NCHC | National Center for High-Performance Computing
www.nchc.org.tw/en/

Networking in Virtual Infrastructure and Future Internet

NCHC
Jen-Wei Hu

Overview

- Virtualization
- Networking in Virtualization
- Future Internet

Hardware virtualization

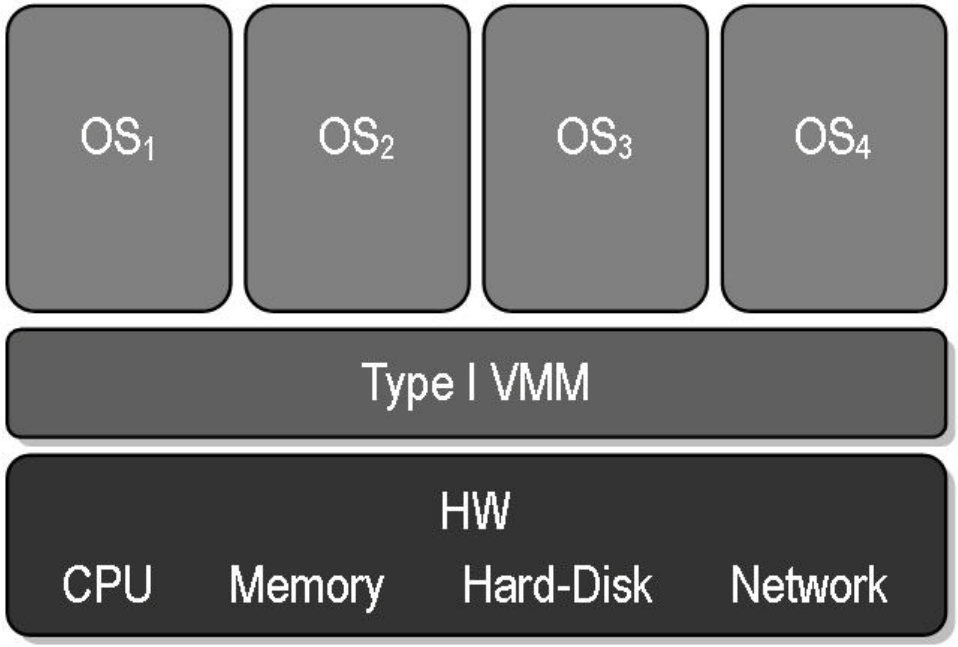
- Hardware virtualization techniques
  - Enable you to run concurrently multiple operating systems on a host computer.
  - Provide isolated execution environments for each virtual machine.

Classification of Hypervisors

- Virtual Machine Monitor (VMM)
- Essentially, hypervisors could be classified into two types according to the resident position of host machine.
  - Type I hypervisor
  - Type II hypervisor

Type I

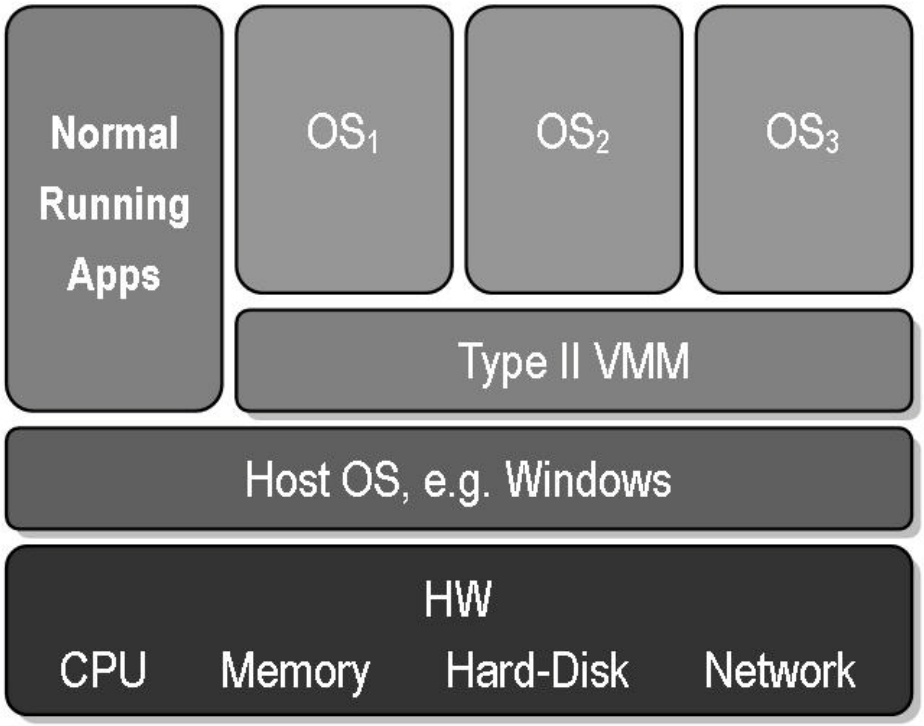
- This type of hypervisors runs directly on the host's hardware to control the hardware.



Source: wikipedi

Type II

- This type of hypervisors likes a conventional software that runs within an operating s



Source: wikipedi

Popular Hypervisors

CITRIX

Windows Server 2008 R2 Hyper-V

Xen Server™

KVM

vmware

Networking in Virtualization

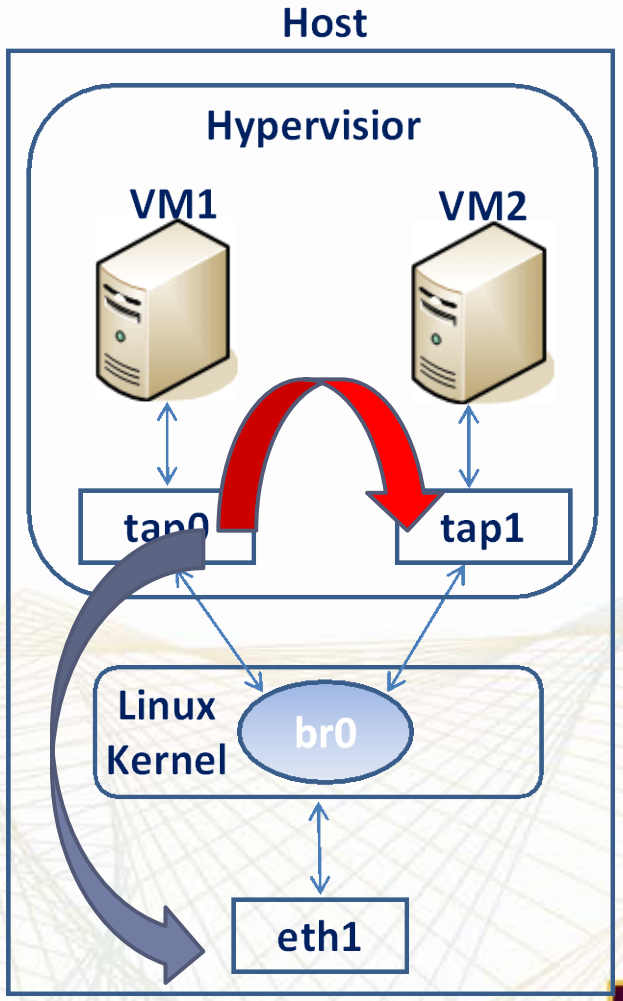
- Compare to CPU, network virtualization has lagged behind.
- Networking is important because a single server will host 40 or more VMs in the near future.

Popular Networking Modes in VMM

- Internal/Host-only networking
- Bridged networking
- Network Address Translation (NAT)

Virtual Ethernet Bridge (VEB)

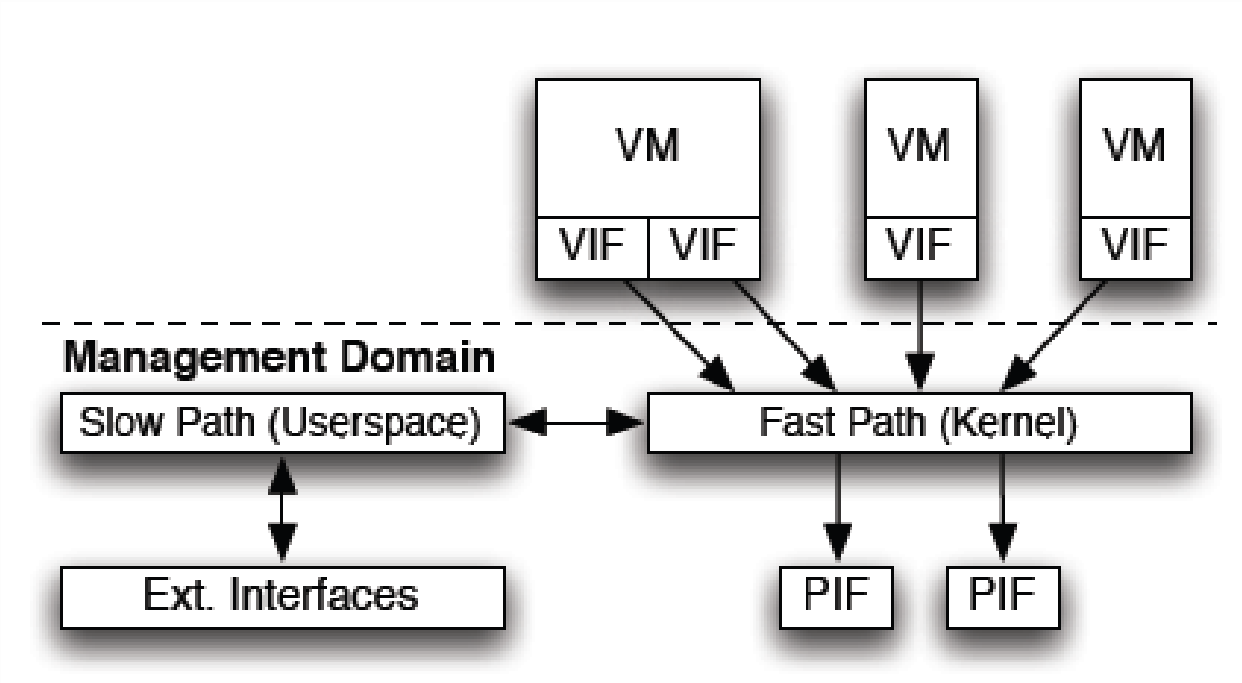
- Supports IEEE802.1d
- Packets can be delivered among VMs and between VM and other machine
- No external hardware required
- Cons:
  - Waste of CPU/memory usage
  - No or lack of traffic visibility
  - Separate policy control from outside network
  - Does not support virtual networks



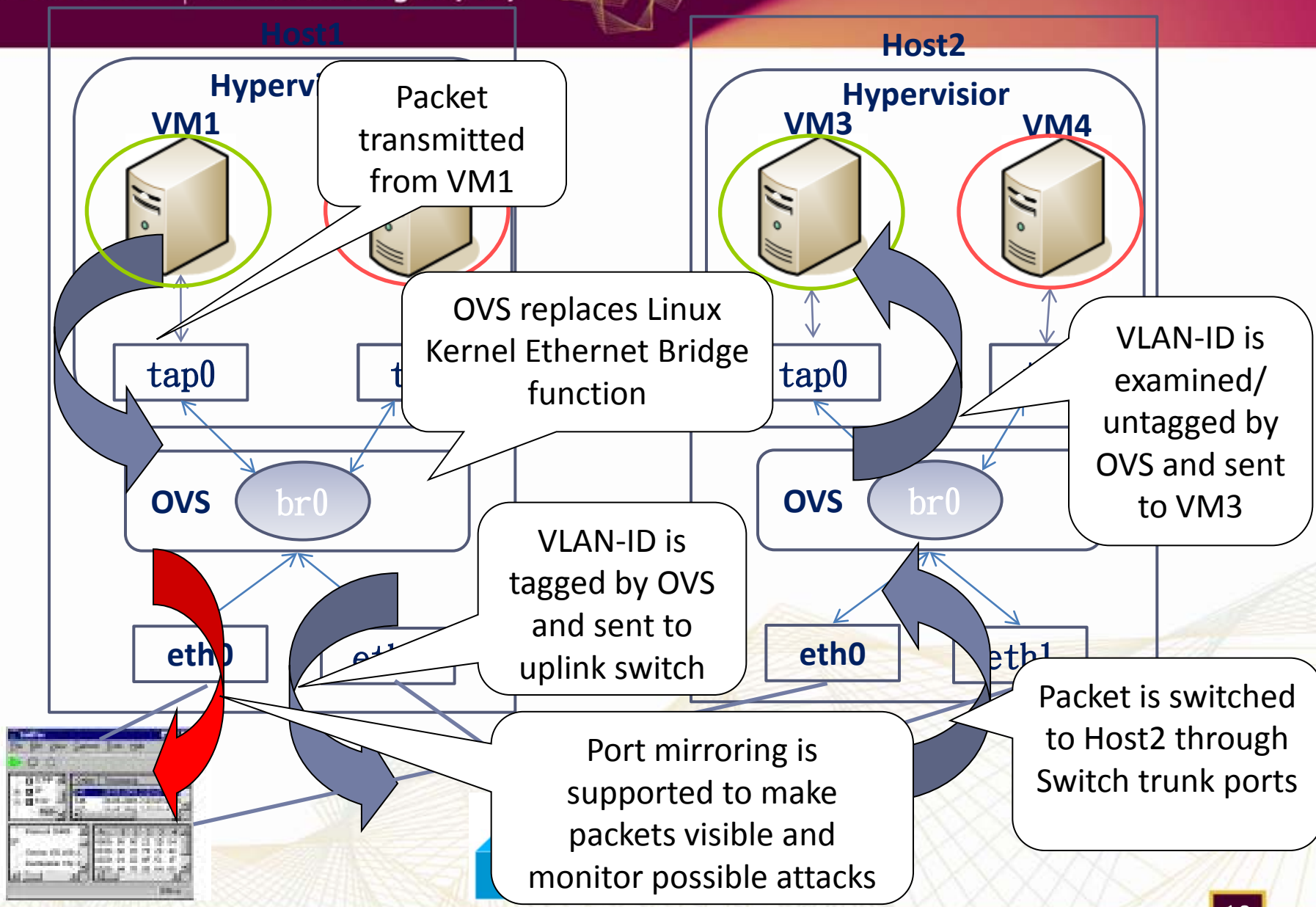
Open vSwitch

- Open source software that well suited to function as a virtual switch in VM environments
  - Visibility into inter-VM communication via NetFlow, sFlow, SPAN and RSPAN
  - Standard 802.1Q VLAN model with trunking
  - Kernel-based forwarding
  - Support for OpenFlow
  - Compatibility layer for the Linux bridging code

Architecture



Source: openvswitch.org


Host1
Hyperv
VM1
Packet transmitted from VM1
tap0
OVS
br0
eth0
Host2
Hypervisior
VM3
VM4
tap0
OVS
br0
eth0
OVS replaces Linux Kernel Ethernet Bridge function
VLAN-ID is examined/ untagged by OVS and sent to VM3
VLAN-ID is tagged by OVS and sent to uplink switch
Packet is switched to Host2 through Switch trunk ports
Port mirroring is supported to make packets visible and monitor possible attacks

Separation of Network Configuration

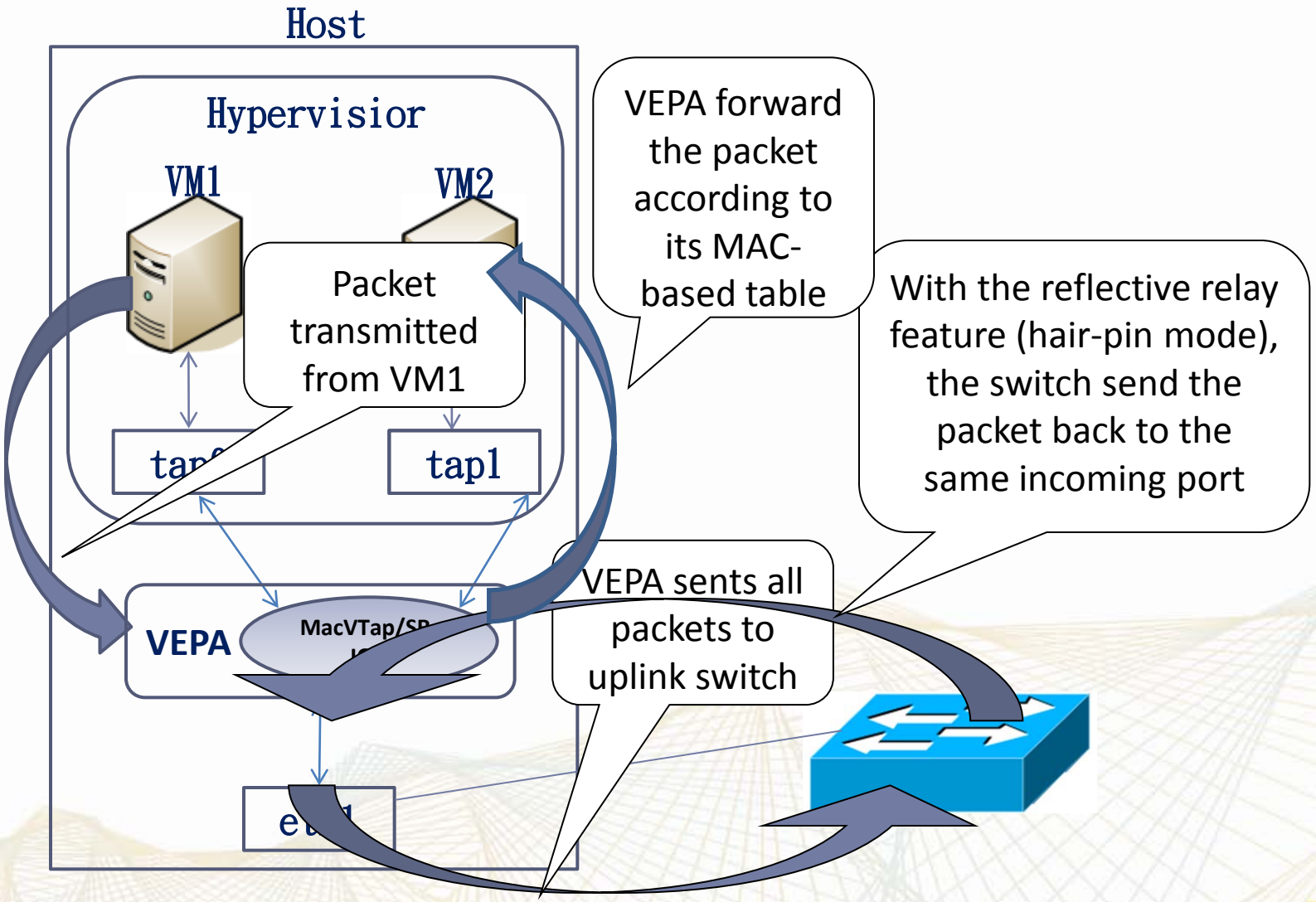
- Configurations of network is now divided into two parts
  - Physical network devices that managed by network team
  - Software virtual switches is configured by server team
- Possible inconsistence of network and server configurations may cause errors and is very hard to troubleshooting/maintenance.

Hardware Edge Virtual Bridging (EVB)

- Two ongoing IEEE standards are working on physical virtual switching environments.
- IEEE 802.1Qbg
  - VEPA (Virtual Ethernet Port Aggregation)
  - lead by HP (HP, IBM, Extreme, Brocade, Juniper ...)
- IEEE 802.1Qbh
  - Bridge Port Extension / VN-Tag
  - proposed by Cisco

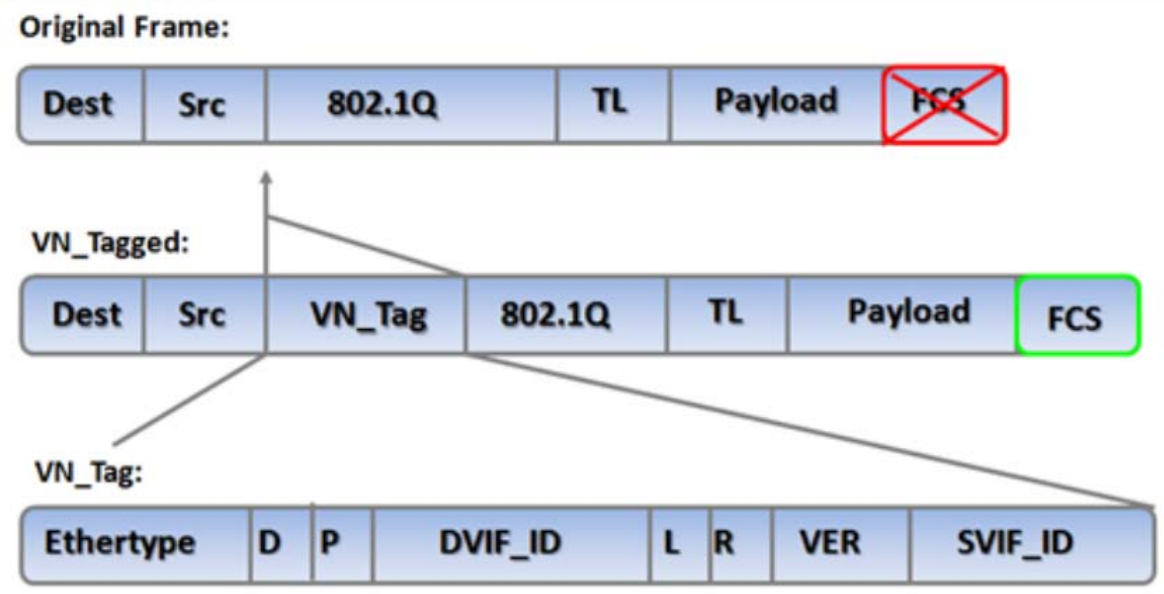
IEEE 802.1Qbg/ VEPA

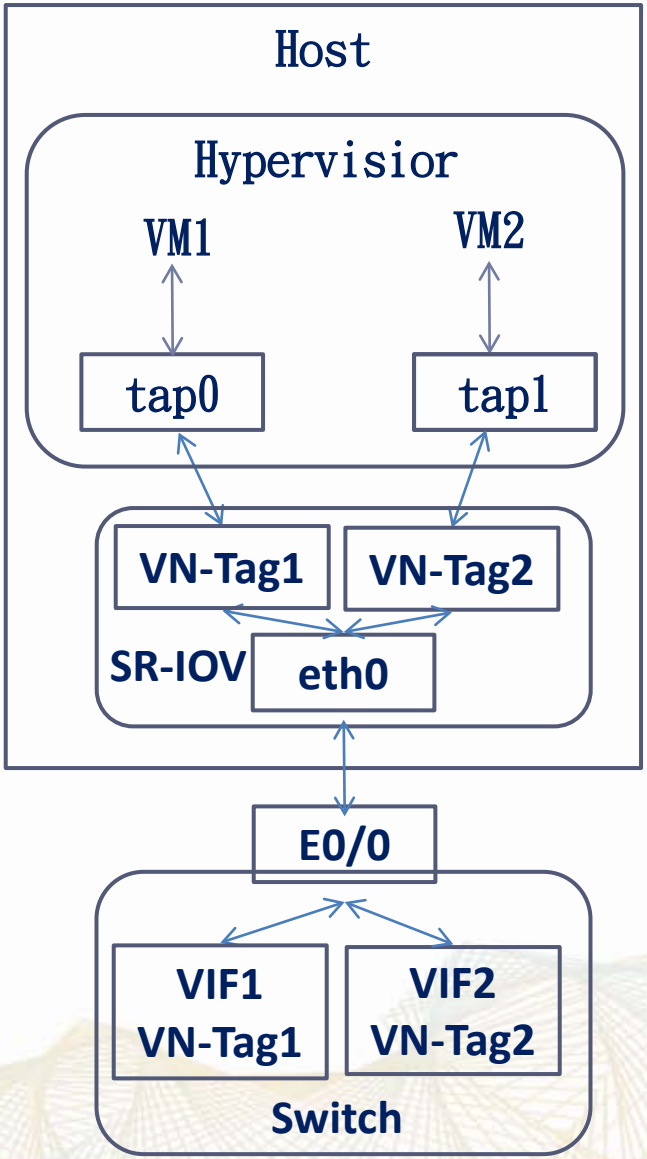
- Minor software update from VEB to VEPA is required in order to force packets transmitted to uplink switches.
  - SR-IOV NICs can also support VEPA with minor update.
- Switches firmware should also be upgraded to support reflective relay (hair-pin mode).
  - Leverage existing hardware
  - No changes to existing frame formats
- QoS, ACL, and monitoring functions remains the same at physical switches layer


Host
Hypervisior
VM1
VM2
Packet transmitted from VM1
tap1
VEPA
MacVTap/SR
VEPA forward the packet according to its MAC-based table
With the reflective relay feature (hair-pin mode), the switch send the packet back to the same incoming port
VEPA sents all packets to uplink switch

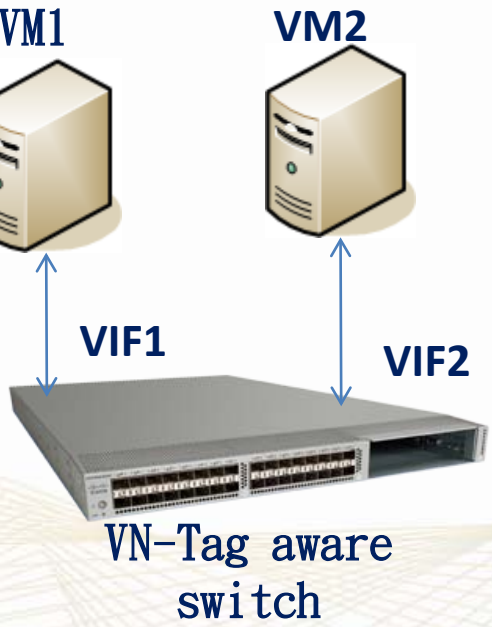
IEEE 802.1Qbh/Bridge Port Extension

- An additional header (VN-Tag) is added into standard Ethernet frame to identify virtual interfaces. (VIF)
- Each VIF can be separately configured as if it were a physical IF
- Switching/Forwarding inside switches only
- Requ ion




Host
Hypervisior
VM1
VM2
tap0
tap1
VN-Tag1
VN-Tag2
SR-IOV
eth0
E0/0
VIF1
VN-Tag1
VIF2
VN-Tag2
Switch

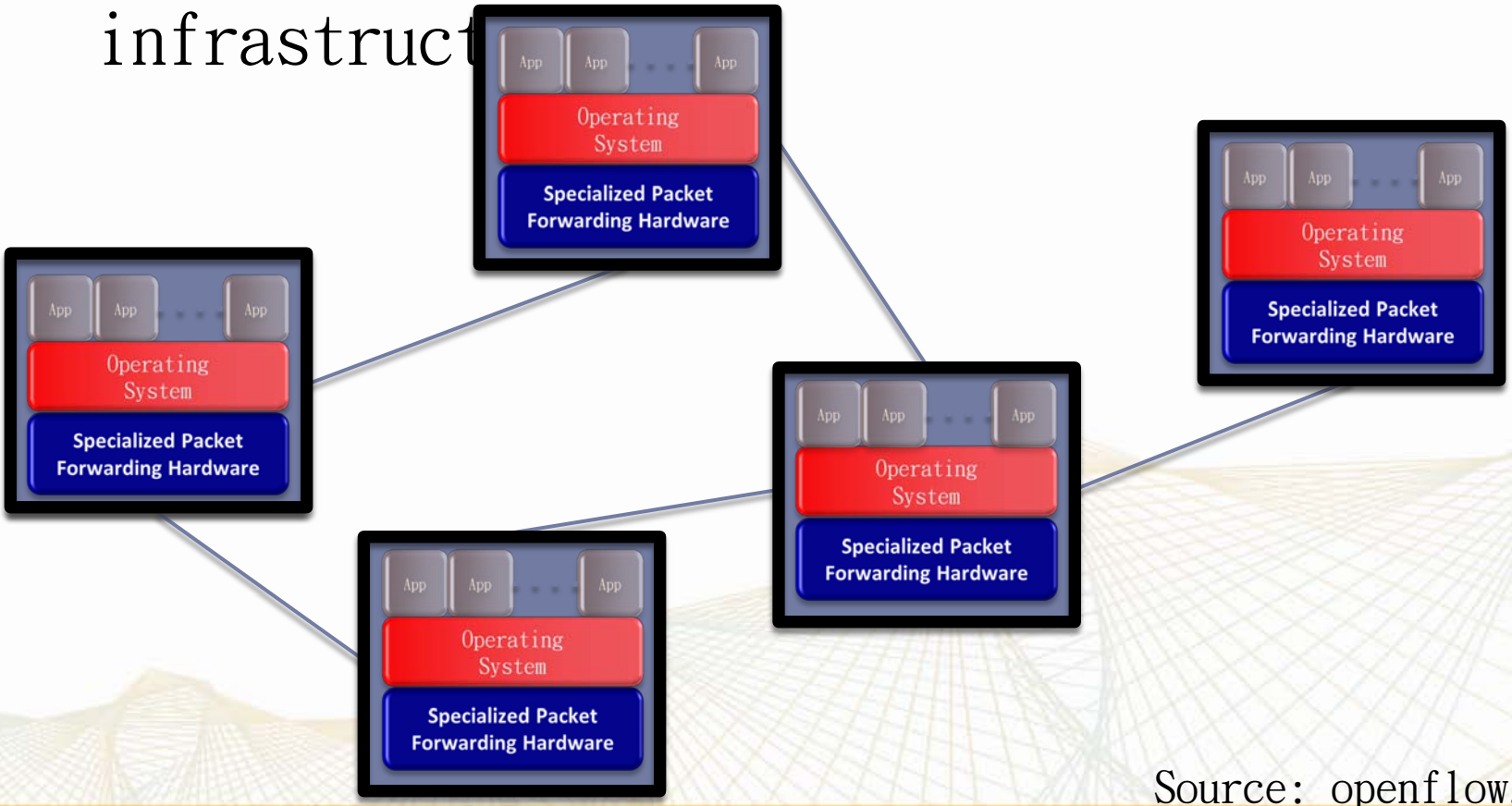
=


VM1
VM2
VIF1
VIF2
VN-Tag aware
switch

	VEB	EVB	
		802.1Qbg	802.1Qbh
Pros	•Host local switching •Software update •No external hardware	•Leverage existing HW •Traffic visibility •QoS and SLA control	•VIF represents as a physical interface – More scalable •No Switching and Forwarding required inside host
Cons	•More CPU/memory usage •Configuration maintenance •Traffic visibility	•Less scalable •Consumes host CPU to forward traffic	•Requires major HW/SW update

Current Internet

- Closed to innovations in the infrastruct



Source: openflow.org

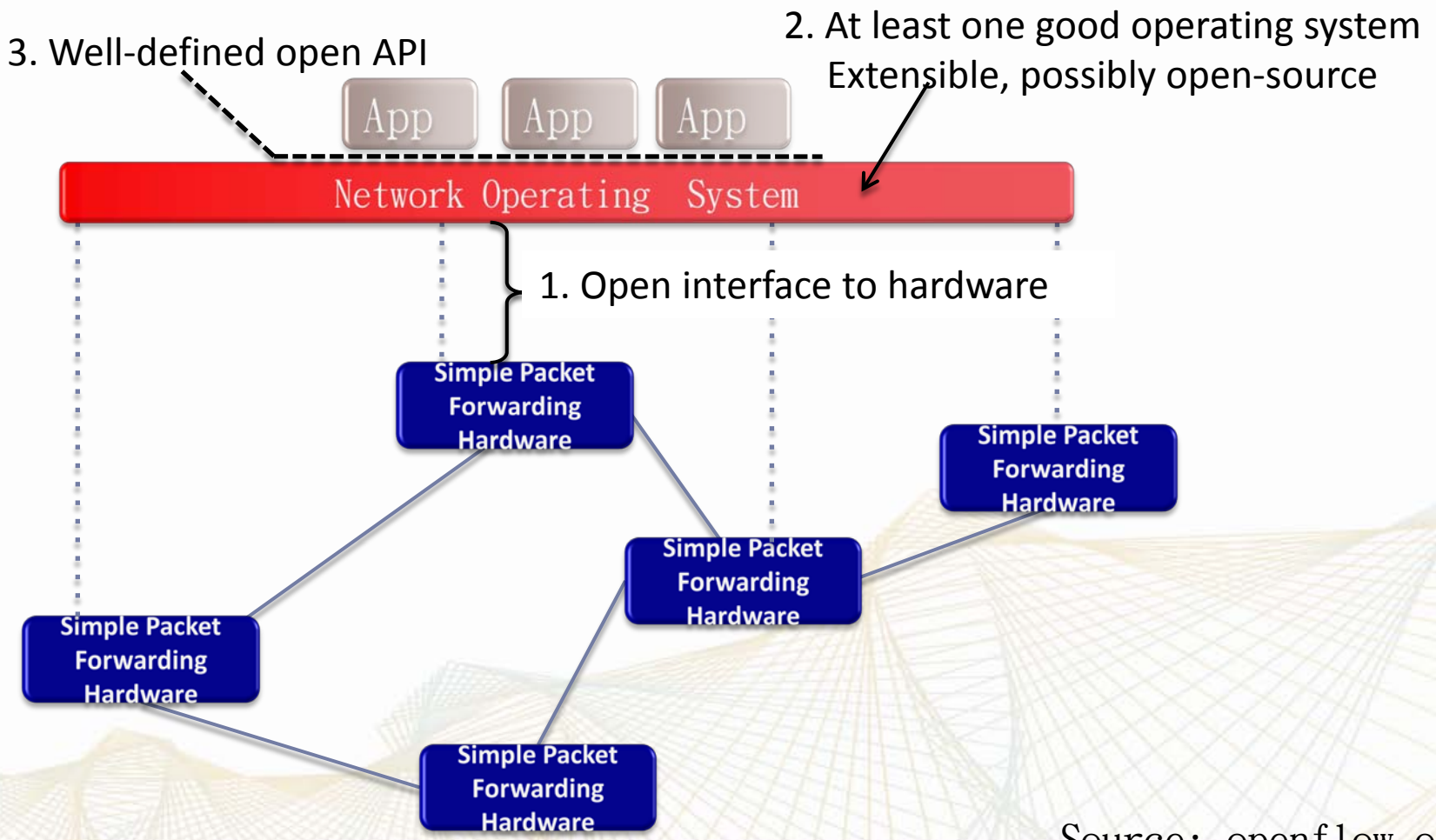
Future Internet

- To solve some limitations in current Internet
  - Scalability
  - Security
  - QoS
  - Virtualization
- Future Internet is a summarizing term for worldwide research activities dedicated to the further development of the original Internet. (From Wiki)

Future Internet Testbed

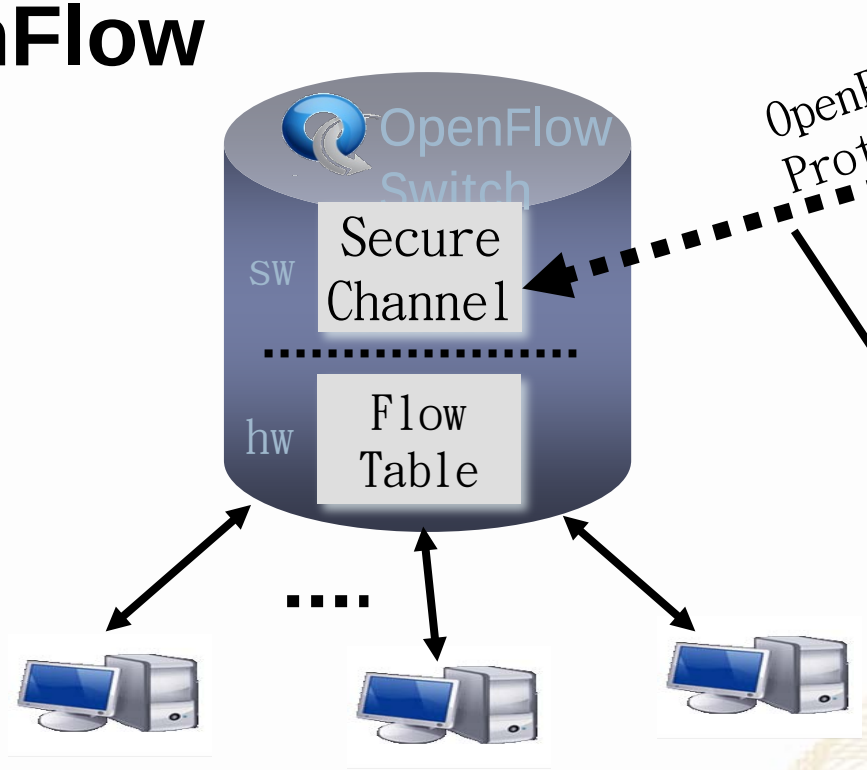
- For innovations and researches in Future Internet, the testbed requires some advanced concepts:
  - Programmability
  - Virtualization
  - End-to-end slice

New Concept



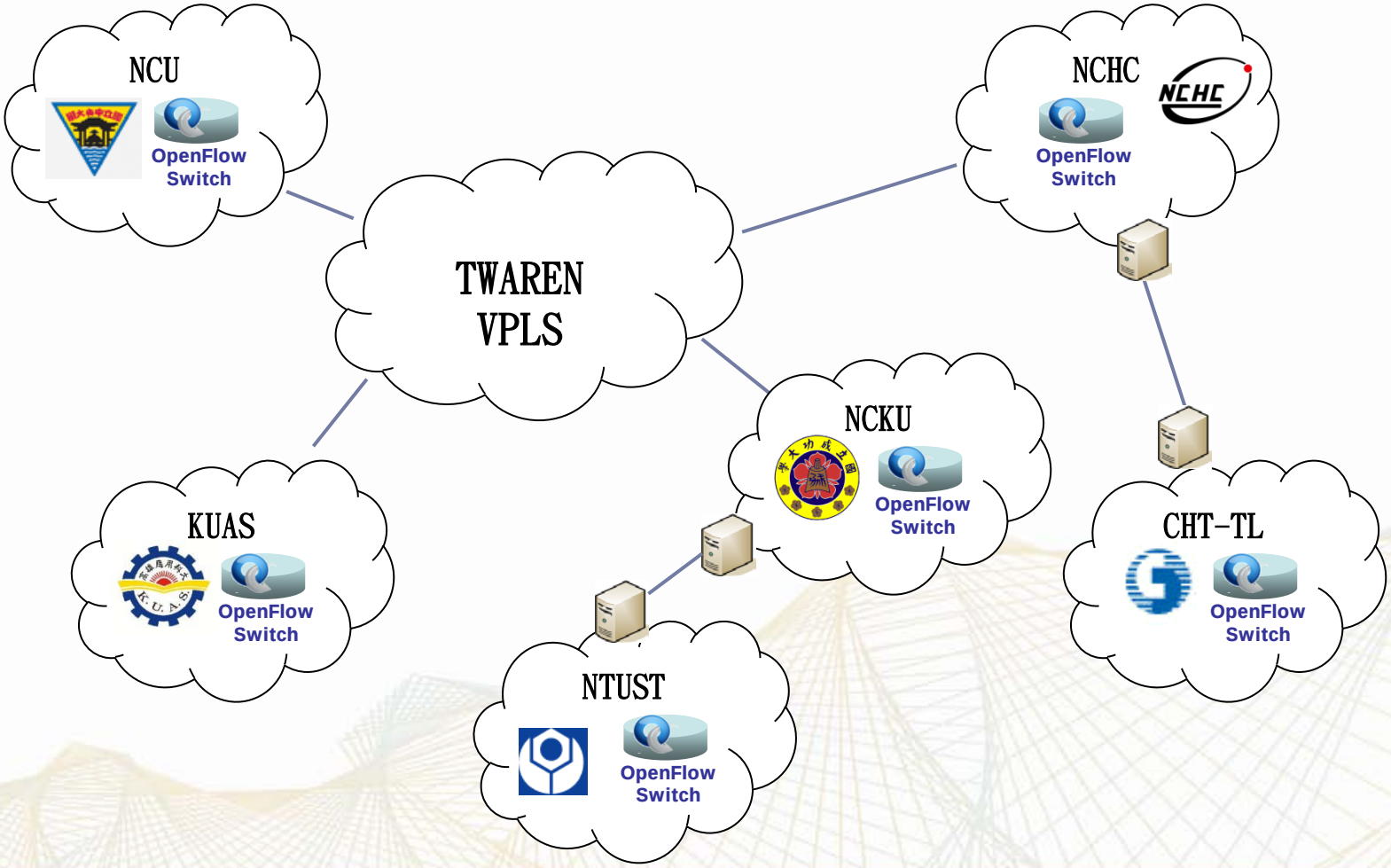
Source: openflow.org

OpenFlow

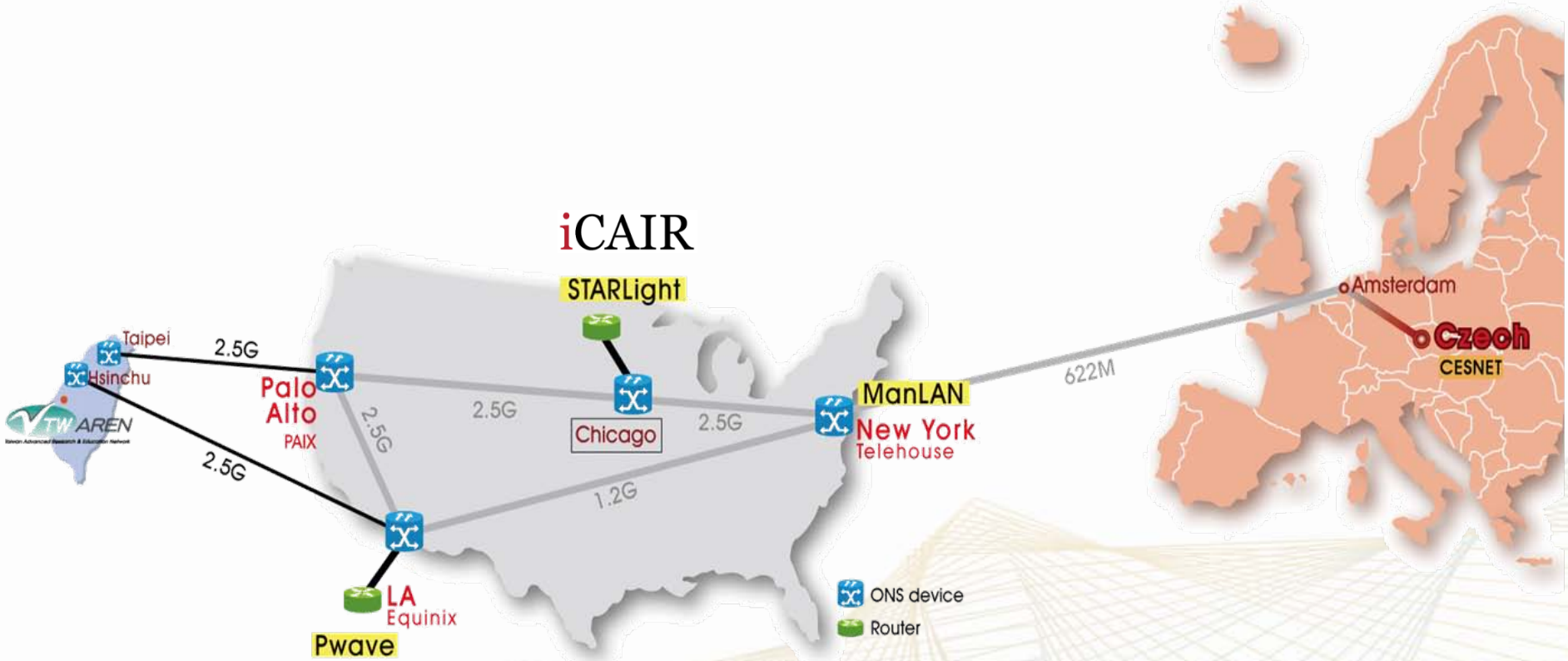


- Make deployed networks programmable
- Makes innovation easier
- Validate your experiments on production network at full line speed

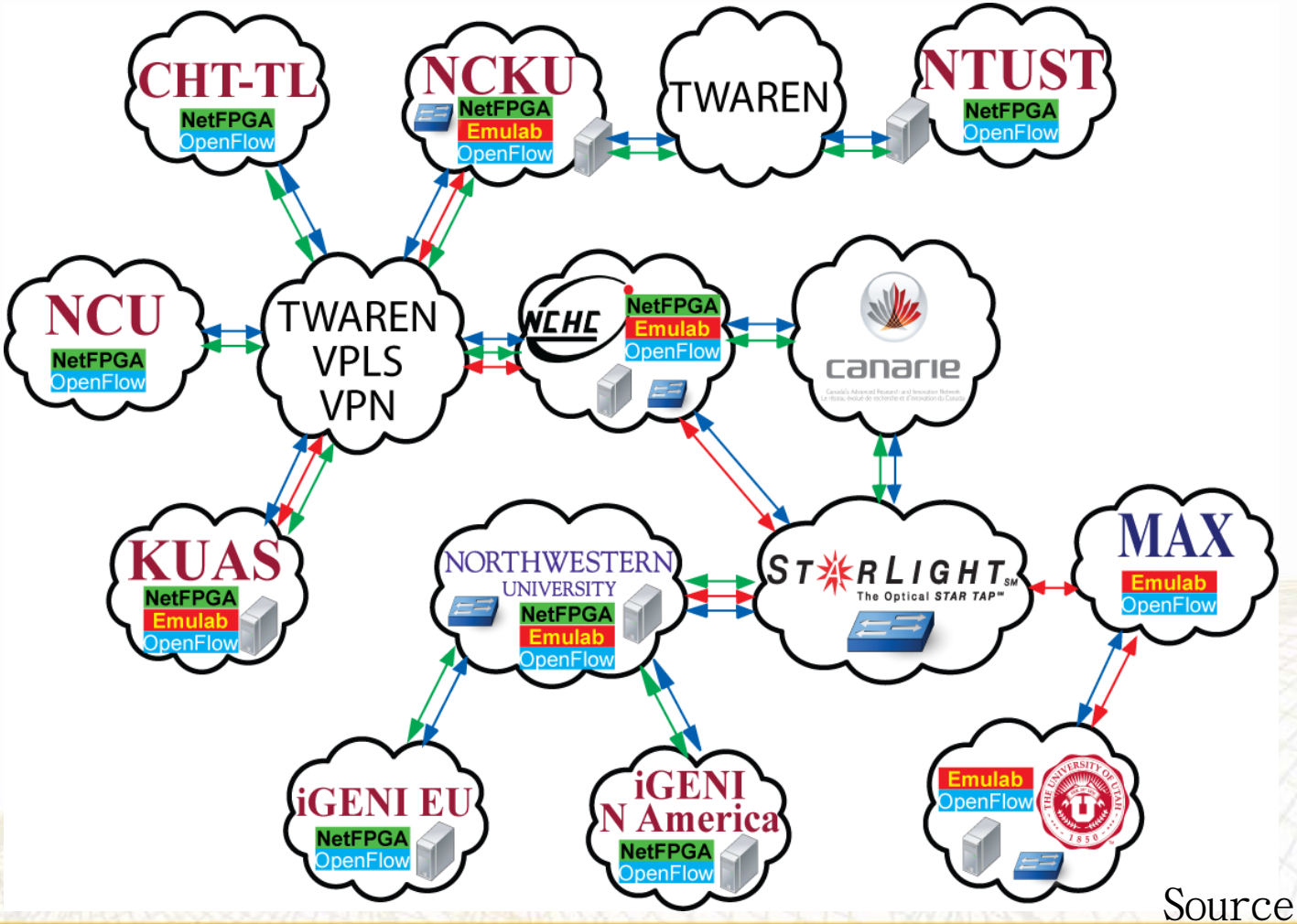
Current OpenFlow Testbed in TWAREN



TWAREN International Circuit



iGENI-Taiwan Integrated Research Network



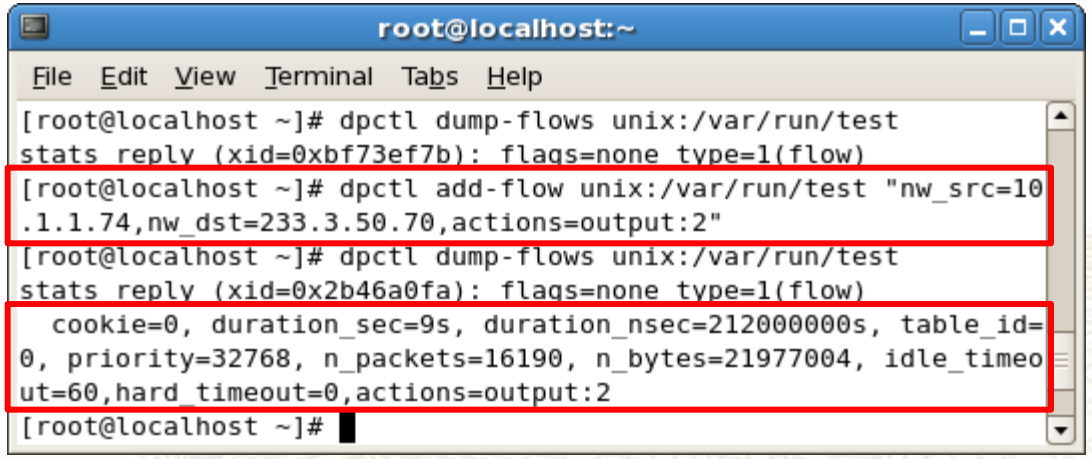
Source: iCAIR

Research 1 - IGMP in OpenFlow

- Video transferred over FI testbed is not as smooth as over legacy Internet.
  - There are mosaics appearing every second.

Research 1 – Proposed Solution

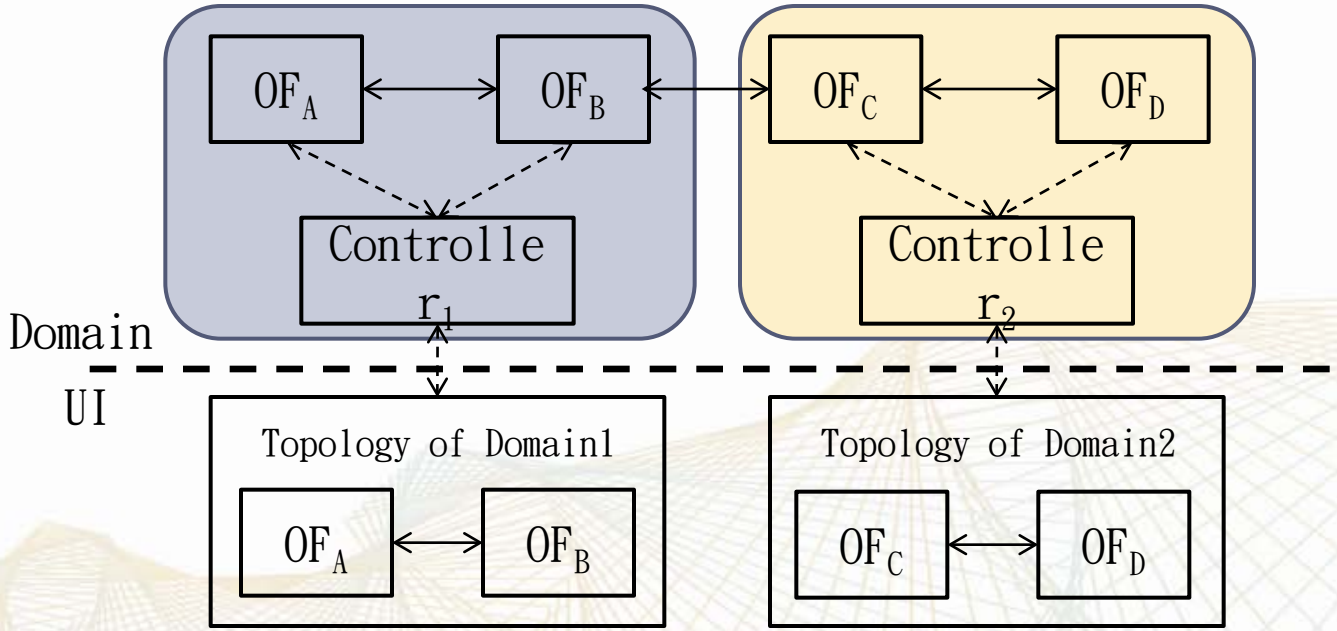
- Because IGMP is not supported in OpenFlow, we have to manually insert multicast streaming flows into the flow table.



Research 1 – Proposed Solution (cont.)

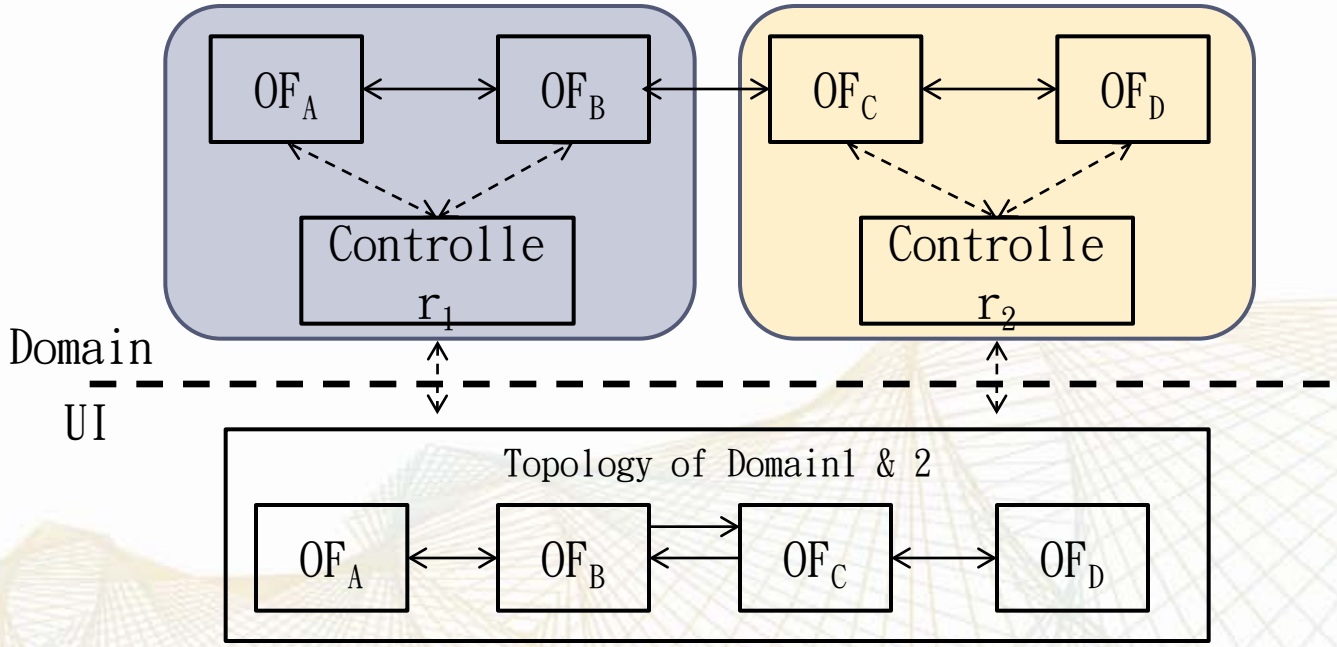
Research 2 – Inter-domain Connection

- OpenFlow Controller just only knows its directly connected switches.
- It will be inconvenient when the environment has more than one OpenFlow domain.

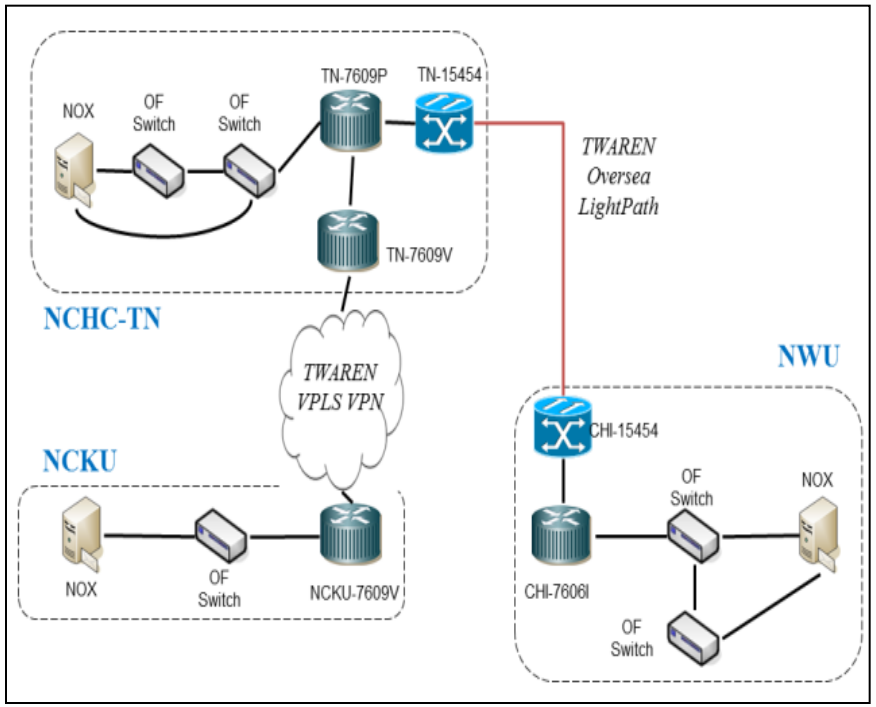


Research 2 – Proposed Solution

- We add additional contents in LLDP packet to let directly connected Controllers have its neighbors' topology.

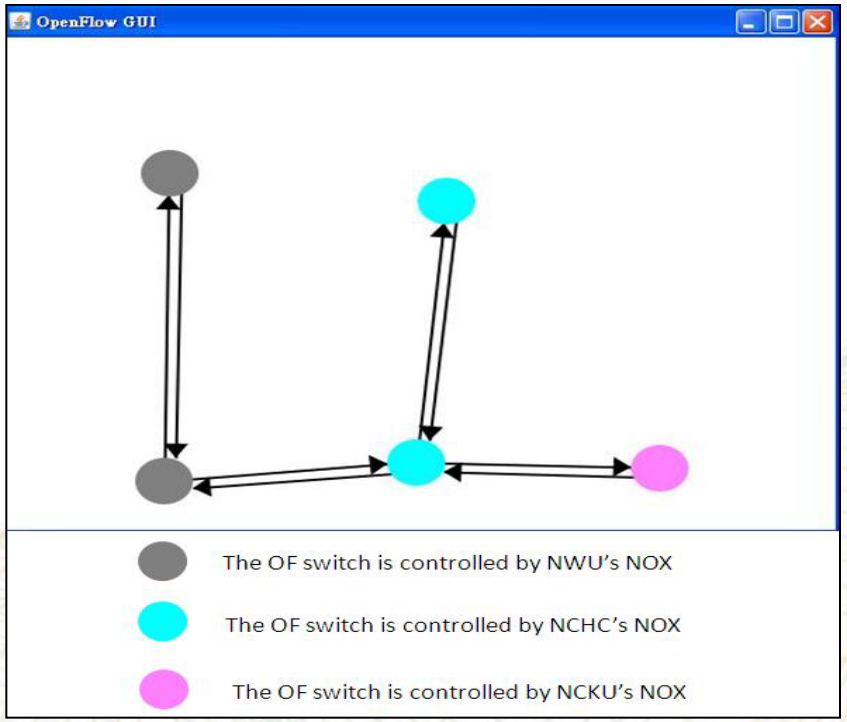


Research 2 – Proposed Solution (cont.)



Physical OpenFlow Network Topology

OpenFlow Network Topology on GUI



Conclusions

- Networking is an important part of Cloud.
- OpenFlow is an API, but it makes the network programmable and implements innovation easier.
- The combination of OpenFlow switches and virtual switches will be an interested develop/research area for control and management the next-generated network.

Thank you