Chair of Communication Networks
Department of Electrical and Computer Engineering
Technical University of Munich

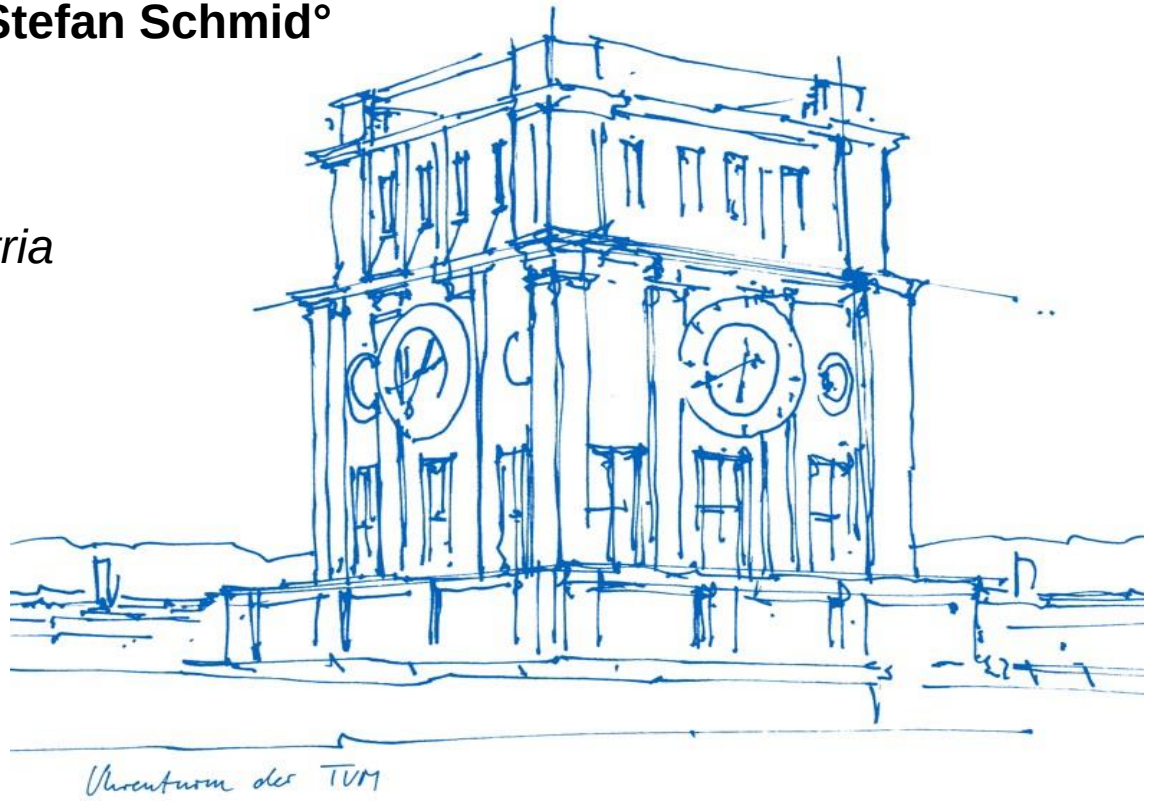
On The Impact of the Network Hypervisor on Virtual Network Performance

Andreas Blenk*, Arsany Basta*, Wolfgang Kellerer*, Stefan Schmid°

**Technical University of Munich, Germany*

°Faculty of Computer Science, University of Vienna, Austria

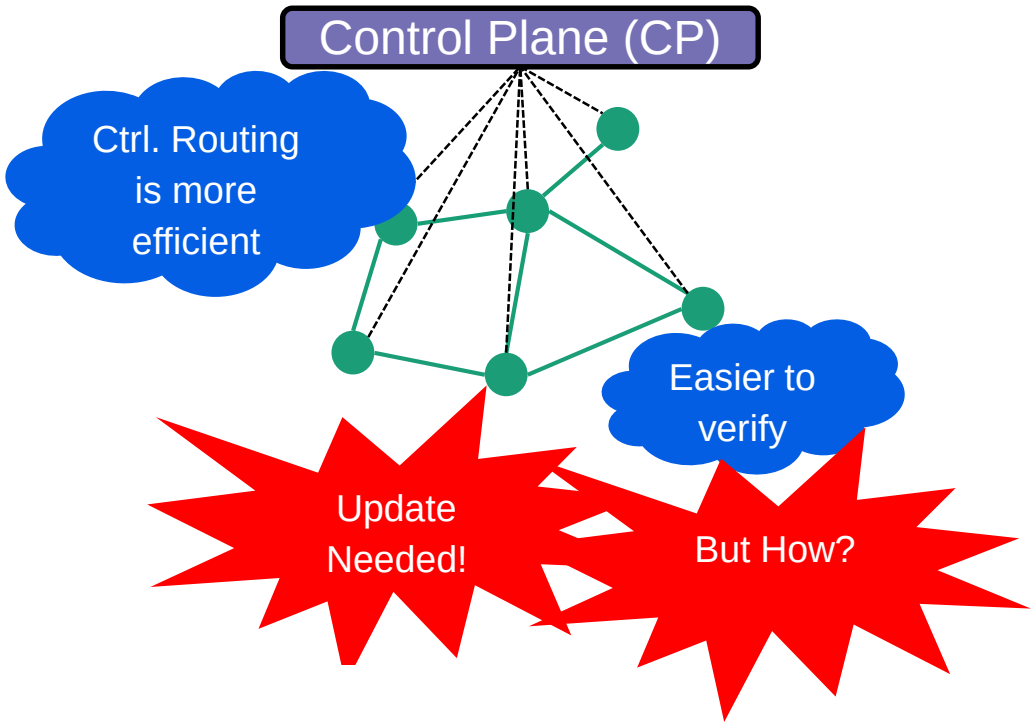
IFIP Networking 2019



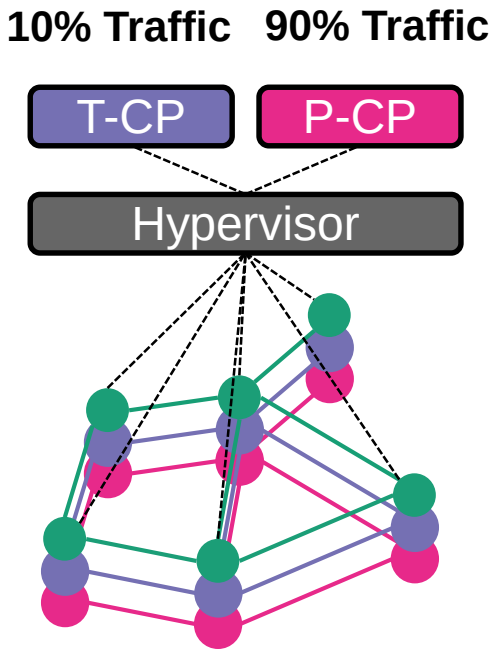
Introducing New Networking Solutions

Canary testing as used by Google [Govindant et al.]

SDN-based Solution

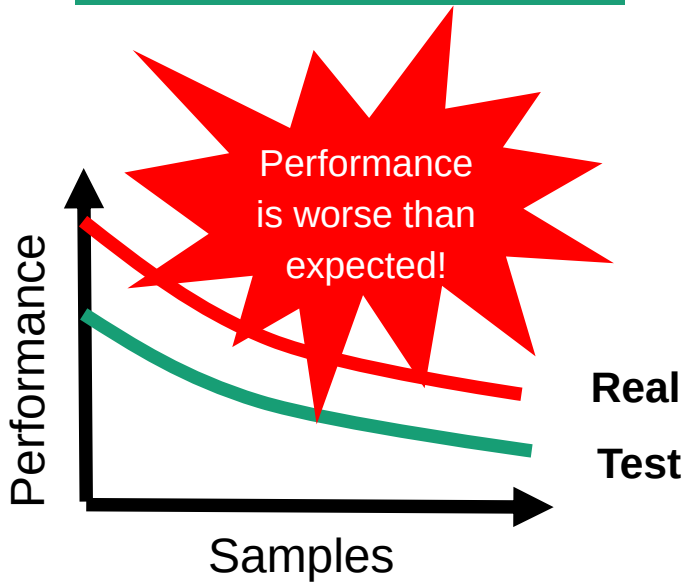


Solution Testing



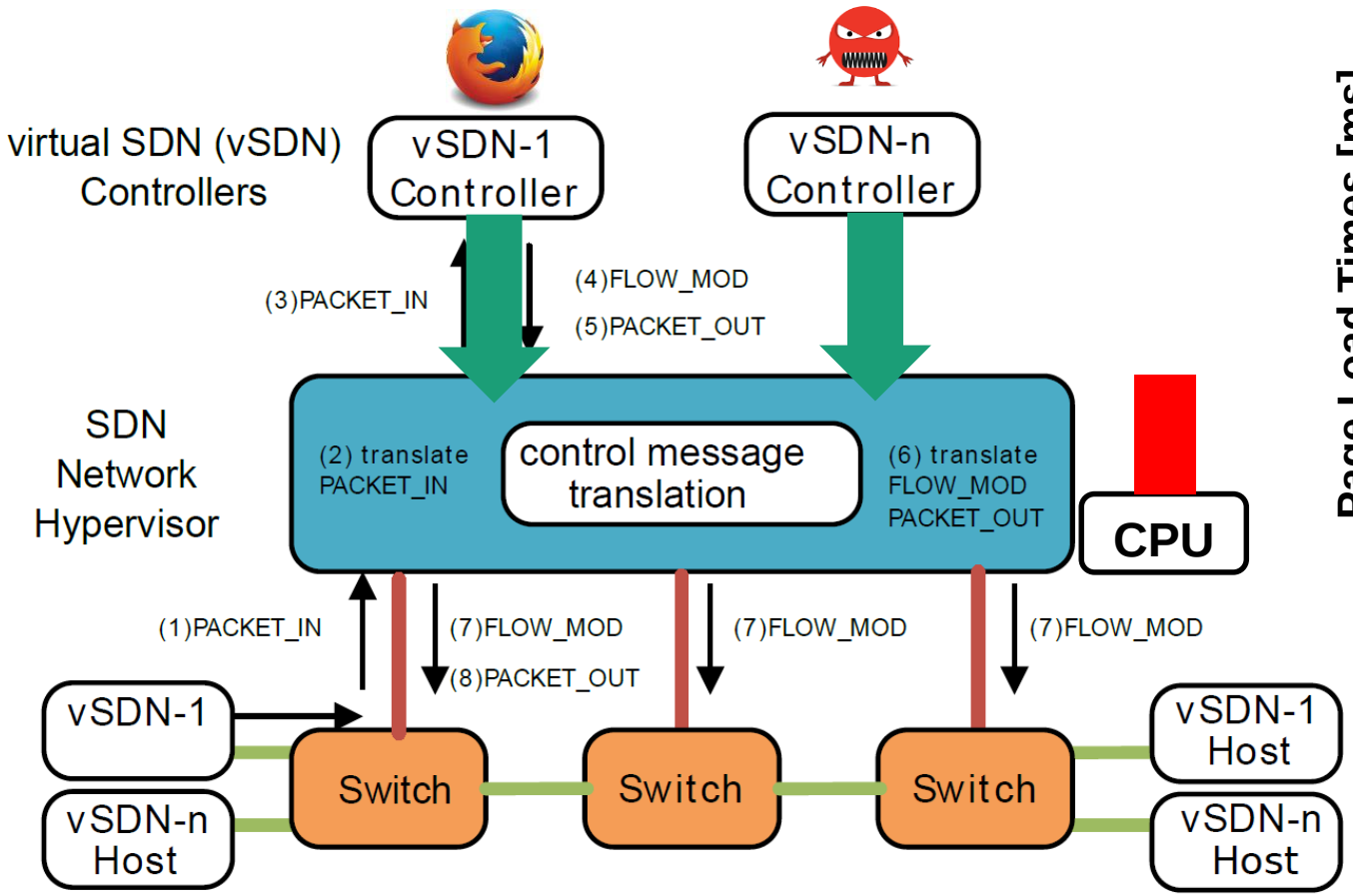
Why???

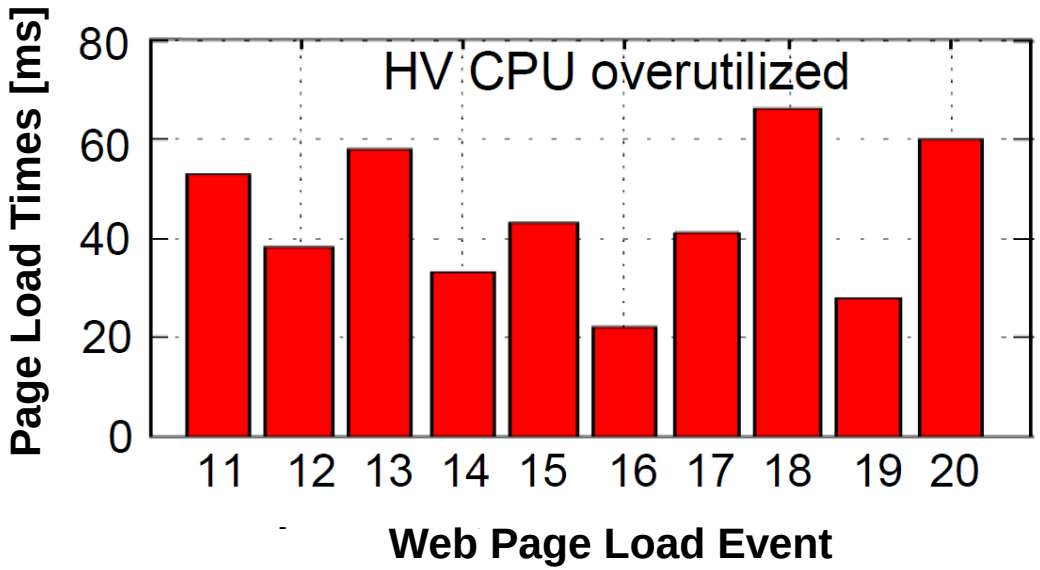
Test Results vs Production Traffic Tests



Govindan et al. 2016. Evolve or Die: High-Availability Design Principles Drawn from Googles Network Infrastructure. In Proceedings of the 2016 ACM SIGCOMM Conference (SIGCOMM '16). ACM, New York, NY, USA, 58–72.

Motivation: What happens in case of overload?





Hypervisor itself can be source of unpredictability!

This talk: A tool to analyze hypervisors and a performance evaluations of hypervisors!

Network Hypervisor Architectures

FlowVisor: A Network Virtualization Layer

Rob Sherwood*, Glen Gibb†, Kok-Kiong Yap†, Guido Appenzeller†,
Martin Casado◦, Nick McKeown†, Guru Parulkar†
* Deutsche Telekom Inc. R&D Lab, Los Altos, CA USA
† Stanford University, Palo Alto, CA USA
◦ Nicira Networks, Palo Alto, CA USA

ABSTRACT

Network virtualization has long been a goal of of the network research community. With it, multiple isolated logical networks each with potentially different addressing and fo… same physical infr… by taking advantag… [20, 23]) or by du… cialized) hardware[…

In this paper w… virtualization in w… plane can be share… each with distinct f… level virtualization… allows multiple net… with production tr… and hardware forw… this approach is co… chipsets and does … hardware such as F…

We build and de… our own productio… in practice by runn… within a campus ne… head of our approa… the isolation between virtual slices.

1. INTRODUCTION

This paper explores how to virtualize a network, and describes a particular system that we prototyped - called FlowVisor - that we have deployed to *slice*[1] our own production network. Similar to computer virtualization [22, 1, 21, 17], network virtualization promises to improve resource allocation, permits operators to checkpoint their network before changes, and allows competing customers to share the same equipment in a controlled and isolated fashion. Critically, virtual networks also promise to provide a safe and realistic environment to deploy and evaluate experimental "clean slate" protocols in production networks.

To better understand virtual networking, we first look closely at computer virtualization. Computer virtualization's success can be linked to a clean abstraction of the … uter virtualiza… permits slicing … operating sys… s it has its own …are abstraction … below the vir… build on a con… operating sys… veral flavors of …ed entirely new …rdware can be …Nvidia's GPU), …are abstraction … have different …rmance, lower …ove and below …tion of options, … efficient mar…

…efits in the net… work. Thus, by analogy, the network itself should have a hardware abstraction layer. This layer should be easy to slice so that multiple wildly different networks can run simultaneously on top without interfering with each other, on a variety of different hardware, including switches, routers, access points, and so on. Above the hardware abstraction layer, we want new protocols and addressing formats to run independently in their own isolated slice of the same physical network, enabling networks optimized for the applications running on them, or customized for the operator who owns them. Below the virtualization layer, new hardware can be developed for different environments with different speed, media (wireline and wireless), power or fanout requirements.

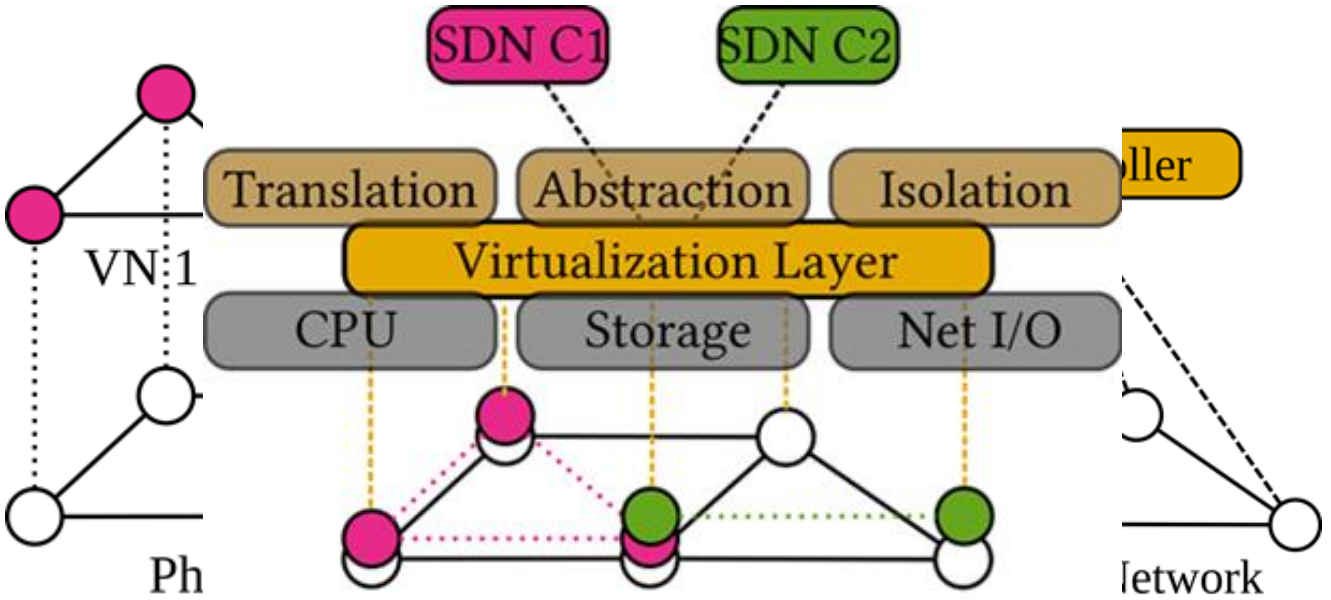
The equipment currently deployed in our networks

[1]Borrowing from the GENI [4] literature, we call an instance of a virtual network a *slice*, and two distinct virtual networks on the same physical hardware *slices*.

1

Idea:
Combine Network Virtualization and Software Defined Networking

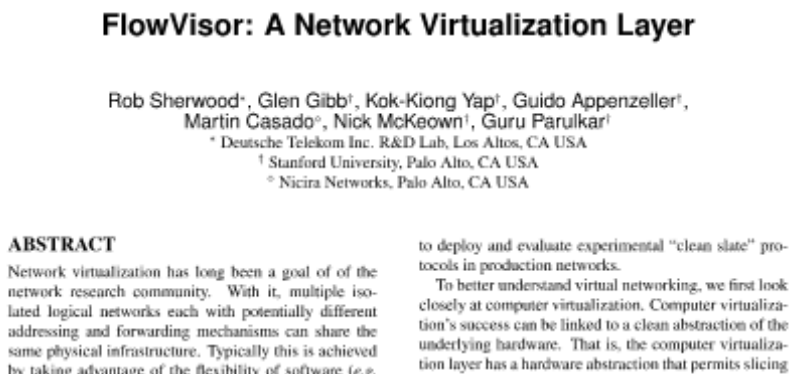
Network Virtualization + **Software Defined Networking**



Interference?

Predictable?

Hypervisor Network Function Implementations: FV vs OVX



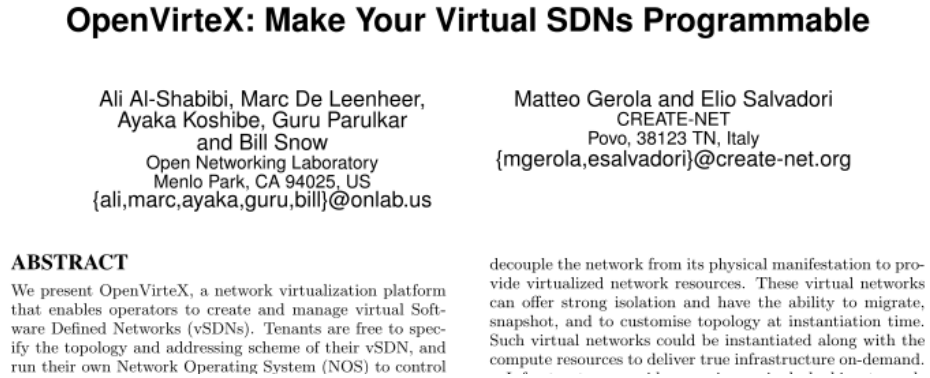
FlowVisor: A Network Virtualization Layer

Rob Sherwood*, Glen Gibb†, Kok-Kiong Yap†, Guido Appenzeller†,
Martin Casado◦, Nick McKeown†, Guru Parulkar†
* Deutsche Telekom Inc. R&D Lab, Los Altos, CA USA
† Stanford University, Palo Alto, CA USA
◦ Nicira Networks, Palo Alto, CA USA

ABSTRACT

Network virtualization has long been a goal of of the network research community. With it, multiple isolated logical networks each with potentially different addressing and forwarding mechanisms can share the same physical infrastructure. Typically this is achieved by taking advantage of the flexibility of software (e.g. to deploy and evaluate experimental "clean slate" protocols in production networks.

To better understand virtual networking, we first look closely at computer virtualization. Computer virtualization's success can be linked to a clean abstraction of the underlying hardware. That is, the computer virtualization layer has a hardware abstraction that permits slicing



OpenVirteX: Make Your Virtual SDNs Programmable

Ali Al-Shabibi, Marc De Leenheer,
Ayaka Koshibe, Guru Parulkar
and Bill Snow
Open Networking Laboratory
Menlo Park, CA 94025, US
{ali,marc,ayaka,guru,bill}@onlab.us

Matteo Gerola and Elio Salvadori
CREATE-NET
Povo, 38123 TN, Italy
{mgerola,esalvadori}@create-net.org

ABSTRACT

We present OpenVirteX, a network virtualization platform that enables operators to create and manage virtual Software Defined Networks (vSDNs). Tenants are free to specify the topology and addressing scheme of their vSDN, and run their own Network Operating System (NOS) to control

decouple the network from its physical manifestation to provide virtualized network resources. These virtual networks can offer strong isolation and have the ability to migrate, snapshot, and to customise topology at instantiation time. Such virtual networks could be instantiated along with the compute resources to deliver true infrastructure on-demand.

- **Translation:** Changes only CP information
- **Abstraction:** 1-to-1 Mapping
- Main work done by single thread

- **Translation:** Rewrites message headers
- **Abstraction:** 1-to-1 Mapping, Big Switch
- Applies multi-threading for tasks

Performance Analysis of Existing Network Hypervisor Architectures

FlowVisor: A Network Virtualization Layer

Rob Sherwood*, Glen Gibb†, Kok-Kiong Yap†, Guido Appenzeller†,
Martin Casado◦, Nick McKeown†, Guru Parulkar†
* Deutsche Telekom Inc. R&D Lab, Los Altos, CA USA
† Stanford University, Palo Alto, CA USA
◦ Nicira Networks, Palo Alto, CA USA

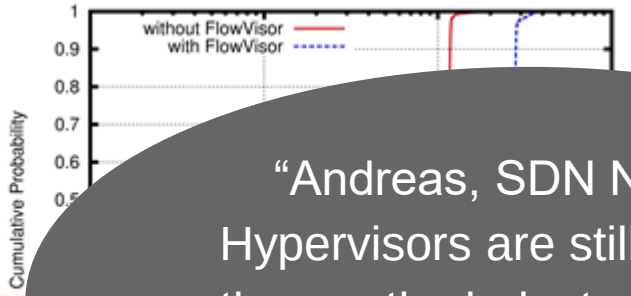
ABSTRACT

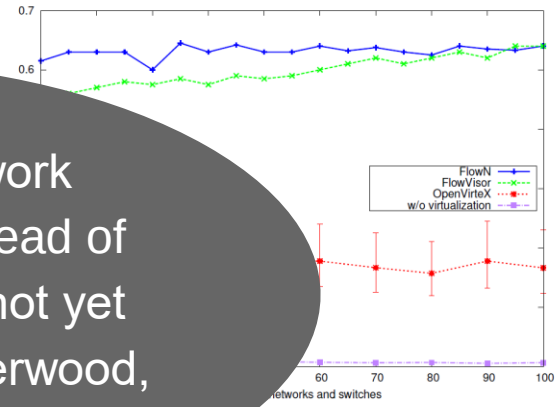
Network virtuali
network researc
lated logical net
addressing and f
same physical int
by taking advant
[20, 23]) or by d
cialized) hardwar
In this paper v
virtualization in
plane can be sha
each with distinct
level virtualizatio
allows multiple n
with production
and hardware fo
this approach is
chipsets and does
hardware such as
We build and
our own product
in practice by ru
within a campus
head of our appro
the isolation betw

1. INTROD

This paper exp
describes a partic
FlowVisor - that
production netwo
tion [22, 1, 21, 1
improve resource
point their netwo
ing customers to
trolled and isolat
also promise to p

[1]Borrowing from t
of a virtual networ
on the same physic





OpenVirteX: Make Your Virtual SDN

Ali Al-Shabibi, Marc De Leenheer,
Ayaka Koshibe, Guru Parulkar
and Bill Snow
Open Networking Laboratory
Menlo Park, CA 94025, US
{ali,marc,ayaka,guru,bill}@onlab.us

Matteo Ge
Pov
{mgerola,esi

ABSTRACT

We present OpenVirteX, a network virtualization platform
that enables operators to create and manage virtual Soft-
ware Defined Networks (
ify the topology and add
run their own Network C
it. Since OpenVirteX lo
infrastructure, it also en
such as link and switch re
and migration of these te
on the design of FlowVis
controller proxy between
ants' network OSes. Our
show that i) OpenVirteX
with configurable vSDNs
to the control channel, a
the introduction of featu
resilience to tenant netw

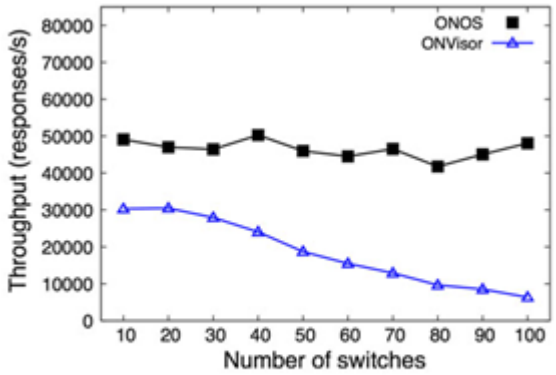
decouple the netv
vide virtualized
can offer strong

Categories and Sub

C.2.1 [Computer-Com
Architecture and Design;
Networks]: Network Op

"Andreas, SDN Network Hypervisors are still ahead of time … the industry is not yet ready for them" Rob Sherwood, Facebook

…n-to-controller only! [OVX]



(B) Throughput comparison

1 Tenant only! [Onvisor2018]

Received: 2 May 2017 | Revised: 9 August 2017 | Accept
DOI: 10.1002/nem.2012

RESEARCH ARTICLE

LEY

ONVisor: Towards a scalable and flexible SDN-based network virtualization platform on ONOS

Yoonseon Han[1,3] | Thomas Vachuska[3] | Ali Al-Shabibi[3] | Jian Li[2,3] | Huibai Huang[4] | William Snow[3] | James Won-Ki Hong[1,2]

No detailed performance study! Why? No Tool available!

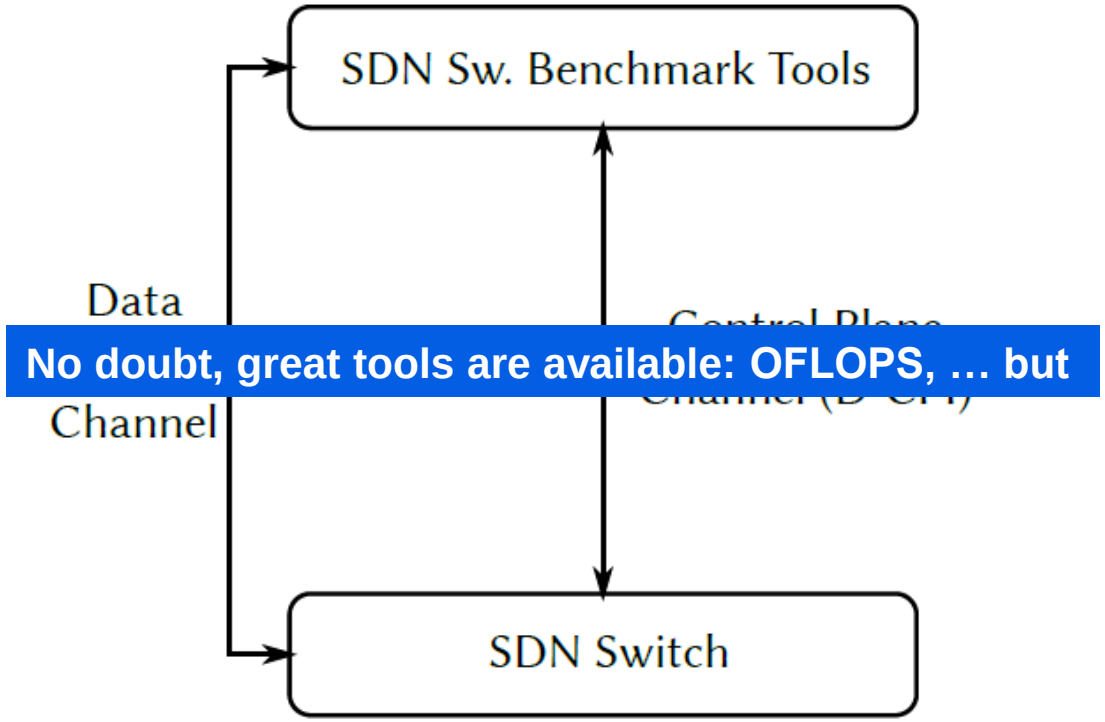
1. INTRODUCTI

Park, California, USA

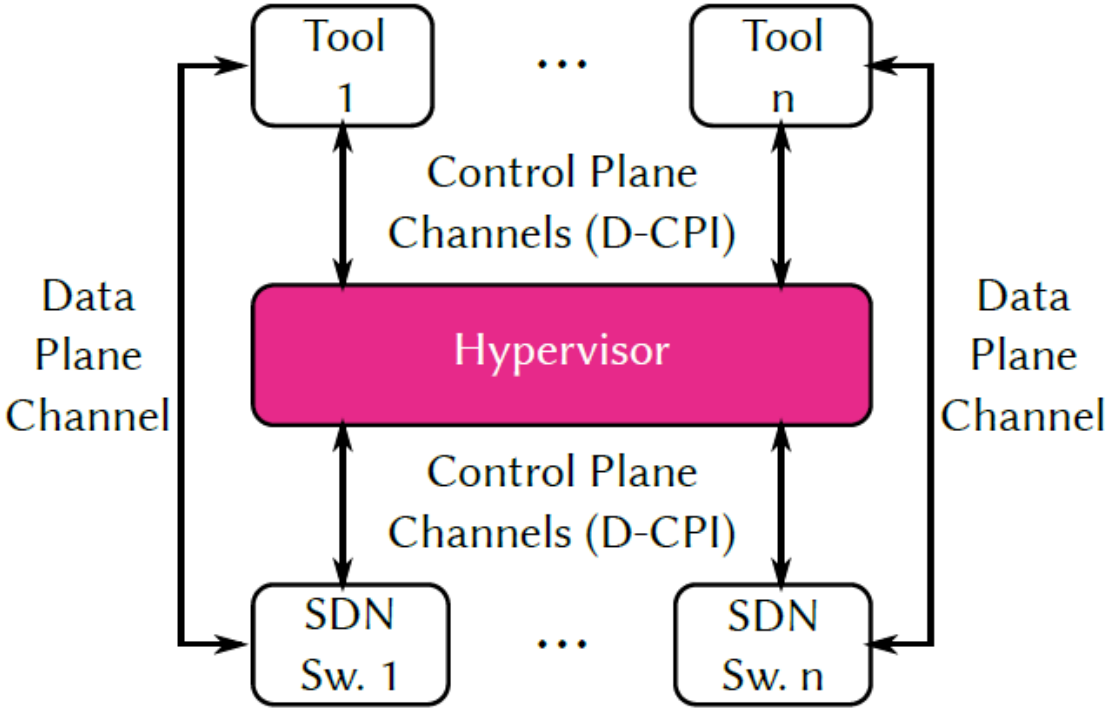
virtual networks (VNs) by decoupling the physical network in terms of topology,

From non-virtualized SDN networks to virtualized SDN networks

Switch Benchmarks

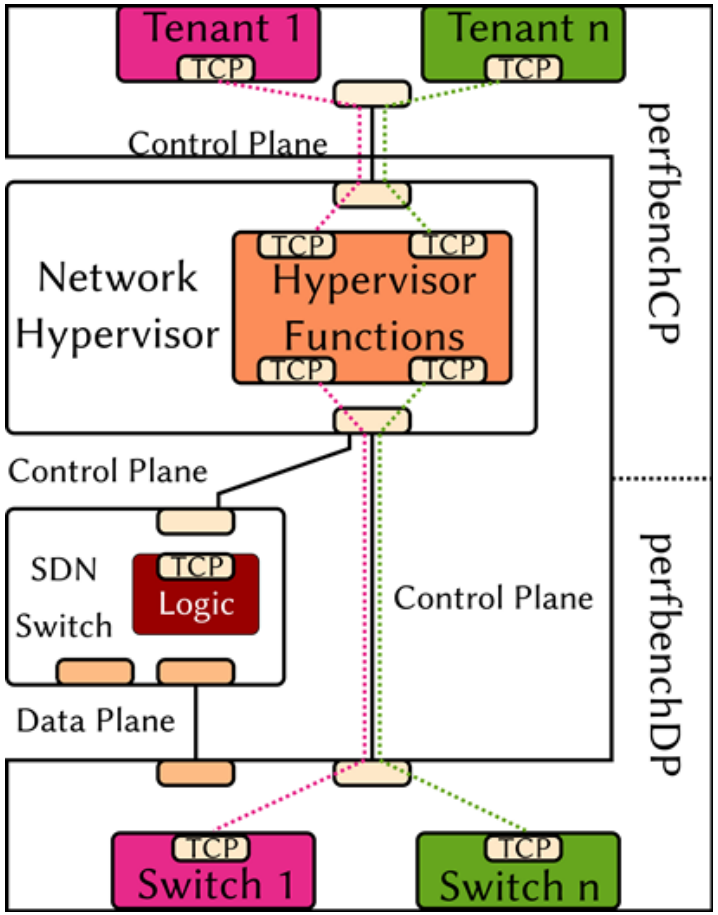


No doubt, great tools are available: OFLOPS, … but



- Challenge: Coordination and emulation complexity
- Goal: One tool emulating single tenant, single switch, multi-tenant, multi-switch

perfbench [1,2,3]



- Multi-tenant/multi-switch e
- Traffic modeling: inter-
- Modular measurements: either
  both entities
- OF 1.0 and 1.3

Available soon!

docker

Open Source!

https://github.com/tum-lkn/perfbench

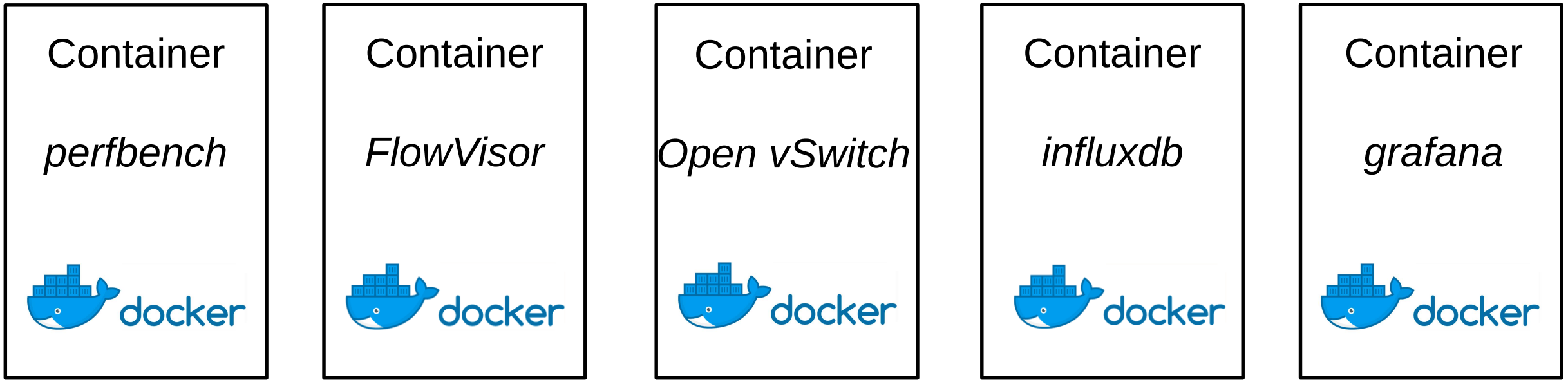
[1] A. Blenk, A. Basta, L. Henkel, J. Zerwas, S. Schmid, W. Kellerer, perfbench: A Tool for Predictability Analysis in Multi-Tenant Software Defined Networks. ACM SIGCOMM 2018 Conference Posters and Demos, 2018,

[2] A. Basta, A. Blenk, S. Dudycz, A. Ludwig, S. Schmid ,Efficient Loop-Free Rerouting of Multiple SDN Flows. IEEE/ACM Transactions on Networking 26 (2), 2018, pp. 948-961

[3] A. Blenk, A. Basta, W. Kellerer, S. Schmid, On the Impact of the Network Hypervisor on Virtual Network Performance .IFIP Networking, Poland, Warsaw, 2019, pp. 1-9

perfbench in action

perfbench in action: setting

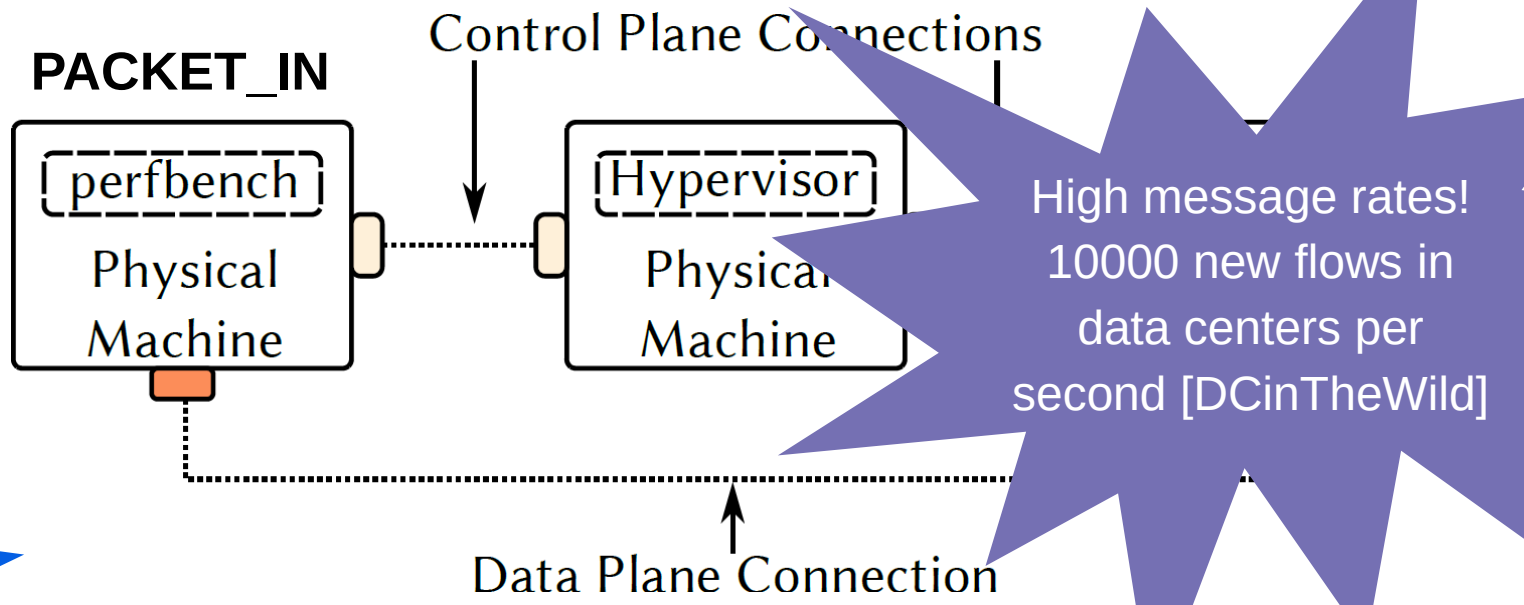


Testbed setups

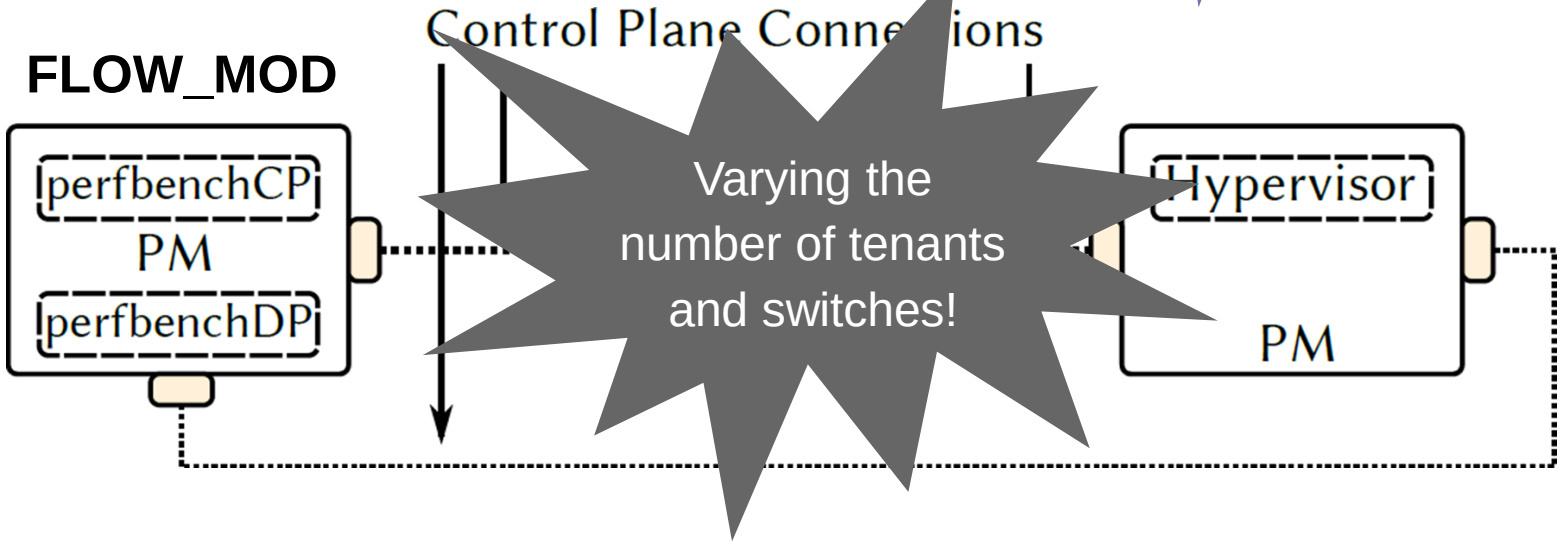
With OVS

Different OF Message types

Perfbench switch emulation

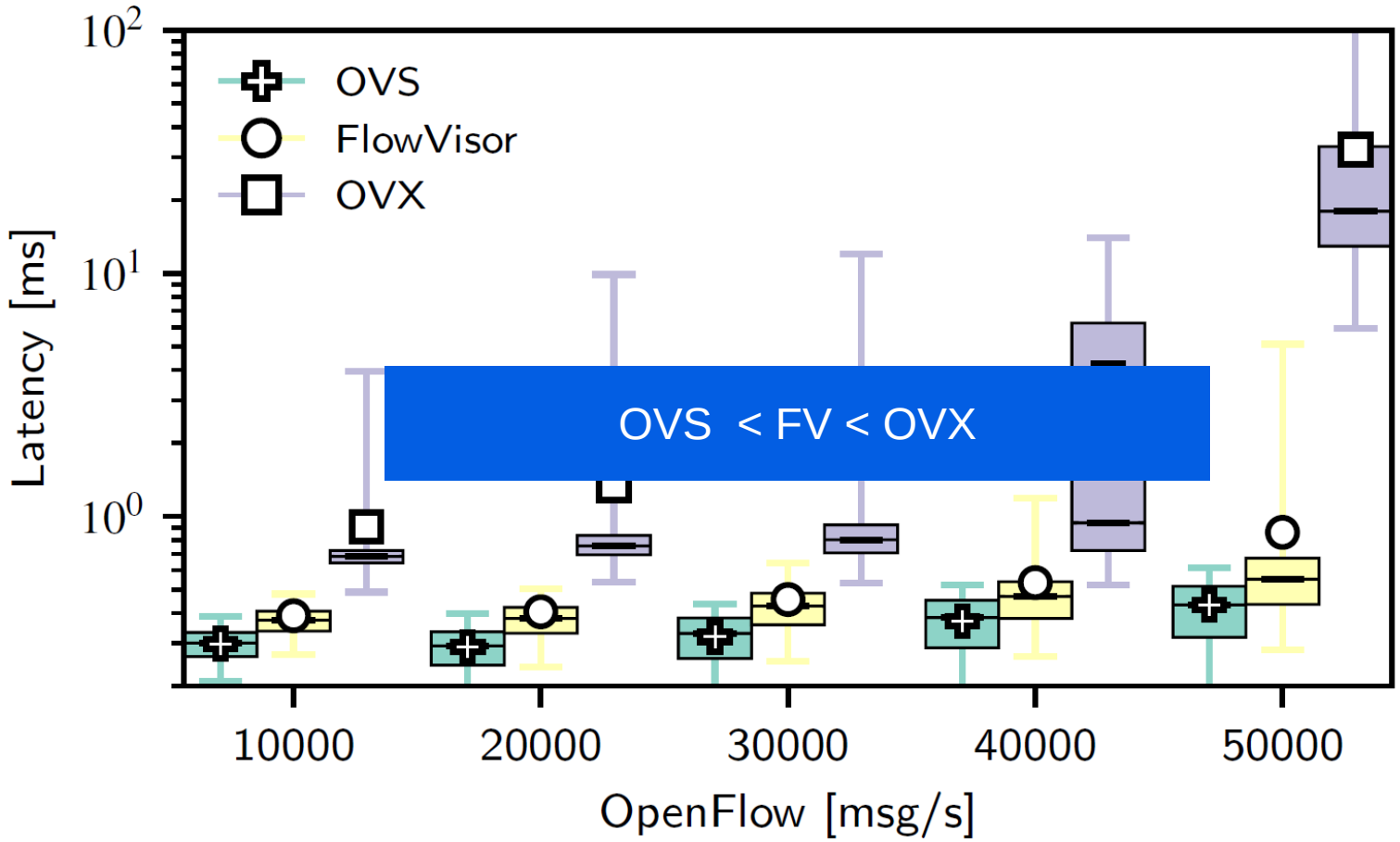


High message rates! 10000 new flows in data centers per second [DCinTheWild]



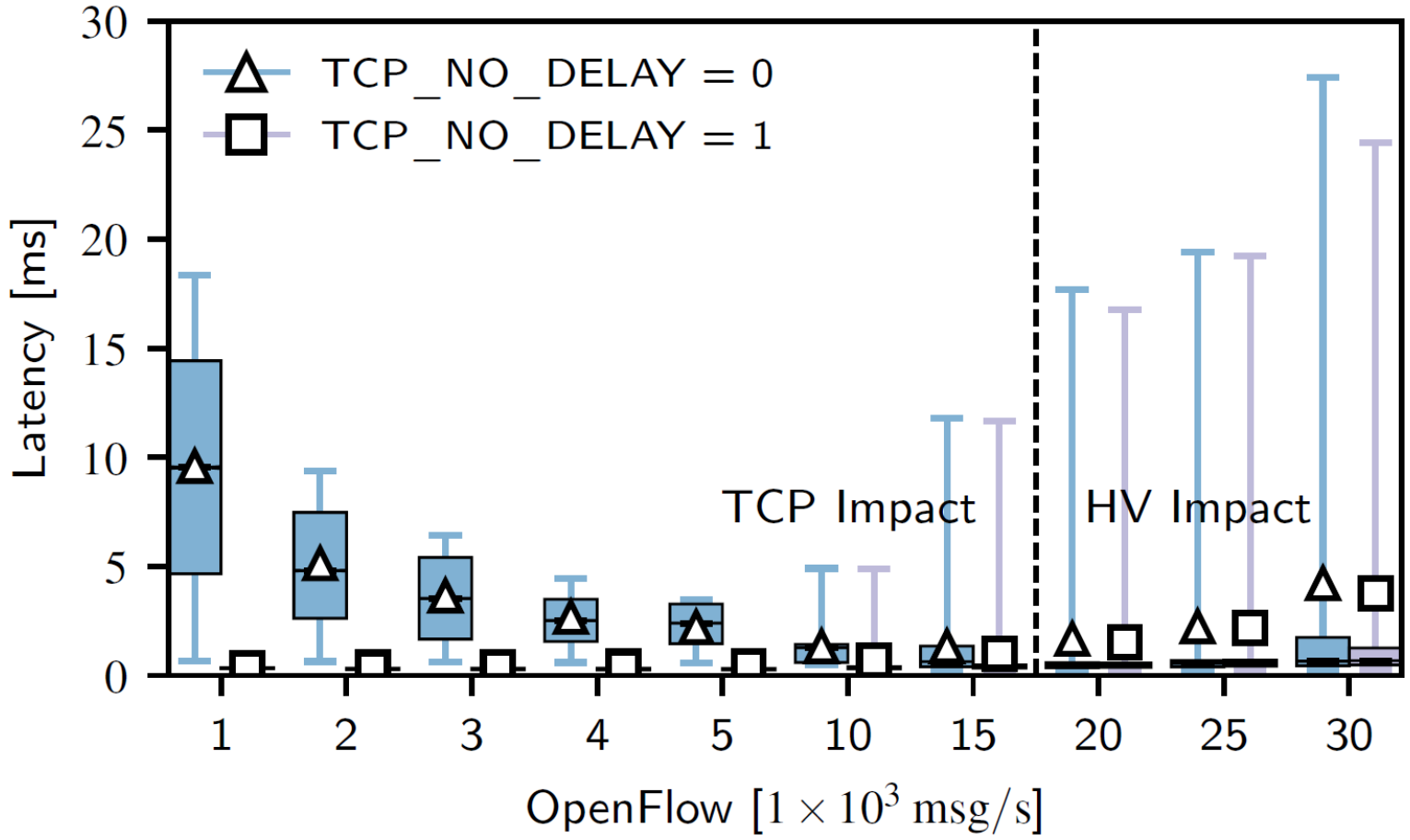
Varying the number of tenants and switches!

Virtualization: What does it cost? (PACKET_IN)



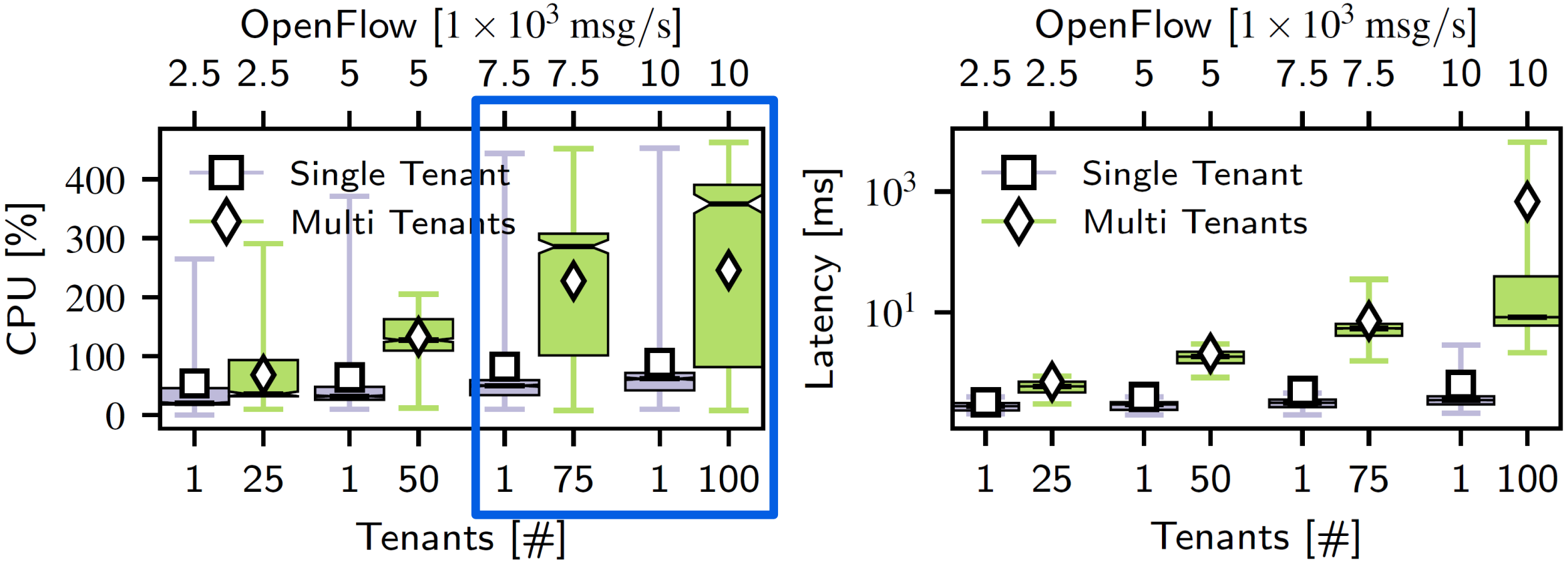
- Latency of PACKET_IN: OVS < FV < OVX
- Not inline with original papers

Impact of tenant controller behaviors (FLOW_MOD)



- Tenant controller behavior (Delay vs No Delay) determines latency
- 15k – 30k: processing of hypervisor determines latency

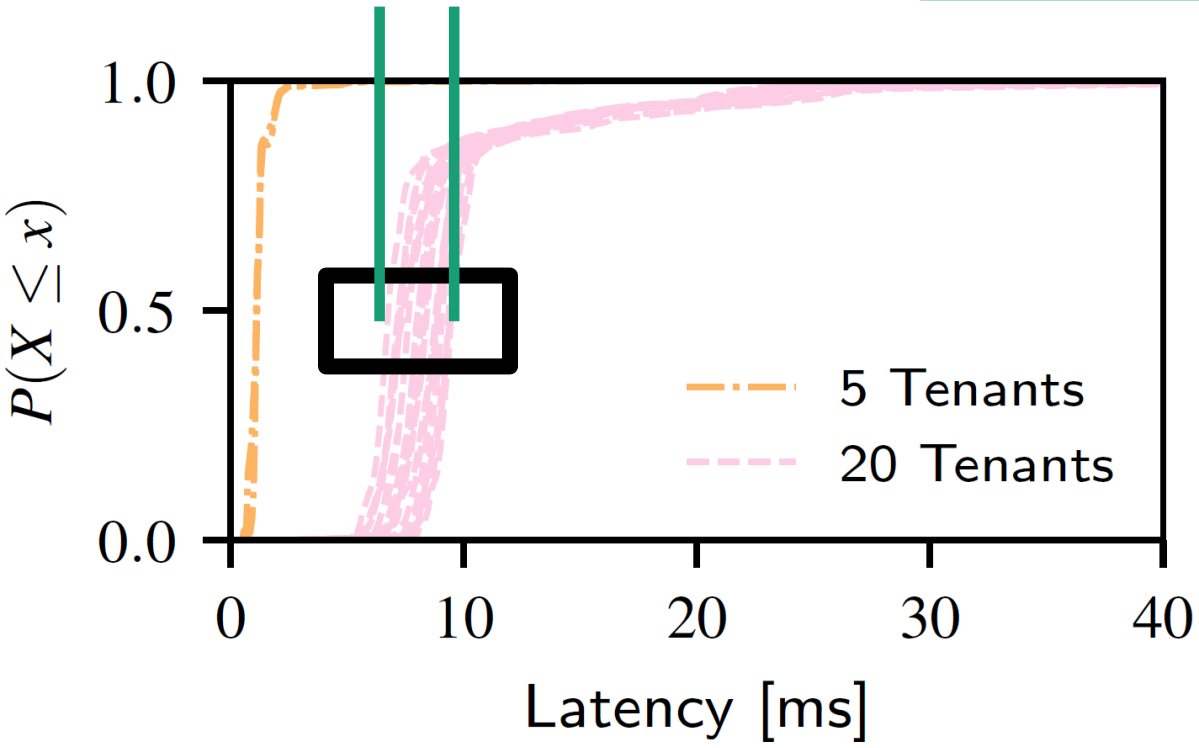
OVX impact of number of tenants (FLOW_MOD)

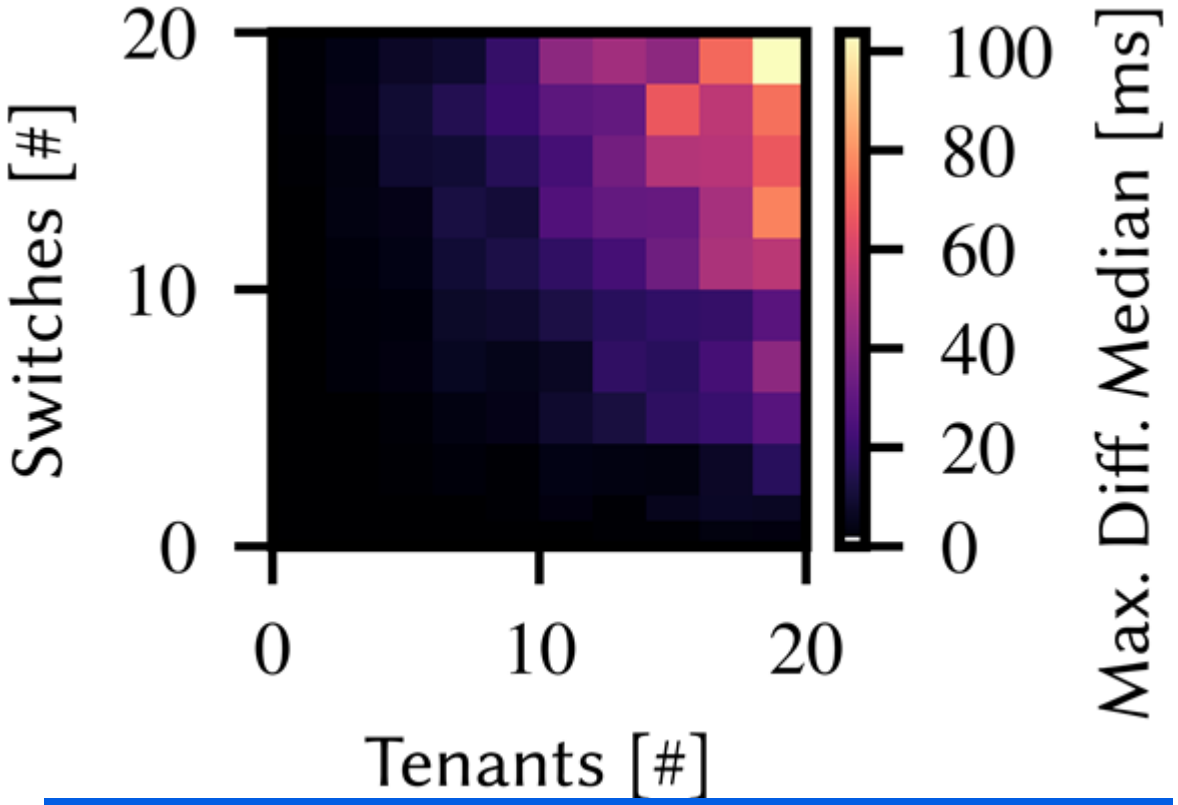


- Overutilization: With 100 tenants and 10000 FLOW_MOD messages per second
- Latency becomes high – worse service level agreements, unpredictable

Future Work: How to evaluate Fairness? (FV, FLOW_MOD)

Take maximum difference of median values





Already 20 tenants show a notable latency gap of 5 ms (high variance)

The more switches and controllers, the less predictable

Conclusion

Programmable network virtualization important: testing, slicing, guarantees, isolation, flexibility

But: Network Hypervisor itself can introduce unpredictability! Potential showstopper!

Hence, **deep understanding of architectures realizing programmable virtual networks important!**

This research:

- Benchmarking virtual environments is important … but not trivial

This paper:

- A tool for benchmarking virtual SDN networks
- Performance insights in hypervisor implementation aspects

Thank you!

Questions?