DPDK | IT大咖说 知识分享平台

DPDK DATA PLANE DEVELOPMENT KIT

THE LINUX FOUNDATION

DPDK SUMMIT CHINA 2017

主办方： intel

参与方： 腾讯云 ZTE 美团云 Panabit 太一星晨 Balance Your Networks UnitedStack有云 云杉网络 Yunshan Networks

协办方： SDNLAB 专注网络创新技术 视频支持方： IT大咖说 知识分享平台

DPDK DATA PLANE DEVELOPMENT KIT

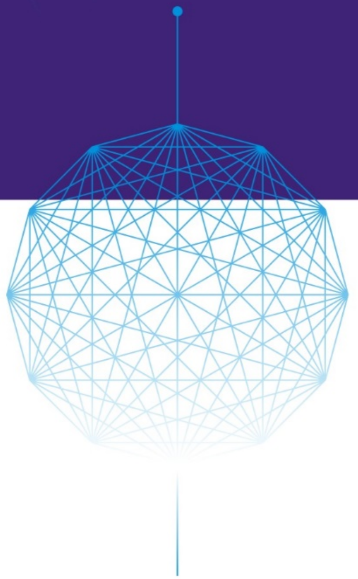
THE LINUX FOUNDATION

DPDK | IT大咖说 知识分享平台

DPDK in container

Status Quo and Future Directions

Jianfeng Tan, June 2017



主办方： intel

参与方： 腾讯云 ZTE 美团云 Panabit 太一星晨 Balance Your Networks UnitedStack有云 云杉网络 Yunshan Networks

协办方： SDNLAB 专注网络创新技术　　视频支持方： IT大咖说 知识分享平台

LEGAL DISCLAIMER

- No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.
- Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.
- This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.
- Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.
- © 2017 Intel Corporation. Intel, the Intel logo, Intel. Experience What's Inside, and the Intel. Experience What's Inside logo are trademarks of Intel. Corporation in the U.S. and/or other countries.
- *Other names and brands may be claimed as the property of others.
- Copyright © 2017, Intel Corporation. All rights reserved.

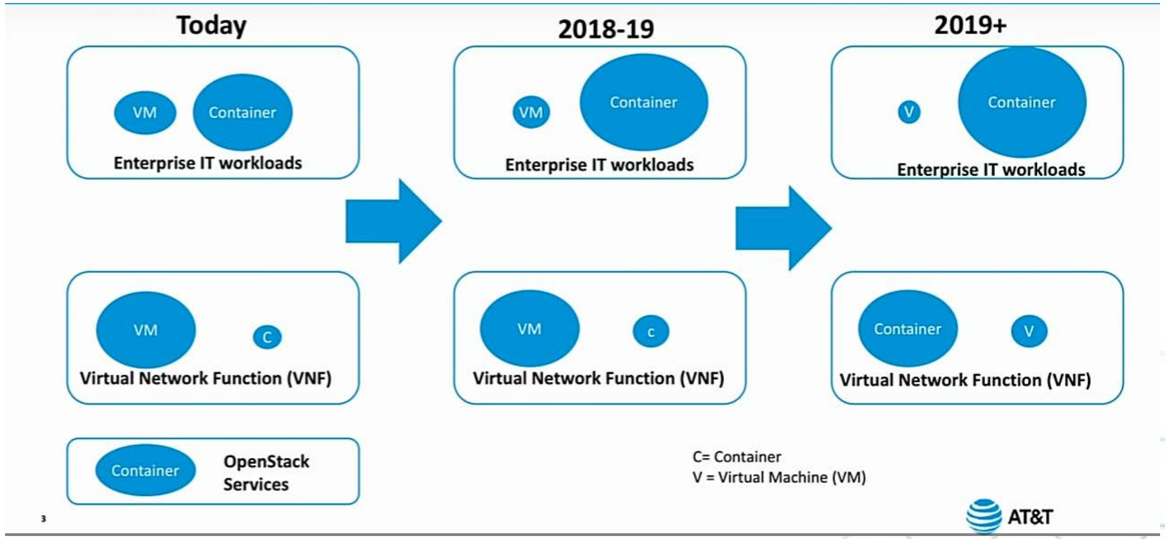
Agenda

- Why containers
- Challenges in container networking
- Data plane
- Control plane
- Summary

Why containers

- Easy-to-deploy
- Lightweight
  - ❑ Deployment time
  - ❑ Footprint (image & memory & CPU)
- Benefits
  - ❑ Service agility
  - ❑ Performance

Trend



Today

VM Container
Enterprise IT workloads

VM C
Virtual Network Function (VNF)

Container OpenStack Services

2018-19

VM Container
Enterprise IT workloads

VM c
Virtual Network Function (VNF)

2019+

V Container
Enterprise IT workloads

Container V
Virtual Network Function (VNF)

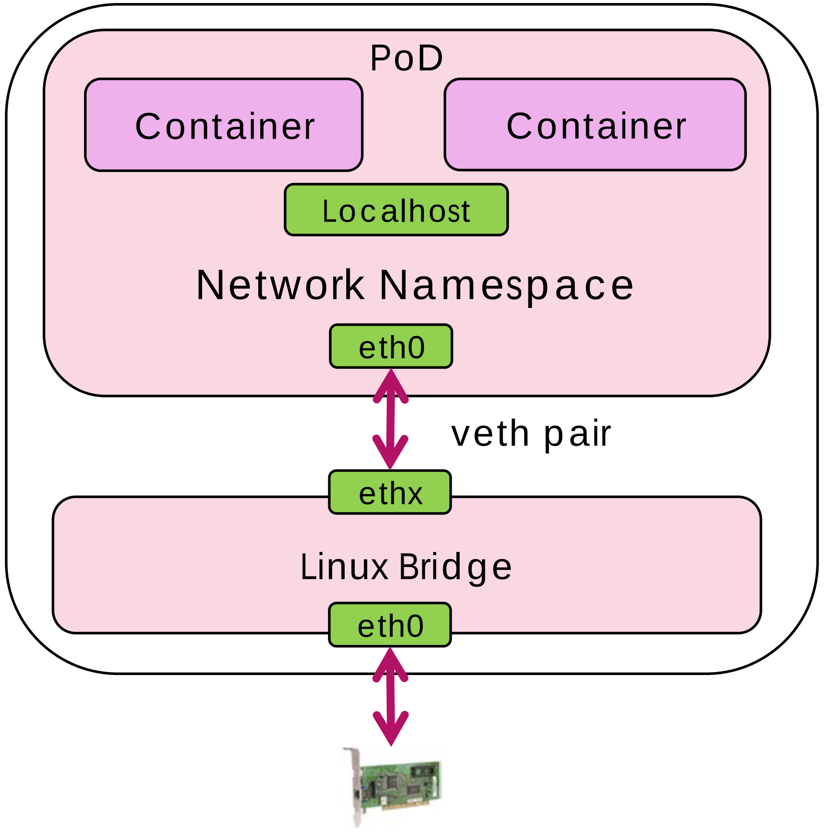
C= Container
V = Virtual Machine (VM)

AT&T

3

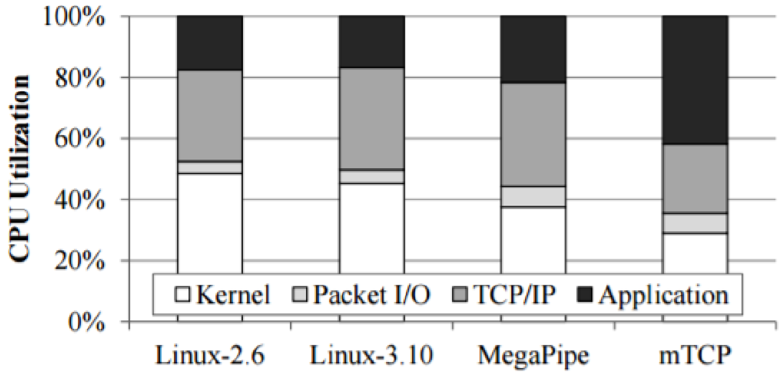
AT&T Container Strategy and OpenStack's Role in It, OpenStack Boston 2017 (bit.ly/2rfftRA)

Challenges in Container networking



HAProxy[1]: 5% user and 95% system

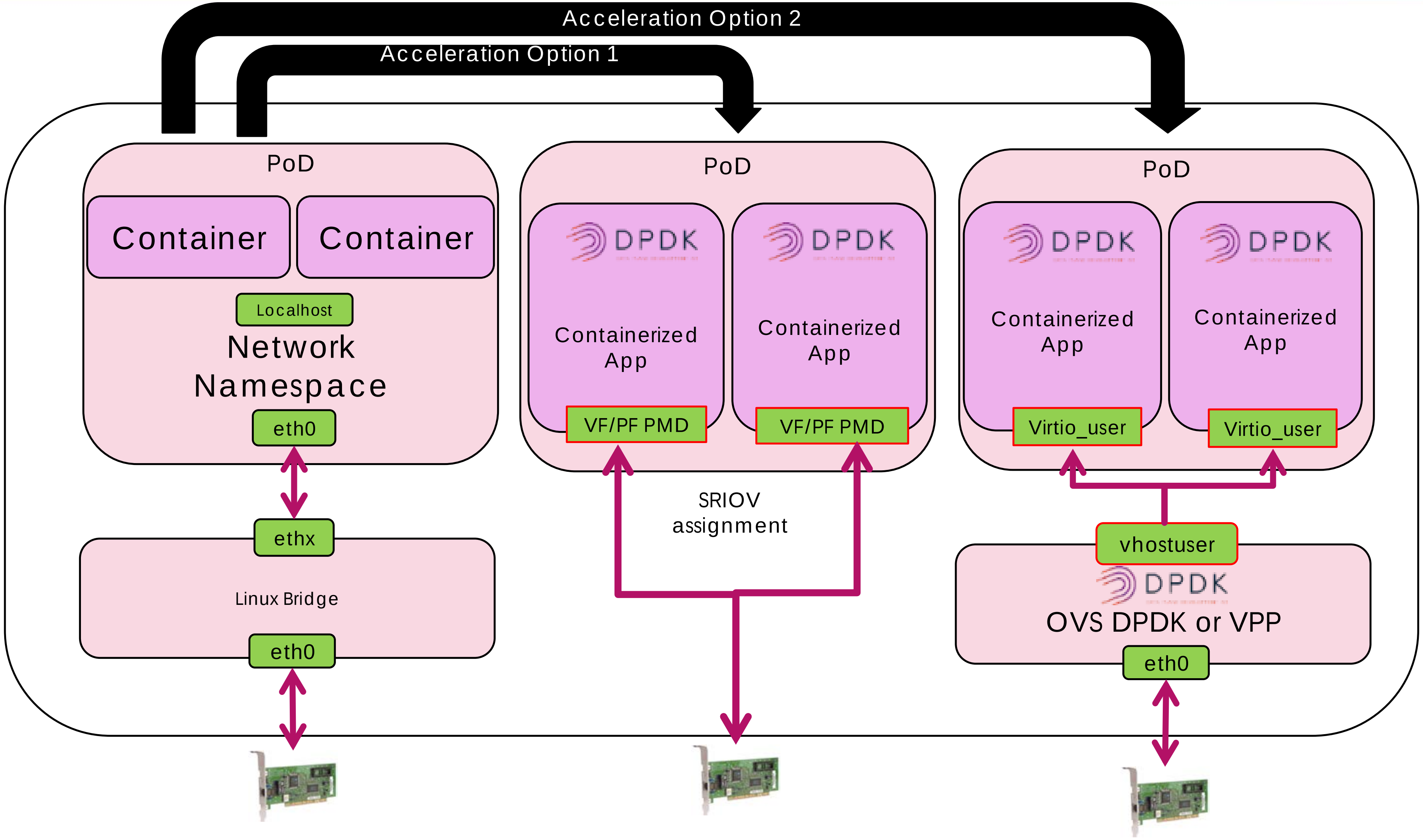
Lighttpd[2]:



How much time spent in percentages in kernel space？

[1] http://www.haproxy.org/
[2] https://www.usenix.org/sites/default/files/conference/protected-files/nsdi14_slides_jeong.pdf

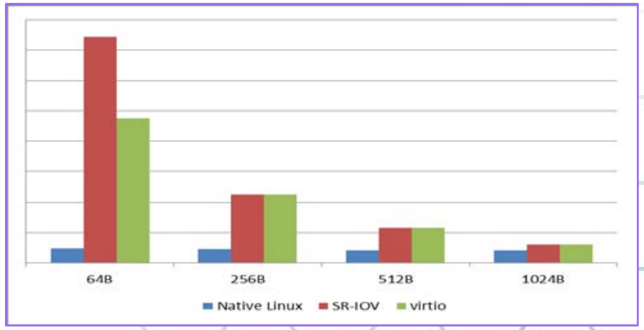
Data plane: DPDK as user space networking



Latency:

Microsecond-level

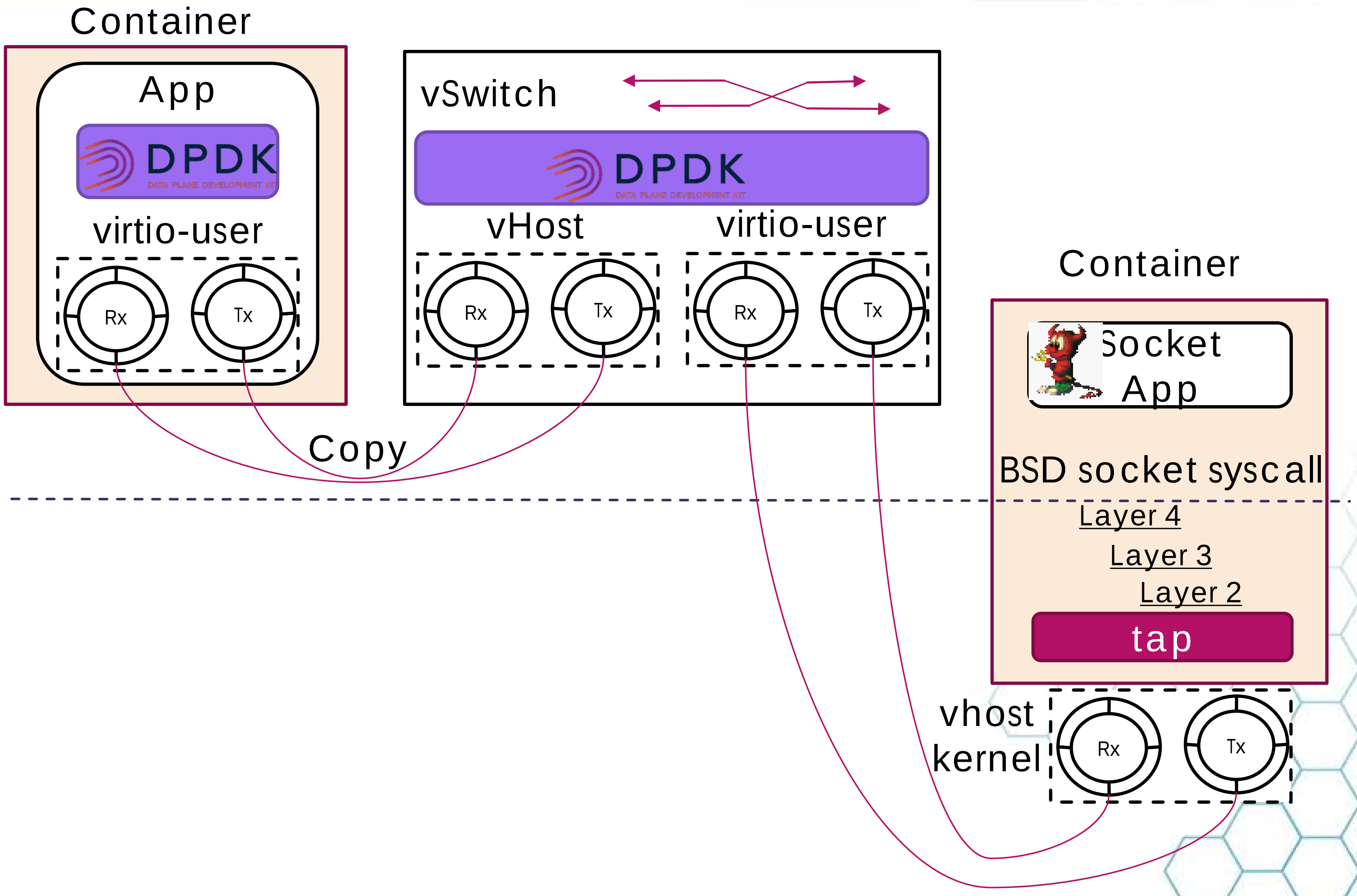
Throughput:



Performance data: https://dpdksummit.com/Archive/pdf/2016USA/Day02-Session02-Steve%20Liang-DPDKUSASummit2016.pdf

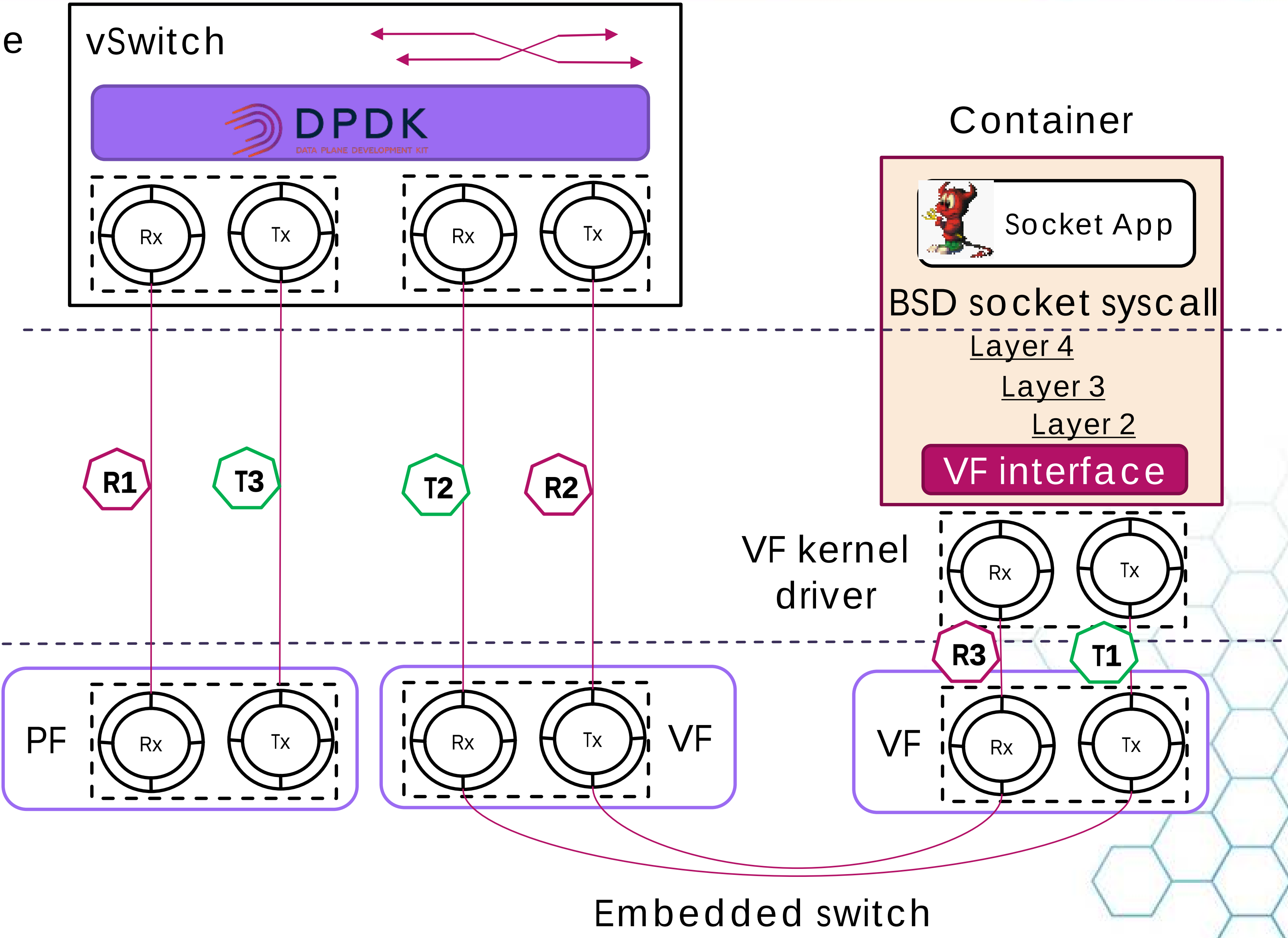
Bridge legacy containers with vSwitch (1)

- Performance?
  - One more data copy
  - Packet mmap?
- Thread model
  - Tx in application context instead of vhost kthread context



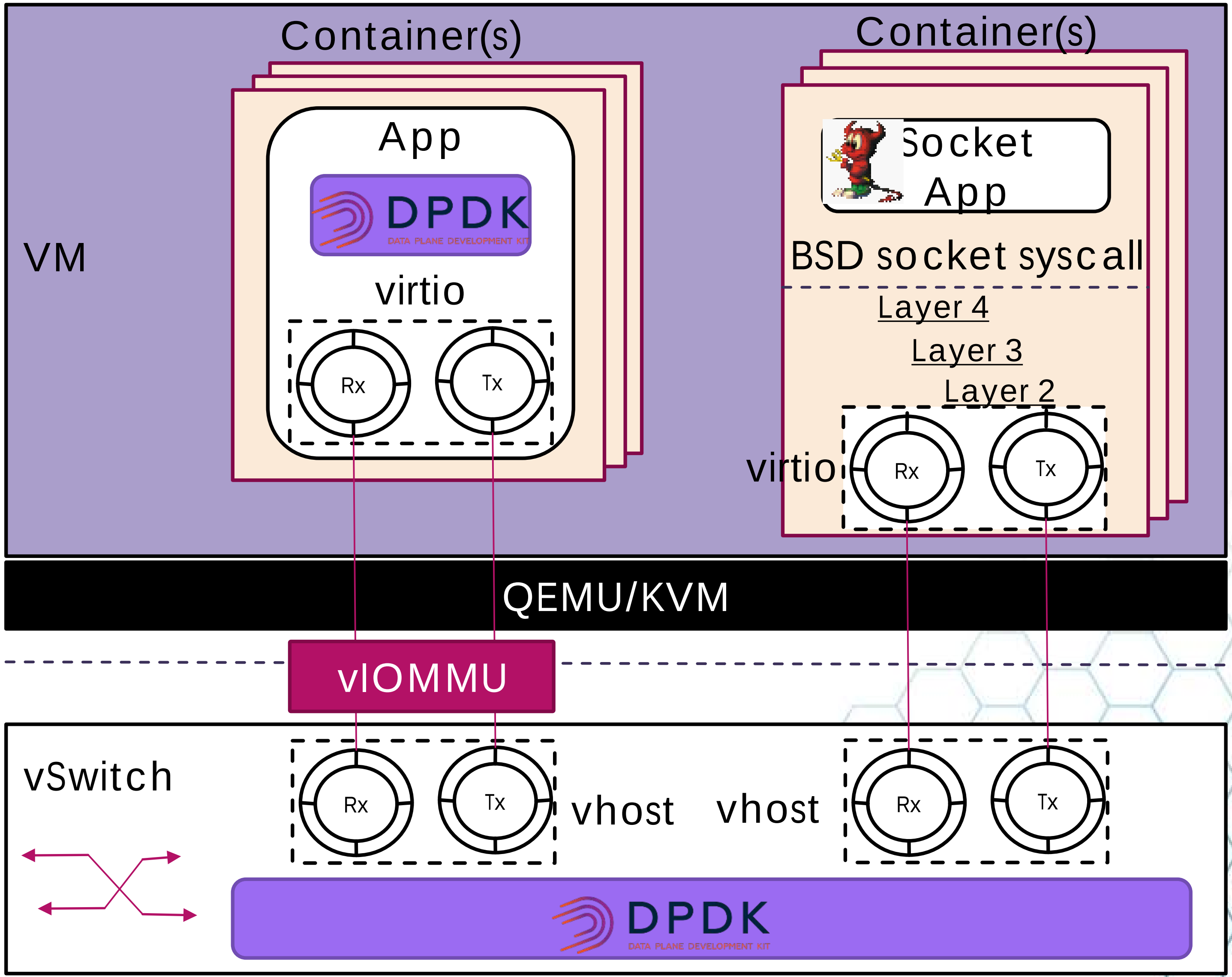
Bridge legacy containers with vSwitch (2)

- Dimidiate throughput for 1x more data crossing PCIe bus
- Pressure on embedded switch



Bridge legacy containers with vSwitch (3)

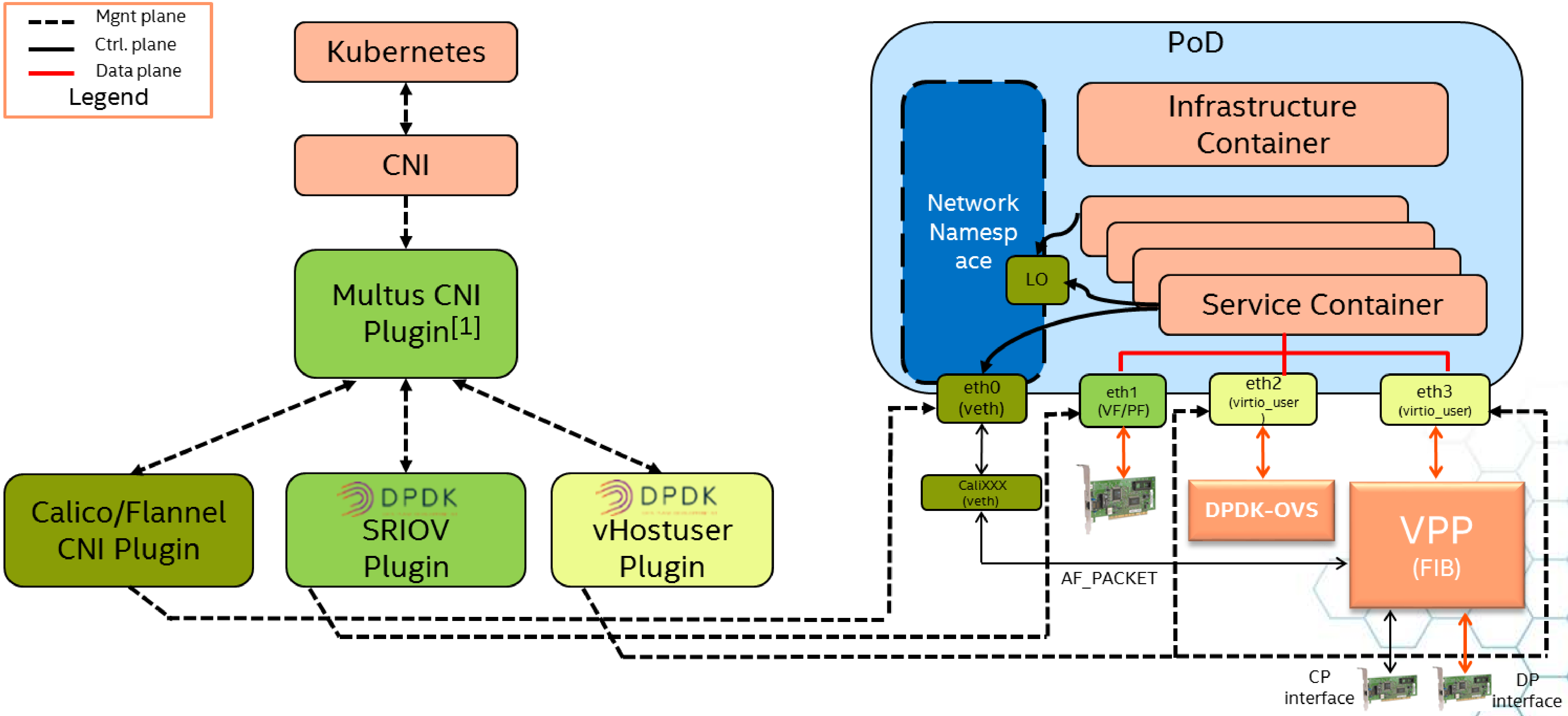
- *1* VM : *n* containers
- Virtio-net device(s) hot-plugged per Container
- VT-x de-privileged host allows radical optimization
- Containers in the same trust zone reside in one VM
- Protect DMA attack from compromised DPDK application through vIOMMU



Data plane: Challenges and opportunities

- Rapid lifecycle
  - No PCI full scan if there is a whitelist param
  - Memory – lazy allocation
- Scalability
  - 4K pages support
  - Core sharing - interrupt mode for vhost-user
  - Device hotadd/hotplug
  - More fine-grained device pass-through
- Stable and performant user space TCP stack

Control plane: status quo



Mgnt plane
Ctrl. plane
Data plane
Legend

Kubernetes
CNI
Multus CNI Plugin[1]
Calico/Flannel CNI Plugin
DPDK SRIOV Plugin
DPDK vHostuser Plugin

PoD
Network Namespace
Infrastructure Container
LO
Service Container
eth0 (veth)
eth1 (VF/PF)
eth2 (virtio_user)
eth3 (virtio_user)
CaliXXX (veth)
AF_PACKET
DPDK-OVS
VPP (FIB)
CP interface
DP interface

[1] https://github.com/Intel-Corp/multus-cni

Control plane: Scheduling

- Intel® Resource Director Technology (RDT)
  - Linux kernel 4.10 introduces L3 CAT, etc
  - Linux kernel 4.12 is on-track to support MBA
- CPU pinning, NUMA aware, huge pages
  - Enhance kubelet service?
- Enhanced Platform Awareness (EPA) feature framework in k8s through Node Discovery pod[1]

[1] https://github.com/Intel-Corp/node-feature-discovery

Acknowledge contribution from:

- Cunming Liang
- Danny Zhou
- Heqing Zhu
- Hongjun Ni
- Huawei Xie
- John DiGiglio
- Johnson Li
- Kuralamudhan Ramakrishnan
- Ray Kinsella

Summary

- DPDK as the user space container networking data plane is ready and still in evolution.
- Control plane of user space container networking is WIP.
- Bridging legacy containers with user space vSwitch is WIP.

You are very welcomed to share ideas and contribute code!

Thanks!