DPDK SUMMIT CHINA 2017
DPDK in container
Status Quo and Future Directions
Jianfeng Tan, June 2017
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

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

How much time spent in percentages in kernel space?

HAProxy\(^1\): 5% user and 95% system

Lighttpd\(^2\):

Data plane: DPDK as user space networking

Acceleration Option 1

Container
Container

Network Namespace

eth0

ethx

Linux Bridge

Acceleration Option 2

Containerized App
Containerized App

VF/PF PMD
VF/PF PMD

SRIOV assignment

DPDK

vhostuser

OVS DPDK or VPP

Containerized App
Containerized App

Virtio_user
Virtio_user

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

![Diagram showing the connection between legacy containers and vSwitch with DPDK integration.](image-url)
Bridge legacy containers with vSwitch (2)

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

![Diagram showing network traffic and container interaction with vSwitch and Embedded switch]
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

[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!