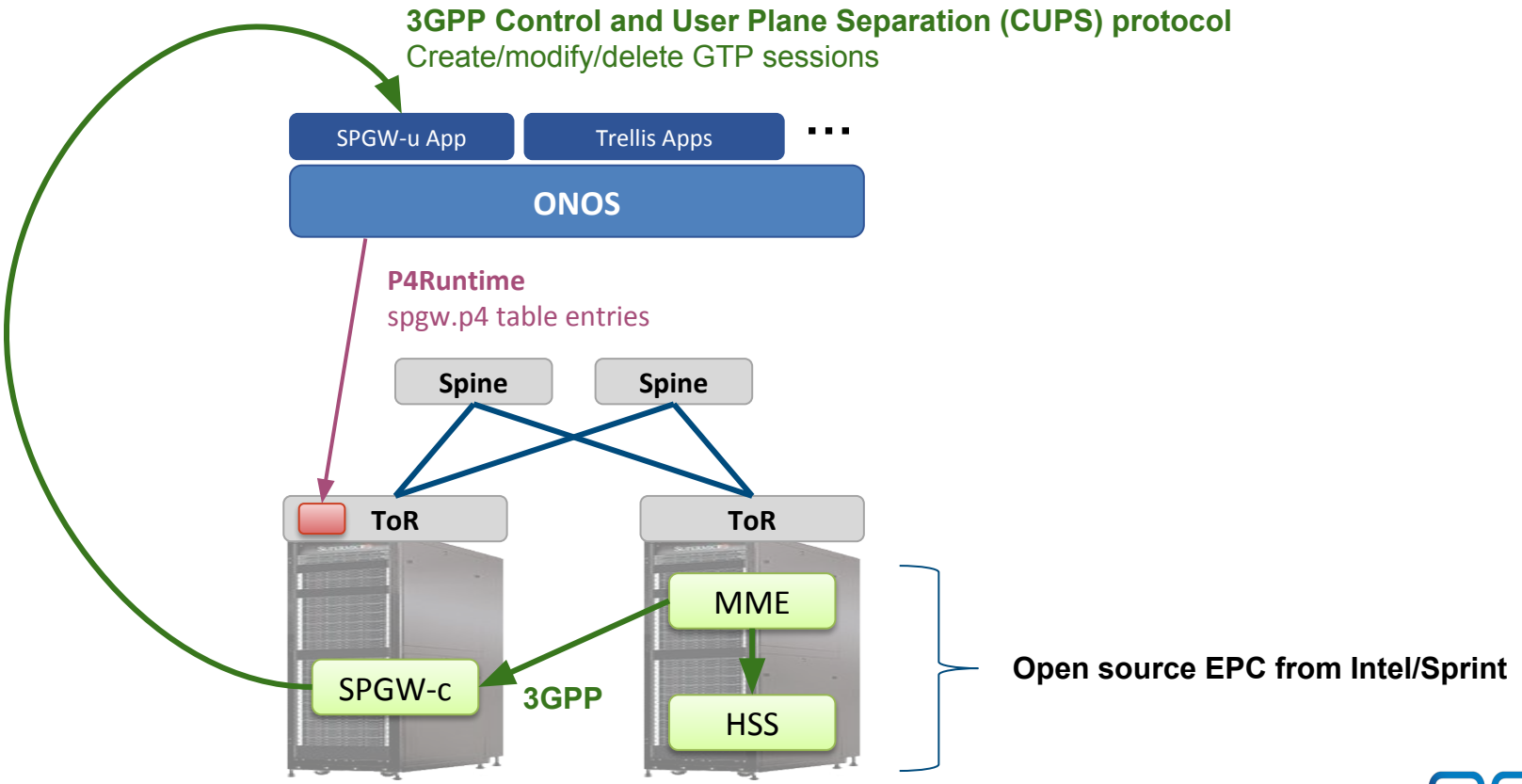
M-CORD with P4 fastpath



spgw.p4

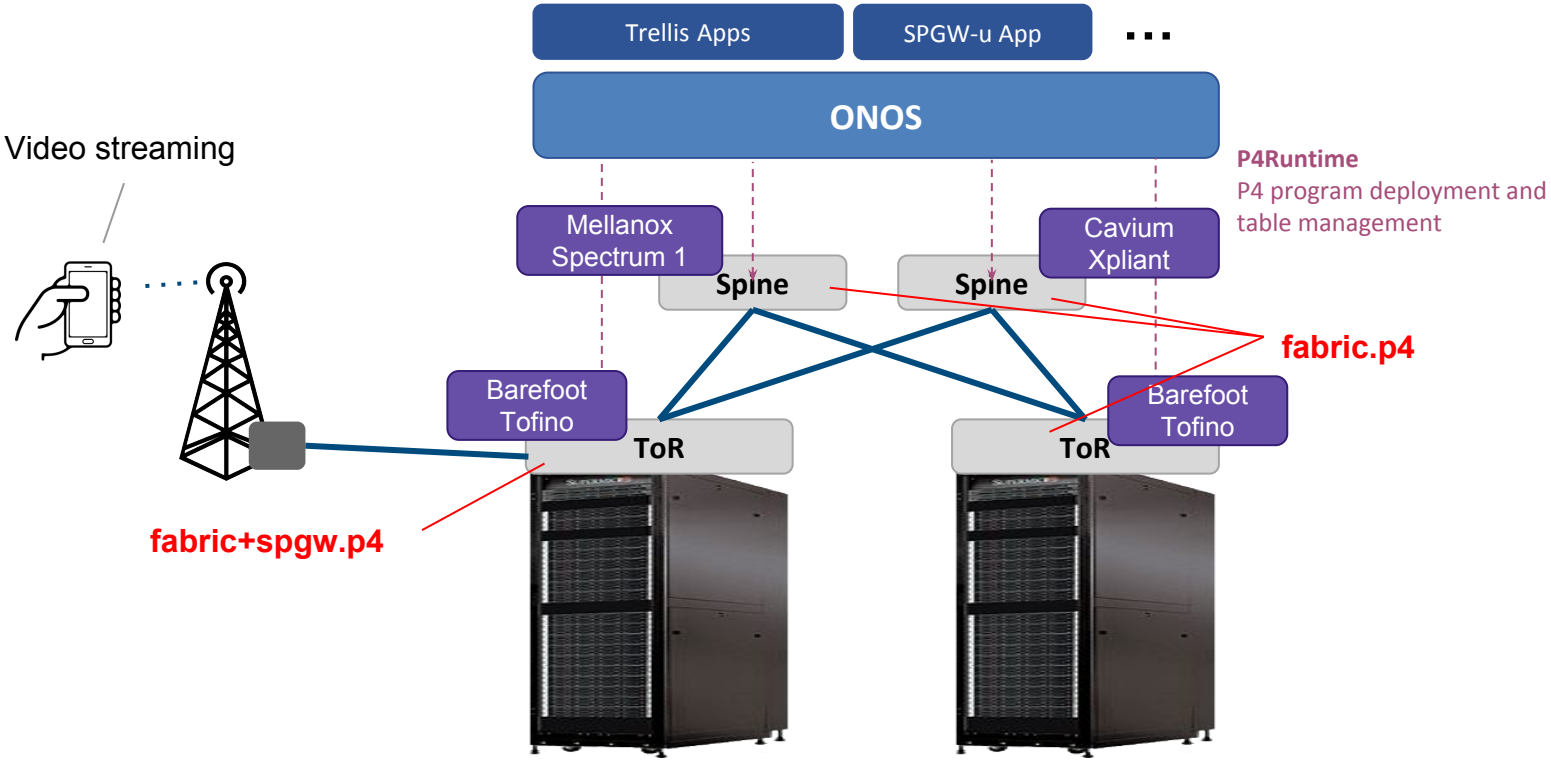
- **PoC P4 implementation of the SPGW-u data plane**
  - ~300 lines of P4_16 code
  - Integrated with fabric.p4
  - https://github.com/opennetworkinglab/onos/.../spgw.p4
- **Good enough to demonstrate end-to-end connectivity**
  - Support GTP encap/decap, filtering, charging functionalities
- **Important missing features (future work)**
  - *Downlink buffering during handovers:* cannot describe in P4, need cooperation of CPU and external storage
  - *Downlink QoS:* rate-limiting easy in P4, but cannot describe scheduling (e.g. per-subscriber/per-bearer hierarchical scheduling)

SPGW-u App


3GPP Control and User Plane Separation (CUPS) protocol
Create/modify/delete GTP sessions
SPGW-u App
Trellis Apps
...
ONOS
P4Runtime
spgw.p4 table entries
Spine
Spine
ToR
ToR
MME
SPGW-c
3GPP
HSS
Open source EPC from Intel/Sprint

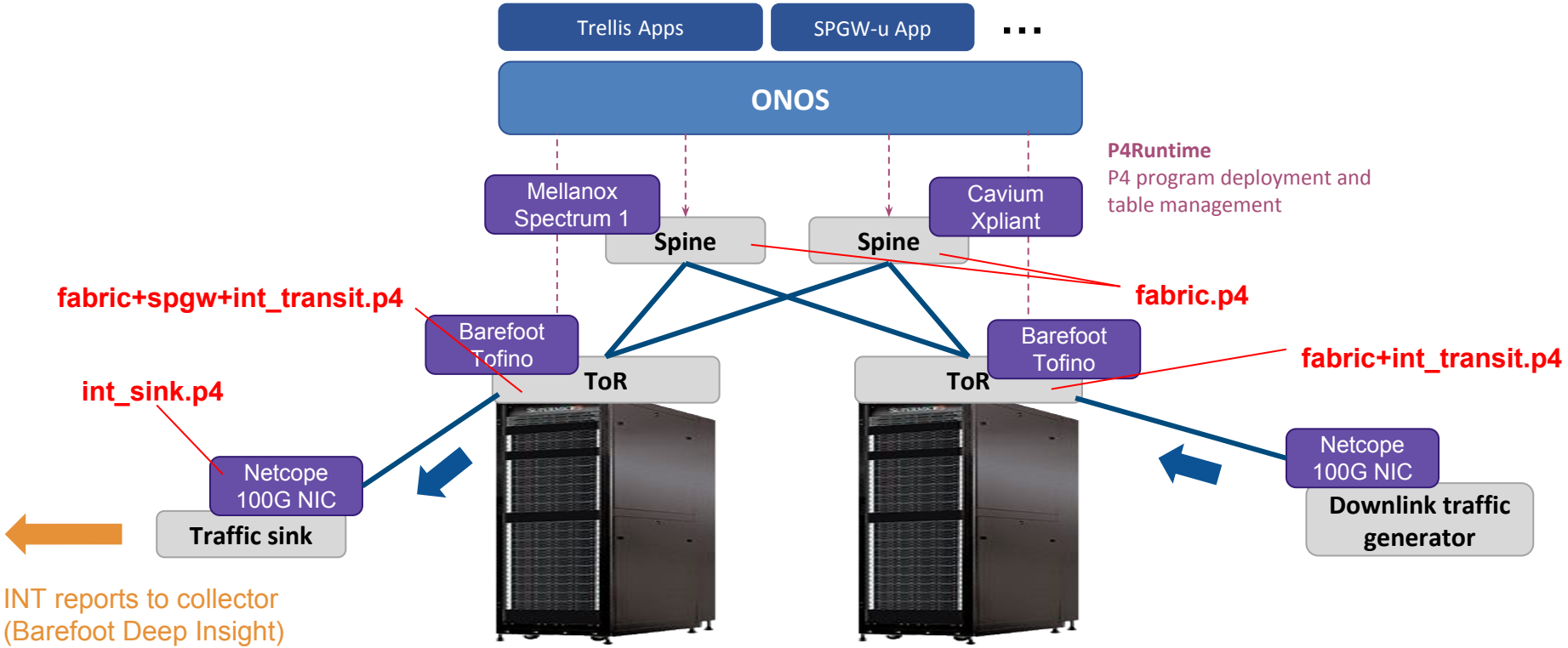
Demo 1 @ MWC and ONS 2018

End-to-end connectivity



Demo 2 @ ONS 2018

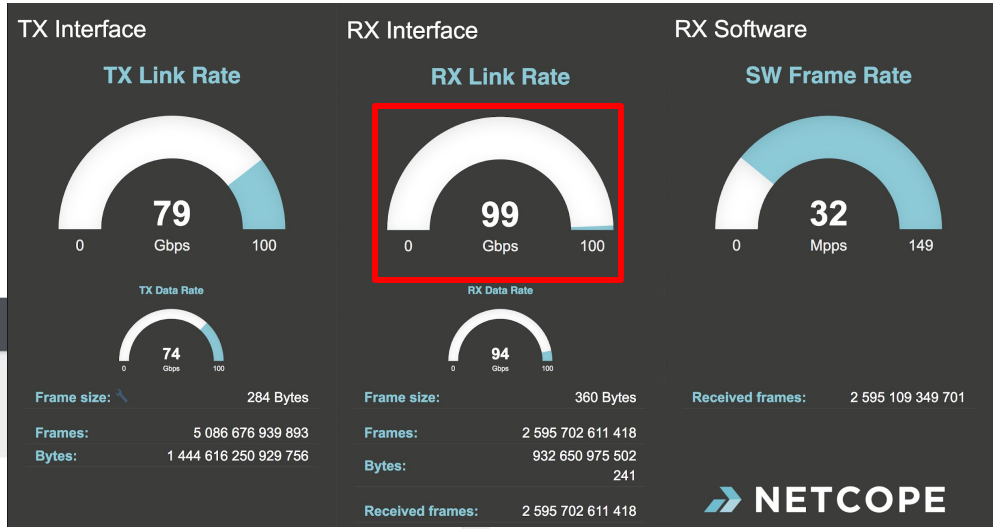
spgw.p4 performance with INT



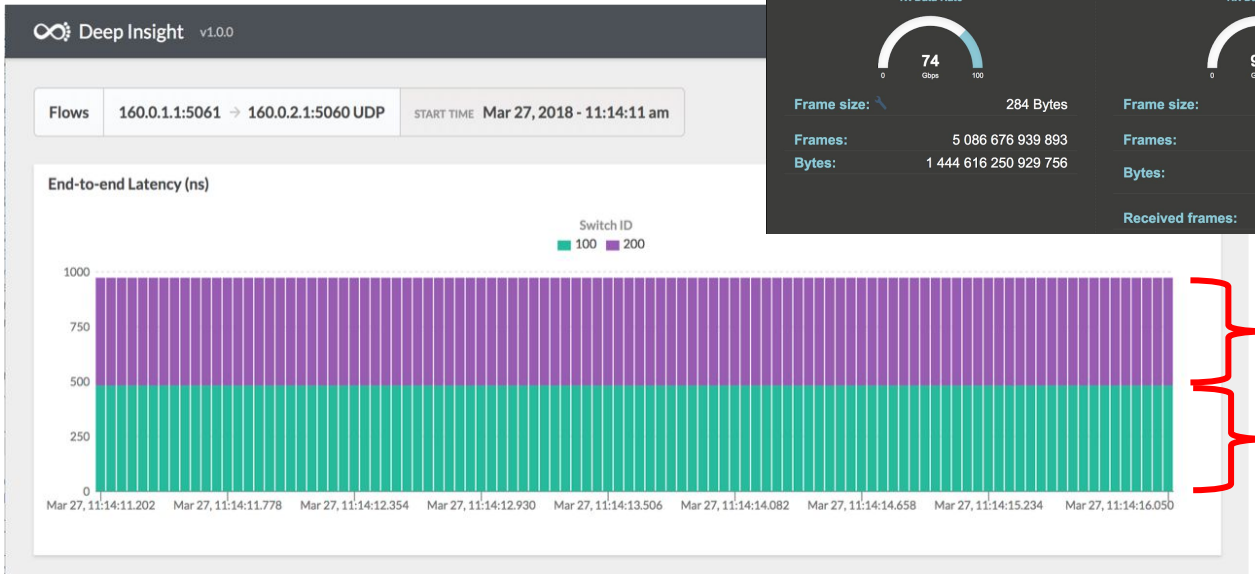
Demo 2 @ ONS 2018

Overhead due to GTP and INT headers

Throughput



Hop latency



~490ns to perform GTP encap plus forwarding (ToR 1)

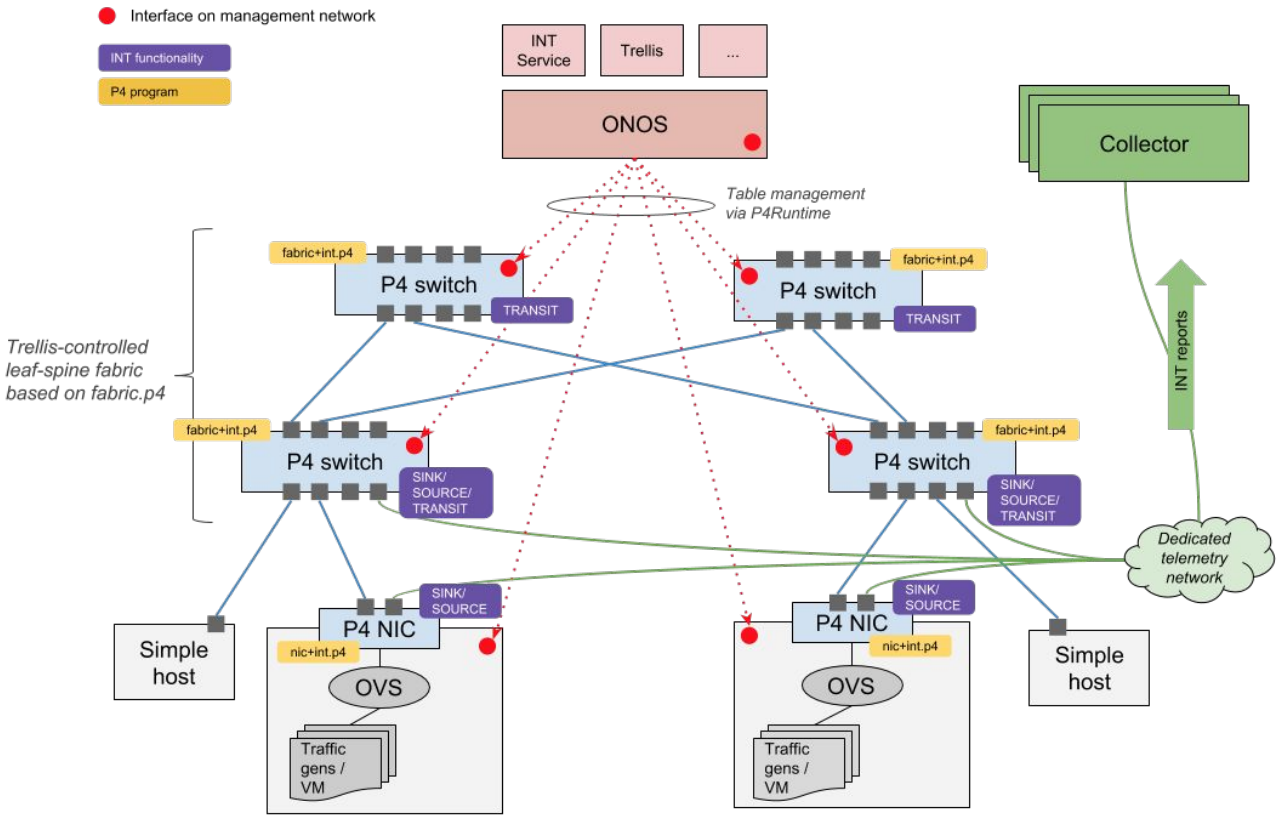
~480ns to perform forwarding (ToR 2)

INT on ONOS

INT on ONOS

- ONOS INT Service
  - ONOS service to control INT-capable devices
  - Expose SDN-like network-level APIs to ONOS applications
    - e.g., Start/stop INT, add/remove INT Intent, set/get INT Config
- ONOS driver for INT control operations (src/sink/transit)
  - Support both P4-capable and non P4-capable devices
  - Implements the same JAVA Interface
- Extend fabric.p4 with INT support
  - ONOS-supported INT reference program
- Open-source implementation of a collector (based on Prometheus)
  - For testing and demonstration purposes

INT on ONOS


Interface on management network
INT functionality
P4 program
INT Service
Trellis
...
ONOS
Collector
Table management via P4Runtime
Trellis-controlled leaf-spine fabric based on fabric.p4
fabric+int.p4
P4 switch
TRANSIT
SINK/ SOURCE/ TRANSIT
INT reports
Dedicated telemetry network
SINK/ SOURCE
P4 NIC
nic+int.p4
OVS
Traffic gens / VM
Simple host

Takeaways

- **ONOS allows centralized control of P4-enabled devices via P4Runtime**
  - Driver subsystem allows re-using existing apps with any P4 program
  - Support for pipeline-specific apps via extensible northbound APIs
- **Fabric.p4 and Trellis as a P4-enabled leaf-spine DC fabric solution**
  - Trellis work driven by service provider
  - Framework to implement advanced functions in the fabric
- **spgw.p4 demonstrates merits of executing VNFs on P4 HW (vs. CPU)**
  - Features missing: downlink buffering, QoS (community help needed)
- **ONOS-INT service to provide common API for managing heterogeneous INT-capable networks**

Next steps

- **Work on INT WIP**
- **Improve support for fabric.p4**
  - Reach parity with OF-DPA based fabric (forwarding capabilities, fault-tolerance)
  - Pipeline optimizations: tweak P4 program to maximize table size
- **Working on other VNF use cases**
  - Residential access: BNG, PPPoE termination
- **Stratum project**
  - High-quality implementation of P4Runtime/gNMI/gNOI agent on the switch

ONOS-P4 Brigade - Join the effort!

P4 Brigade mailing list:

brigade-p4@onosproject.org

P4 Brigade Wiki:

https://wiki.onosproject.org/display/ONOS/P4+brigade

Contributions from many brigade members - Thank you!

Carmelo Cascone, Andrea Campanella, Jonghwan Hyun, Uyen Chau, Brian O'Connor (**ONF**), Esin Karaman, Serkant Uluderya, Mehmed Mustafa, Ekber Aziz (**Netsia**), Frank Wang (**Inspur**), Yi Tseng, Kevin Chuang, Nate Tang, Iver Liu (**NCTU**), Wu Shaoyong, Jian Tian, Ke Zhiyong (**ZTE**), Tom Tofigh (**AT&T**), Phil Huang, Ahbee Wai (**Edgecore**)

Thanks!

Switching ASIC vs CPU - What are the benefits?

- **Maximized, deterministic throughput**
  - Always process traffic at line rate, with any traffic workload
- **Minimized, deterministic processing latency (and jitter)**
  - In the order of nanoseconds, with any traffic workload
- **Reduced power consumption**
  - Use less CPU resources, instead use switch that is there anyways

Achieved effortlessly!
Writing P4 code is easier than writing C code optimized
for throughput/latency/power consumption