XenSummit Asia

November 2-3, 2011

Seoul, Korea

Link Virtualization based on Xen

ShinHyoung Lee, Chuck Yoo

shlee@os.korea.ac.kr,
hxy@os.korea.ac.kr

Sponsored by:

SAMSUNG & LIBERTAS JUSTITIA VERITAS KOREA UNIVERSITY

Contents

introduction

The Future Internet

- Requirement
  - Various network protocols can be existed in the Future Internet
- Challenge
  - How to isolate different networks

Network Virtualization is a good solution

- Virtualization layer is an innovative substrate for Future Internet
  - allowing multiple virtual networks
  - isolating virtual networks

Virtual Network

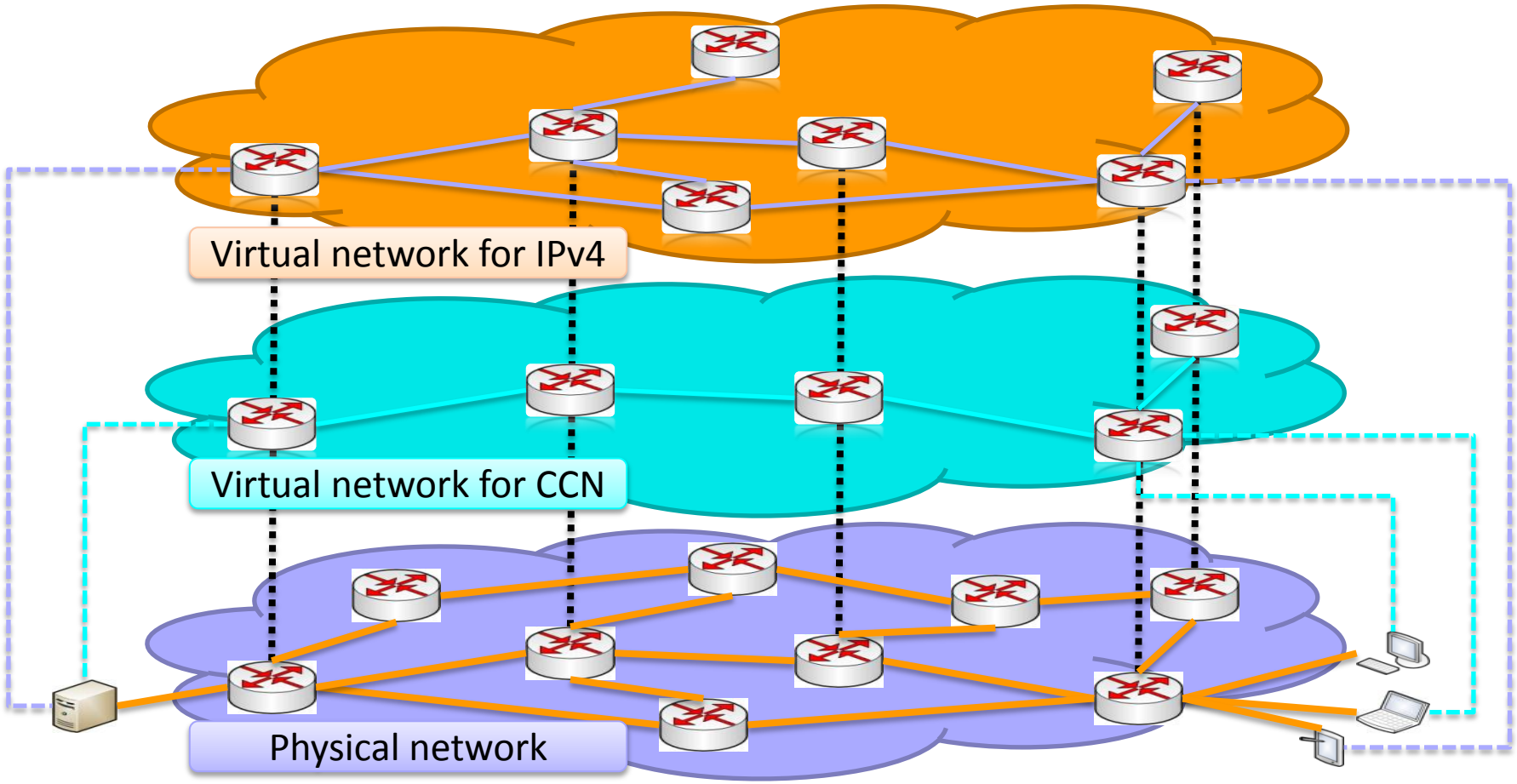
Node Virtualization

- Implemented by Router Virtualization
- e.g.) Xen

Link Virtualization

- Implemented by NIC Virtualization
- e.g.) Paravirtualization on Xen, SR-IOV

Example of Virtual Network



Performance network virtualization

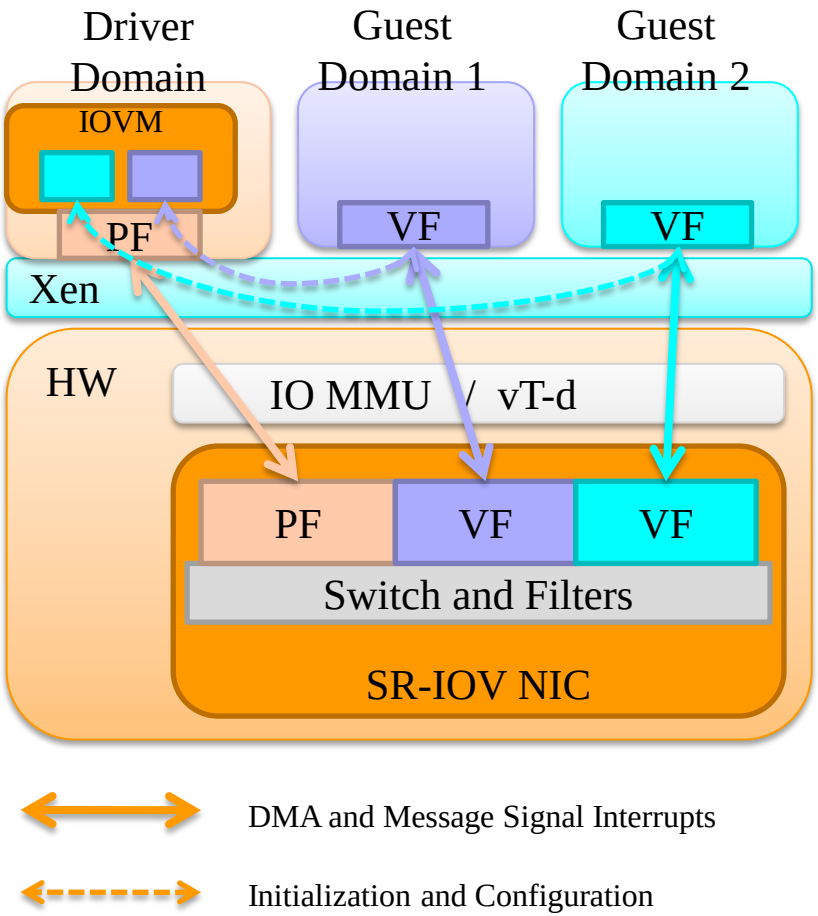
Cannot support over 10Gbps traffic

Some virtualization techniques are try to solve performance problem

- Paravirtualization with Xen
- SR-IOV with PCIe

We use both of them

SR-IOV

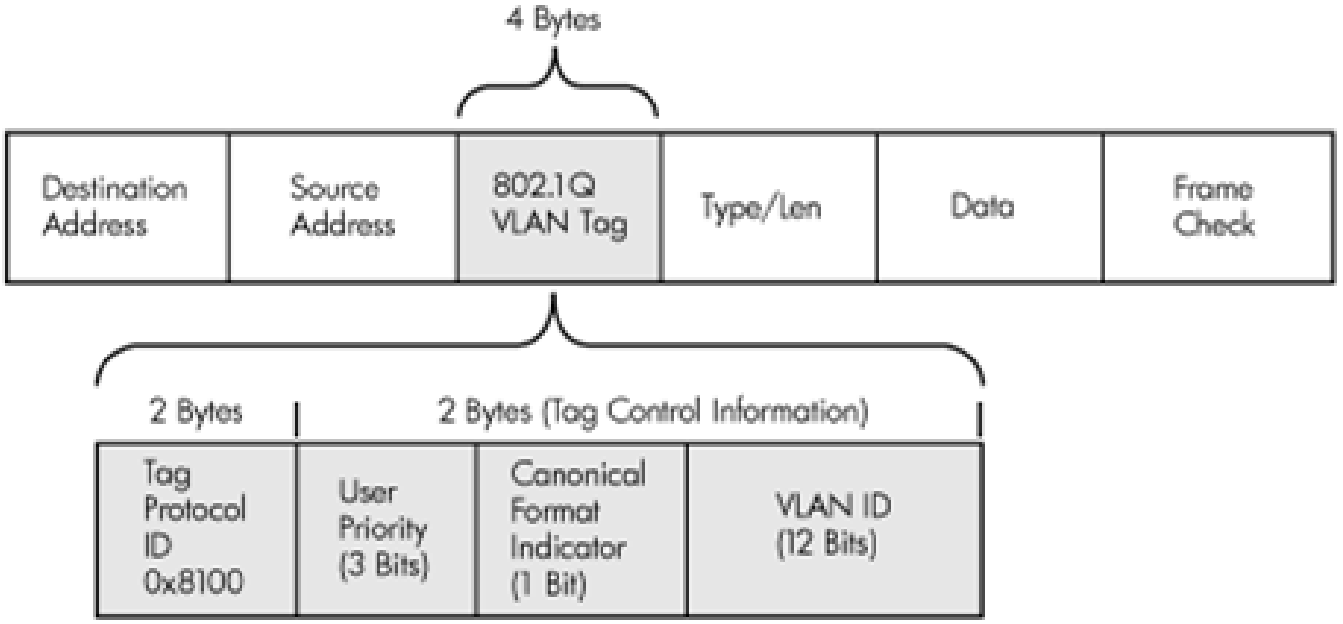


- SR-IOV minimize I/O virtualization overhead
- SR-IOV device has physical function (PF) and virtual function (VF)
- Physical function
  - Use all function of device
  - Initialize and configure device
- Virtual function
  - Communication directly with domain

Related Work

802.1q vlan

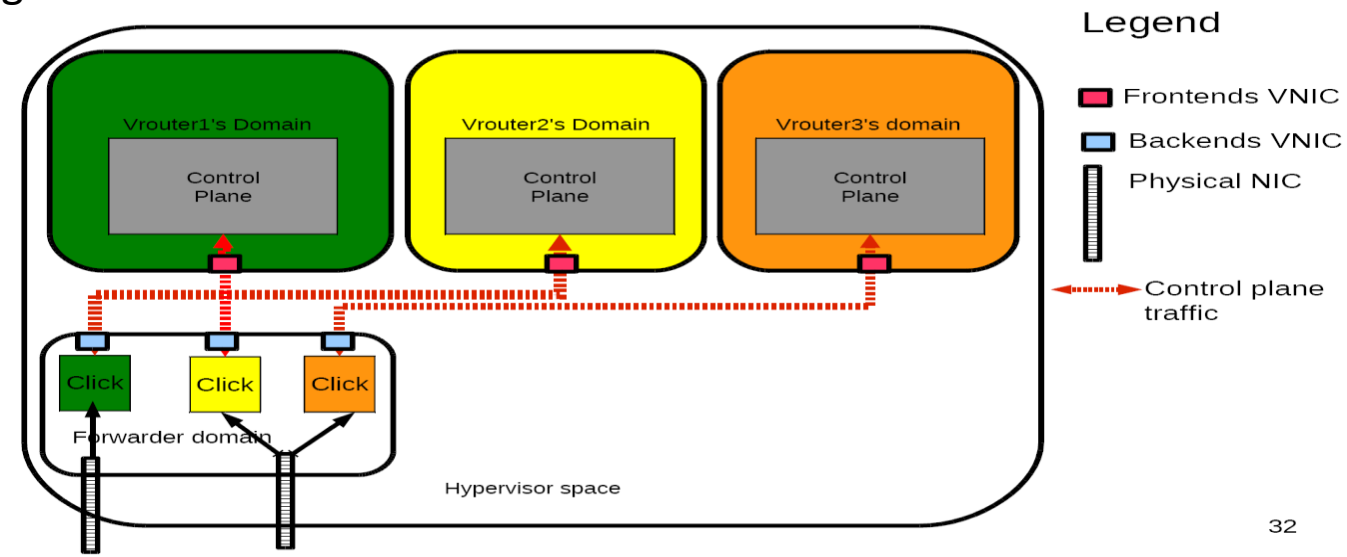
vlan tag in MAC header



Related Work (cont.)

Vrouter at Lancaster University

- Virtualized router based on Xen
- Control plane is placed in guest domain
- Data plane is placed in driver domain because of performance
- Cannot guarantee network isolation



Related Work (cont.)

- Trellis
  - Container based virtualization
  - Use Ethernet over GRE tunneling
  - Hard to guarantee network isolation
- Geni and FIRE
  - No definition of link virtualization
  - Support network and bandwidth isolation indirectly through virtual router resource control

Link Virtualization

- Connect between Virtual Nodes through virtualized NIC
- Network Isolation
  - Node that is member of a virtual network cannot see other virtual networks packets
- Bandwidth Isolation
  - Virtual link shared physical link's bandwidth
  - A virtual link cannot intrude other virtual links' bandwidth

Network Isolation

802.1q vlan

- vlan tag is placed in MAC header and it cannot deliver across the node that do not support 802.1q
- Every node must support virtual network

Tunneling

- Encapsulation/decapsulation overhead
- Not every node must support virtual network

We choose tunneling

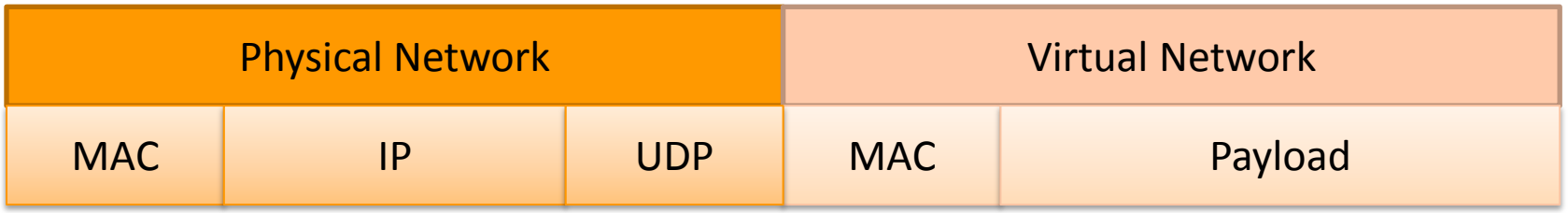
- Evolution of processing power
- It is impossible that every node support virtual network

MAC-in-UDP Tunneling

We use SR-IOV for performance

- SR-IOV NIC support 5-tuple filter
  - Source/destination IP address, source/destination TCP/UDP port number, and protocol
- We use UDP port number over 50K as virtual network id
- SR-IOV NIC can filter via virtual network id through hardware
  - Minimize filtering overhead

MAC-in-UDP tunneling header



Mac-in-UDP Tunneling (cont.)

- Encapsulation/decapsulation is done in guest domain
  - Driver domain do not process the packets
  - Avoid domain switch
  - Minimize performance overhead
- Guest domain must know all information for tunneling
  - Physical network information
    - MAC, IP and UDP header
  - Virtual network information
    - MAC header

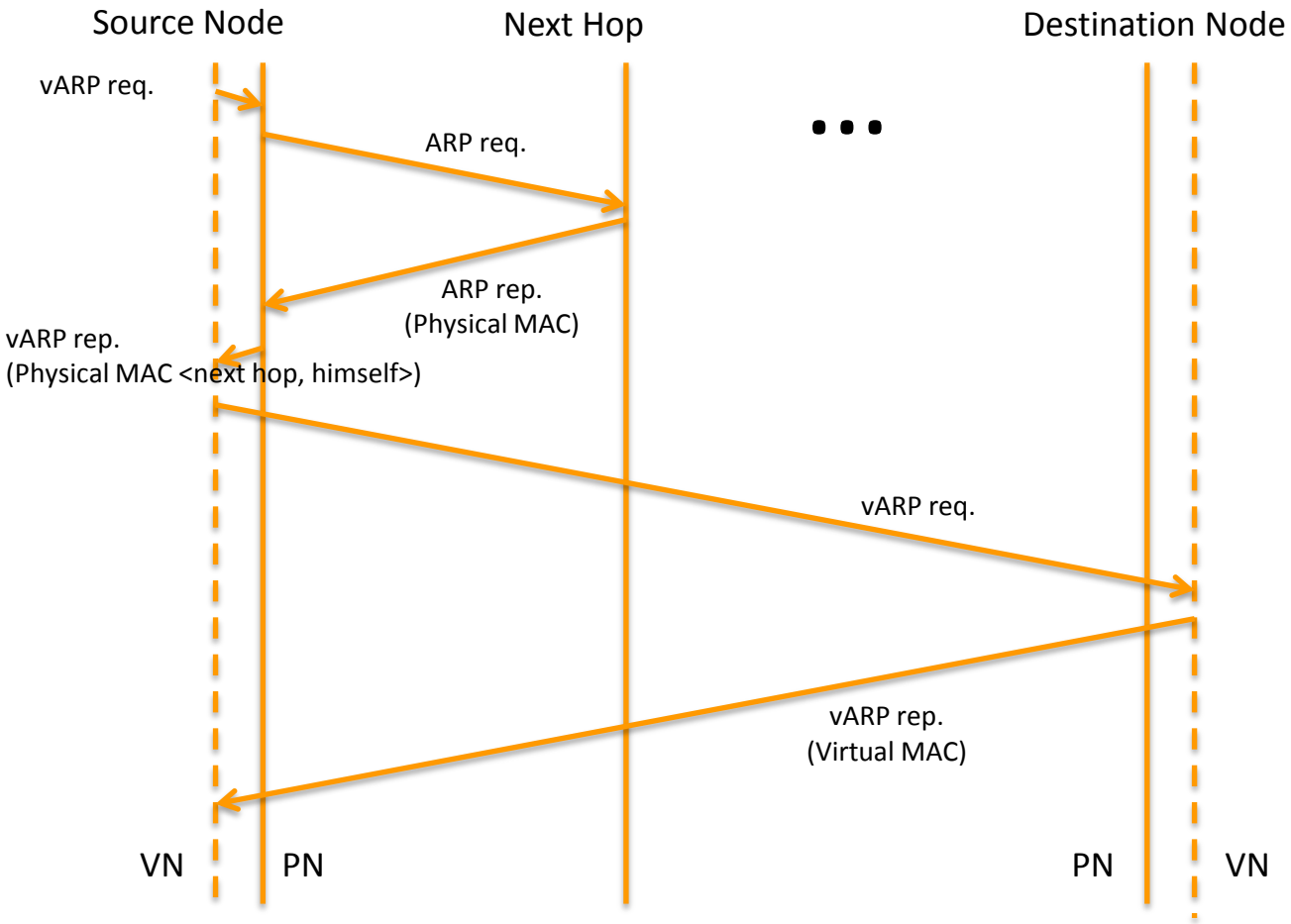
vARP

ARP – matching between MAC address and IP address

vARP – matching between virtual network MAC address and physical network IP address

- User input only physical IP address of other side node on virtual link
- vARP protocol gets physical source/destination MAC address, virtual MAC address automatically

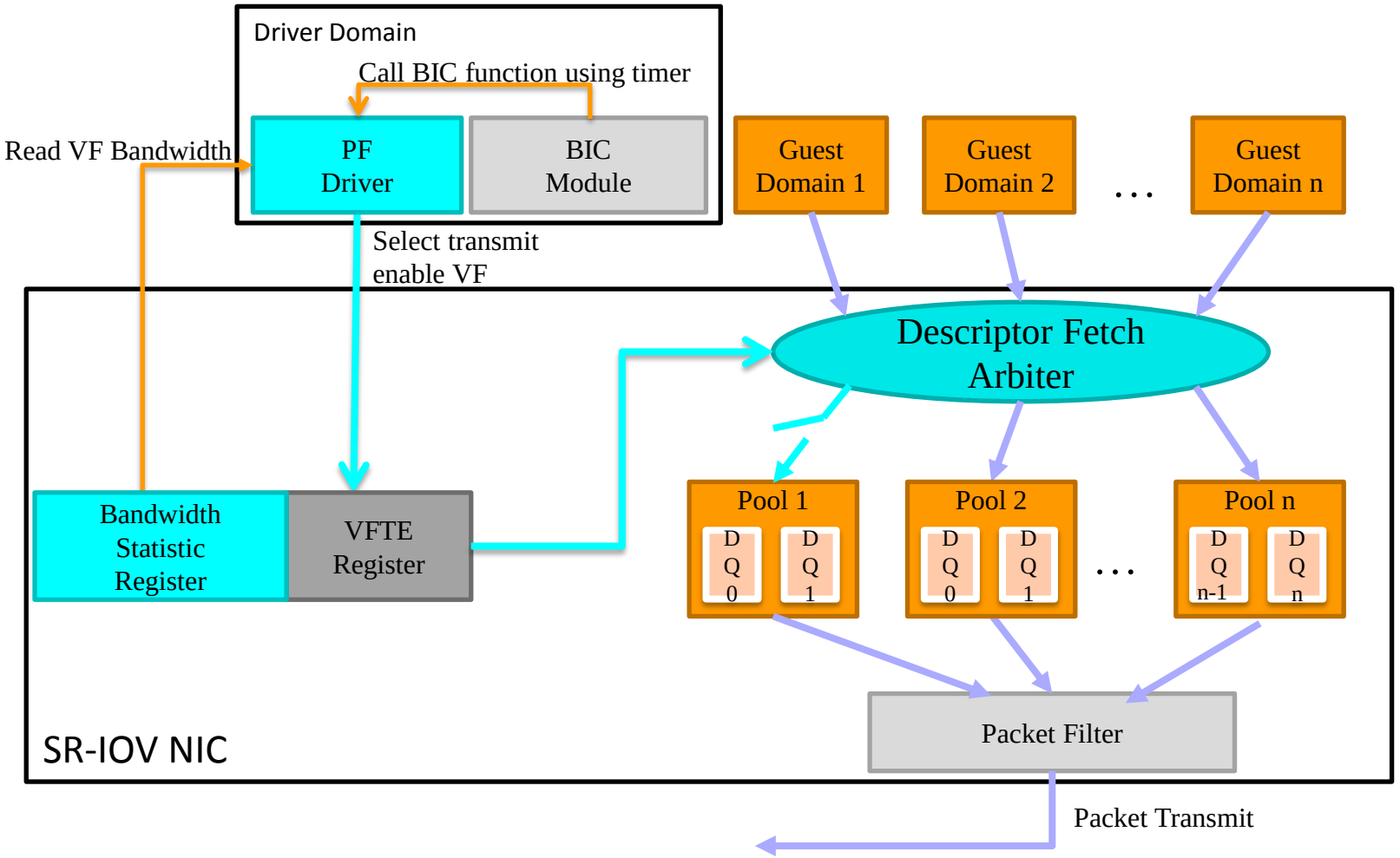
vARP flow


Source Node
Next Hop
Destination Node
vARP req.
ARP req.
...
ARP rep.
(Physical MAC)
vARP rep.
(Physical MAC <next hop, himself>)
vARP req.
vARP rep.
(Virtual MAC)
VN
PN
PN
VN

Bandwidth Isolation

- It is possible that many virtual networks share one physical network
  - Many virtual networks share physical bandwidth
  - Total bandwidth has upper cap by physical link
- Driver domain control guest domains sending rate
  - the received packets have already used the network resource
  - the transmission rate is the same as the receive rate of the next node

Bandwidth Isolation Structure



Policy of sharing physical bandwidth

Weight Based Control

Bandwidth Based Control with priority

Weight Based Control

Each virtual machines assign weight

Virtual machines are send as assigned weight

Example (Total 1Gbps)

	VM1	VM2	VM3
Weight	1	2	3
Used BW	166Mbps =1/6Gbps	333Mbps =2/6Gbps	500Mps =3/6Gbps

Bandwidth Based Control with priority

Each virtual machines assign bandwidth

Virtual machines cannot send more data than assigned bandwidth

Priority is used when summary of virtual networks bandwidth is more than physical network

To avoid disconnect virtual link that have low priority, we guarantee minimum bandwidth

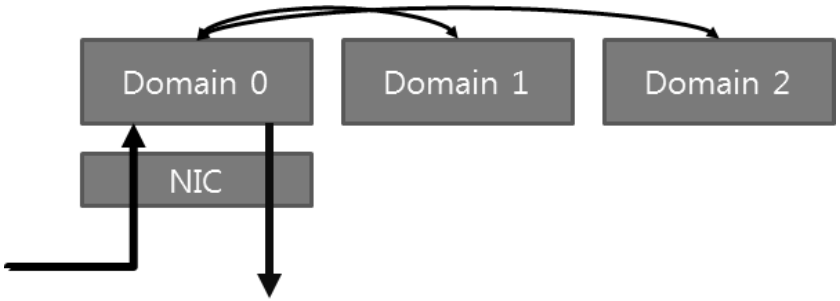
Performance Evaluation

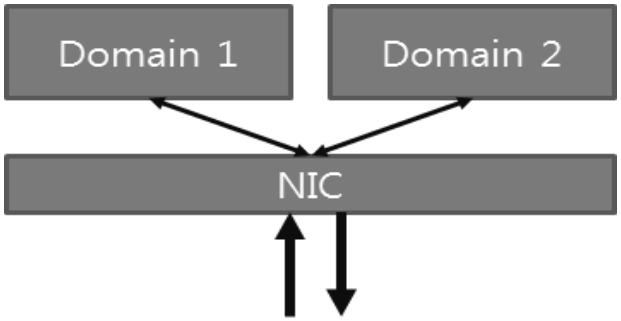
- Physical machine environment
  - Intel XEON X5650 (2.67GHz, 6-cores) * 2
  - 12GB physical memory
  - Intel 82576 NIC (1Gbps with SR-IOV support)
- Software environment
  - Xen 4.0
  - Guest OS Ubuntu 10.04 LTS with Paravirtualization (Kernel ver 2.6.37.1)
  - 4 cores VCPU
  - 2GB memory
  - NIC
    - Igbvf (Intel 82576 VF) 1.1.3 for SR-IOV
    - E1000 (Xen PV NIC model)

Tunneling Overhead

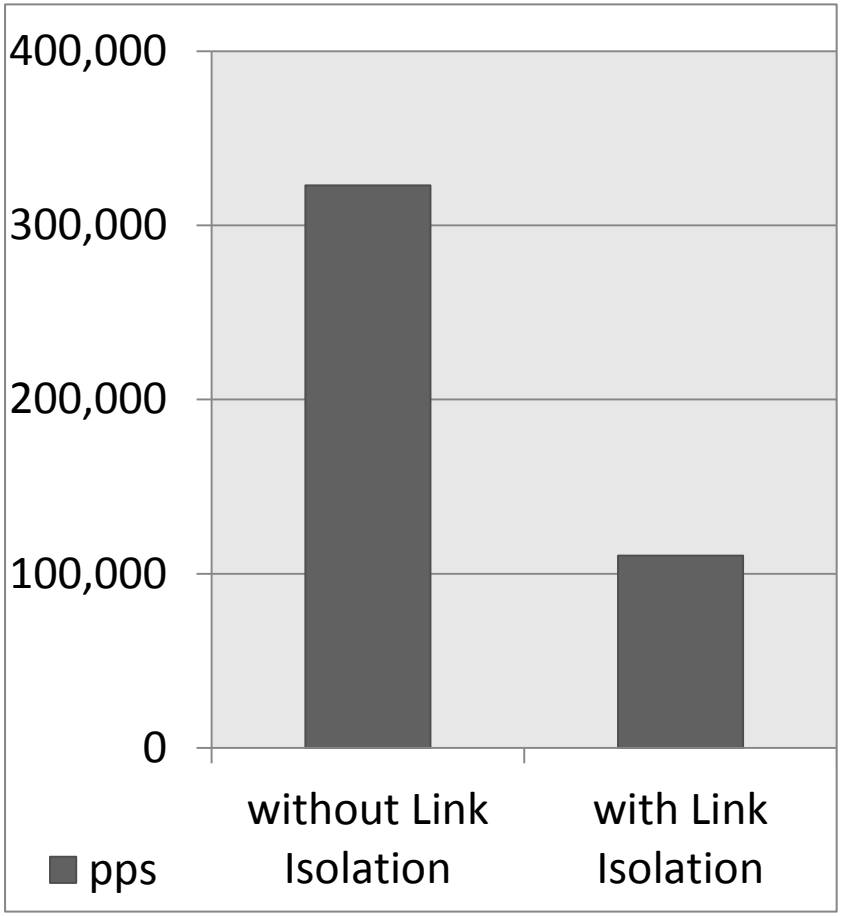
We compare PV model and SR-IOV model

Flow of packets are different

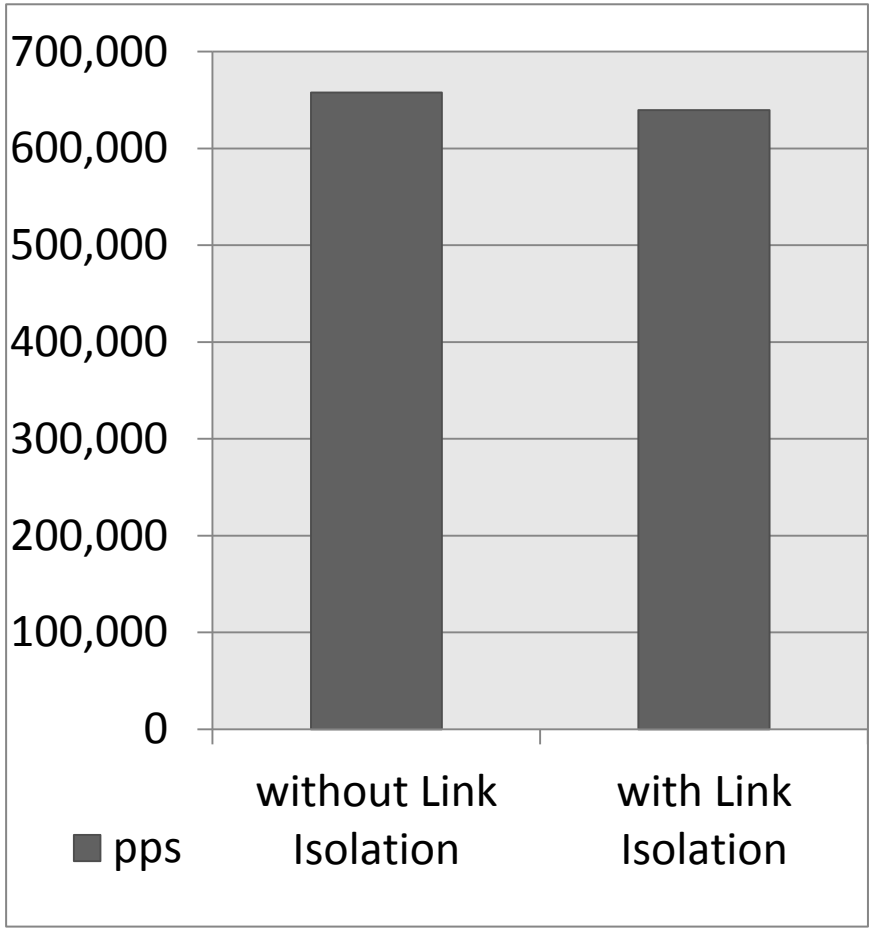




Tunneling Overhead



Performance of E1000 (PV)



Performance of Intel 82576 (SR-IOV)

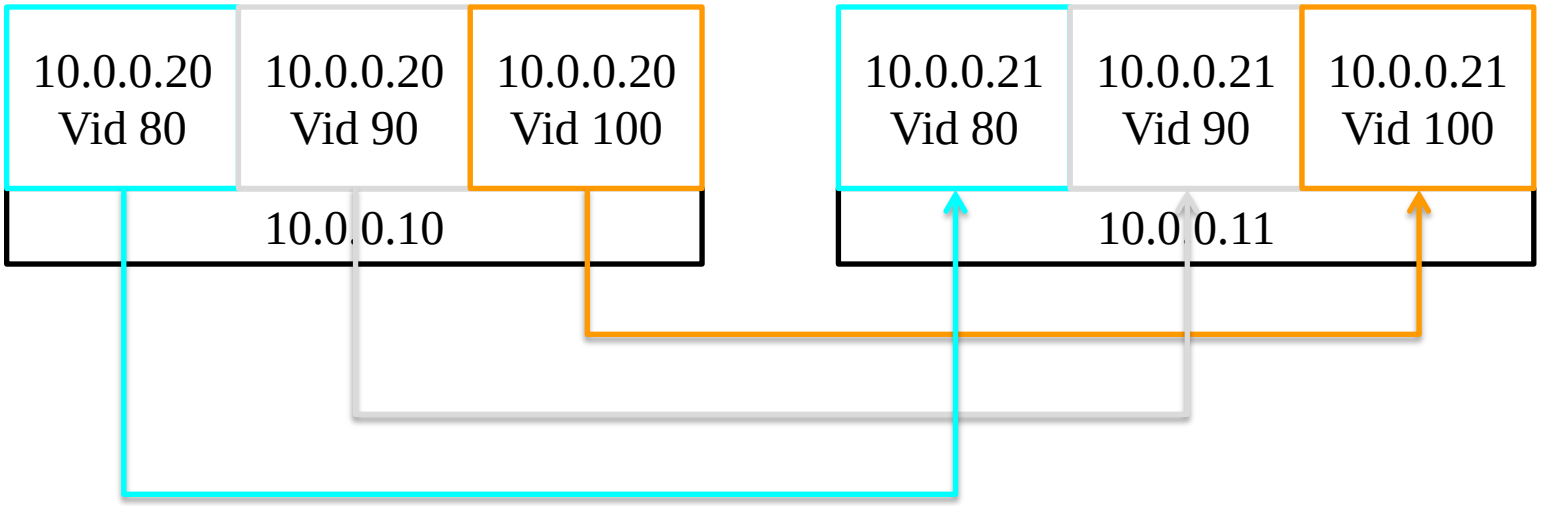
Weight Based Control Performance

Weight	BW (Mbps)				Ratio
VM1:VM2:VM3	VM1	VM2	VM3	total	VM1:VM2:VM3
NA	314	314	314	942	
1:1:1	314	314	314	942	1:1:1
1:1:8	94	94	752	940	1:1:8
1:2:3	157	314	471	942	1:2:3
1:2:4	135	268	538	941	1:1.99:3.99
1:3:3	135	403	403	941	1:2.99:2.99
1:3:6	94	283	565	942	1:3.01:6.01
1:4:5	95	377	471	943	1:3.97:4.96
2:2:3	269	269	403	941	2:2:3
2:3:4	209	314	418	941	2:3:4
3:3:4	283	283	376	942	3:3:3.99

Bandwidth Based Control Performance

Assigned Bandwidth	Priority	Measured Bandwidth (Mbps)			
VM1:VM2:VM3	VM1:VM2:VM3	VM1	VM2	VM3	total
300:200:100	1:2:3	286	191	95.4	572.4
600:400:200	1:2:3	571	273	95.3	939.3
600:400:200	3:2:1	366	381	191	938
1200:600:300	1:2:3	751	95.4	95.4	941.8

Link Virtualization Performance



We use 2 physical machines

Each physical machine has 3 guest domains (virtual networks)

Link Virtualization with Weight Based Control Performance

Weight	BW (Mbps)				Ratio
VM1:VM2:VM3	VM1	VM2	VM3	total	VM1:VM2:VM3
1:1:1	305	305	305	915	1:1:1
1:1:2	228	229	457	914	1:1:2
1:1:8	92	92	730	914	1:1:7.93
1:2:3	152	305	457	914	1:2.01:3.01
1:2:4	130	261	522	913	1:2.01:4.02
1:3:3	131	392	392	915	1:2.99:2.99
1:3:6	92	274	548	914	1:2.98:5.96
1:4:5	92	366	457	915	1:3.97:4.97
2:2:3	261	261	392	914	2:2:3
2:3:4	203	305	406	914	2:3:4
3:3:4	274	274	366	914	3:3:4.01

Link Virtualization with Bandwidth Based Control Performance

Bandwidth	Priority	BW (Mbps)			
VM1:VM2:VM3	VM1:VM2:VM3	VM1	VM2	VM3	Total
300:200:100	1:2:3	278	185	92.7	555.7
600:400:200	1:2:3	556	264	92.8	912.8
600:400:200	3:2:1	356	371	185	912
1200:600:300	1:2:3	728	92.8	92.8	913.6

Conclusion

Network virtualization is the core technology of the future Internet

Link virtualization is necessary for network virtualization

We propose and implement link virtualization on Xen with SR-IOV

We minimize virtualize overhead through Xen and SR-IOV

THANK YOU