vDPA : On the road to production

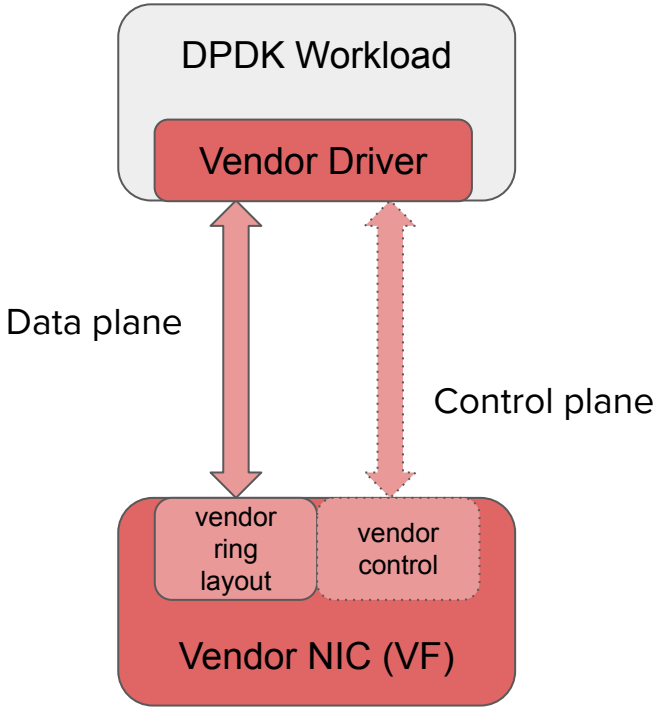
Maxime Coquelin (Red Hat)
Adrián Moreno (Red Hat)

Red Hat

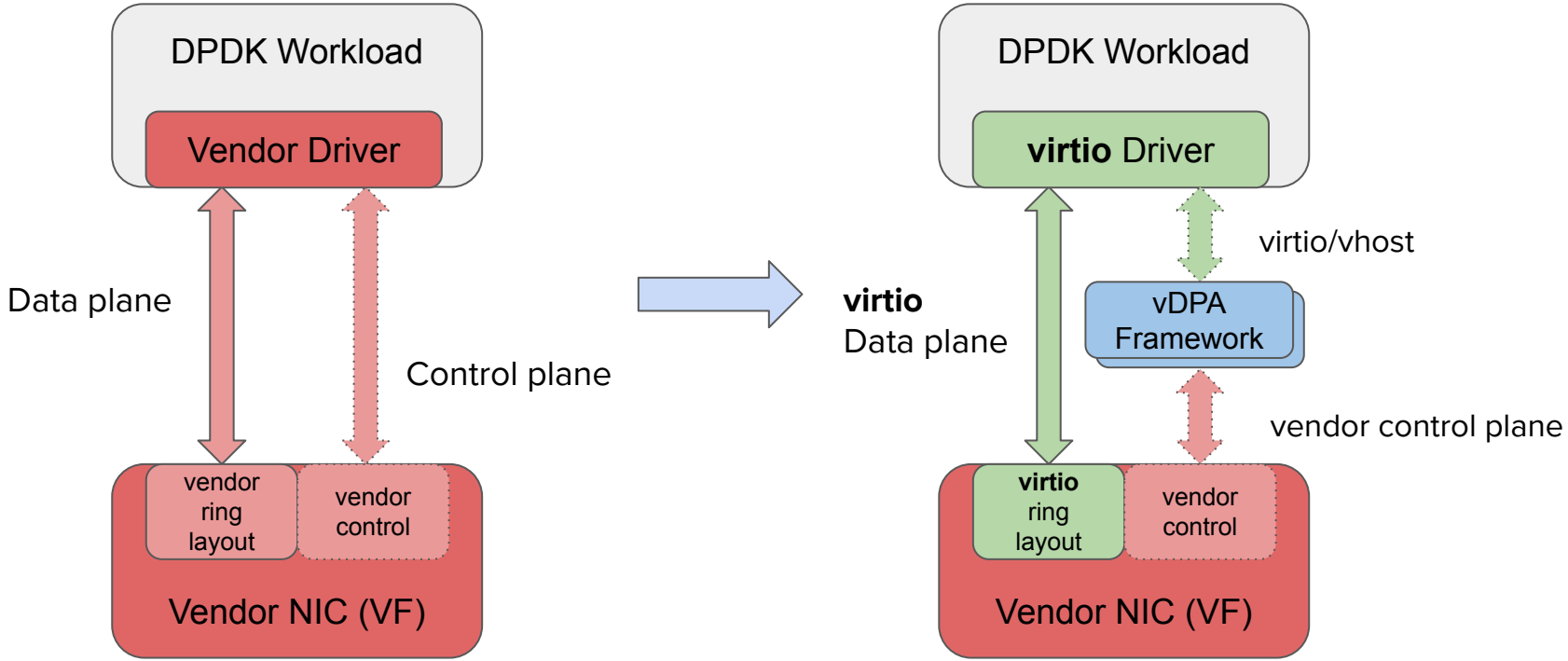
Agenda

- vDPA technology overview
- Kernel vDPA overview
- DPDK vDPA overview
- Virtio-user PMD updates
- DPDK framework updates
- DPDK vDPA daemon
- vDPA looking forward
- Q&A

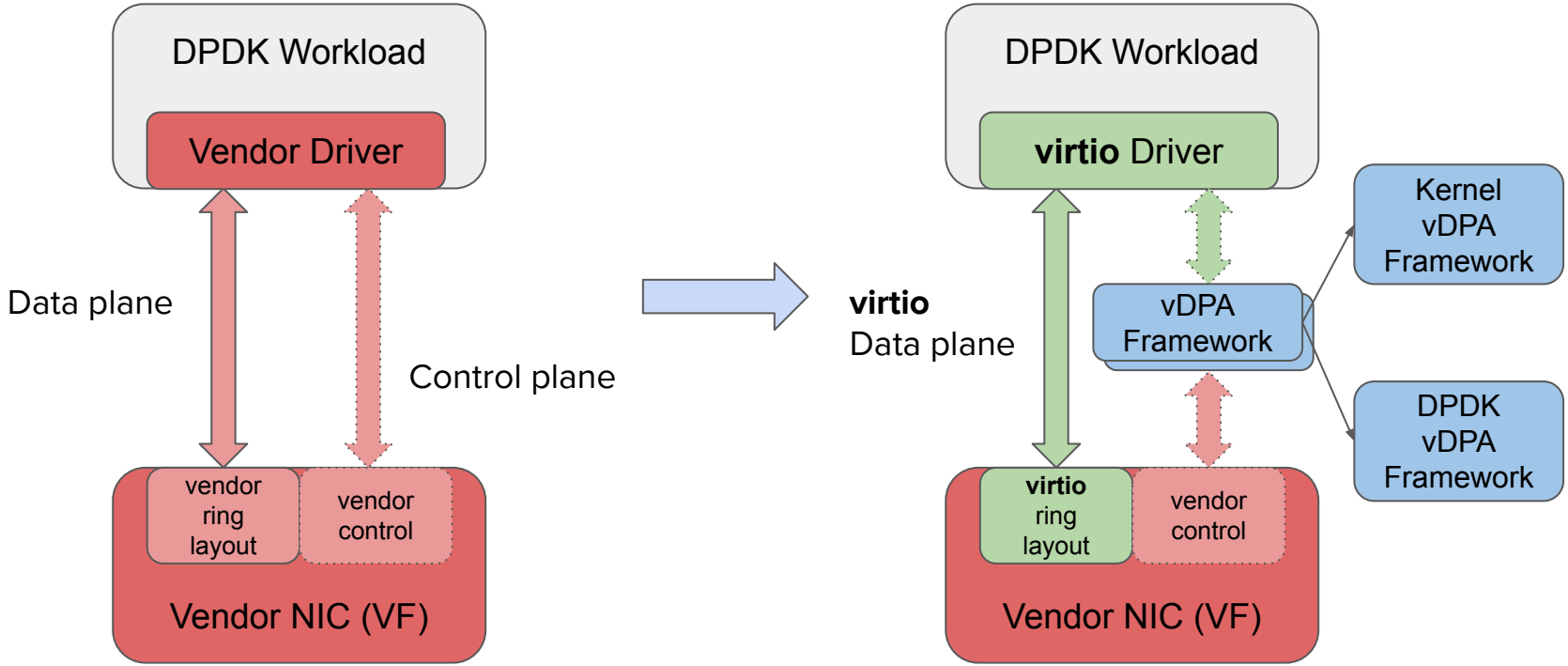
vDPA: Virtio DataPath Acceleration



vDPA technology overview

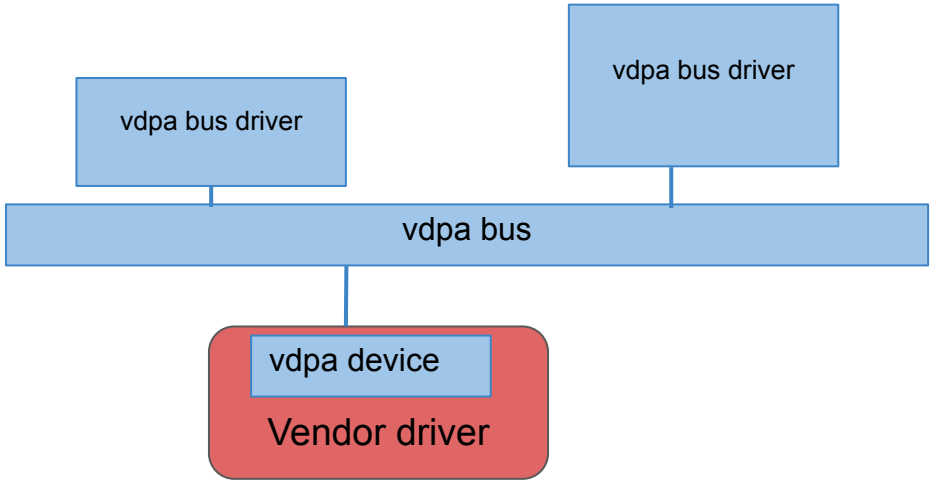


vDPA technology overview



Red Hat

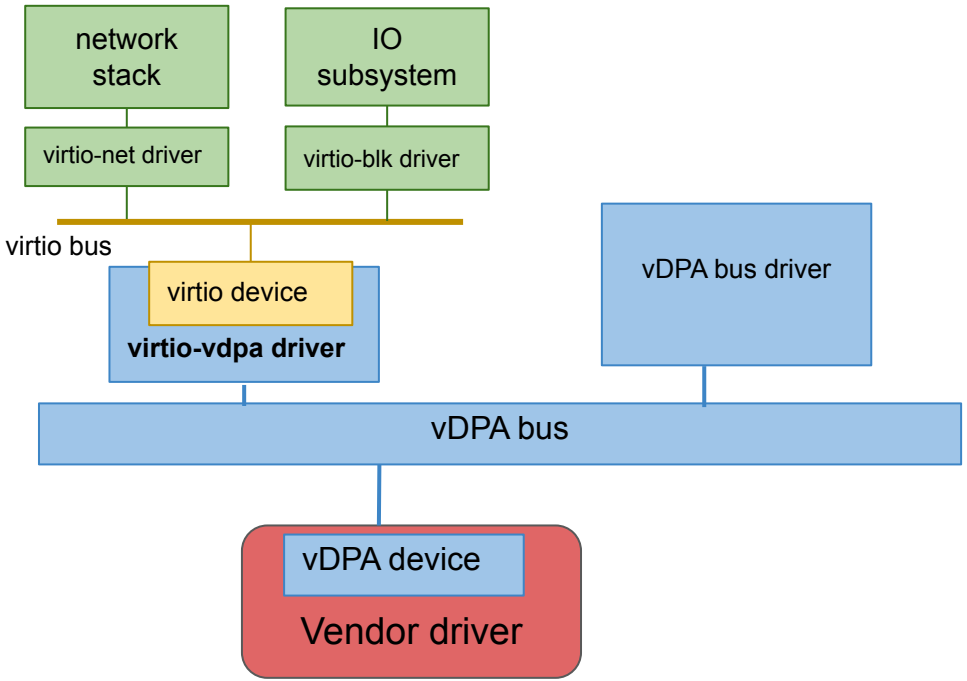
Kernel vDPA overview: the vDPA bus



Vdpa bus operations

- Virtio specific ops
- Interrupt management
- Doorbell mapping
- Migration helpers
- DMA operations

Kernel vDPA overview: the virtio-vdpa driver

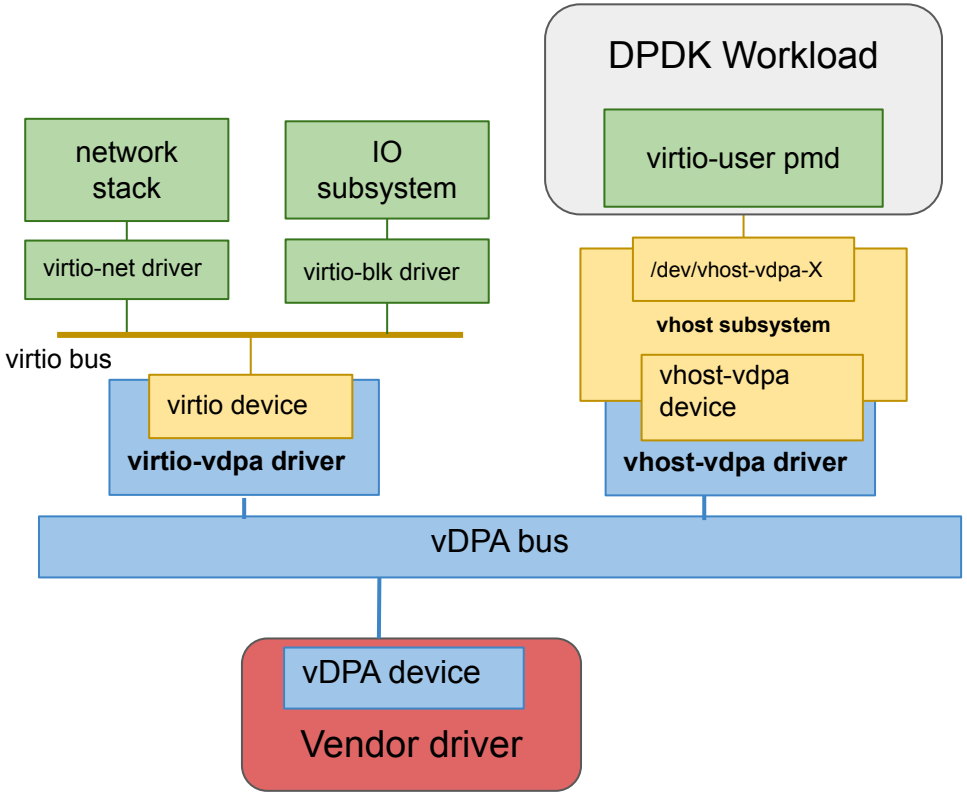


New Virtio-vdpa transport

- Implement virtio-bus ops
- vDPA devices probing and registration
- Interrupt forwarding

Red Hat

Kernel vDPA overview: the vhost-vdpa driver



vhost-vdpa driver

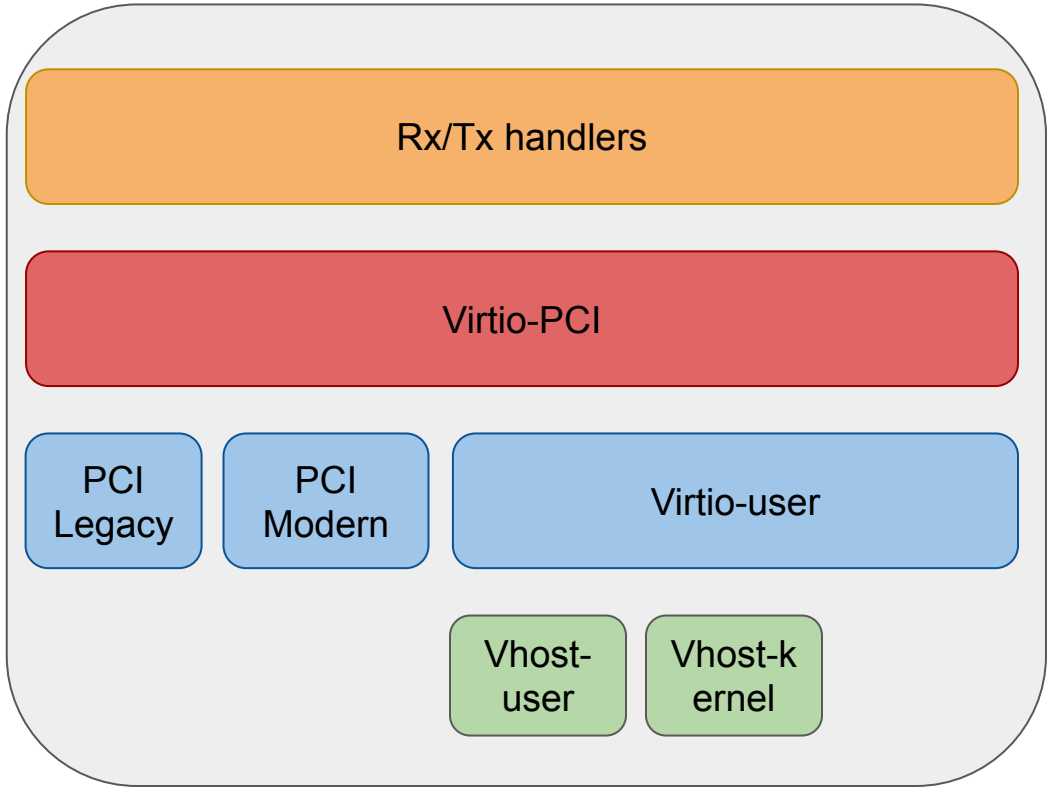
- Supports both on-chip and platform IOMMU
- Reuses most of vhost-net ioctl uAPI
- Uses vhost IOTLB uAPI (char device read/write)
- Some vdpa-specific extensions added to uAPI

Red Hat

Vhost-vdpa uAPI Extensions

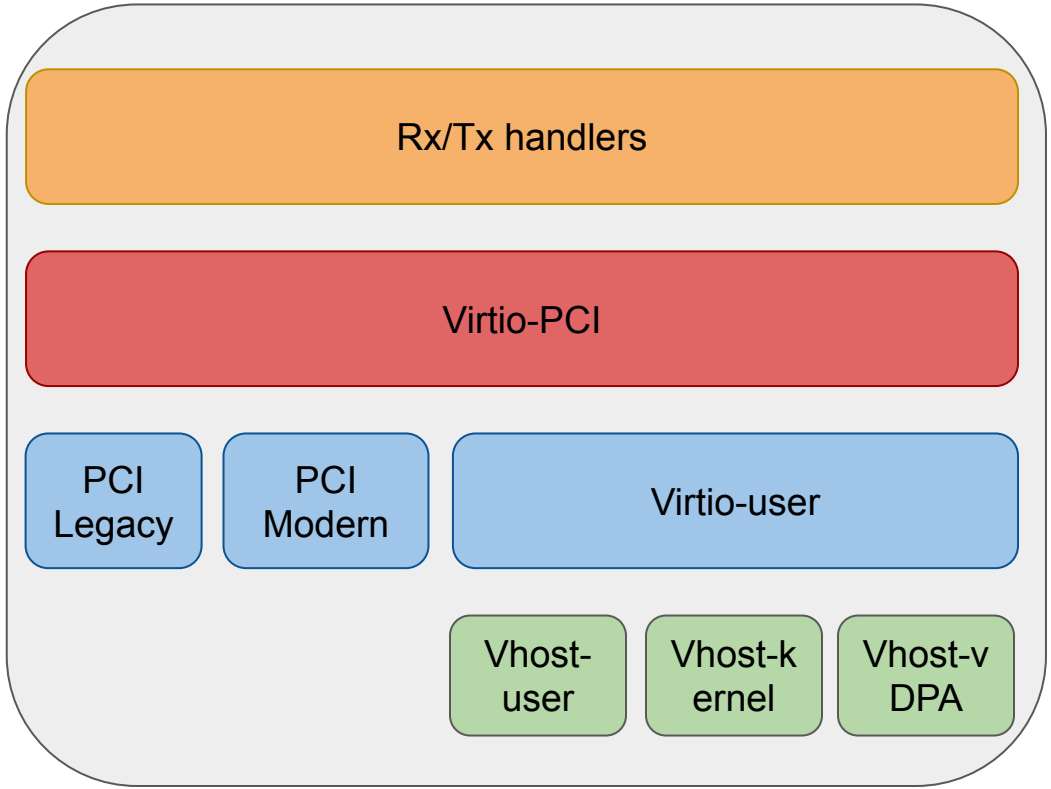
- VHOST_VDPA_GET_DEVICE_ID
  - E.g: match a specific virtio device type
- VHOST_VDPA_{GET,SET}_STATUS
  - E.g: start / stop the device
- VHOST_VDPA_{GET,SET}_CONFIG
- VHOST_VDPA_SET_CONFIG_CALL
  - vDPA devices can emulate or relay config interrupts via eventfd
- Doorbell mapping: mmap()
  - The doorbell register is located at page boundary and does not share page with other registers

Virtio-user PMD updates: Overview



- Virtio-user part of Virtio PMD
  - Shares same data-path
- Plugs into the Virtio-PCI layer
- Currently supports two backends
  - Vhost-user
  - Vhost-kernel

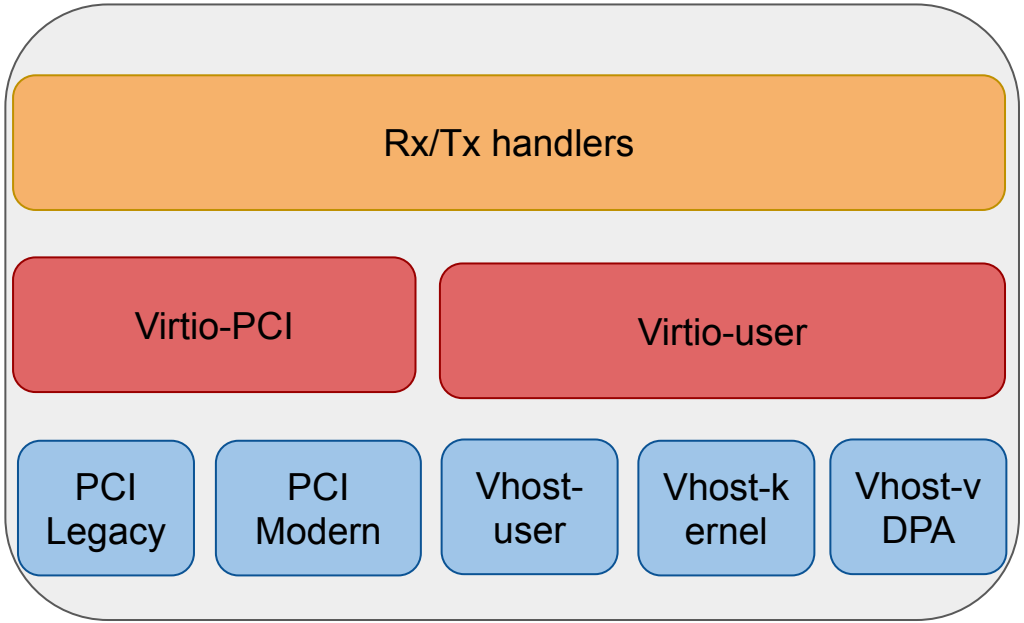
Virtio-user PMD updates: Vhost-vDPA



- Vhost-vDPA backend support
  - Targets DPDK v20.11
  - V1 posted
- Implements the frontend side of the Vhost-vDPA uAPI
- No code sharing with Vhost-Kernel for now

Red Hat

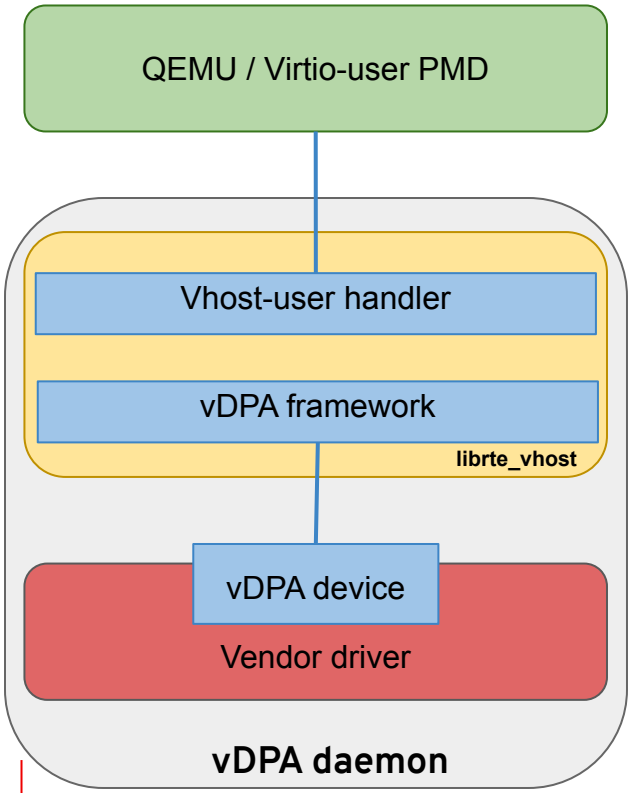
Virtio-user PMD updates: Refactoring



Virtio PMD

- Virtio PMD refactoring
  - Targets DPDK v21.XX
- New layer for bus operations
- Virtio-user layer with generic ops
- Code sharing between Vhost-Kernel and Vhost-vDPA

DPDK vDPA overview



- Introduced in DPDK v18.05
- Relies on Vhost-user protocol
  - Vhost-user requests to vendor control plane
  - Seamless migration from SW backend
- Part of DPDK Vhost-library
  - Reuse of Vhost-user protocol handling
- Live-migration support
  - SW assisted if not supported in HW

DPDK vDPA overview: API

- User-facing API for application
  - Bind/unbind vDPA device to a Vhost-user socket
  - Query vDPA devices information
    - Supported Virtio and Vhost protocol features
    - Number of queue pairs supported by the device
    - ...
  - vDPA device statistics collection and reset

DPDK vDPA overview: API

- Driver API
  - vDPA device registration
  - Set of operations implemented by the driver
    - vDPA device start/stop
    - Queue pairs enablement
    - Virtio and Vhost-user protocol features setting
    - Doorbell mapping
    - Statistics collection/reset
  - SW-assisted live-migration

DPDK framework updates

- Statistics collection API addition
  - Generic Virtio counters (e.g. number of descriptors processed)
  - Extended statistics support for vendor-specific counters
- vDPA is now a DPDK device class
  - RTE_DEV_FOREACH(dev, "class=vdpa", &dev_iter) {...}
- Framework is now bus-agnostics (only PCI before)
  - Relies on rte_device
- Framework internals and API simplified (No more vDPA device ID)
- User & driver API split to avoid layers violations
  - struct rte_vdpa_device no longer exposed to the user application

DPDK vDPA daemon

- Currently, only application using vDPA API is DPDK's vdpa example
  - Good for testing, but not meant for production
- New userspace-vdpa daemon development initiated (**uvdpad**) [1]
  - Exposes a JSON-RPC API to management layers (k8s, OpenStack)
  - Client implementation in Go (both library and CLI tool) [2]
  - Device hot-plug and hot-remove
- Still Alpha stage

[1] https://gitlab.com/mcoquelin/userspace-vdpa
[2] https://github.com/amorenoz/govdpa

DPDK vDPA daemon: API

- JSON-RPC v2.0 API currently exposed by the daemon
  - *list-interfaces*
  - *create-interface*
  - *destroy-interface*
  - *list-methods*
  - *version*
- More to come
  - Statistics collection
  - Device configuration (MAC,...)

vDPA looking forward

- Virtio-user PMD
  - IOVA range reporting & IOTLB Batching (for device with on-chip IOMMU)
  - Multiqueue support in Vhost-vDPA backend
  - Virtio PMD refactoring
- DPDK vDPA Framework
  - K8s & Openstack integration
  - VHOST_USER_PROTOCOL_F_VDPA (“vhost-user 2.0”)
  - Virtio-vDPA driver
  - Control queue interception / relaying

Q&A

- Virtio networking blogs https://www.redhat.com/en/virtio-networking-series
- Virtio networking community mailing list virtio-networking@redhat.com