Software-defined network monitoring with data stream platforms

Drissa HOUATRA, Orange Labs – Châtillon

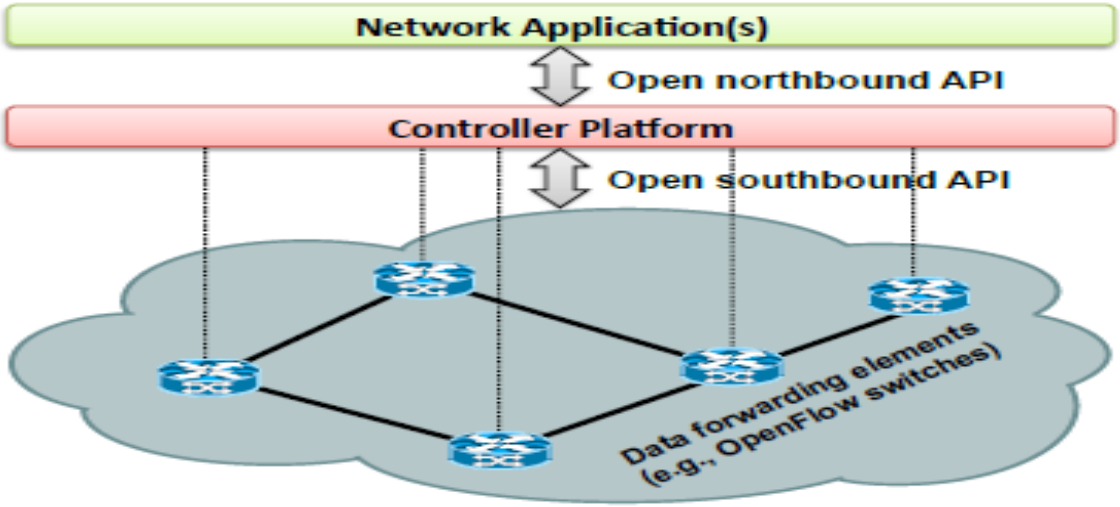
Yuchia TSENG, IRT SystemX – Palaiseau

SDN Day 2017, Nov. 23, SystemX Nano-INNOV, Palaiseau

Summary

- Monitoring and dataflow programming in SDN
  - SDN and network monitoring
  - Principle of network monitoring with dataflow programs
- Monitoring in controllers and the data plane
  - Monitoring requirements
  - Dataflow programs for controllers and forwarding engines
- Monitoring applications in SDN environments
  - Relations with controllers, data distribution requirements
  - Dataflow graphs in SDN monitoring applications

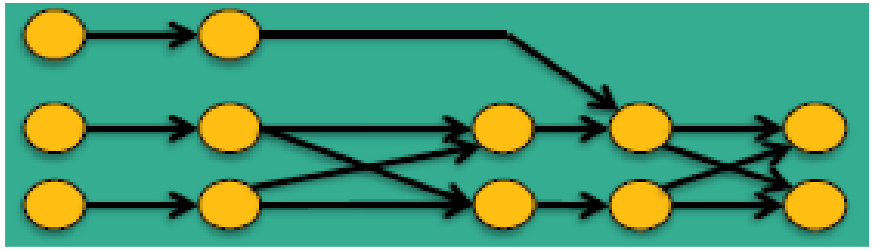
Monitoring and dataflow programming in SDN



- SDN environments
  - External and central control
  - Open, global state information
  - Monitoring needs at forwarding engines, controllers, applications
- How to monitor SDN environment?
  - Classical network monitoring problems are still present
  - Expect new problems specific to SDN environments
  - Enhanced visibility & more accurate state information
  - More powerful monitoring instruments

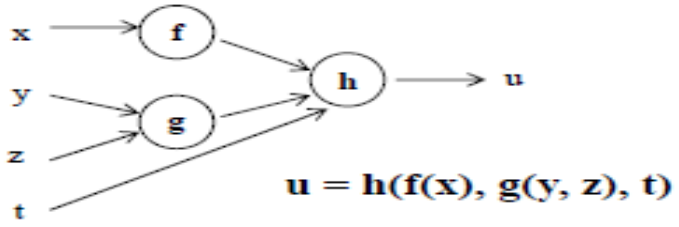
Monitoring and dataflow programming in SDN

- Stream processing in monitoring applications
  - Standard: systematic collection of monitoring data in
    - central database,
    - managements server
  - Becomes difficult, sometimes impossible (data and network size)
  - Not adapted to real-time analysis, live or forecast event detection
  - Mainstream trend
    - Adopt stream processing
    - Batch processing (databases) = particular case of stream processing
    - Unified view of batch & stream processing
- Monitoring with dataflow streams
  - Dataflow programming
  - Distributed dataflows as DAGs, with functions to process data streams



Monitoring and dataflow programming in SDN

- Functional programming



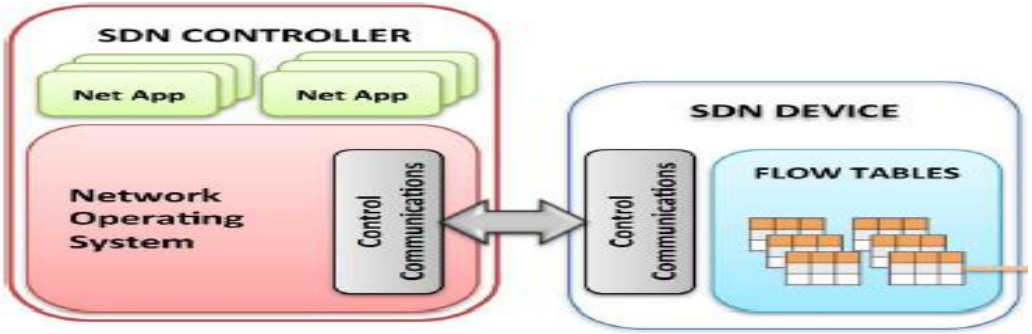
- Benefits of functional programming with dataflow stream
  - No side effect
  - Atomic operations
  - Stateless services
- Monitoring applications are designed as HPDC programs, with:
  - Relatively simple exploitation of parallelism in distributed environments
  - No time waste with control flow synchronization
  - Distributed stateless services: no state mutation problem at nodes

Monitoring in controllers and the data plane

- Some notices about SDN controllers
  - Software system running on GPPs
  - Can be deployed on several GPP nodes, e.g. cluster(s)
- Monitor an open distributed software system
- Common monitoring requirements
  - Controllers
    - Program failures due to memory usage (troubleshooting)
    - Measurements and reports for time-critical controller processes
  - Controller-to-switch
    - Communication failures between switches and SDN controllers
    - Vertical (switch-to-controller) reliability and performance information
  - Switches
    - Packet loss and retransmission statistics
    - Horizontal (switch-to-switch) reliability and performance information

Monitoring in controllers and the data plane

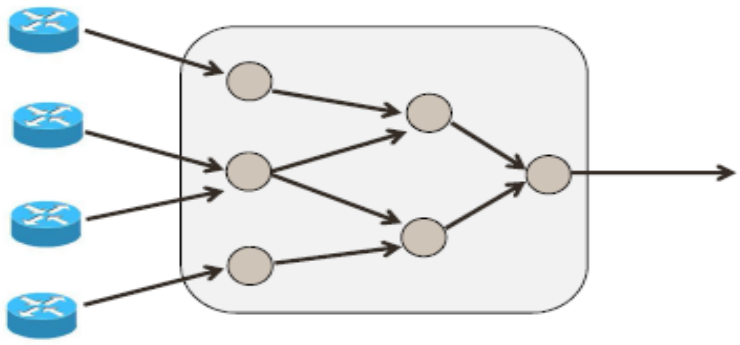
- More on the controller-switch relations (vertical links)
  - Links created with a configuration that should be monitored
  - Relation evolving with distributed interactions: problems with delays, reliability, synchrony, etc.
  - A key differentiator with non-SDN systems
- Example with an OpenFlow switch



- Important monitoring requirements for SDN visibility:
  - Controller monitoring
  - Switch monitoring
  - Controller-switch relation configuration monitoring

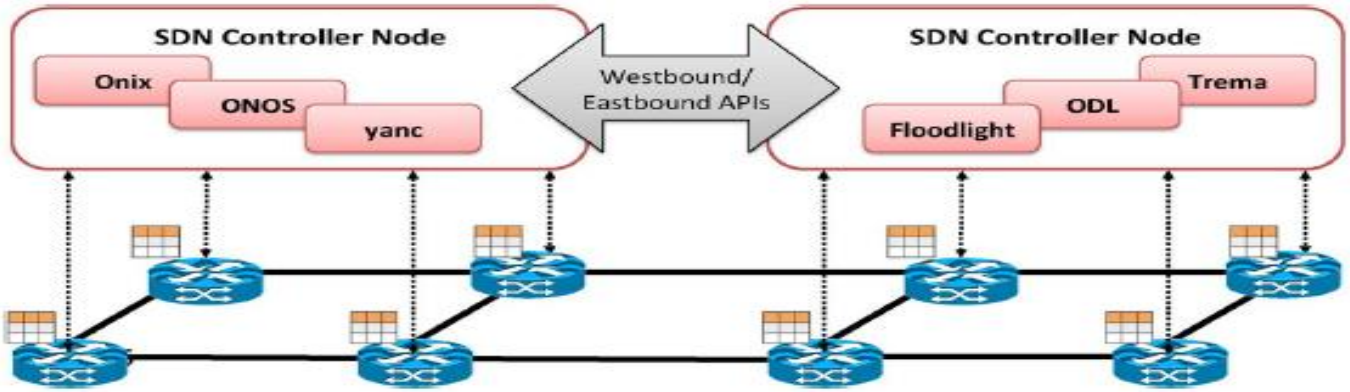
Monitoring in controllers and the data plane

- Streamlining the monitoring data
  - If the controller is distributed on several nodes, we may need to
    - Propagate the monitoring information
    - Process that information on several nodes
  - Distributed monitoring fabric
- Principles of a dataflow-based monitoring fabric
  - Inputs: monitoring streams from forwarding engines
  - Dataflow graph
    - Controller cluster
    - Monitoring data streams from forwarding nodes
    - Data is processed and distributed between controller nodes
    - Example



Monitoring applications in SDN environments

- SDN monitoring
  - Challenging network applications
  - Agility and complexity of the SDN environment
  - Multiple SDN controllers
- Monitoring with distributed (multiple) controllers
  - Multiple controllers for resilience, scalability, performance



  - Data collection via northbound APIs
  - Data distribution, merging and analysis of monitored data
  - Big & fast data application, with performance and real-time constraints

Network applications in SDN environments

- Data from main visibility points
  - Forwarding engines (switches)
  - Switch-controller links
  - Controllers
- Large scale SDN monitoring with
  - Distributed monitoring architecture
  - Control network of multiple controller cluster nodes
- Data distribution approach
  - A dataflow graph for each controller
  - Merge monitored data obtained from different controllers
    - Via the northbound APIs
    - Maybe via eastbound and westbound APIs – connecting dataflow graphs