NTT Software Innovation Center

NTT

# OpenStack/Quantum SDN-based network virtulization with Ryu


Kei Ohmura
NTT

May 31, 2013

# Outline

- Introduction to Ryu
- OpenStack Quantum and Ryu
- Demo
- Summary

# What is “Ryu”

流
(ryu)

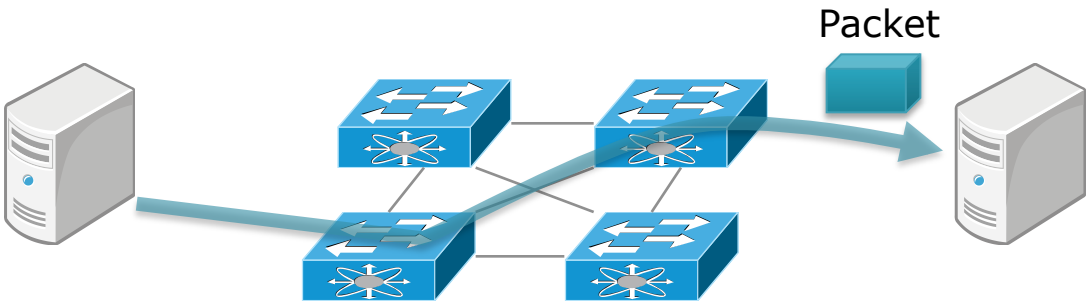
means “flow”

龍
(ryu)

means “japanese dragon”,
one of warter gods

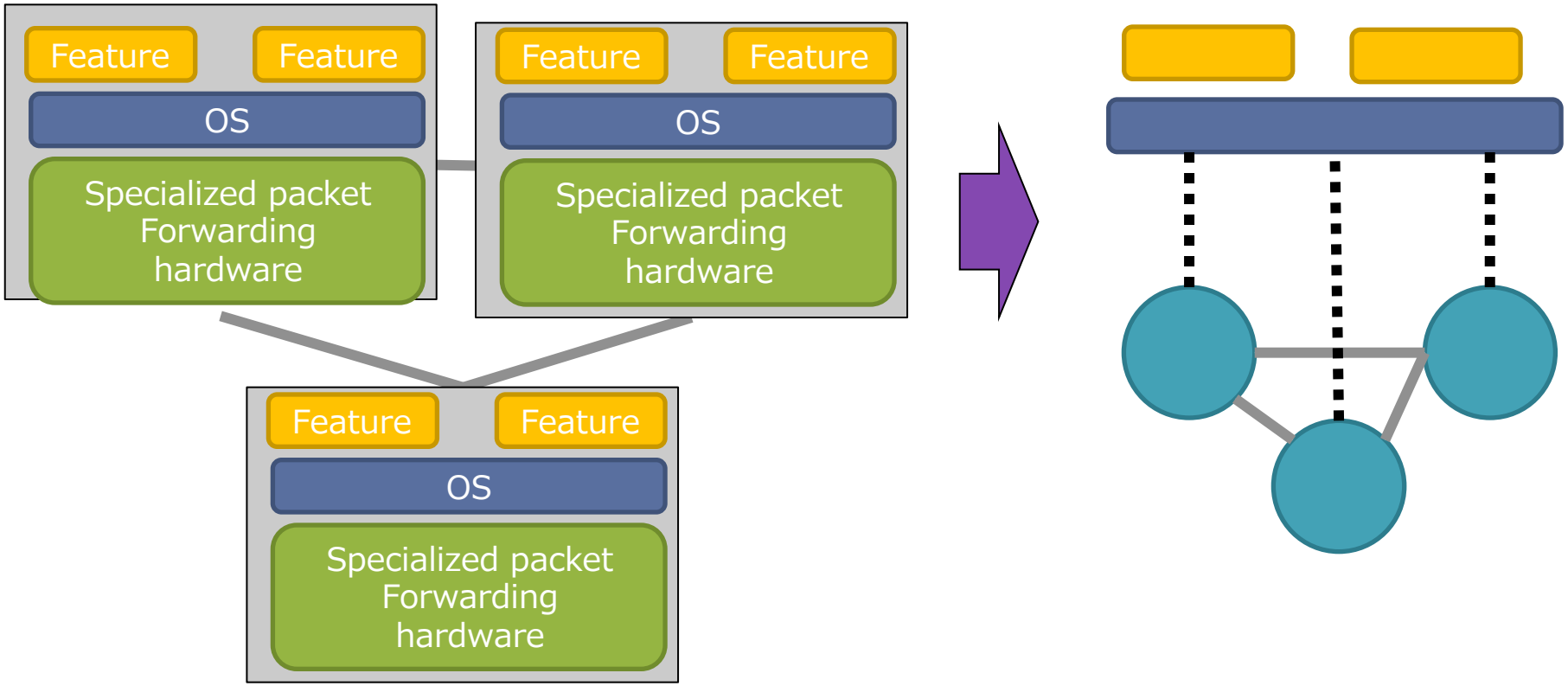
# What is “Ryu”


**manages “flow” control to enable intelligent networking**



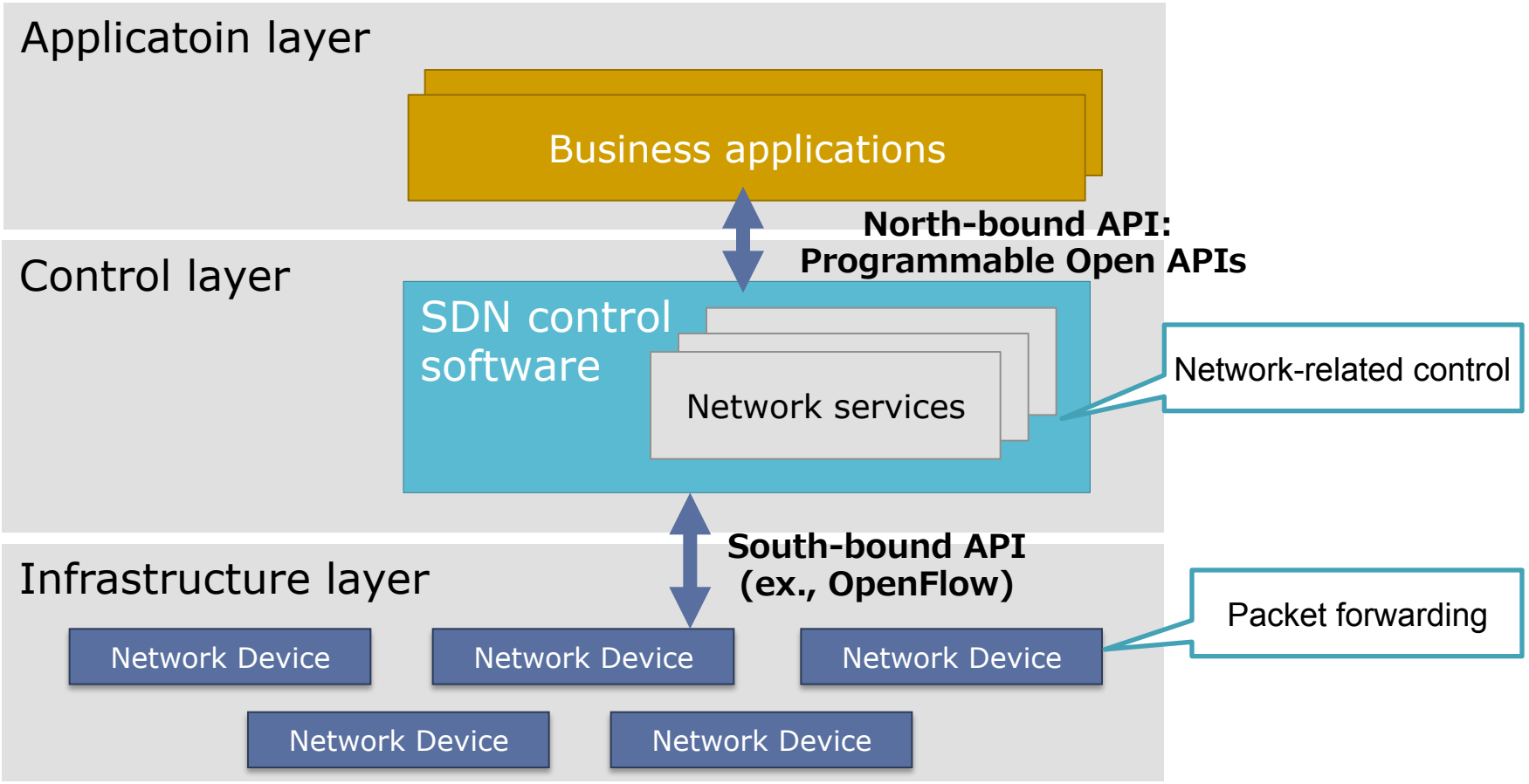
# SDN(Software Defined Networking)

- **Separates control and data plane:**
  - Open interface between control and data plane
  - Network control and management features in software



# SDN(Software Defined Networking)

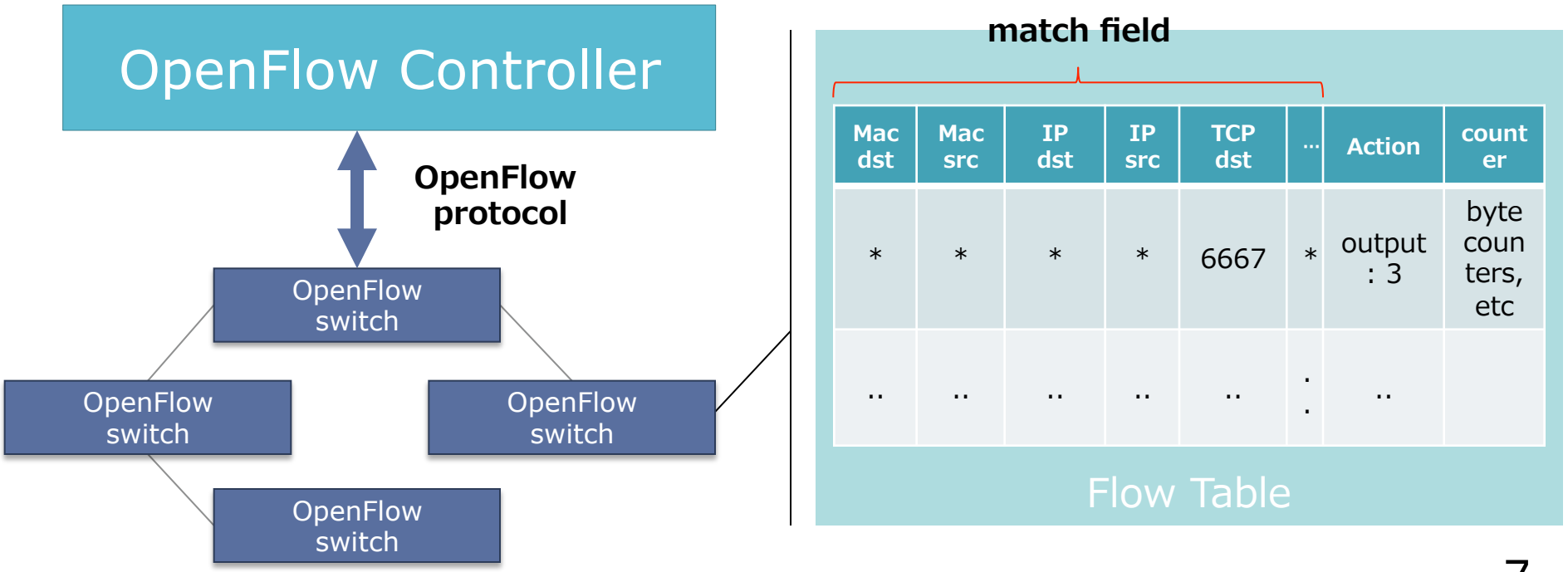
- **Separates control and data plane:**
  - Open interface between control and data plane
  - Network control and management features in software




http://www.opennetworking.org/sdn-resources/meet-sdn

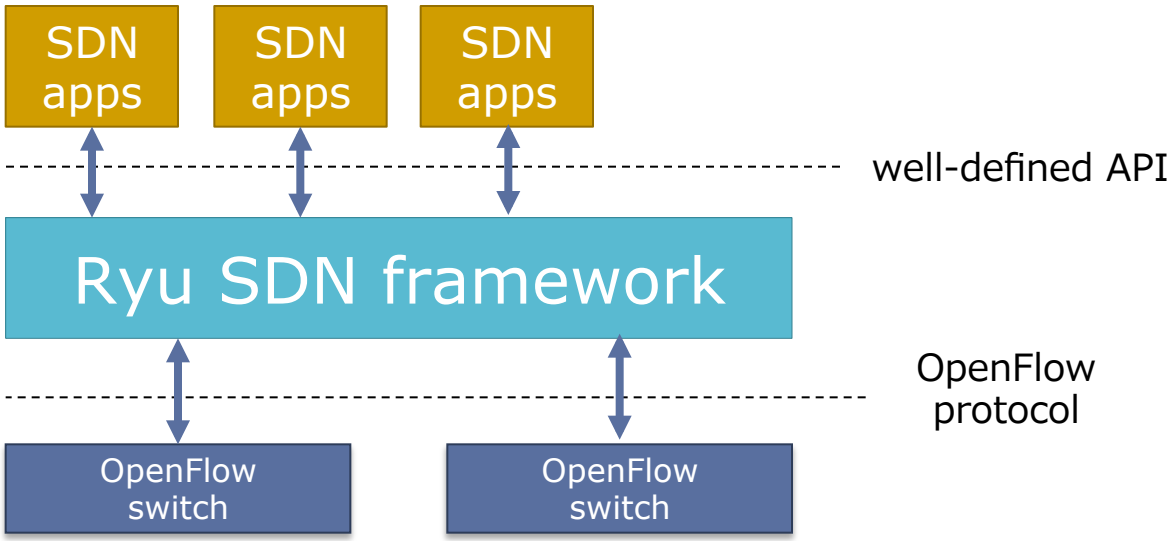
# OpenFlow Overview

- One of the key technologies to realize SDN
- Open interface between control and data plane



# Ryu SDN framework

- **SDN Framework**
  - A platform for building SDN applications
  - Provides useful libraries and well-defined API
- **Open source software (Apache v2)**
  - Fully written in Python
  - Project site: http://osrg.github.com/ryu/



# Our goals

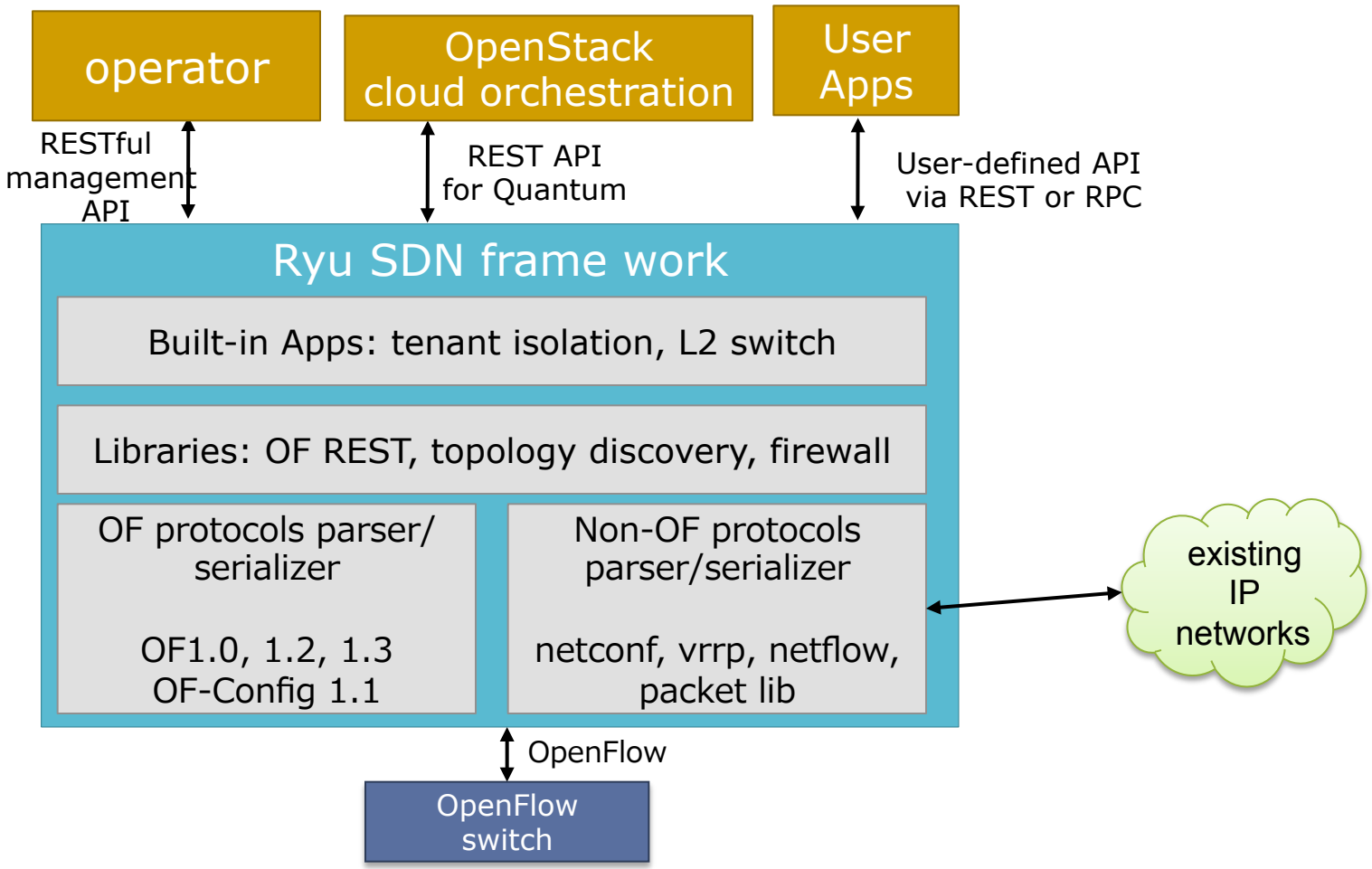
- **De facto SDN platform**
  - Standard network controller for cloud orchestrators, e.g. OpenStack
  - Default network controller for Linux Distributions, e.g. RHEL/feadora/ ubuntu
- **High quality for commercial deployment**
  - code quality, functionality, usability

# Features

- **Generality**
  - Vendor-neutral
  - Supports open interface (eg., OpenFlow)
  - Used by some switch vendors
- **Agile**
  - Framework for SDN application development instead of all-purpose big monolithic ‘controller’.

# Architecture

- Implement your apps by using Ryu SDN Framework



# Current status

- **OpenFlow protocol**
  - OF1.0 + nicira extensions, OF1.2, OF1.3
  - OF-Config 1.1
- **Other protocols**
  - netconf, vrrp, xFlow, snmp, ovsdb
- **Ryu applications/libraries**
  - Topology viewer
  - OF REST
  - Firewall
  - Some sample apps are in the ryu/app directory.
- **Switch Interoperability**
  - Referenced by some switch vendors
  - Open vSwitch
    - Integration testing with Open vSwitch (OF1.0, OF1.2)
    - nicira extensions, OVSDB
- **Integration with other components**
  - HA with Zookeeper
  - IDS (Intrusion Detection System)
  - OpenStack Quantum

# How to use

- **Install Ryu from pip**

```
$ sudo pip install ryu
```

- **Install Ryu from the source code**

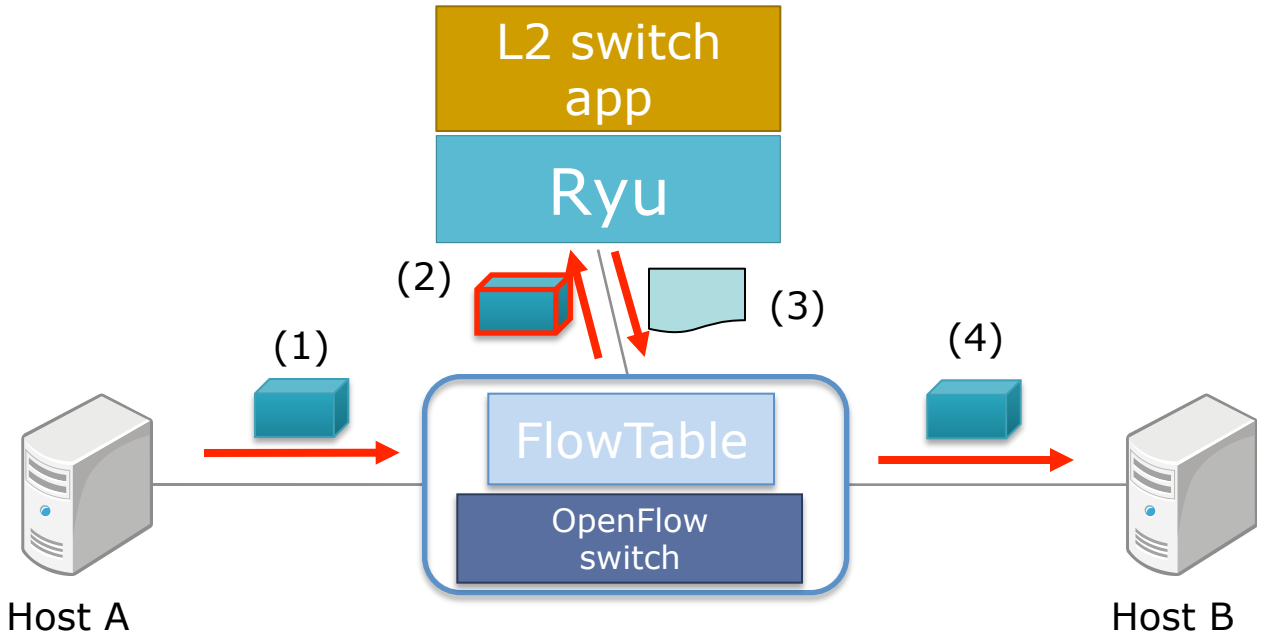
```
$ git clone git://github.com/osrg/ryu.git
$ cd ryu; sudo python ./setup.py install
```

- **Run your application**

```
$ ryu-manager yourapp.py
```

# Mac learning switch

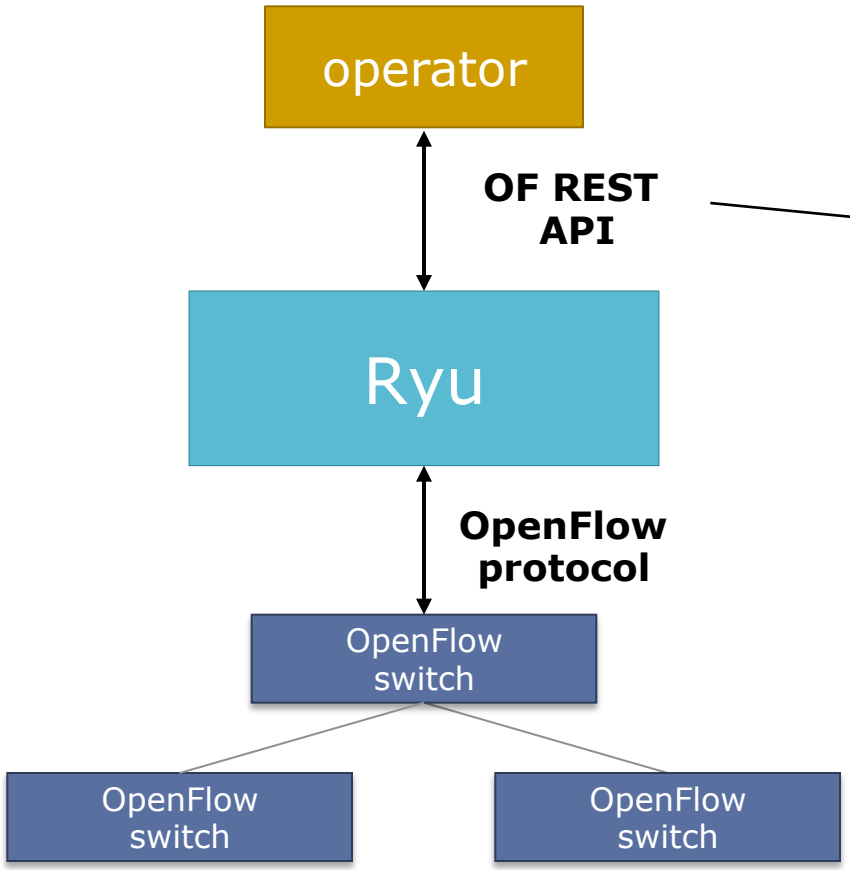
```
$ ryu-manager ryu/app/simple_switch.py
```




tutorial: https://github.com/osrg/ryu/wiki/OpenFlow_Tutorial

# OF REST

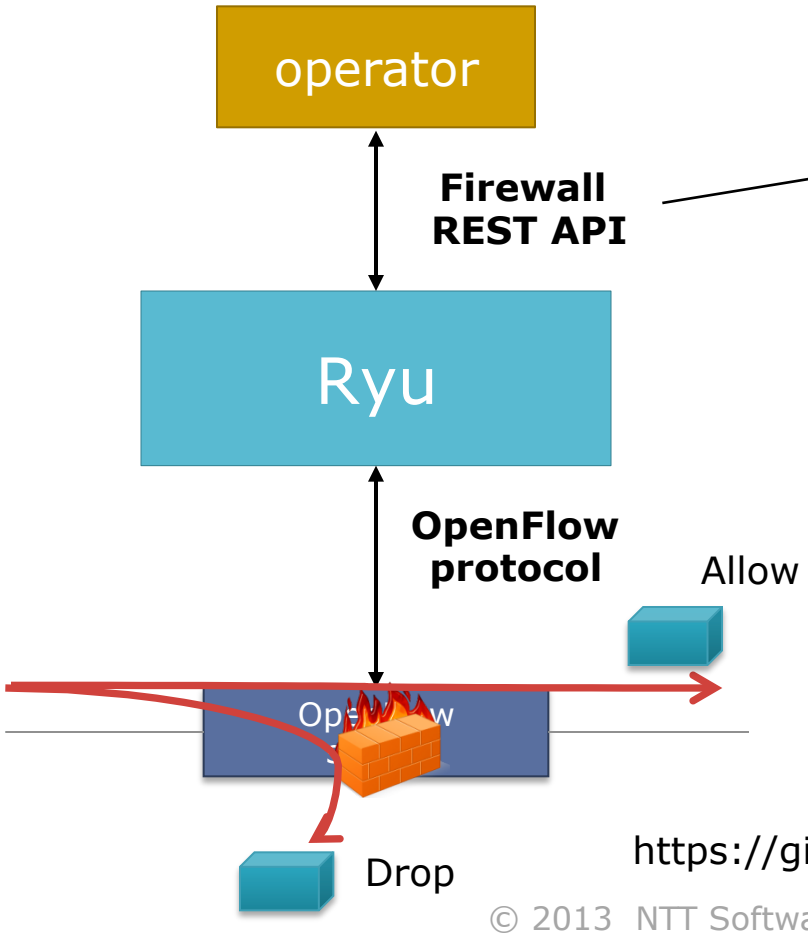
```
$ ryu-manager ryu/app/ofctl_rest.py
```




OF REST API
・add a flow entry
  POST http://example.org/stats/flownetry/add
・delete  flow entries
  DELETE http://example.org/stats/flowentry/delete
・get flow stats
  GET http://example.org/stats/flow/{dpid}

# Firewall REST

```
$ ryu-manager ryu/app/rest_firewall.py
```

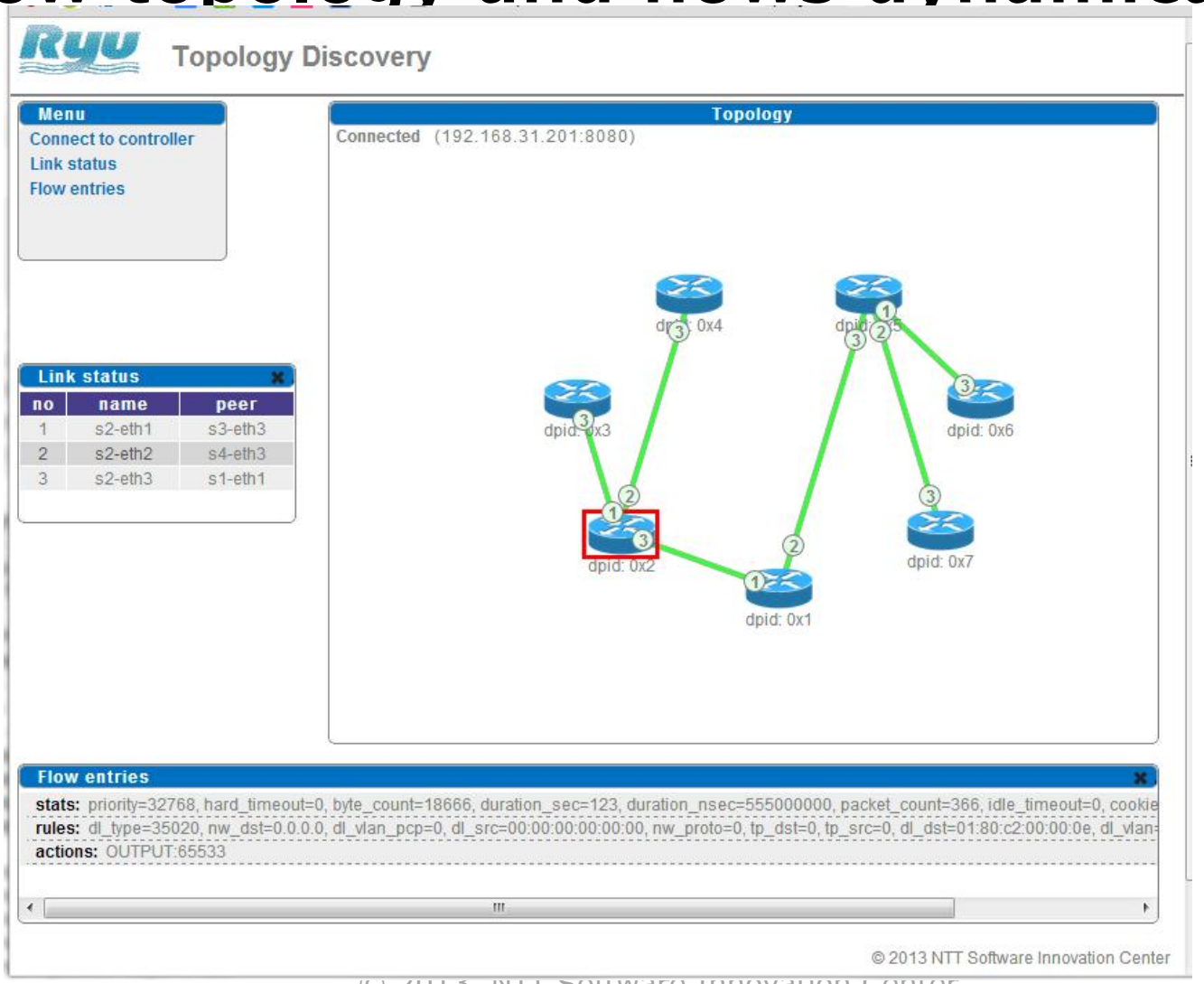



Firewall REST API
- add a rule
  POST http://example.org/firewall/rules/{switch-id}
- delete a rule
  DELETE http://example.org/firewall/rules/{switch-id}
- get rules
  GET http://example.org/firewall/rules/{switch-id}

https://github.com/osrg/ryu/wiki/Third-Party-Tools,-Etc.

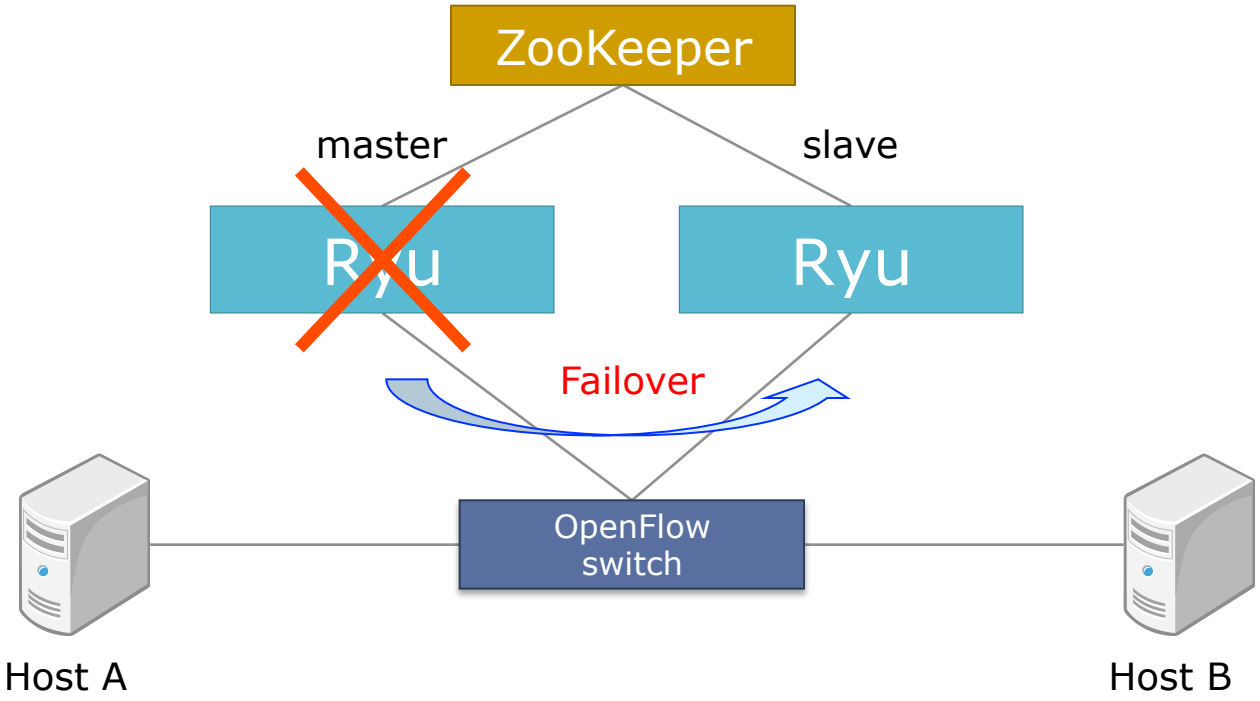
# Topology viewer

- Show topology and flows dynamically



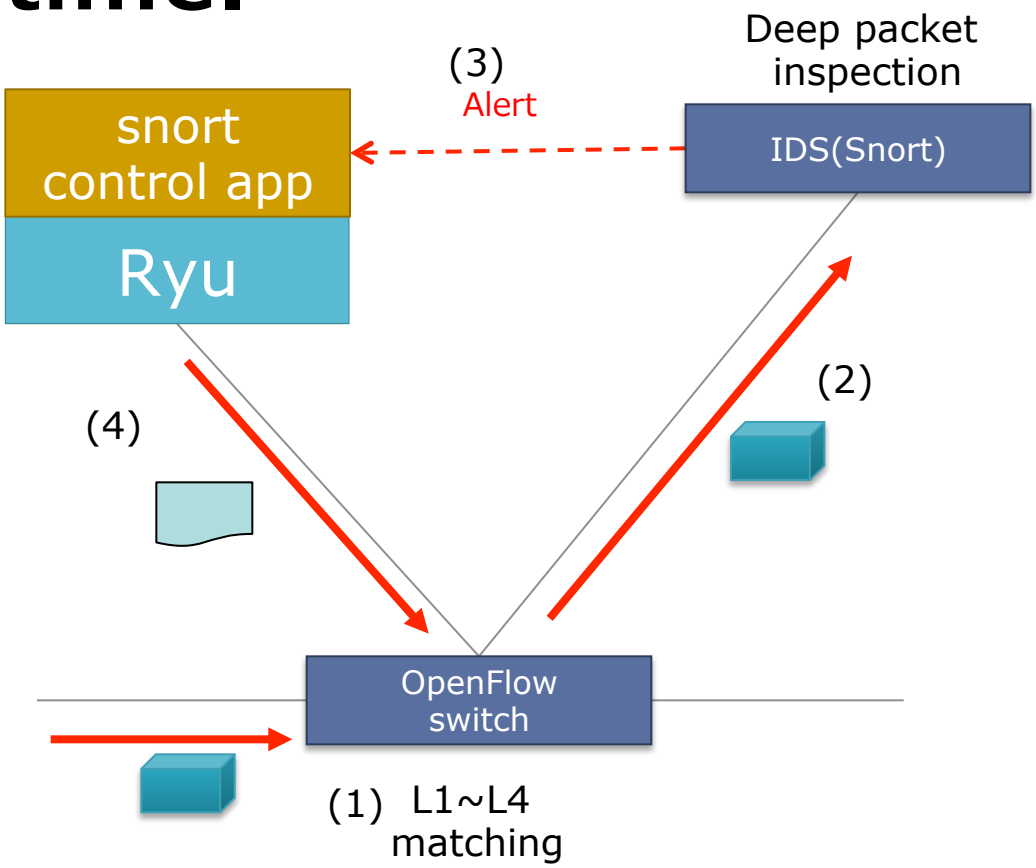
# HA with Zookeeper

- **Centralized controller is single point of failure (SPOF)**
- **Ryu + ZooKeeper is able to avoid SPOF**



# IDS integration

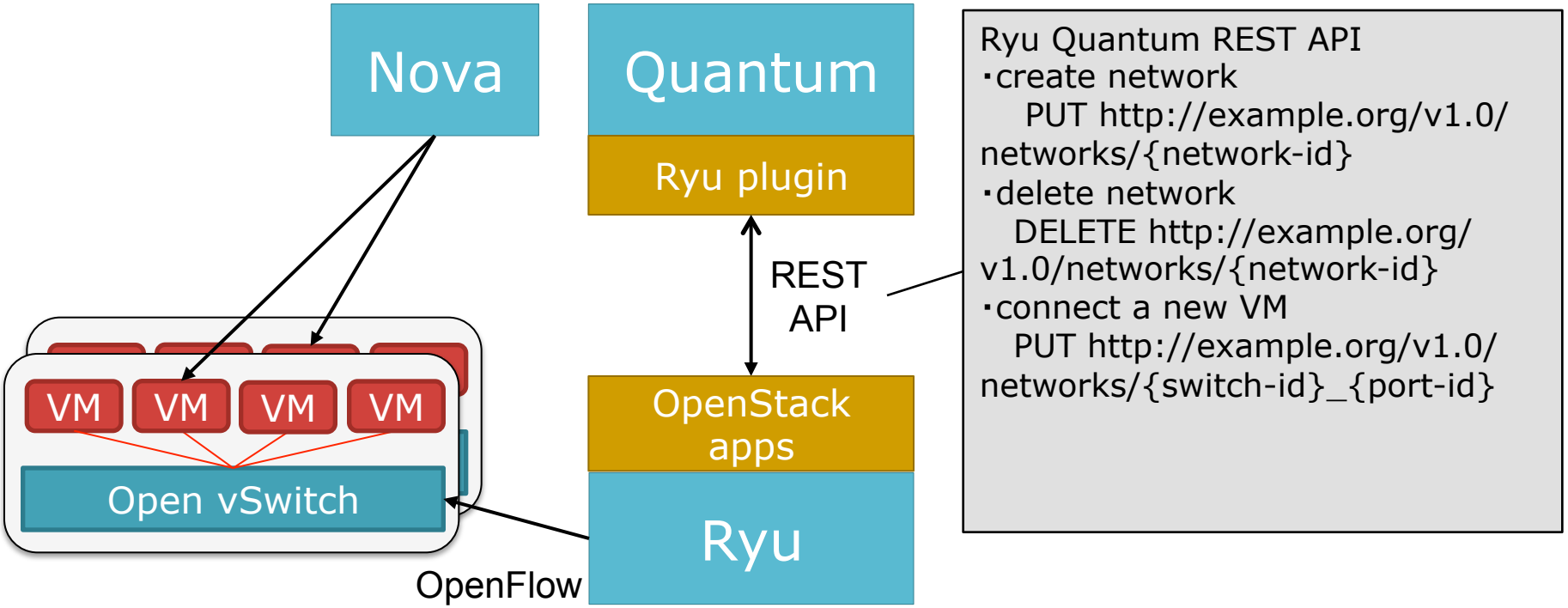
- Ryu + IDS can cope with threats in real time.




https://github.com/osrg/ryu/wiki/Snort-Integration

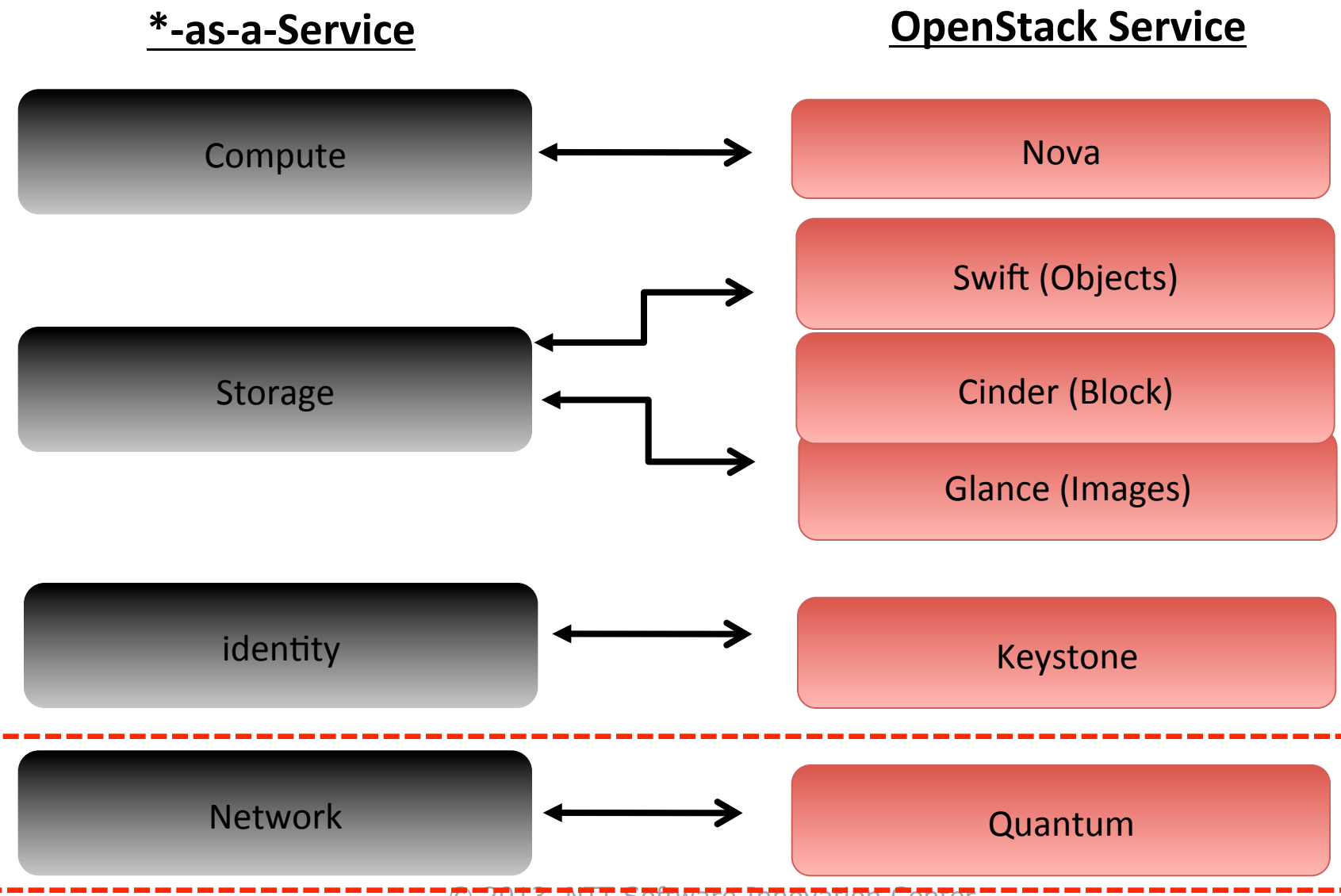
# Ryu plugin for OpenStack Quantum

- **Ryu plugin was merged into OpenStack Quantum Grizzly release**




https://github.com/osrg/ryu/wiki/OpenStack

# OpenStack



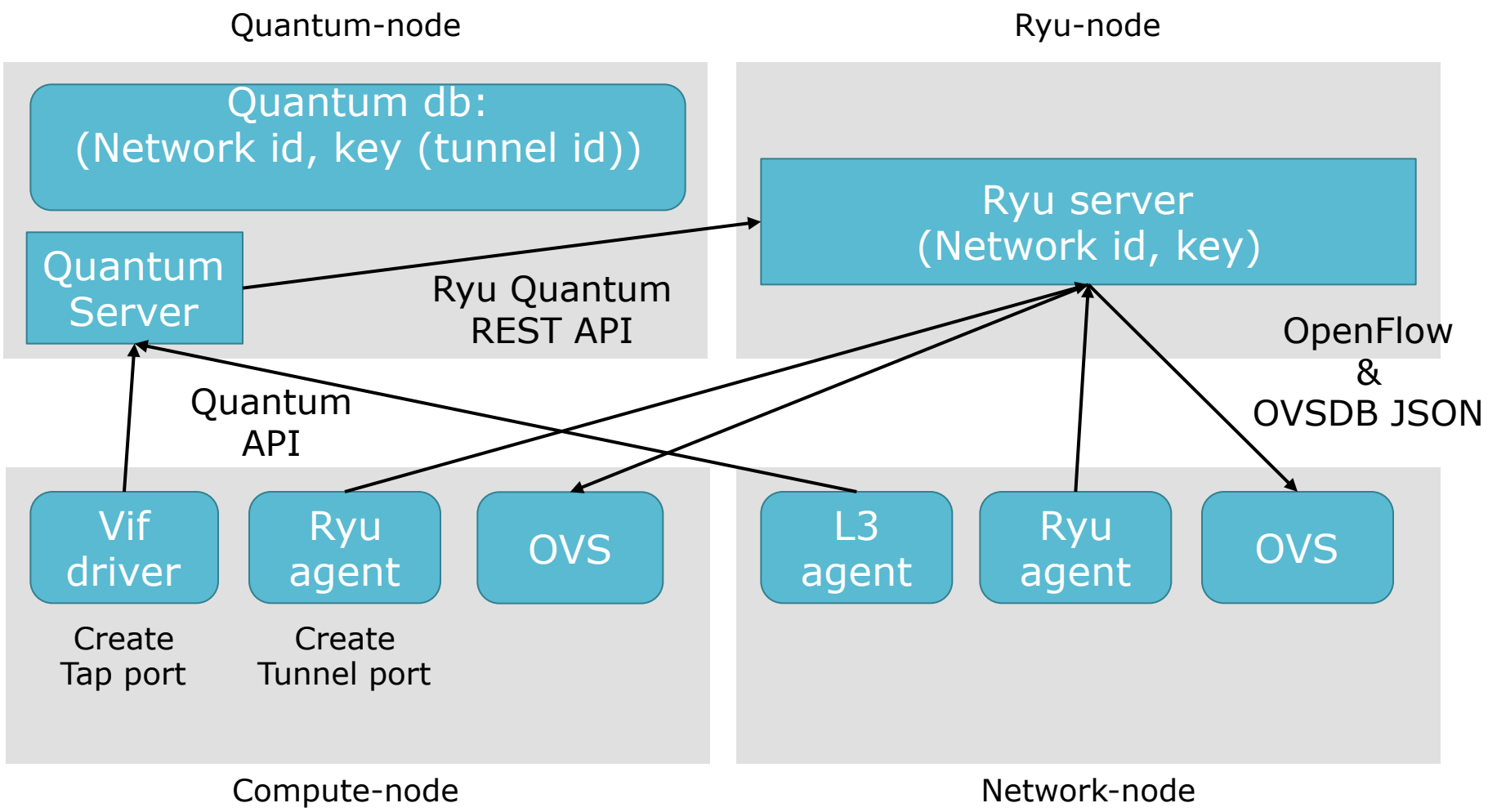
# OpenStack Quantum

- **Provides networking-as-a-service**
  - Quantum controls network virtualization like Nova controls server virtualization
- **“plugin” mechanism**
  - Enable different technologies
    - Ryu, Open vSwitch, Cisco UCS, Linux Bridge, NVP

# What does Ryu bring to OpenStack

- **Flat L2 networks regardless of the underlying physical network**
  - We don’t need high-end switches
- **Scalable multi-tenant isolations**
  - Ryu provides tunneling based isolations
  - Virtual networks that Ryu provieds are decoupled from VLAN limitations

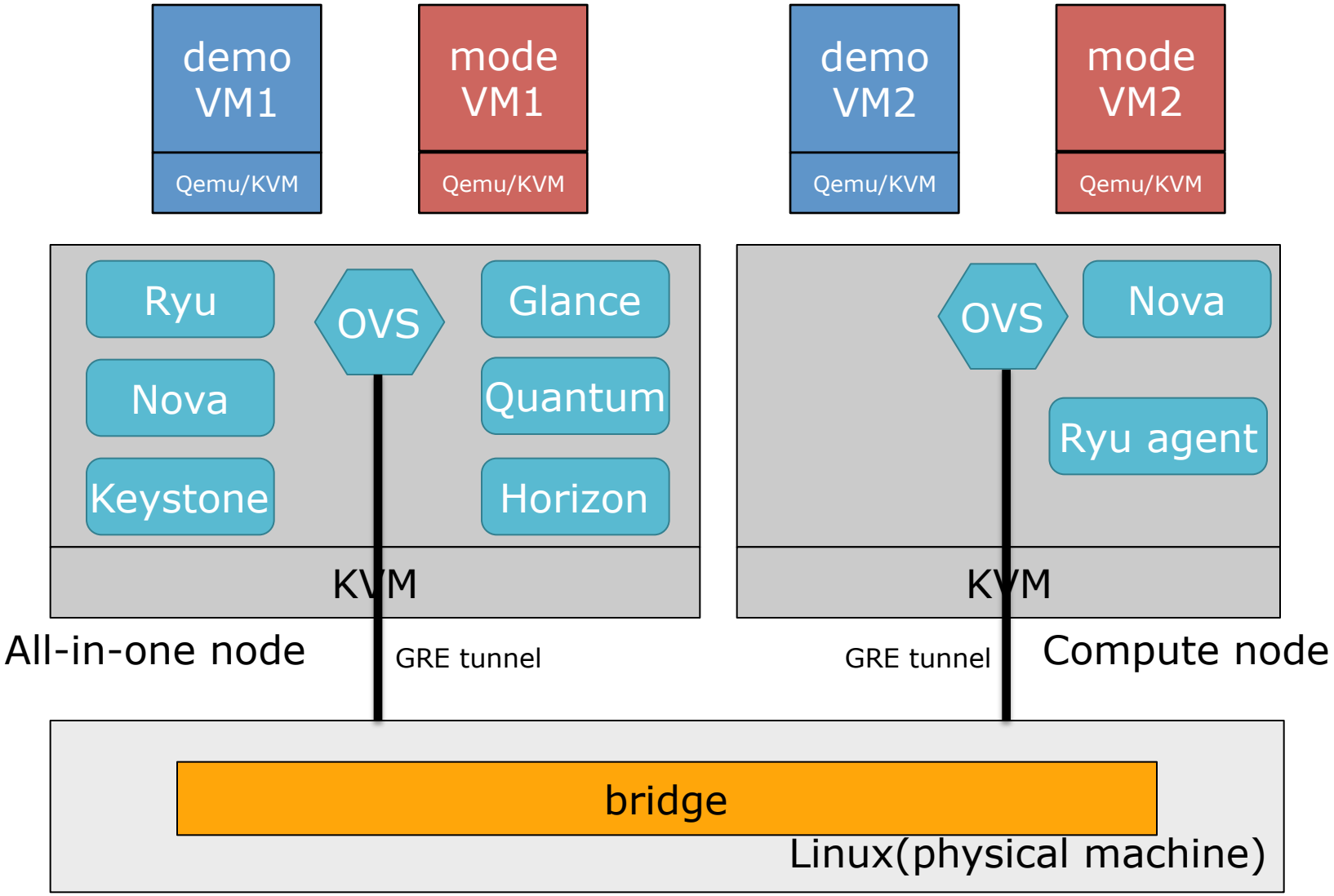
# How Ryu works with OpenStack



# Demo

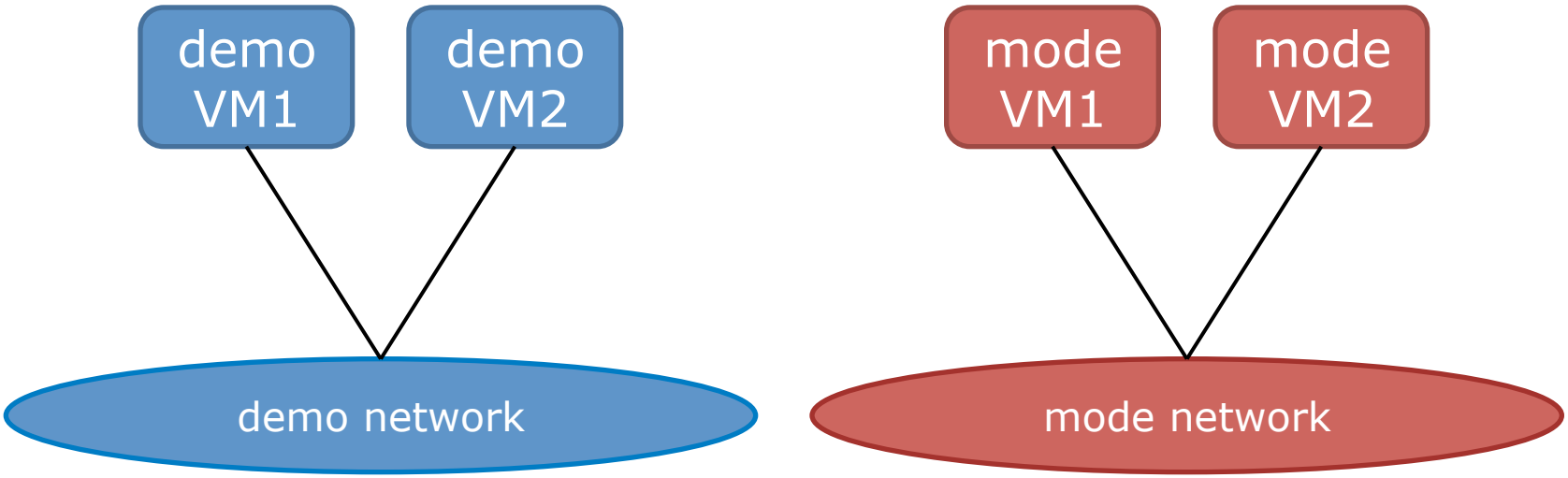
- **Ryu and OpenStack (GRE tunneling)**

# Ryu and OpenStack: physical view



# Ryu and OpenStack: logical view

Tenant demo ID -> 0x2
Tenant mode ID -> 0x4



# Future works

- Adds more components(protocols, IaaS, stats, security, etc).
- Improves distributed deployment component(cluster support)
- New testing methods (Ryu has more than 15,000 lines test code).

# Summary

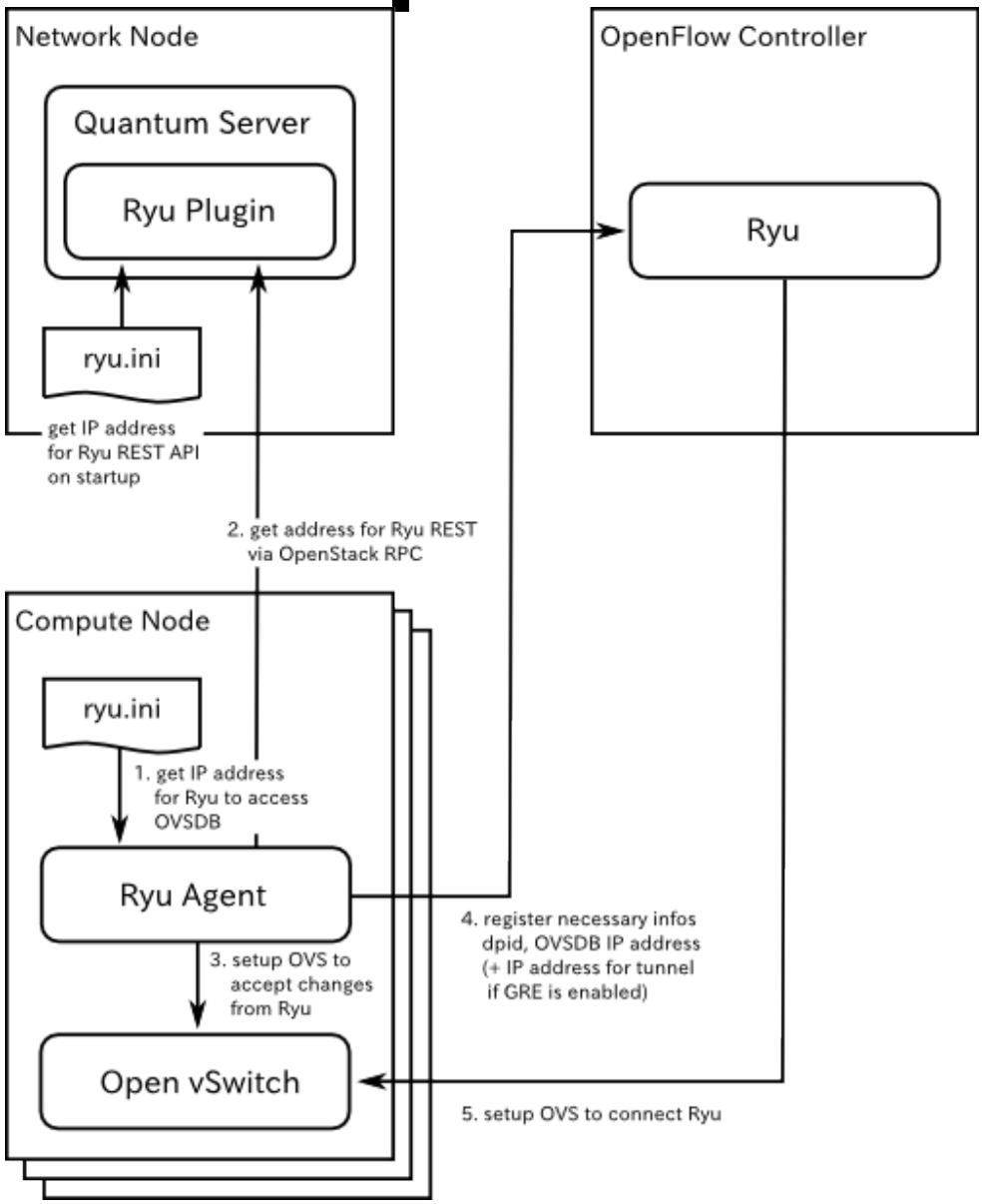
- **Ryu is an ongoing project**
  - Ryu project needs more developers
  - site: http://osrg.github.com/ryu/
  - wiki:https://github.com/osrg/ryu/wiki/_pages
  - ML: ryu-devel@lists.sourceforge.net

NTT Software Innovation Center

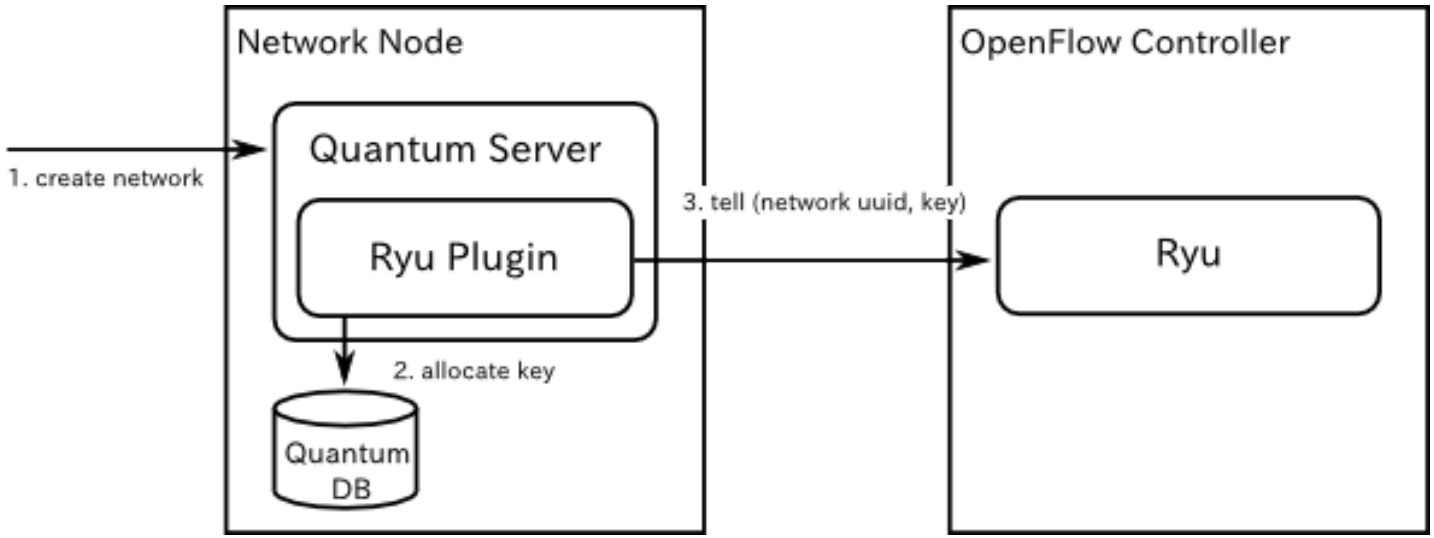
NTT

# Appendix

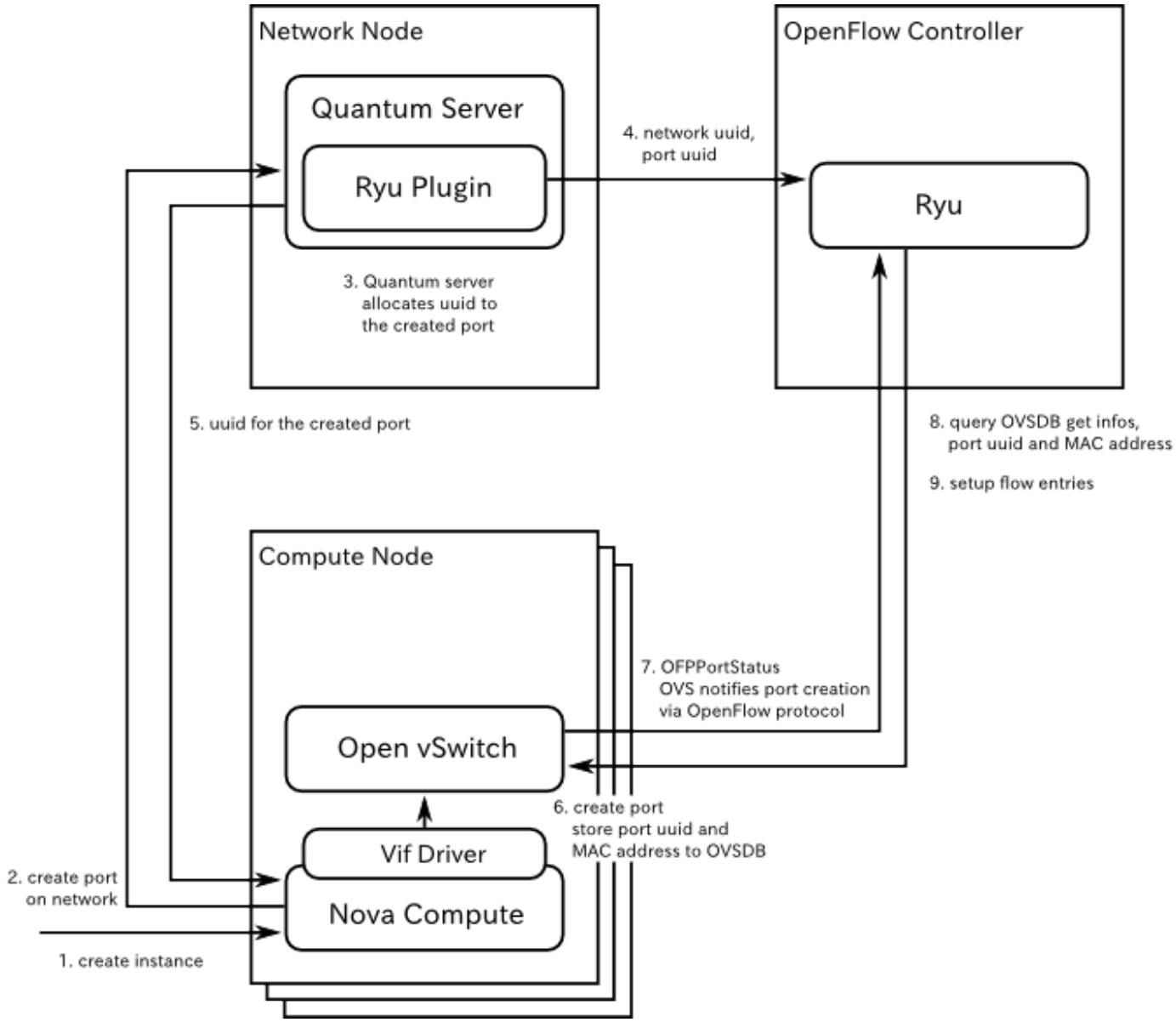
# Node boot up



# Network creation



# Instance creation



# Flow table usage