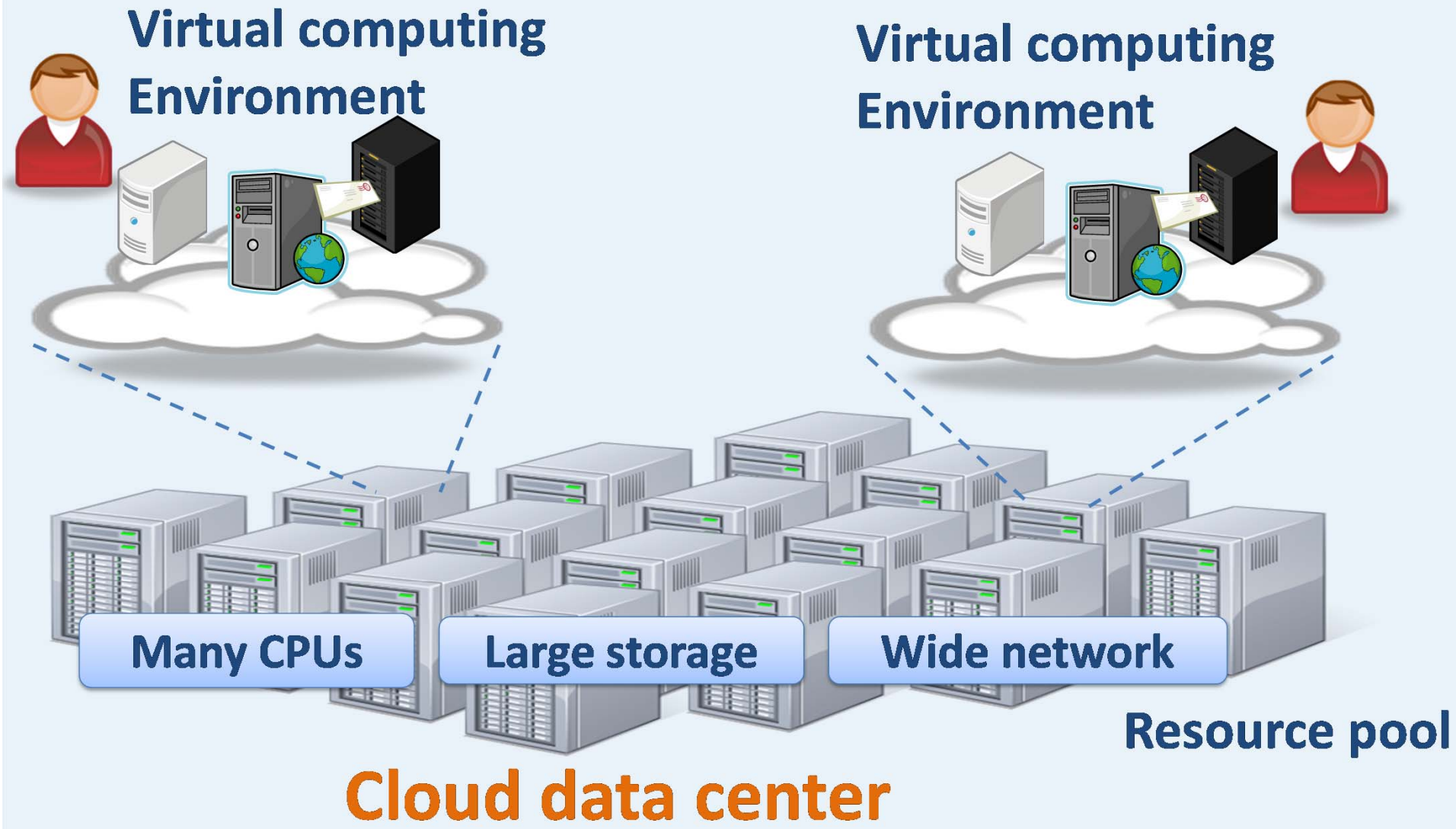
International Clouds using OpenFlow

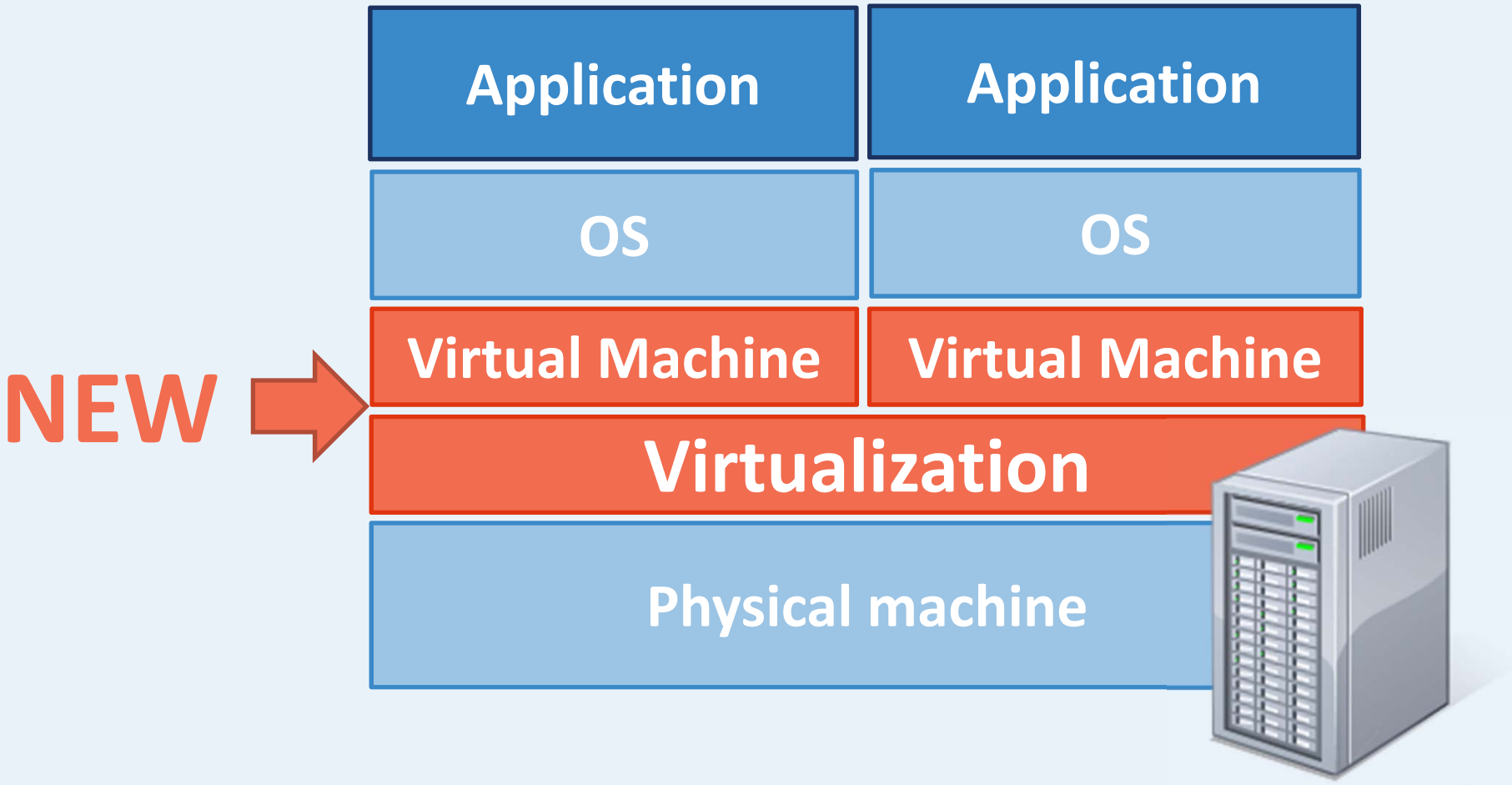
Kohei Ichikawa (ichikawa@naist.jp)
Nara Institute of Science and Technology, Japan

March 20, 2013
Pragma24


Cloud


Virtual computing Environment
Virtual computing Environment
Many CPUs
Large storage
Wide network
Resource pool
Cloud data center

Virtualization technology brings New Paradigm



Virtualization technology allows to control machines by software

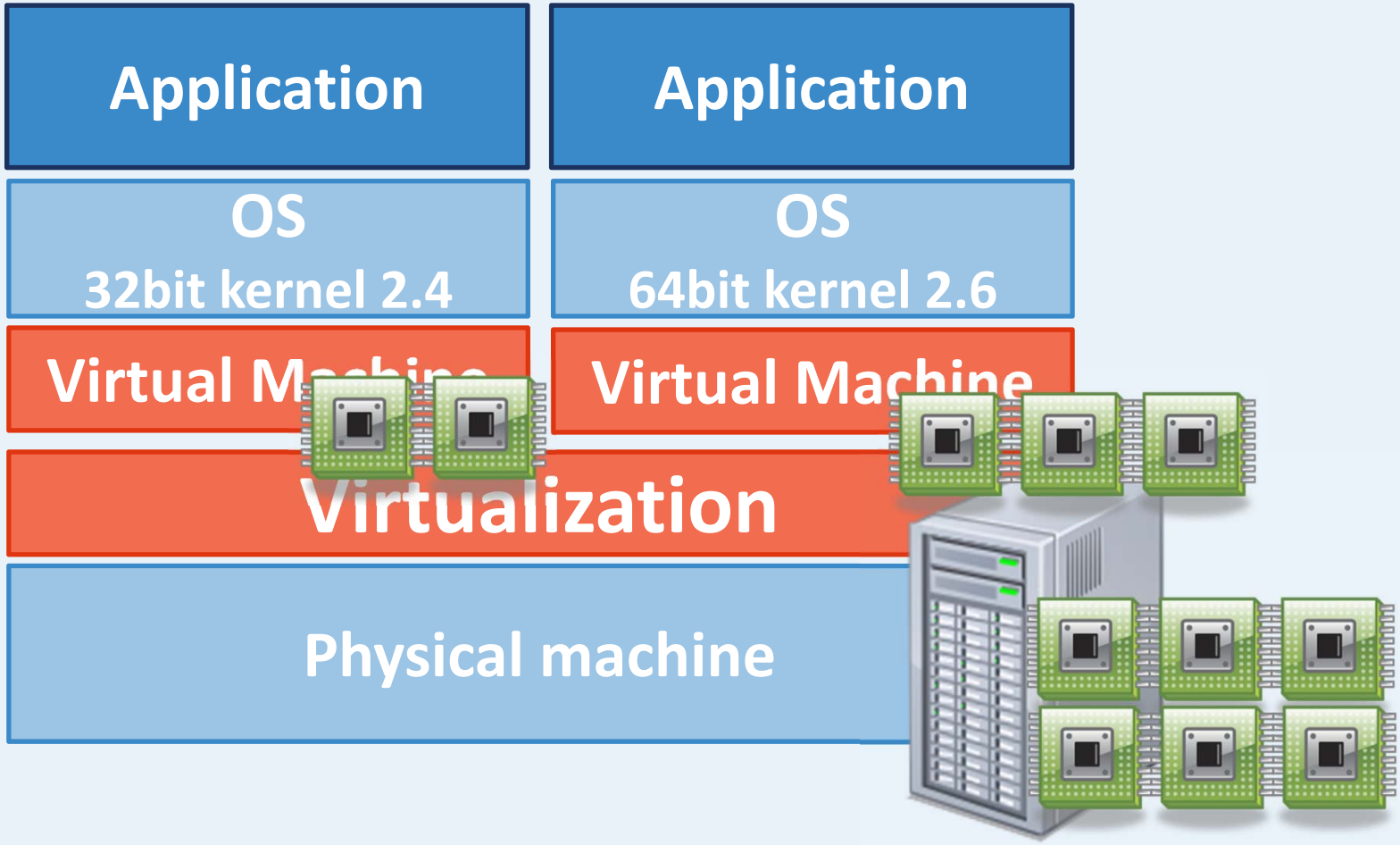
```
# xm  start  myGuestOS
```


```
# xm  reboot  myGuestOS
```


```
# xm  shutdown myGuestOS
```

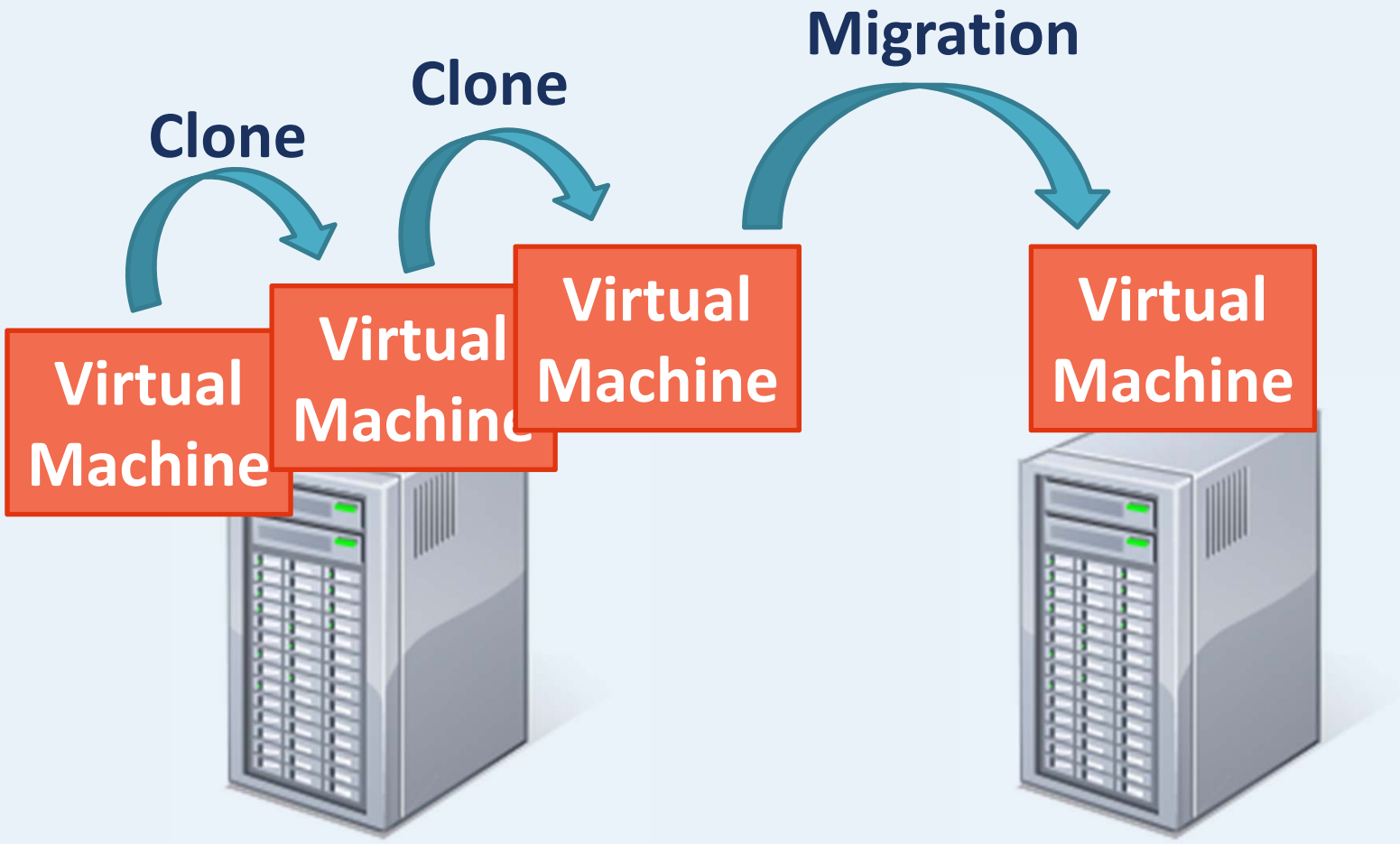
Virtual machine is

dynamic and Flexible

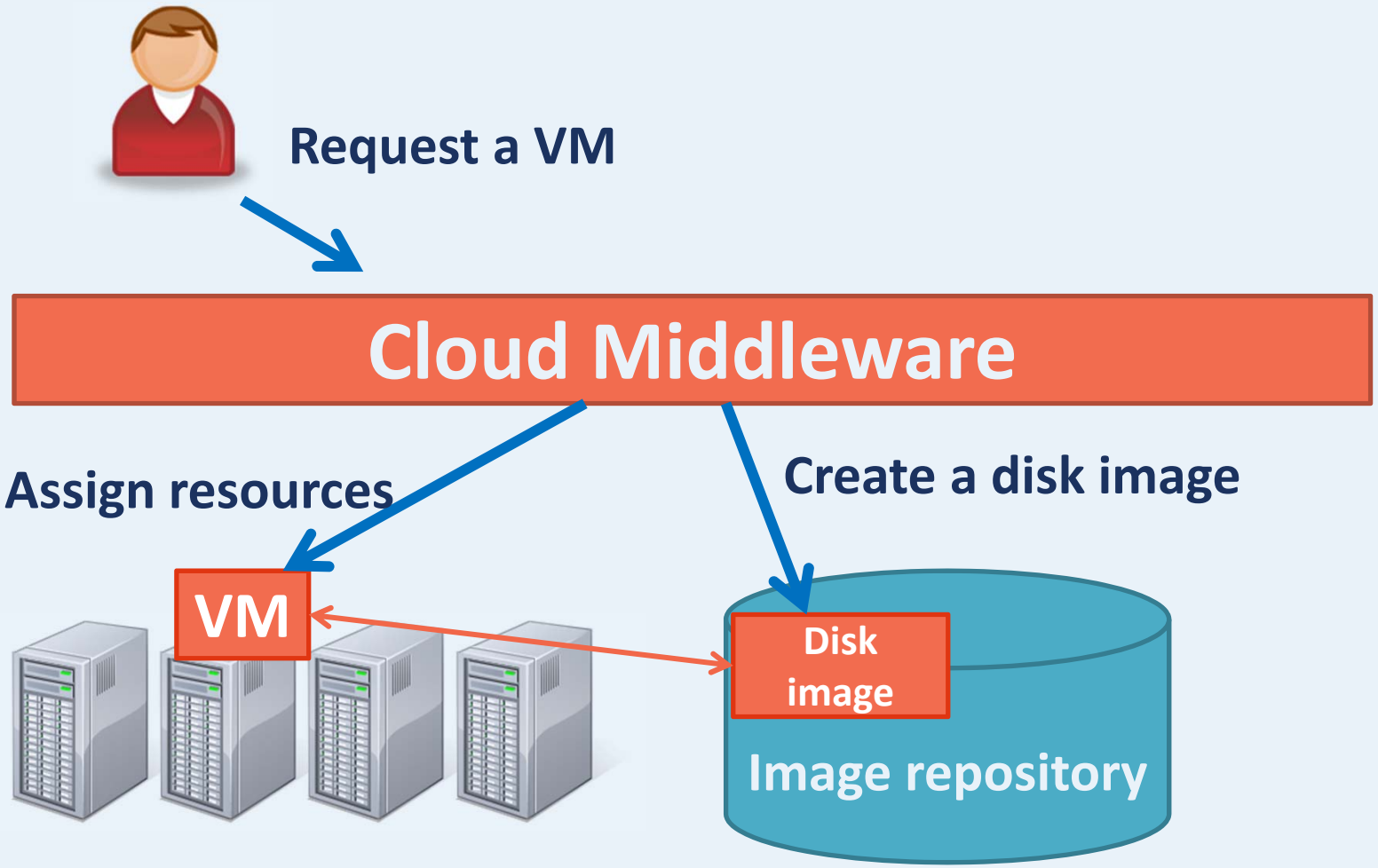


Virtual machine is

easy to deploy



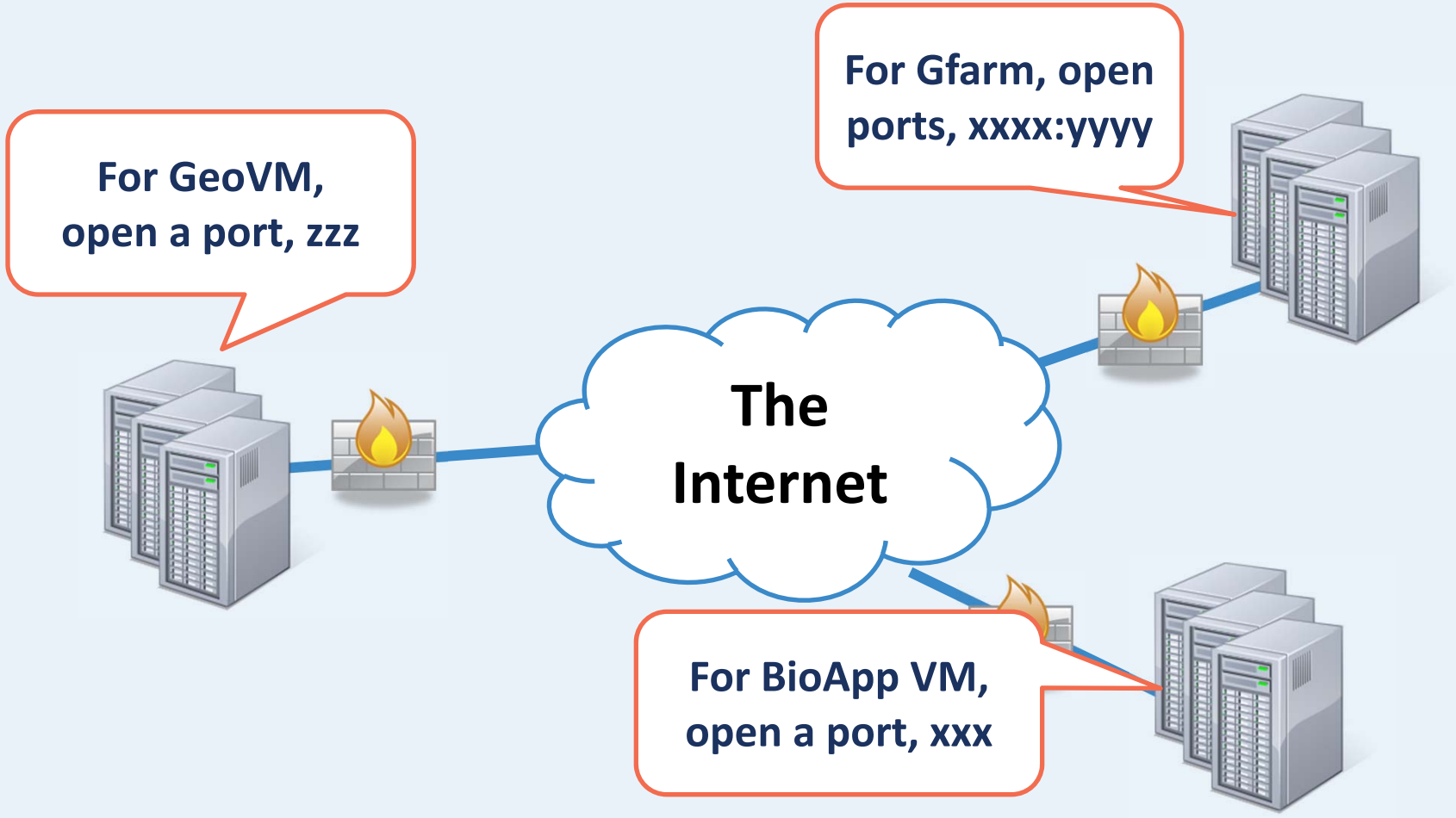
Deployment of VMs is fully controllable from Software



But, …

Network is still not fully controllable from software

Different sites have different policies



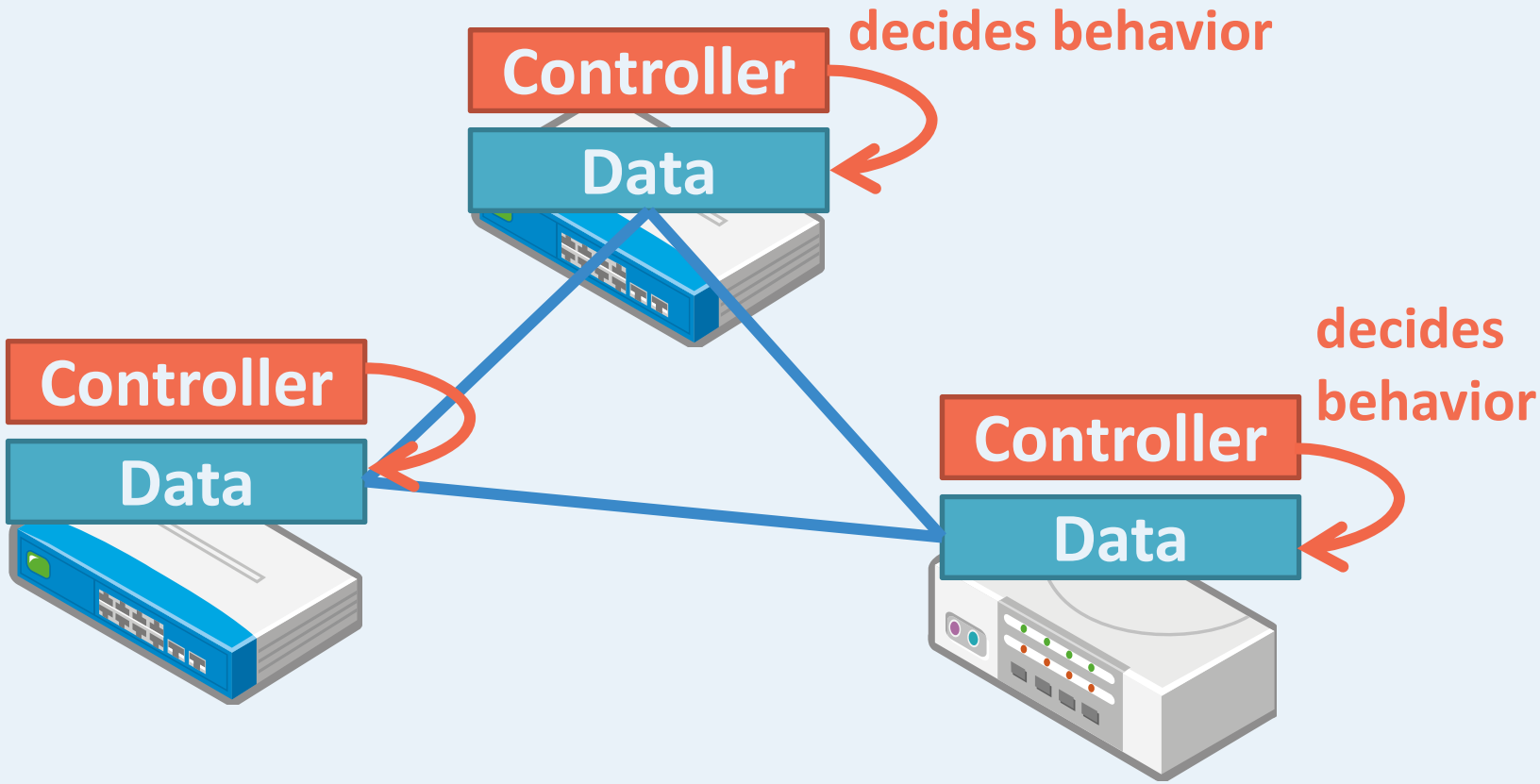
Every time manual operations are required to change the policies

SDN:

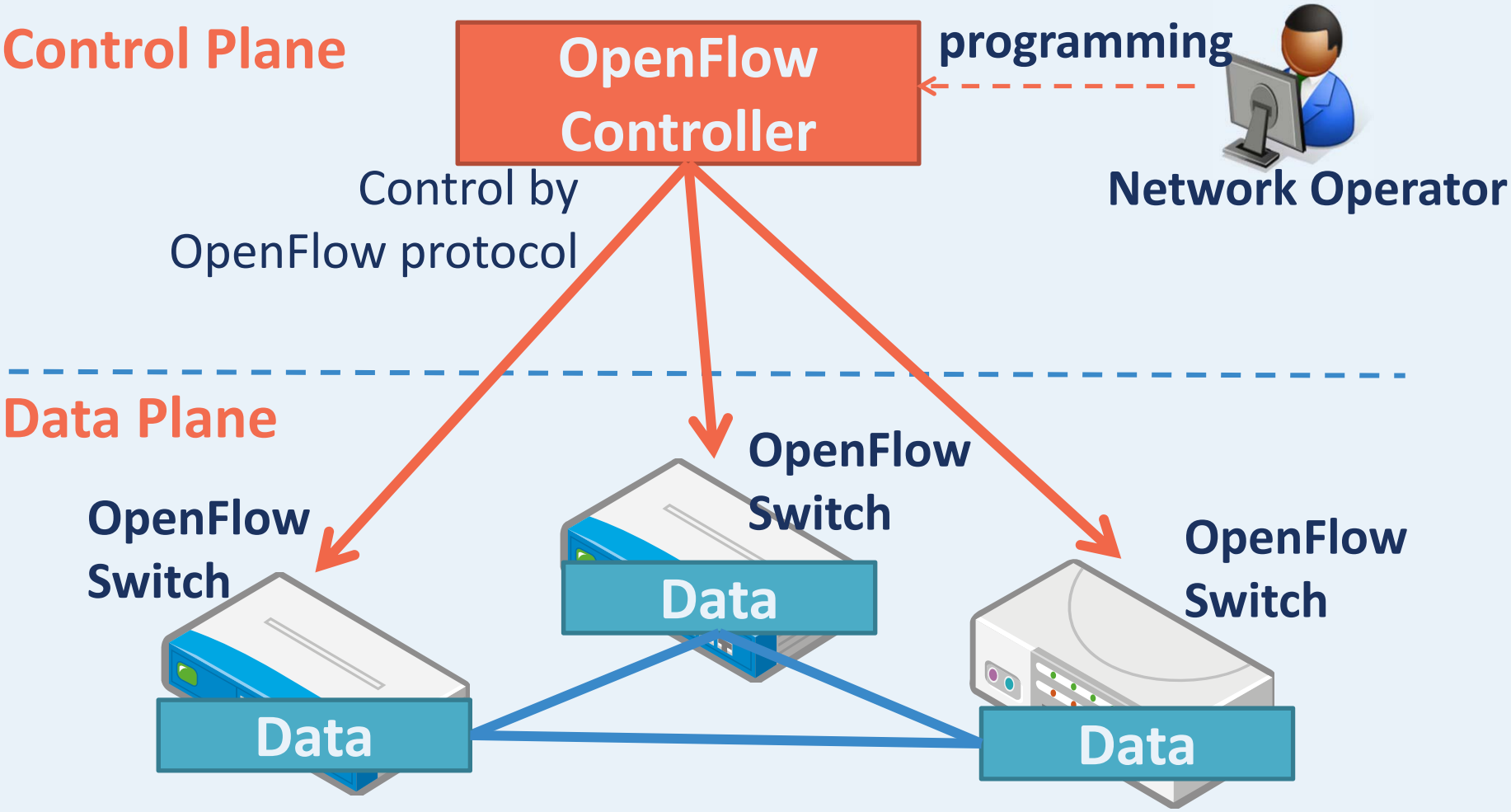
Software-Defined Network

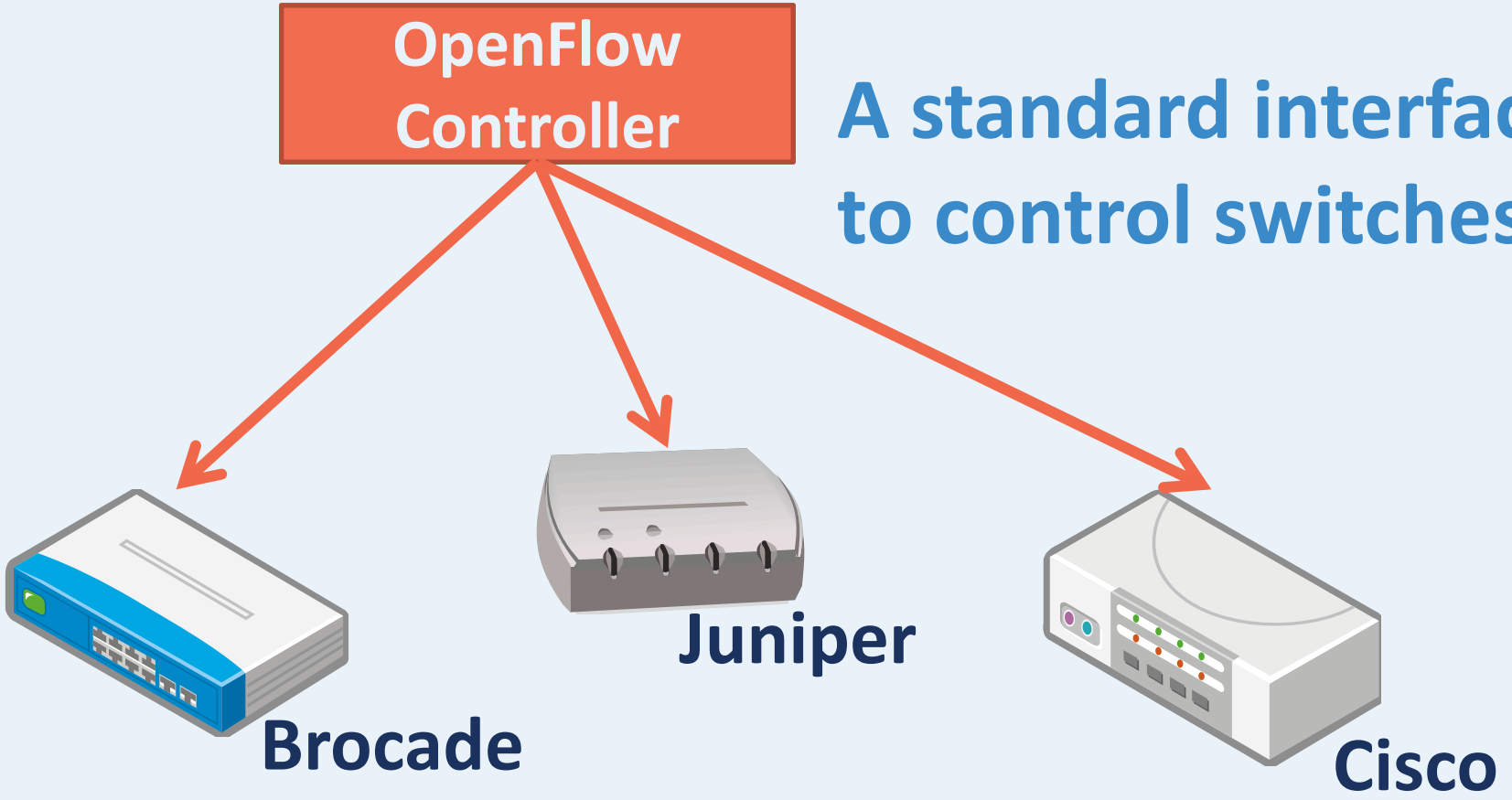
An open standard enables SDN

Before OpenFlow:

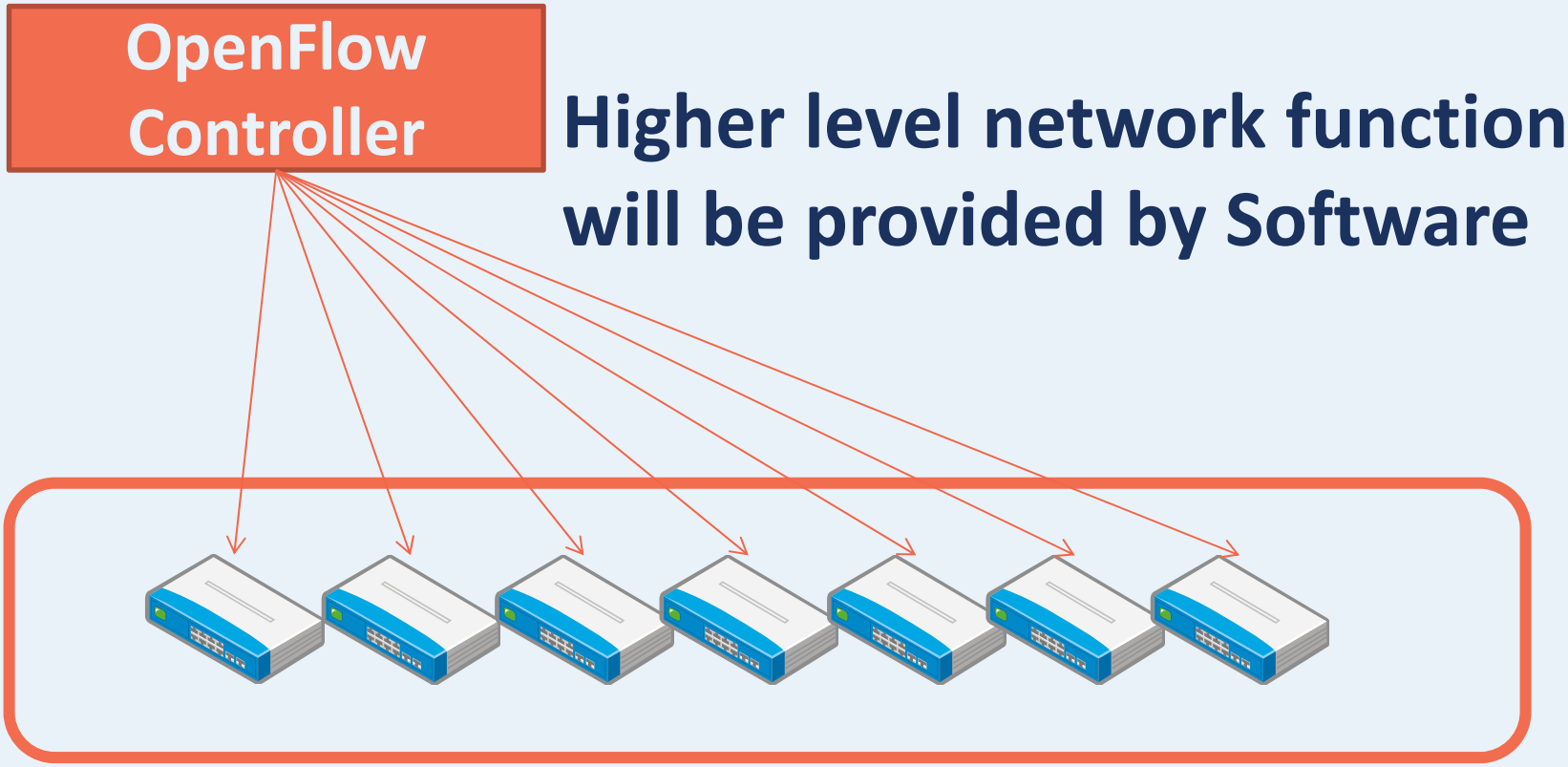


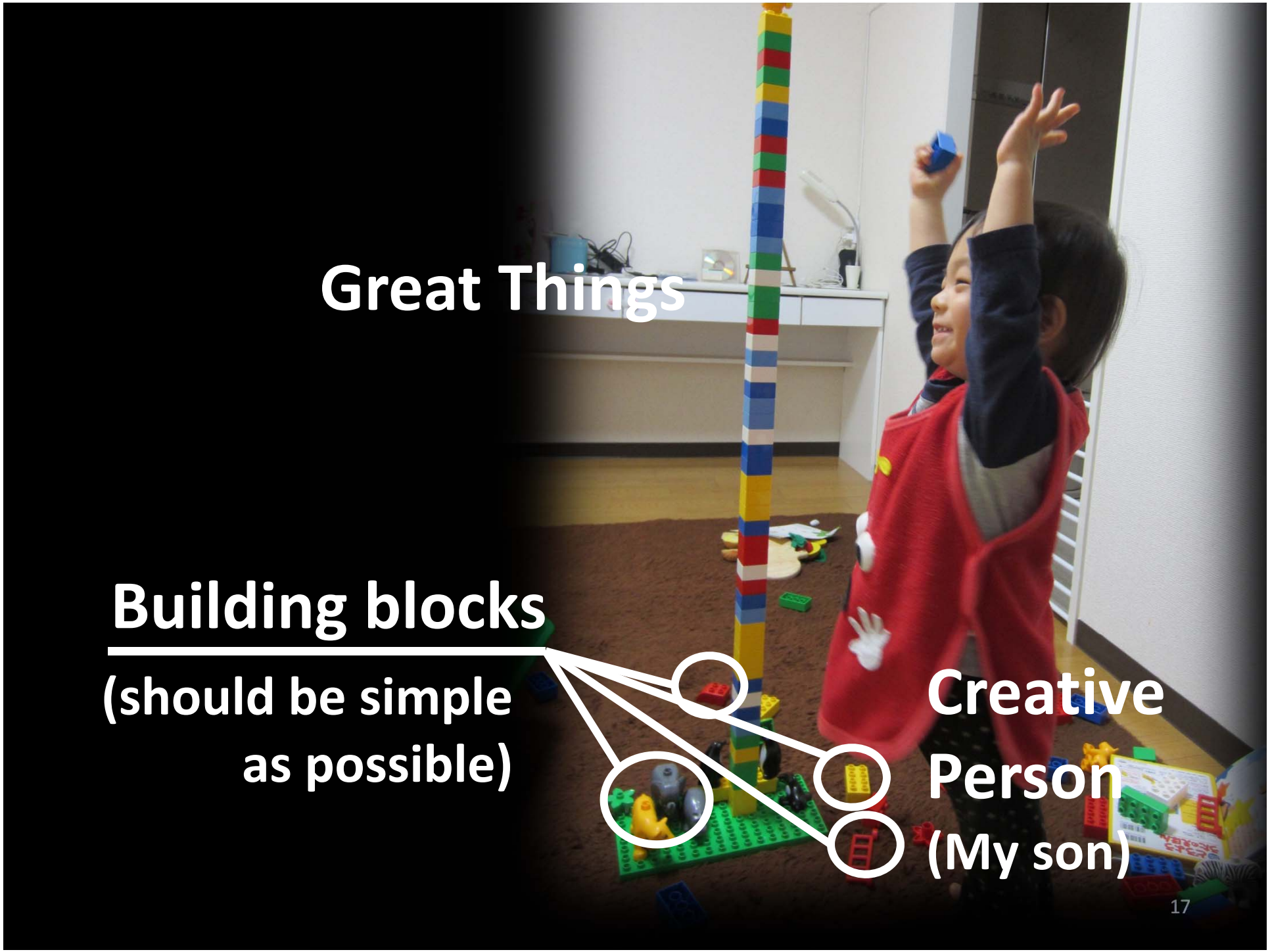
OpenFlow:


Control Plane
OpenFlow Controller
programming
Network Operator
Control by
OpenFlow protocol
Data Plane
OpenFlow Switch
Data
OpenFlow Switch
Data
OpenFlow Switch
Data



It is hard to differentiate from the others.
Switches are going to be simple and commodity.


OpenFlow Controller
Higher level network functions will be provided by Software
Simple, cheap, commodity switches
Switch pool

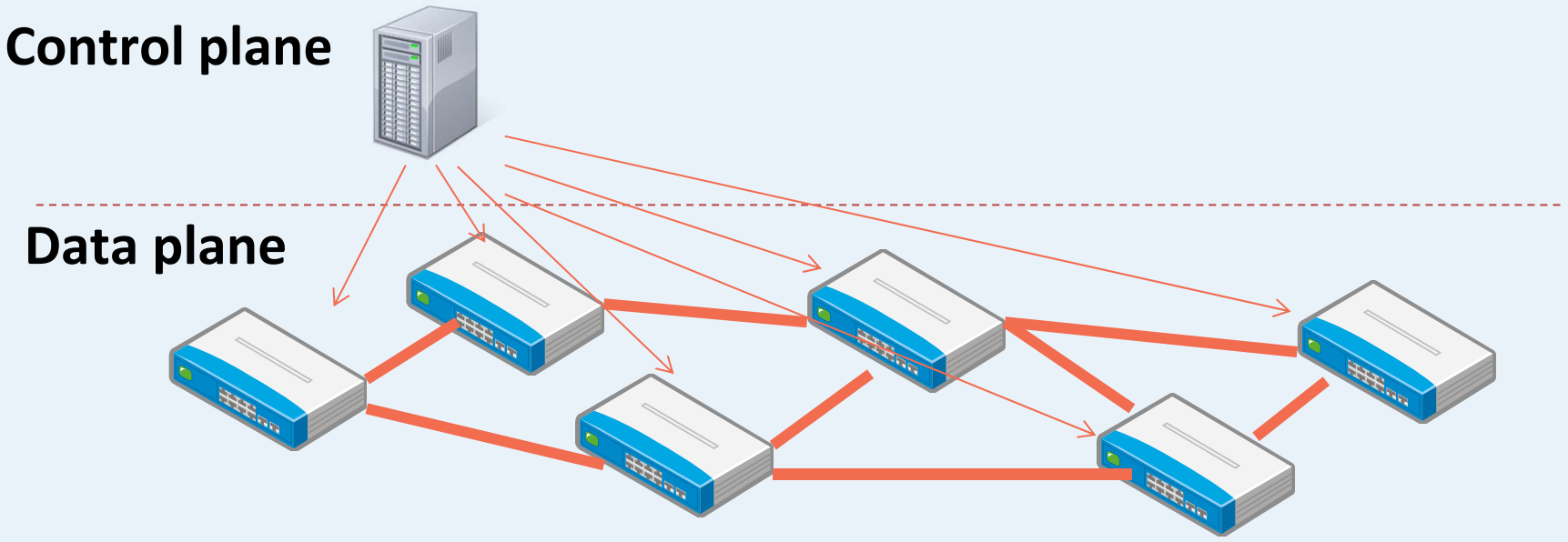

Great Things
Building blocks
(should be simple as possible)
Creative Person (My son)

How to move into OpenFlow?

- Hop by Hop Style
- Overlay Style

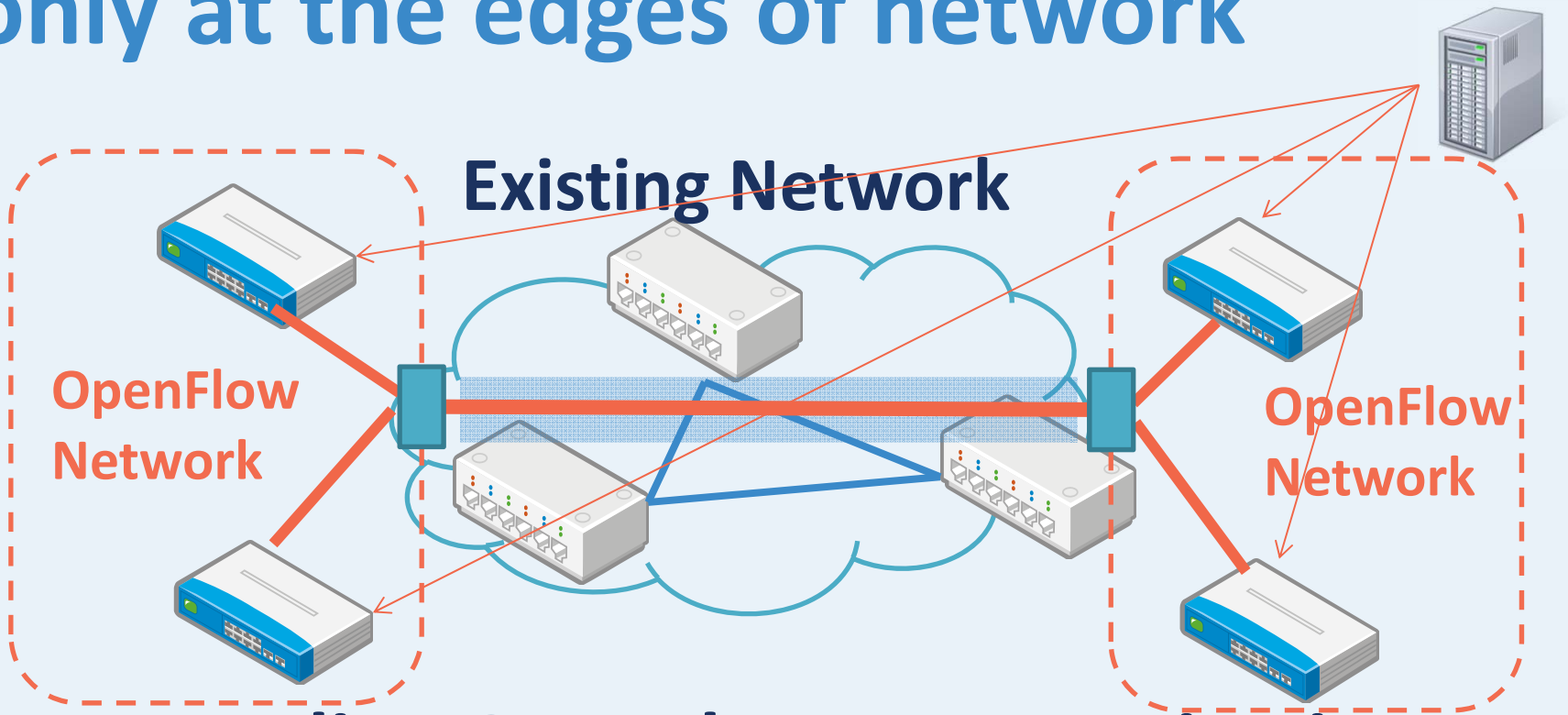
Hop by Hop style:

Replace existing network with OpenFlow network completely



Overlay style:

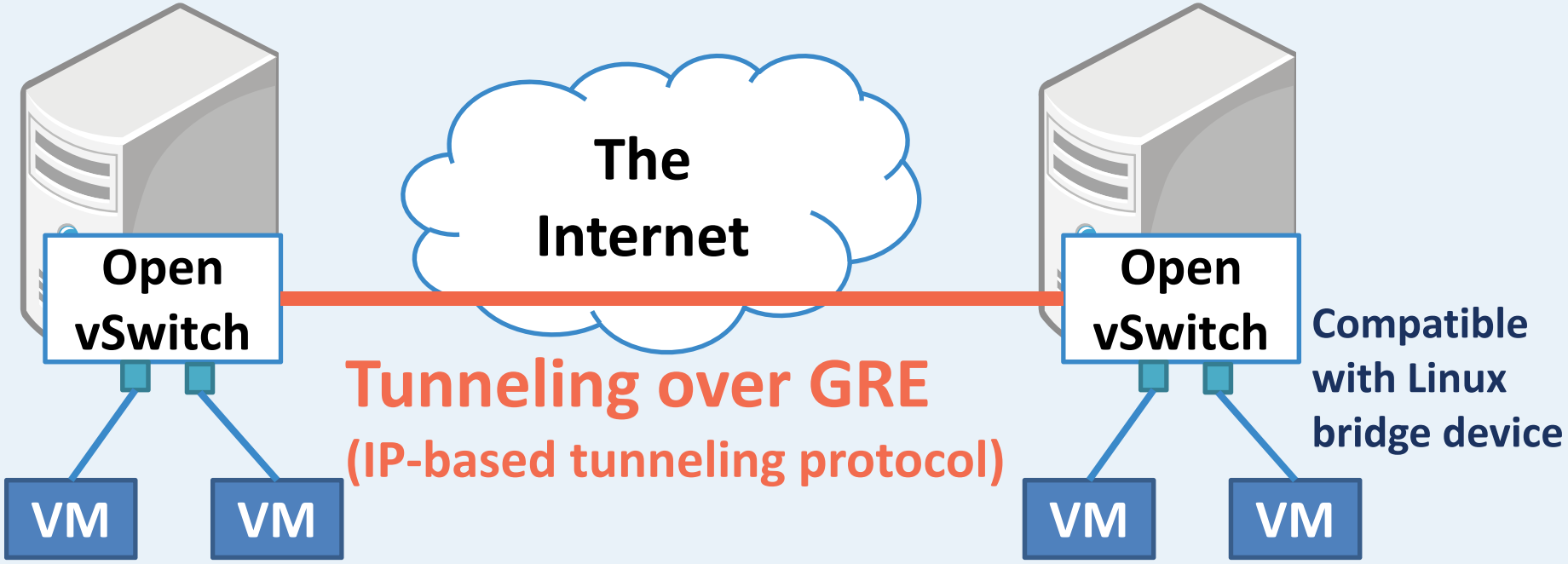
Introduce OpenFlow switches only at the edges of network



Tunneling OpenFlow communications over existing network

Overlay style with Open vSwitch

A software-based virtual switch implementation of OpenFlow



OpenFlow controller development frameworks:

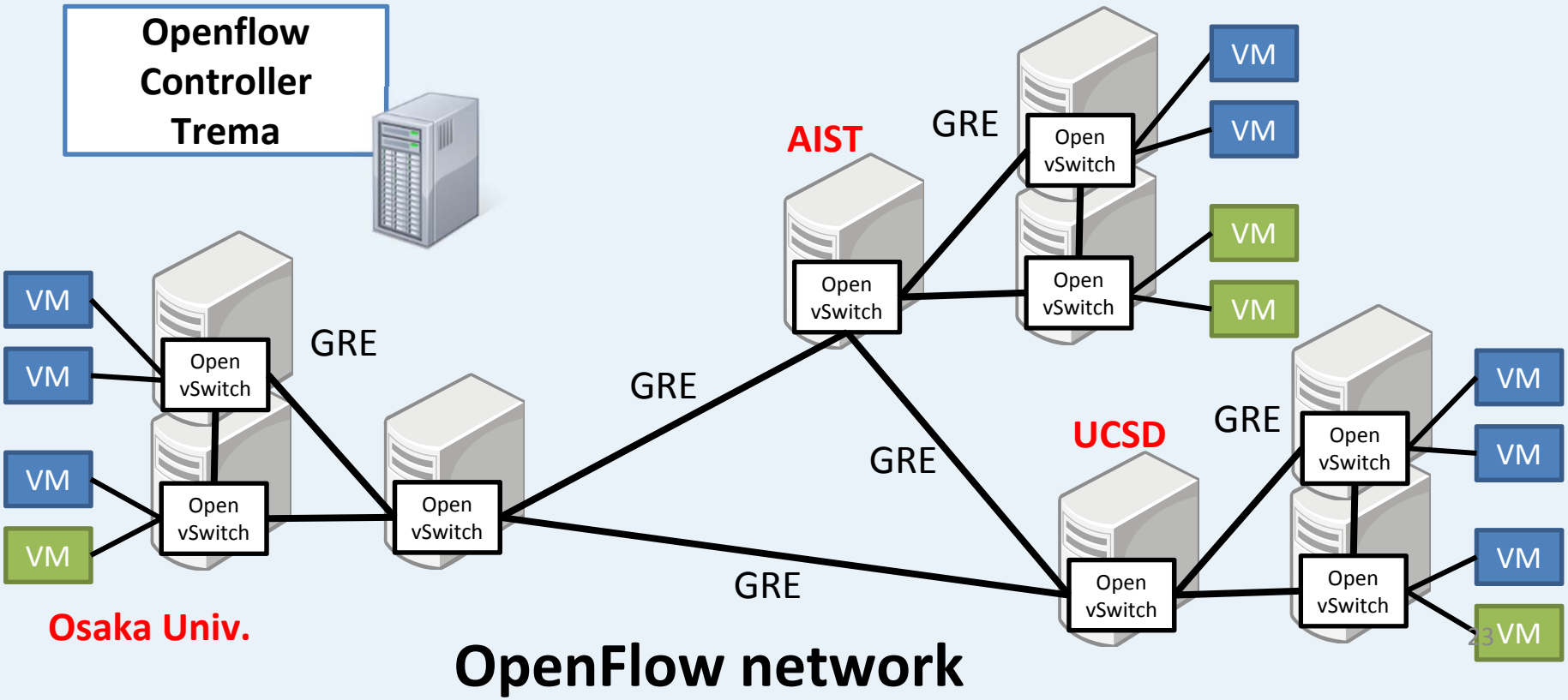
Trema (Ruby)

POX (Python)

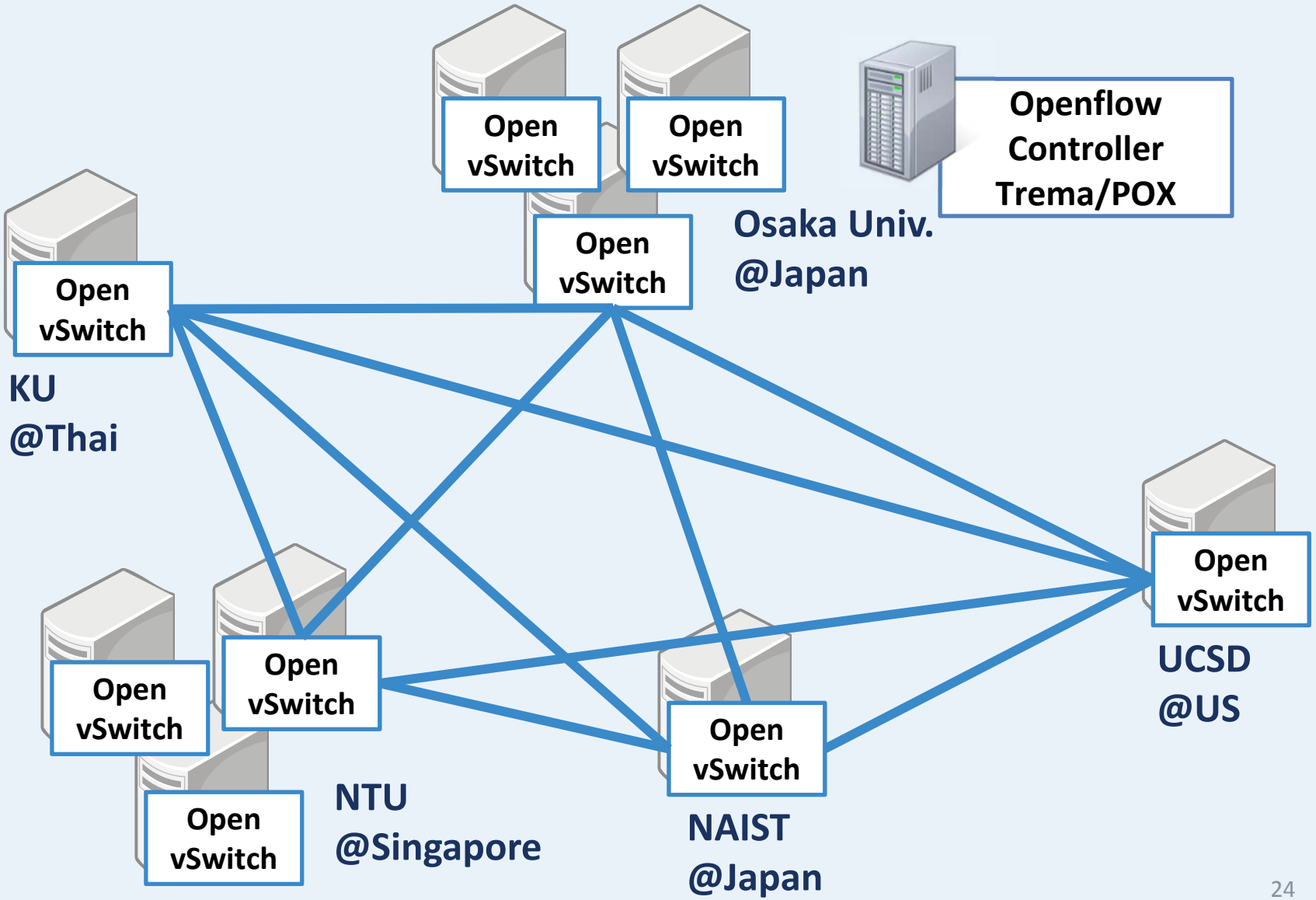
NOX (C++)

Floodlight (Java)

Demo environment at Pragma 22



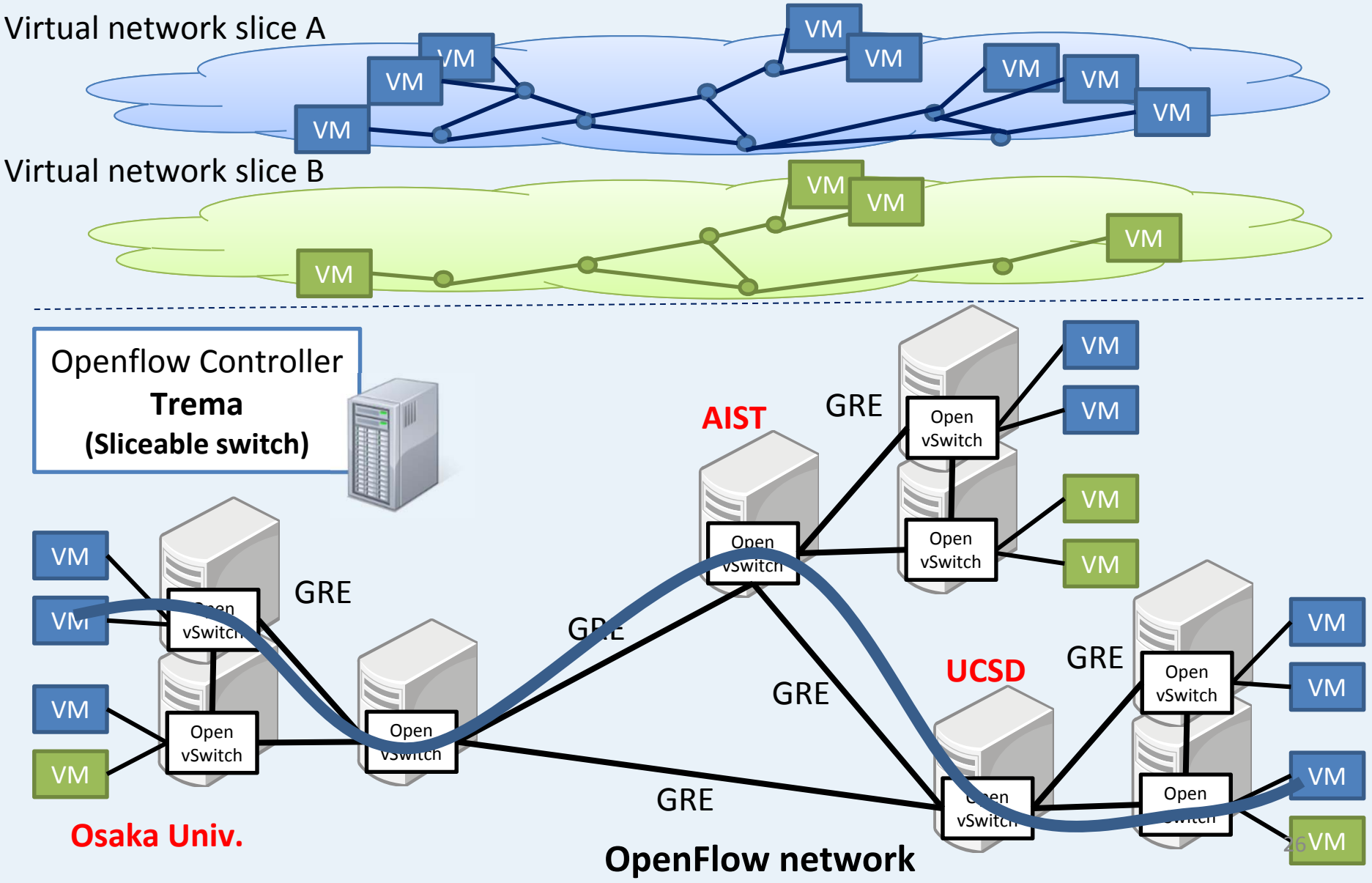
Pragma 24


Open vSwitch
Open vSwitch
Open vSwitch
Openflow Controller Trema/POX
Osaka Univ. @Japan
Open vSwitch
KU @Thai
Open vSwitch
Open vSwitch
Open vSwitch
Open vSwitch
UCSD @US
NTU @Singapore
NAIST @Japan

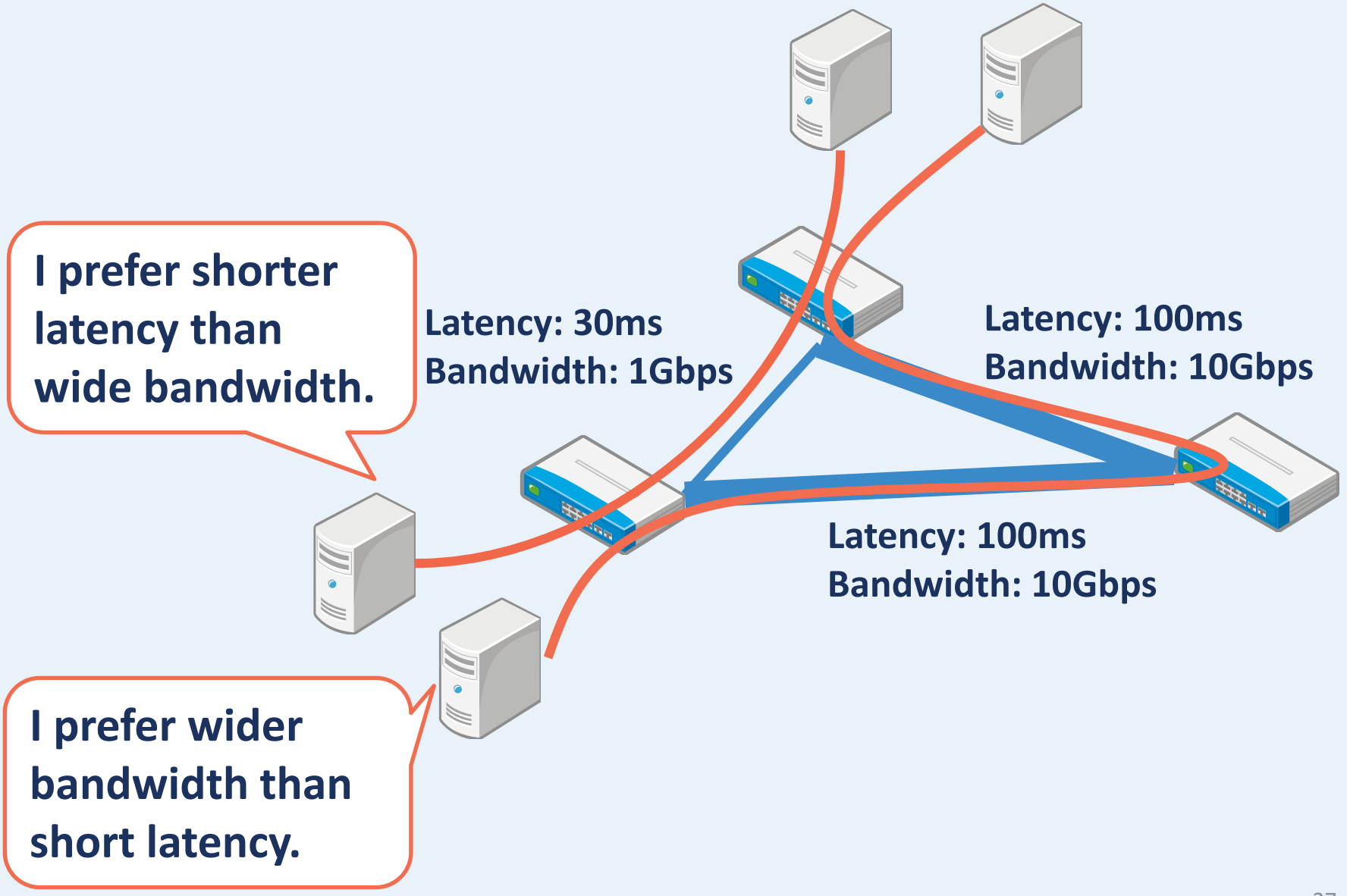
Current Experiments

- Virtual Sliceable Network
- Network throughput-aware routing
- Quick failure recovery

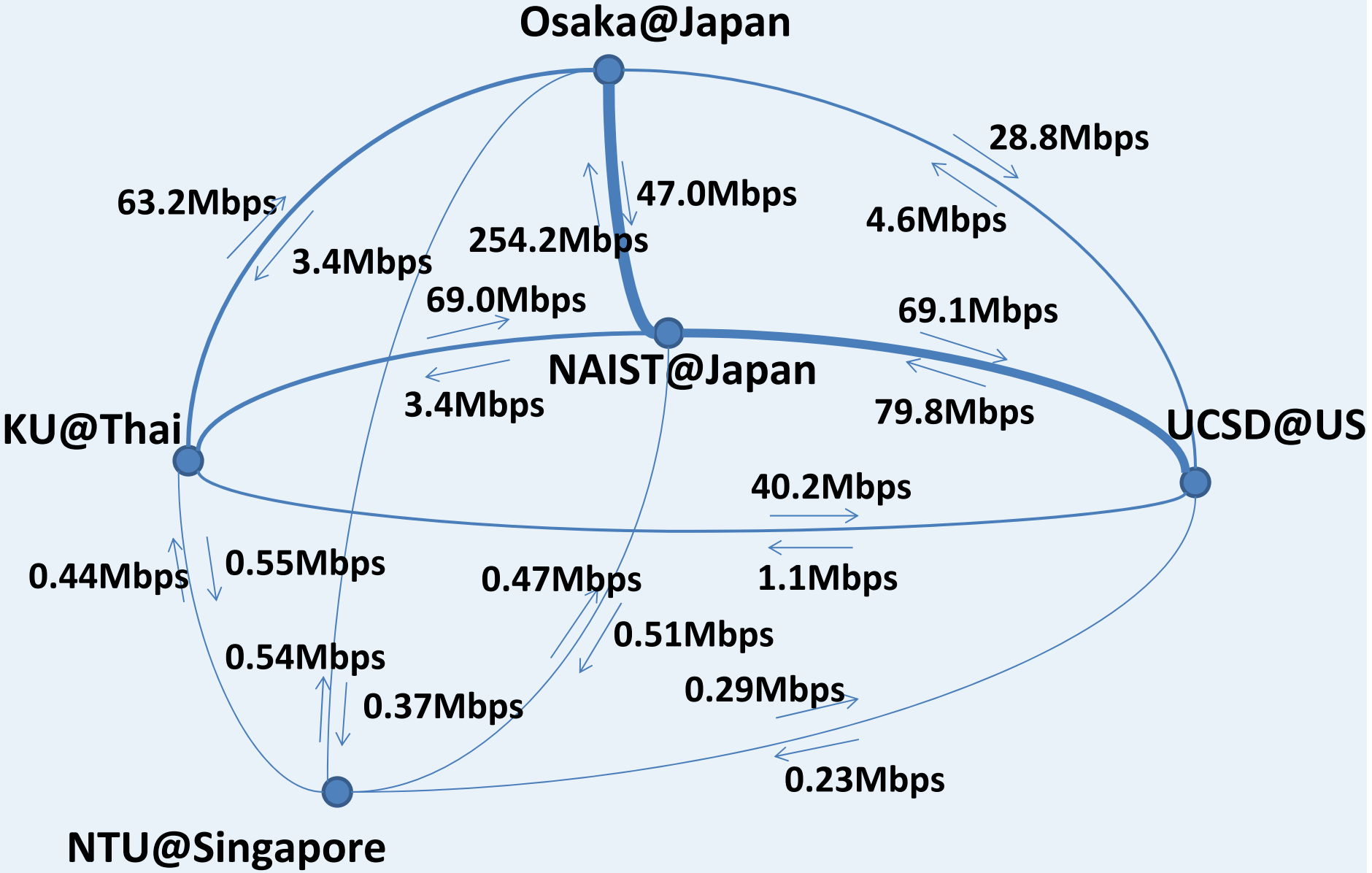
Virtual Sliceable Switch



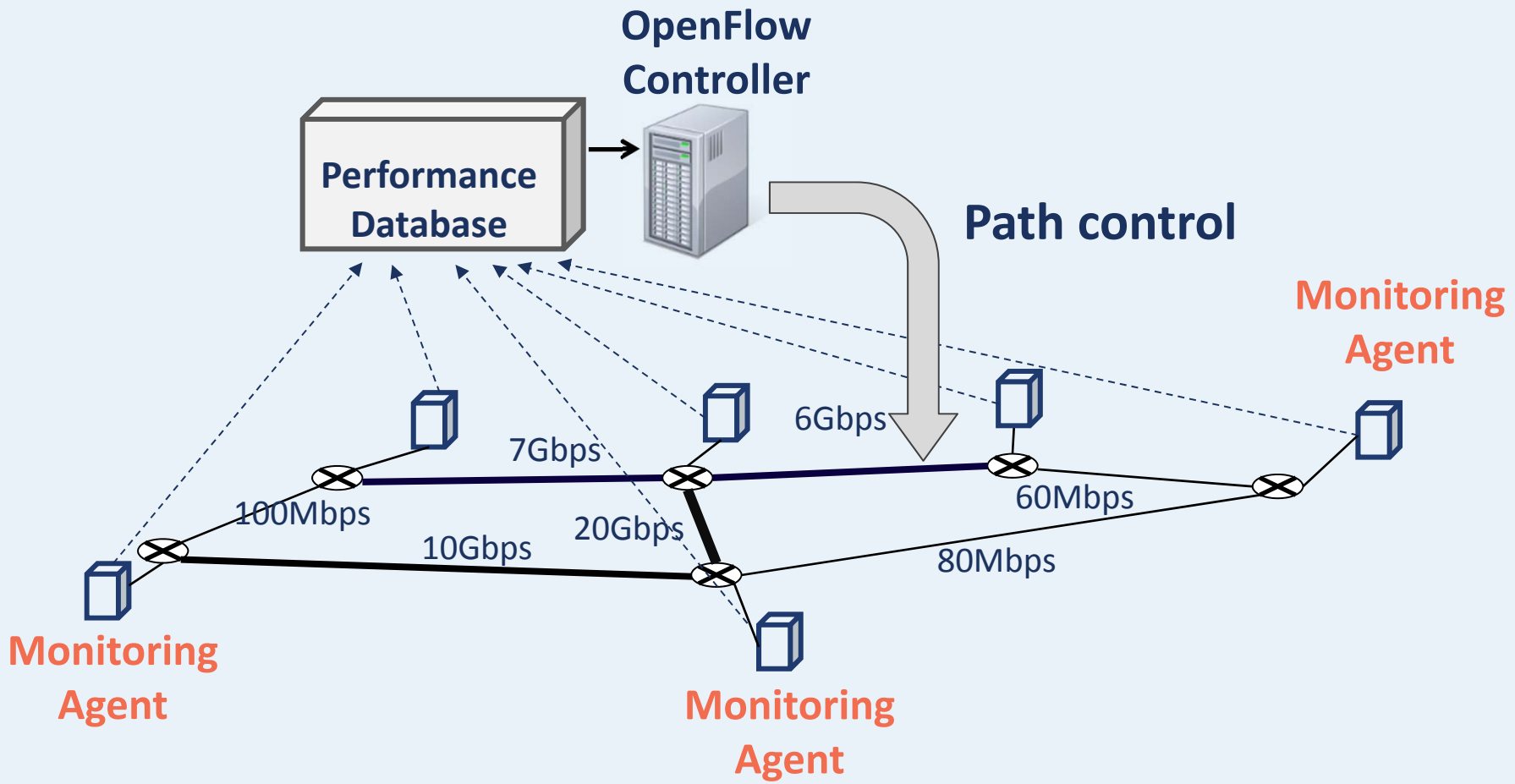
Network throughput-aware routing



Actual throughput in wide-area

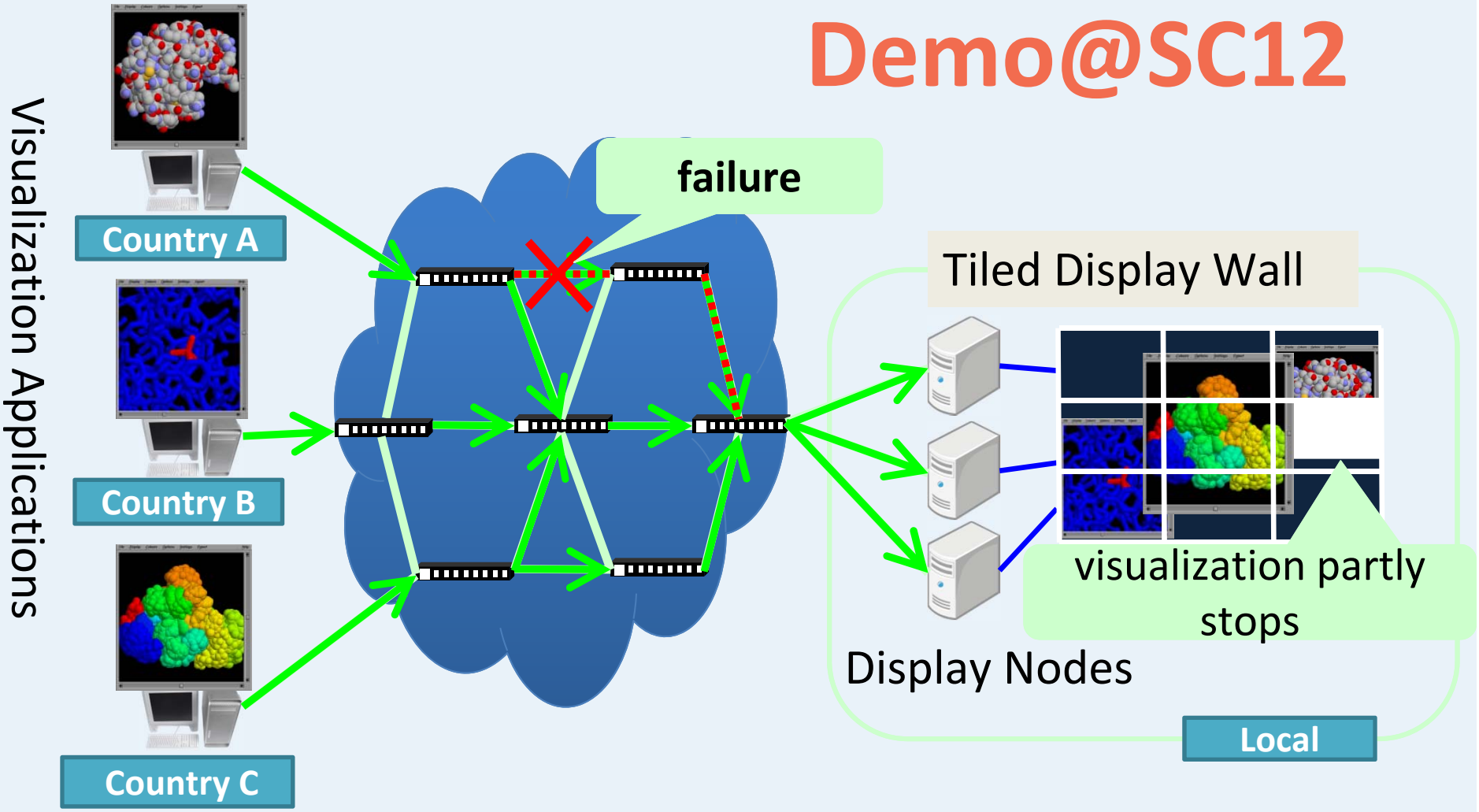


Implementation of Throughput-aware routing switches



Quick failure recovery for remote visualization system

Demo@SC12

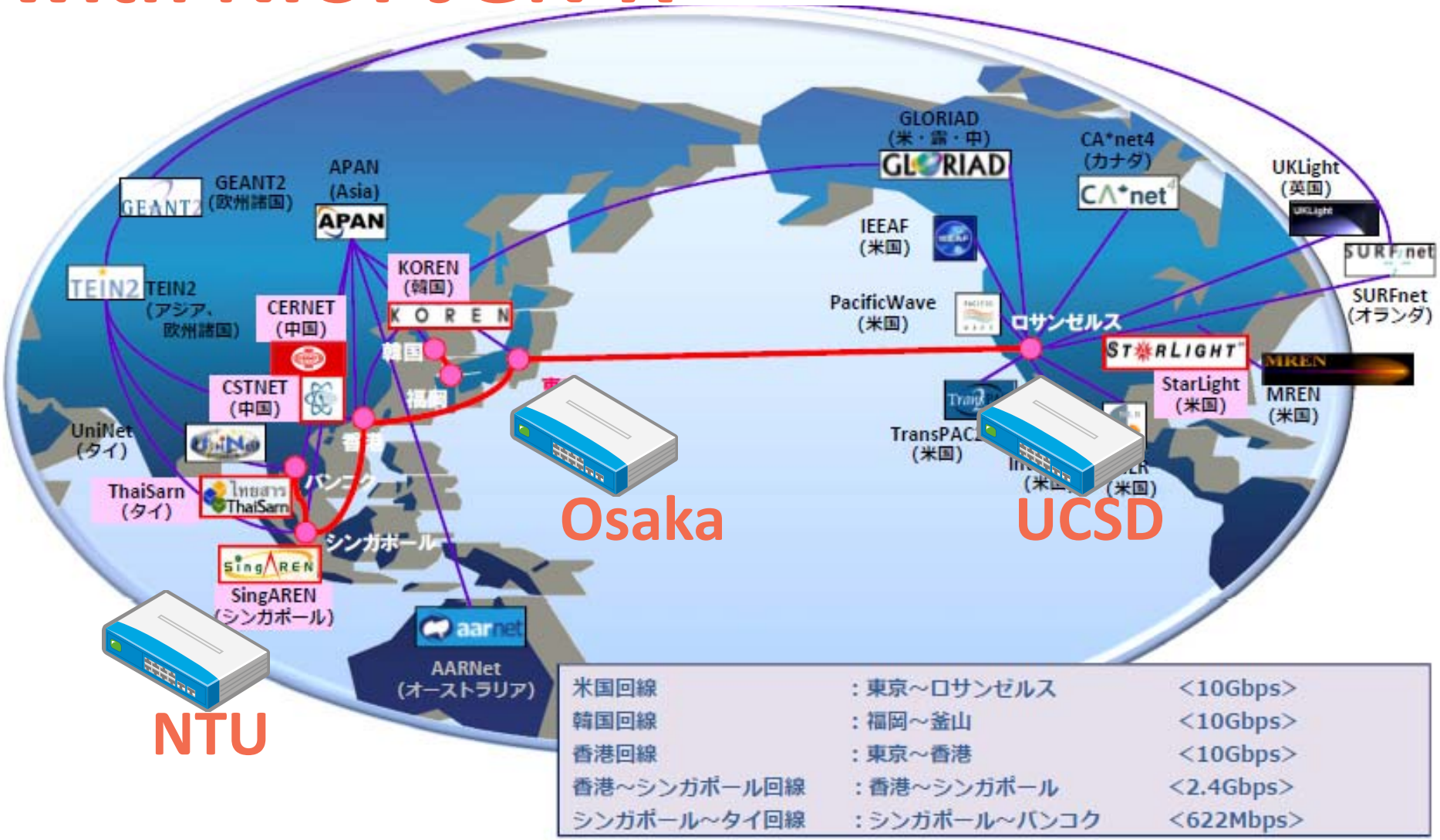


Demo video

Future Plans

- **Network throughput-aware routing**
  - develop more smarter controller
    - Dynamic
- **Hadoop cluster**
  - NTU will try to build Hadoop cluster on the OpenFlow network of Pragma
- **Hop by Hop style OpenFlow**
  - Using NICT RISE network with hardware OpenFlow switches

Hop by Hop style OpenFlow with NICT JGN-X



Other related works on OpenFlow

New generation HPC:

accelerating computation by using OpenFlow

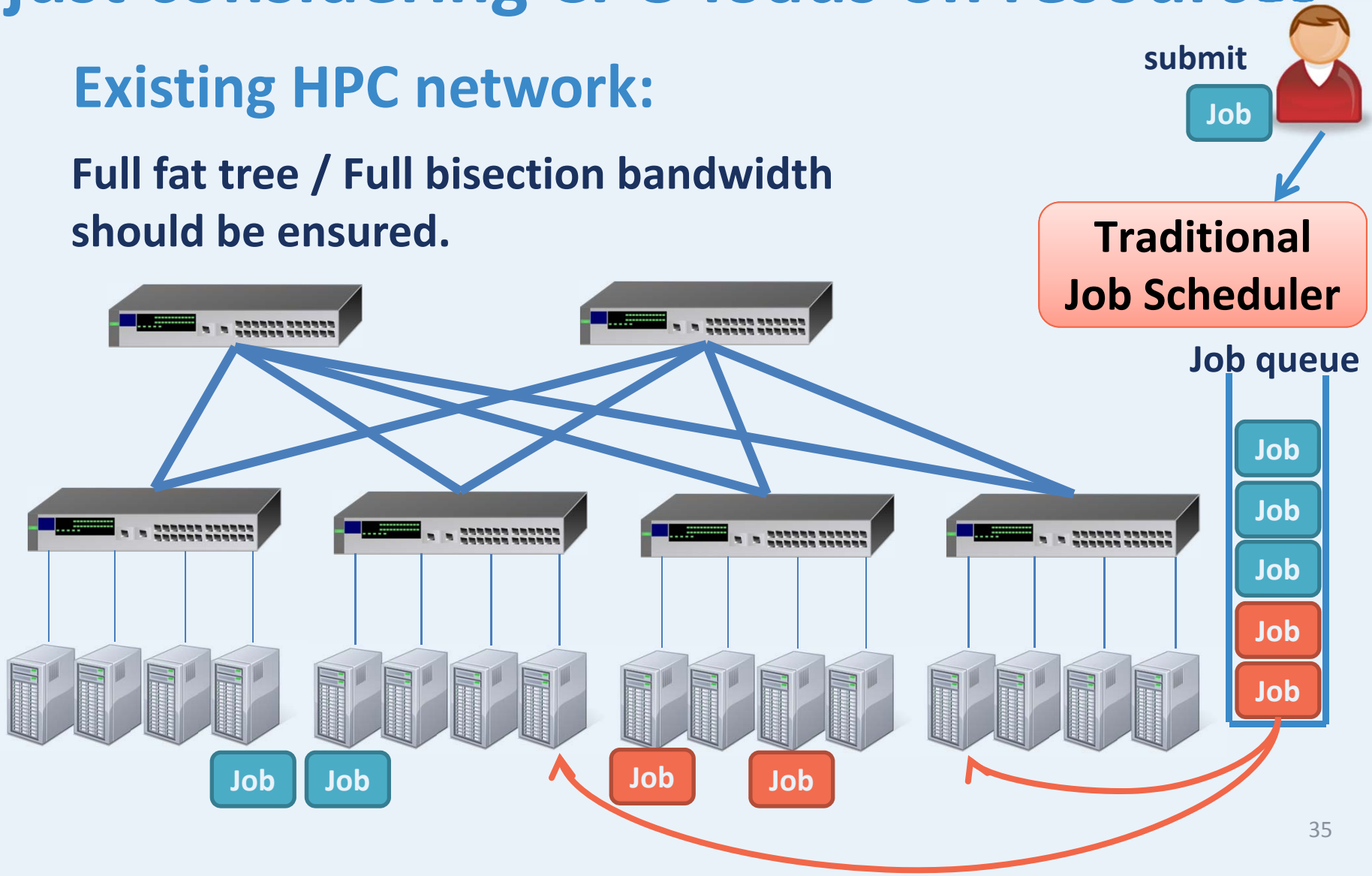
- **Network-aware job scheduler**
  schedule and assign network resources as well as computing resources
- **OpenFlow-enabled MPI**
  accelerating collective communication using OpenFlow

Traditional Job Scheduler

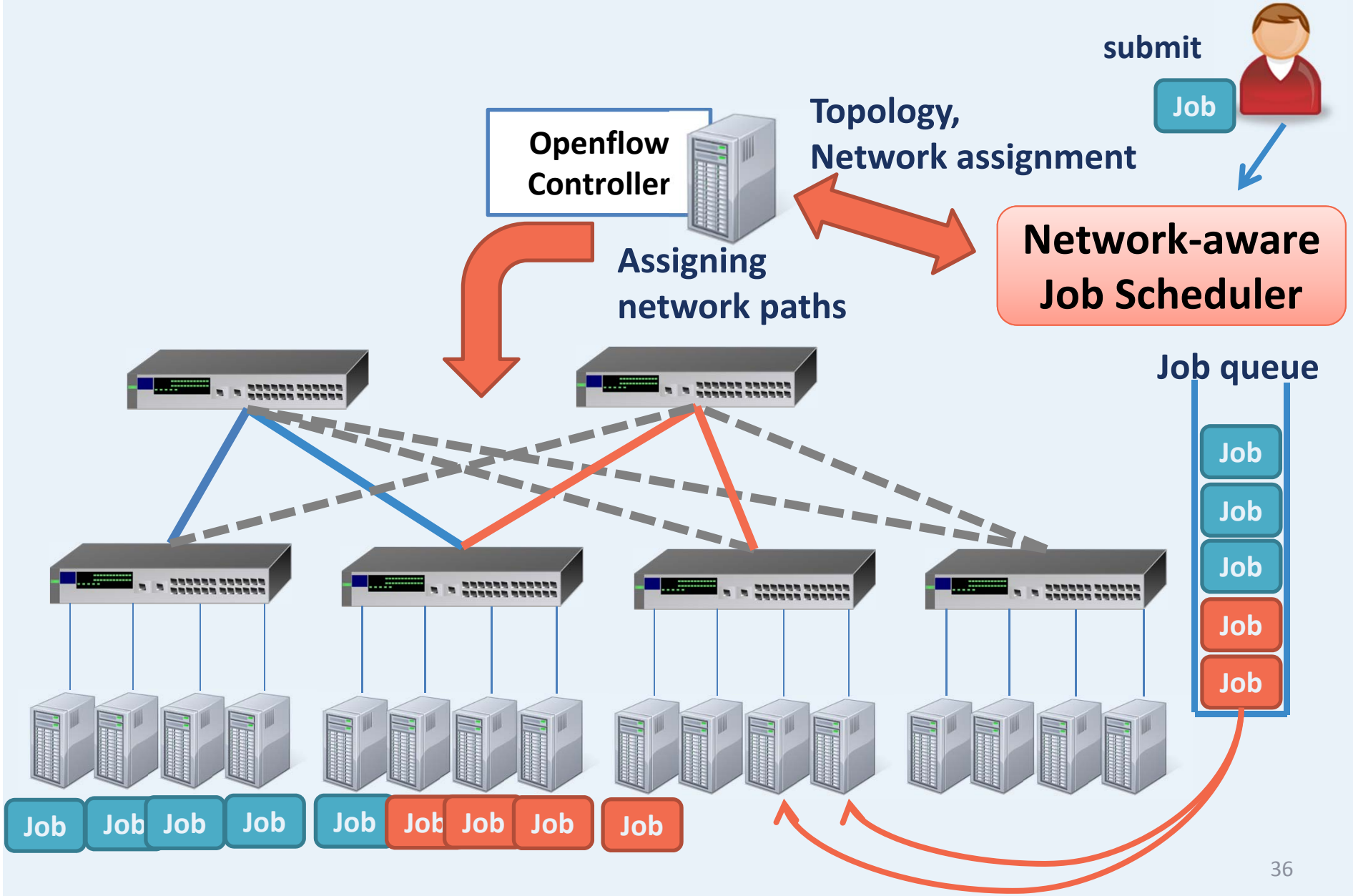
just considering CPU loads on resources

Existing HPC network:

Full fat tree / Full bisection bandwidth should be ensured.



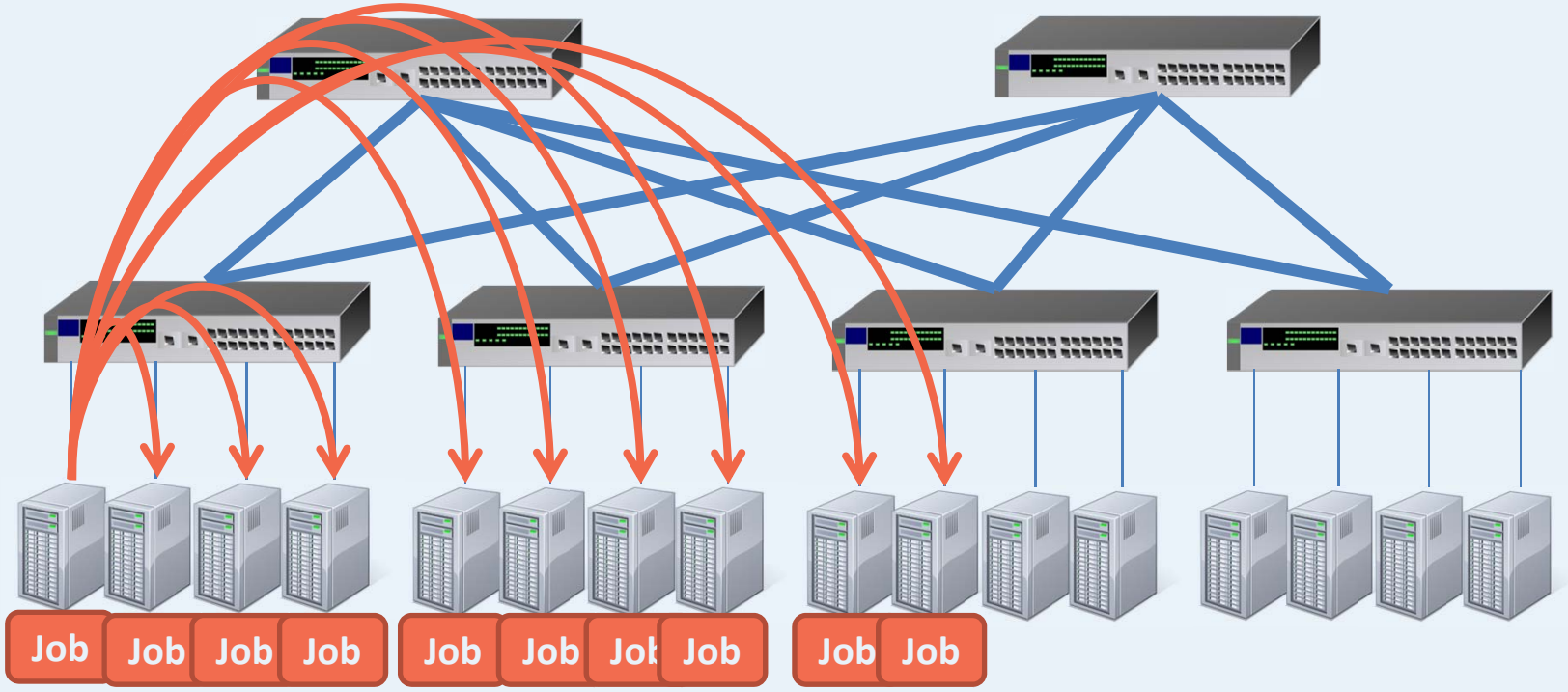
Network-aware Job Scheduler



OpenFlow-enabled MPI

accelerating collective communication using OpenFlow (MPI_Bcast, MPI_Allgather)

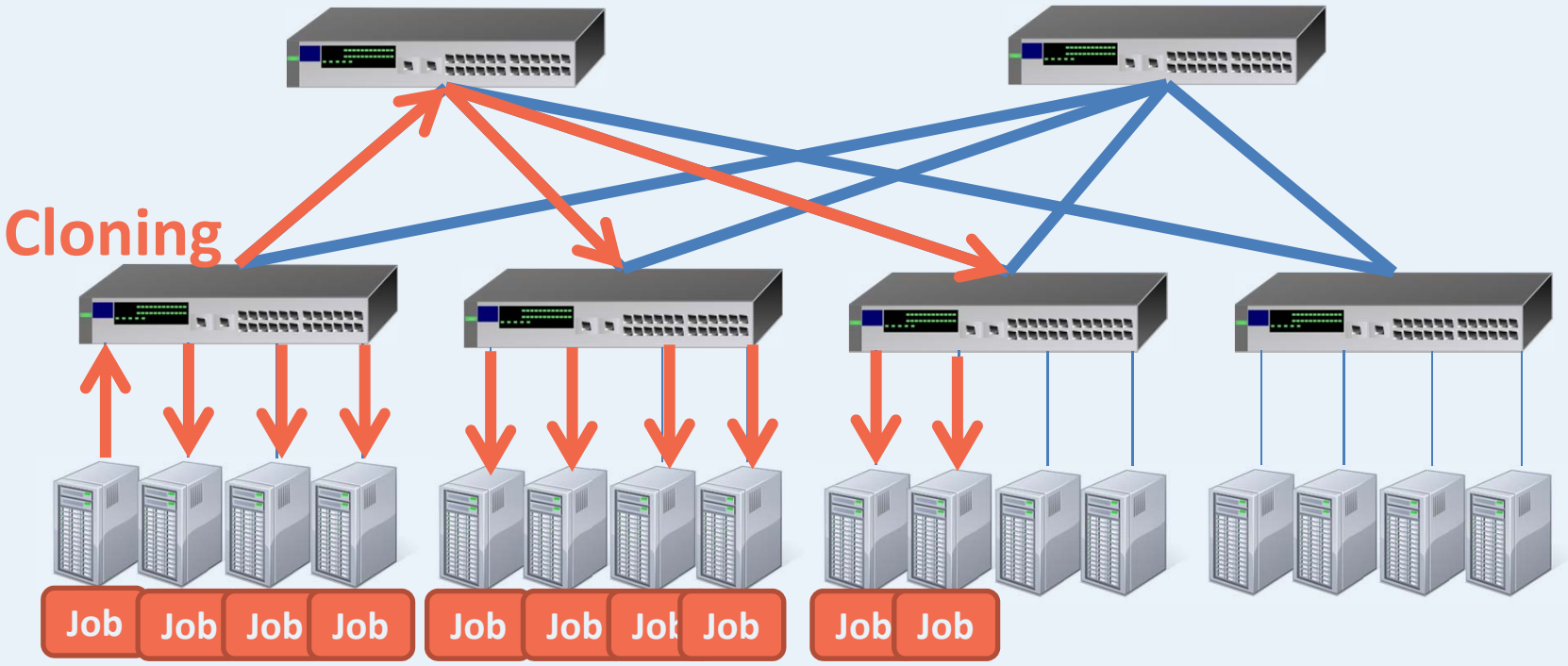
Primitive implementation



OpenFlow-enabled MPI

accelerating collective communication using OpenFlow (MPI_Bcast, MPI_Allgather)

OpenFlow-enabled version



Summary

- Development of OpenFlow network in Pragma Testbed
  - Overlay style OpenFlow (using vSwitch & GRE)
  - Need more App examples
    - NTU will try Hadoop cluster
  - Hop by Hop style OpenFlow
    - Using NICT JGN-X
    - Calling for joining the experiment
- Some examples of other applications using OpenFlow in this area
  - Network-aware job scheduler
  - OpenFlow-enabled MPI