

# Ryuo

## Using High Level Northbound API for Control Messages in SDN

Shaoyu Zhang<sup>1</sup>, Yao Shen<sup>1</sup>, Matthias Herlich<sup>2</sup>, Kien Nguyen<sup>3</sup>,  
Yusheng Ji<sup>2</sup>, Shigeki Yamada<sup>2</sup>

<sup>1</sup>Shanghai Jiao Tong University, China

<sup>2</sup>National Institute of Informatics, Japan

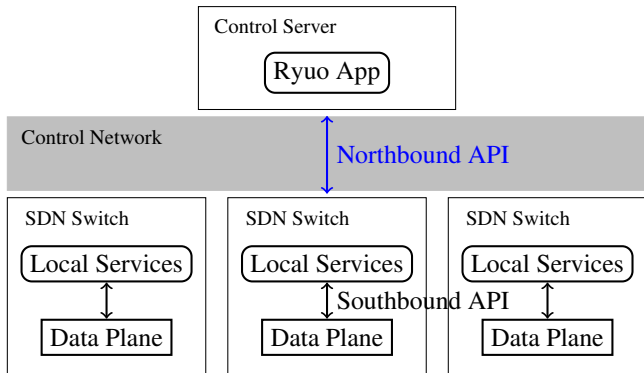
<sup>3</sup>National Institute of Information and Communications Technology, Japan

August 18, 2015

# Issues of OpenFlow

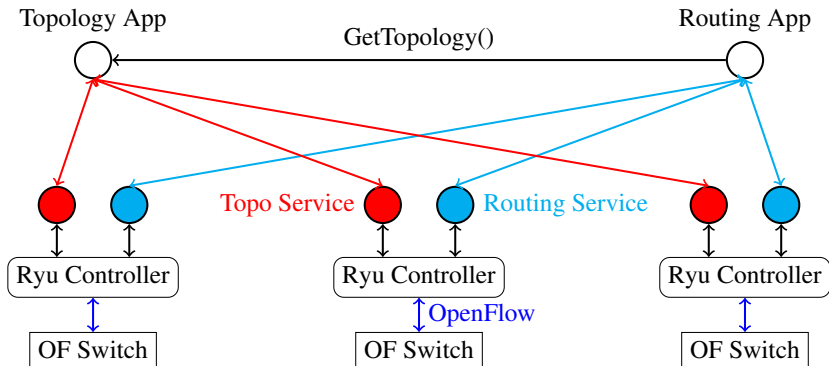
- Control latency
- Applications couple with OpenFlow at some degree
- All control logic centralized

# Ryu: Architecture



- Ryu App: Focus on business logic
- Local Service: Provides high level API, ensures compatibility
- Domain specific control message

# Example: Topology Discovery and Routing



# Evaluations

- Control Traffic
- Control Latency
- Throughput of handling local events

# Evaluation Environment



- Pica8 P-3295 OpenFlow switch
  - ▶ CPU: 825MHz PowerPC
  - ▶ Memory: 512MB
- PC: For host, Ryu controller, Ryuo Application
  - ▶ CPU: Intel Core i5-3470, 3.2 GHz
  - ▶ Memory: 4GB

# Control Traffic Evaluation

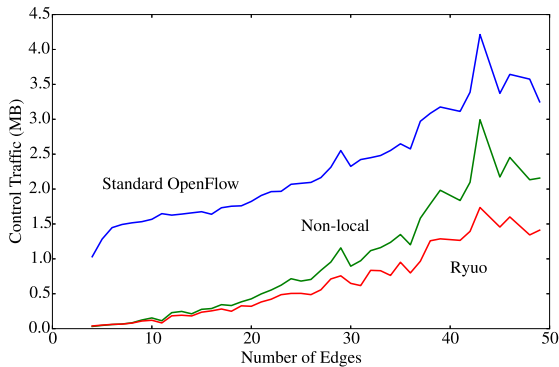
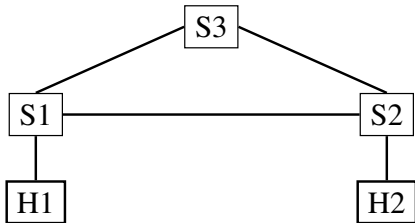


Figure: OpenFlow vs Local Controller Approach (approx.) vs Ryuo

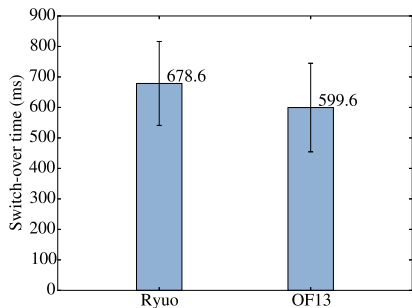
# Evaluation: Fast Failover with OpenFlow 1.0



- H1: 1K packets/s.
- H2: Wireshark.
- S1-S2 → S1-S3-S2
- Count packets lost

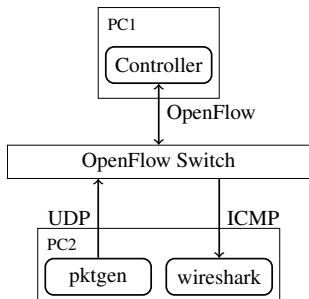
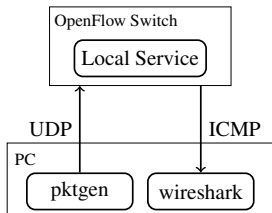


# Evaluation: Fast Failover with OpenFlow 1.0



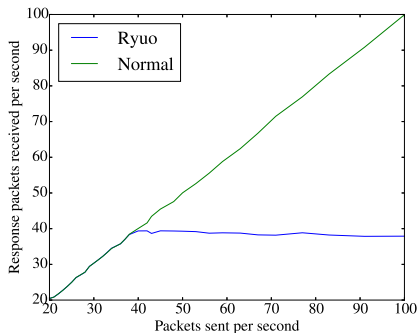
- Local Service:  $\sim 0.1$ s slower.

# Throughput Evaluation: Setup



- Throughput: Number of events handled in one second.
- Send UDP packets, record responses.

# Throughput Evaluation: Result

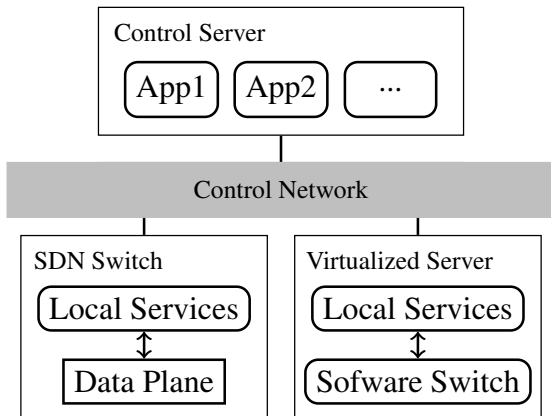


- Switch can only handle about 40 packets per seconds.
- Not a problem on a virtualized server.
- New switches with more powerful CPUs.<sup>[1]</sup>

# Ryuo: Implementation

- Ryuo Apps and Local Services based on Ryu.
- Global Local Communication: Pyro4<sup>[2]</sup>
- Available Online: <https://github.com/epcc-networking/ryuo>

# Ryu: Deployment



- Two ways to deploy:
  - ▶ Directly on OpenFlow switches.
  - ▶ On servers with multiple VMs and Open vSwitch.

## Related Work

- Kandoo<sup>[3]</sup>: Local and Root Controller
- Beehive Netctrl<sup>[4]</sup>: Distributed Network Controller
- Orion<sup>[5]</sup>: Hierarchical Control Plane

# Conclusion

- Advantages:
  - ▶ Reusable Local Services provide high level API.
  - ▶ Easy to extend southbound API.
  - ▶ Less control traffic.
- Disadvantages:
  - ▶ Limited by resources on the switch.
  - ▶ Local Service deployment.

Questions?



Thank you!

# References I

- [1] AS5712-54X with ONIE / 10GbE Data Center Switch. [Online]. Available:  
<http://www.edge-core.com/ProdDtl.asp?sno=457&AS5712-54X>
- [2] Pyro - Python Remote Objects. [Online]. Available:  
<http://pythonhosted.org/Pyro4/>
- [3] S. Hassas Yeganeh and Y. Ganjali, "Kandoo: A framework for efficient and scalable offloading of control applications," in *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*. ACM, 2012, pp. 19–24.

## References II

- [4] —, “Beehive: Towards a simple abstraction for scalable software-defined networking,” in *Proceedings of the 13th Workshop on Hot Topics in Networks*. ACM, 2014, pp. 13:1–13:7.
- [5] Y. Fu, J. Bi, K. Gao, Z. Chen, J. Wu, and B. Hao, “Orion: A hybrid hierarchical control plane of software-defined networking for large-scale networks,” in *IEEE 22nd International Conference on Network Protocols (ICNP)*, 2014, pp. 569–576.