The state of DMA offload in OVS-DPDK

Sunil Pai G, Cian Ferriter, Harry Van Haaren
Intel

December 2021
OVERVIEW

• The Problem
• The Solution
• Datapath design
• Scalability
• Packet walkthrough
• Future Work
The problem...

**PHY-VM-PHY DATAPATH**

```
OVS Datapath

Miniflow extract

DPDK

NIC RX

NIC HW

DPDK

vHost

Virtio

Testpmd (csum fwd)

Guest (VM)

_PHY-VM-PHY DATAPATH_
```
The solution...

**PHY-VM-PHY DATAPATH WITH DMA**

OVS Datapath

- **Miniflow extract**
- **DPDK NIC RX**
- **NIC HW**
- **EMC**
- **SMC**
- **DPCLS**
- **DPIF**
- **Testpmd (csum fwd)**
- **Virtio**
- **DMA**
- **Actions (TX)**

Guest (VM)

- **DPDK vHost**
Offload types...

Useful Work

Offload Work (Descriptor/completion handling)

Completion Polling

Core

Accelerator

Synchronous Operation
– cycles wasted waiting for accelerator

Asynchronous Operation
– find other useful work to do while waiting on accelerator to avoid wasted cycles.
Datapath Design - Southbound

1) Dequeue_burst
2) Poll_completions (Guest)

VirtQ

(Guest)
Datapath Design - Northbound

1) vHost_TX
2) Poll Completions

How and when the jobs completed
Datapath Design – Southbound (altered)

OVS

1) vHost_RX

DMA HW CHOICE

2) Poll Completions for previous TX

VHost Library

1.a) Dequeue_burst
1.b) Poll_completions

(Guest)

2.a) Poll_completions
Mapping DMA to OVS?
Per Dataplane Thread

= Scalability!

Add PMD thread

= Add DMA resource.

Balanced Switch/Copy performance.

https://patches.dpdk.org/project/dpdk/patch/20211122105437.3534231-2-jiayu.hu@intel.com/
Packet walkthrough

OVS Datapath

Miniflow extract
DPDK
Virtio
Virtio

Actions (TX)

Testpmd (csum fwd)

DMA

Vhost_TX [DMA submit]

NIC_RX – Vhost_TX [DMA submit]
Packet walkthrough

DMACopy phase (TX)

NIC_RX – Vhost_TX [DMA submit]

OVS Datapath

Miniflow extract

DPDK

EMC

SMC

DPCLS

Actions (TX)

DPDK

vHost

Virtio

Testpmd (csum fwd)

DMA

DPIF

Guest (VM)

DMA copy phase (TX)

NIC_RX – Vhost_TX [DMA submit]
Packet walkthrough

Vhost_RX
[DMA submit] + [Poll completed]
+ Vhost_TX_Poll_completed

DMA copy phase(TX)
Packet walkthrough

Vhost_RX
[DMA submit] + [Poll completed]
+ Vhost_TX_Poll_completed
Packet walkthrough

OVS Datapath
Miniflow extract
EMC
SMC
DPCLS
Actions (TX)
DPDK
NIC RX
NIC HW
DPDK
vHost
Virtio
Testpmd (csum fwd)
DMA

Guest (VM)

DMA copy phase (RX)

Guest Processing

DPIF

Guest (VM)
 DMA
 Testpmd (csum fwd)

DMA copy phase (RX)

Guest Processing
Packet walkthrough

Vhost_RX
[DMA submit] + [Poll completed] + Vhost_TX_Poll_completed [No packets]
Packet walkthrough

Vhost_RX – NIC_TX

DMA copy phase(RX)
Packet walkthrough

Vhost_RX
[DMA submit] + [Poll completed] + Vhost_TX_Poll_completed
Packet walkthrough

Vhost_RX – NIC_TX
Next steps

- Upstream Design after community review

Open vSwitch 2.18
POC

Open vSwitch 2.19
UPSTREAM
Thanks!

Questions?

Sunil Pai G
sunil.pai.g@intel.com

Cian Ferriter
cian.ferriter@intel.com

Harry van Haaren
harry.van.haaren@intel.com
Enabling asynchronous Para-virtual IO in OVS, OVS conference 2020

- Talk: https://www.youtube.com/watch?v=jj-0OQLe2oU

V2 RFC for vHost async implementation in OVS:
- http://patchwork.ozlabs.org/project/openvswitch/patch/20210907120021.4093604-2-sunil.pai.g@intel.com/

V1 RFC for vhost async implementation with DMAdev in vHost library in DPDK:
- https://patches.dpdk.org/project/dpdk/patch/20210823095355.2478423-2-sunil.pai.g@intel.com/
- https://patches.dpdk.org/project/dpdk/patch/20211122105437.3534231-2-jiayu.hu@intel.com/

Previous presentations in DPDK conferences:
- Asynchronous CBDMA Enqueue Framework for vHost-User, 2019
- Accelerating Para-Virtual I/O with CBDMA, 2018
BACKUP
Why Defer of Work is Needed

- Packet copy into guest can be slow
  - Accelerate this using DMA Engine

- Don’t stall the CPU while the DMA Engine is active
  - Asynchronous acceleration requires defer of work
    - Call DPDK APIs to start transfer
    - Check back later for completion

Asynchronous work achieved
Where Defer of Work Fits into OVS

```
Where Defer of Work Fits into OVS

dpif-netdev  netdev  work defer

pmd_thread_main loop

call process Rxq port

netdev_rxq_recv  miniflow_extract  EMC/SMC/DPCLS  flush_output  netdev_send

return 0

delay work item

return -EINPROGRESS

process each RXQ in poll list

All RXQs processed

async work? yes

netdev process async

no
```
• Para-virtual I/O is a virtualization technique to enhance VM I/O performance.

• **VirtIO** is a standard of para-virtual I/O, which consists of **VirtIO front-end in VM** and **backend in hypervisor**.

• Back-end communicates with front-end by **copying packet buffers** between hypervisor's and VM's memory.

• **Copying large bulk of data** between backend and frontend becomes a **hotspot**
**vHost async API's (vHost Library)**

Northbound API's [Host to Guest]
- `rte_vhost_submit_enqueue_burst`
- `rte_vhost_poll_enqueue_completed`

Southbound API's [Guest to Host]
- `rte_vhost_async_try_dequeue_burst`

/* DMA callbacks */
- `struct rte_vhost_async_channel_ops {
  transfer_data(...);
  check_completed_copies(...);
};`

- `rte_vhost_async_channel_register_thread_unsafe`
- `rte_vhost_async_channel_unregister_thread_unsafe`
- `rte_vhost_async_get_inflight`
- `rte_vhost_clear_queue`

- set `RTE_VHOST_USER_ASYNC_COPY` capability during `rte_vhost_driver_register`

Note: **All vHost async and DMAdev API’s are experimental**

RFC based on above API set: [http://patchwork.ozlabs.org/project/openvswitch/list/?series=261277](http://patchwork.ozlabs.org/project/openvswitch/list/?series=261277)
Datapath Design - Southbound

1) Dequeue_burst
   Enqueue()
2) Poll_completions

OVS
vHost_RX
VHost Library
DMA HW

DMA HW

DMA HW

DMa-Dev DPDK API

VirtQ

(Guest)
Datapath Design - Northbound

OVS

1) vHost_TX

DMA HW

CHOICE

2) Poll Completions

Enqueue_burst

Poll_completions

DMA-Dev DPDK API

Enqueue()

Check Completions

Guest

How and when the jobs completed

Poll TX from RX context
Datapath Design – Southbound (altered)

1) vHost_RX

VHost Library

a) Dequeue_burst
b) Poll_completions

Check Completions

(Guest)

2) Poll Completions for previous TX

DMA HW

CHOICE

OVS

DMA-Dev DPDK API

Enqueue()

Check Completions

DMA HW

DMA HW

VirtQ