DPDK as microservices in ZTE Paas

Yong Wang
ZTE

DPDK Summit - San Jose – 2017

#DPDKSummit
Overview: ZTE Paas

- run in docker, deployed in vm or ironic
- implement service discovery mechanism
- support ICT applications
- multi-tenant, multi-networking-plane
Packet Flow to/from PaaS

1. Packets from outside of PaaS Cloud

2. Inside PaaS Cloud, communications based on FastMQ

3. Packets Output

#DPDKSummit
Message Service - Publish/Subscribe

- Message Service crosses over the K8S clusters
- High speed message traffic between Publisher and Subscriber
- Traffic load balance policy is defined on Publisher by Application
- Service will support the conversation HA
Model without DPDK

POD1

C1

C2

eth0

192.168.0.1

POD2

C1

C2

eth1

192.168.0.2

POD3

C1

C2

eth2

192.168.0.3

#DPDKSummit
one DPDK container per Pod
Model with DPDK(2)

All pods share a DPDK container
Communications between containers(1)

POD1

C1

C2

POD2

C1

C2

POD3

1

C2

POD0

DPDK

eth0

#DPDKSummit
Key Technology

- master/secondary mechanism
- sharing memory
- queue(rte_ring)
Benefits vs vhost-user

- Bidirectional zero copy
- 2M & other size hugepage support

vhost-user:
- only dequeue zero copy
- only support 1G size hugepage
Communications between containers(2)

POD1

C1 → C2
zero copy

POD2

C1 → C2

POD0

DPDK
virtio

POD0

DPDK
virtio

POD0

DPDK
virtio

POD0

DPDK
virtio

openvswitch/vpp/ovs-dpdk

copy! how to avoid?

#DPDKSummit
combine pod0 with ovs
5G and Network Slicing

.dpdk as microservices brings the following benefits to network slicing:
- zero copy
- high performance
- low latency
Network Slice Orchestration

MicroService2

POD1
C1 C2
share memory queue
queue

POD2
C1 C2
share memory queue
queue

POD0
OVS DPDK

Network Slice

NF1 NF3 NF4
Service1 Service2 Service3 Service4
MS1 MS2 MS3
Pod1 Pod2 Pod0

NFVO VNFM VNPM
Acknowledge contribution from

- Chenggang Li
- Jian Gu
- Chunhua Qin
- Songming Yan
- Binbin Xu

#DPDKSummit