New File Accelerator and Recovery Feature in SPDK Vhost

Changpeng Liu, Intel

Xiaodong Liu, Intel

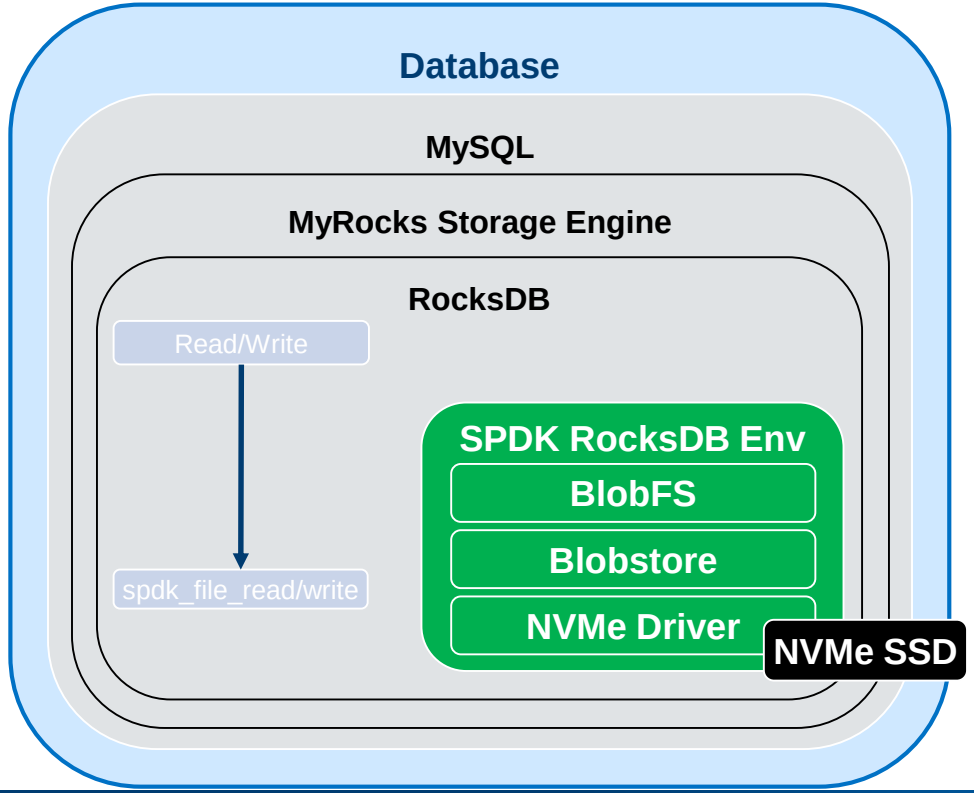
Jin Yu, Intel

Agenda

- SPDK Vhost-fs
- SPDK Vhost Live Recovery

SPDK Vhost-fs

Application Acceleration (Local Storage)

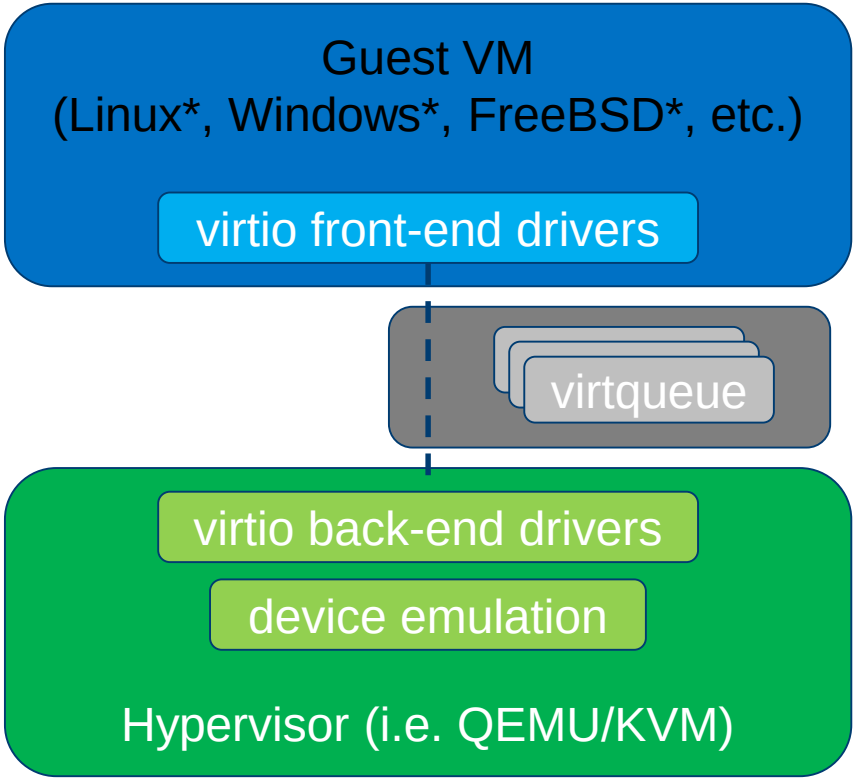


Implementation of RocksDB “env” abstraction

- Drop-in storage engine replacement
- Accelerate application access to local storage
- Benefits: removes latency and improves I/O consistency

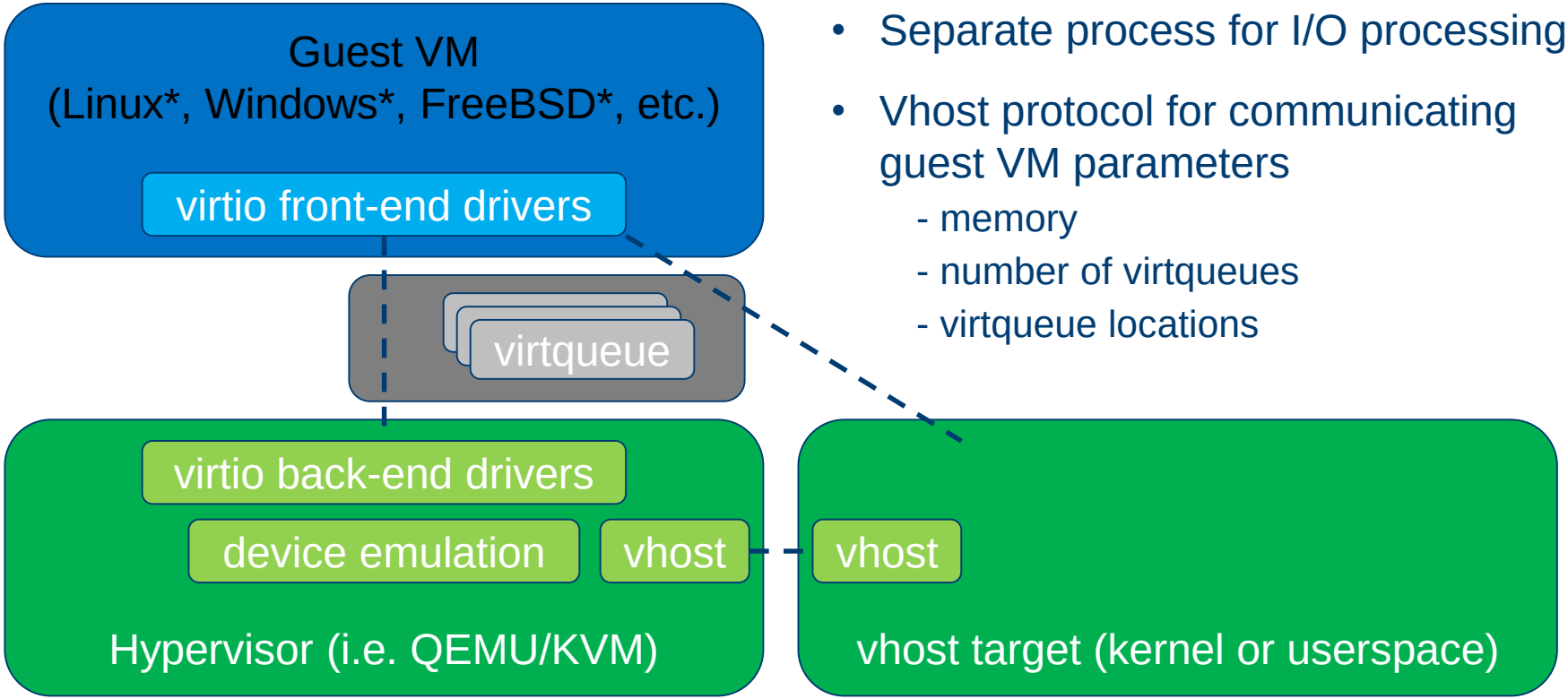
What if running RocksDB in a virtual environment? Is there any protocol can transfer file APIs between VM and Host ?

virtio



- Paravirtualized driver specification
- Common mechanisms and layouts for device discovery, I/O queues, etc.
- virtio device types include:
  - virtio-net
  - virtio-blk
  - virtio-scsi
  - virtio-9p
  - virtio-fs

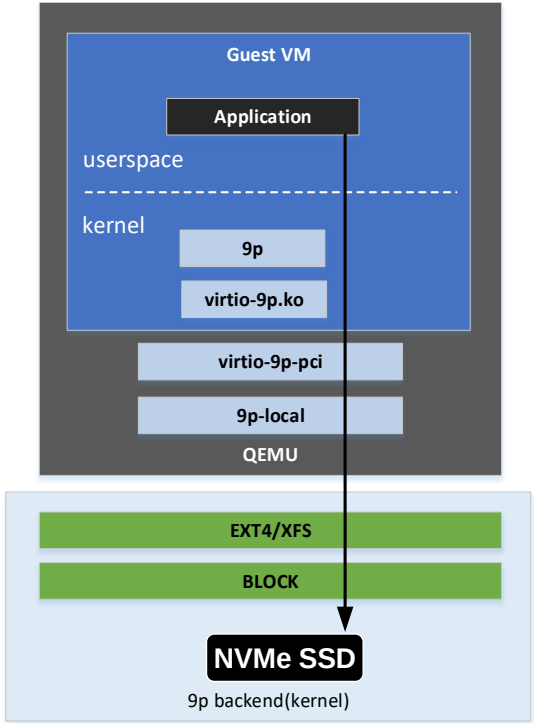
vhost

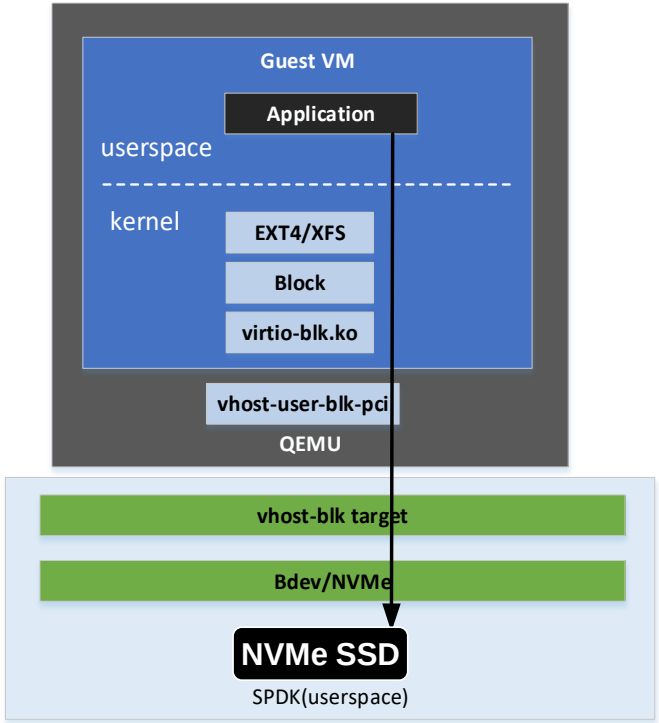


- Separate process for I/O processing
- Vhost protocol for communicating guest VM parameters
  - memory
  - number of virtqueues
  - virtqueue locations

Optional solutions using file APIs in VM

- Using 9p as the file transport protocol
- Format file system with block device



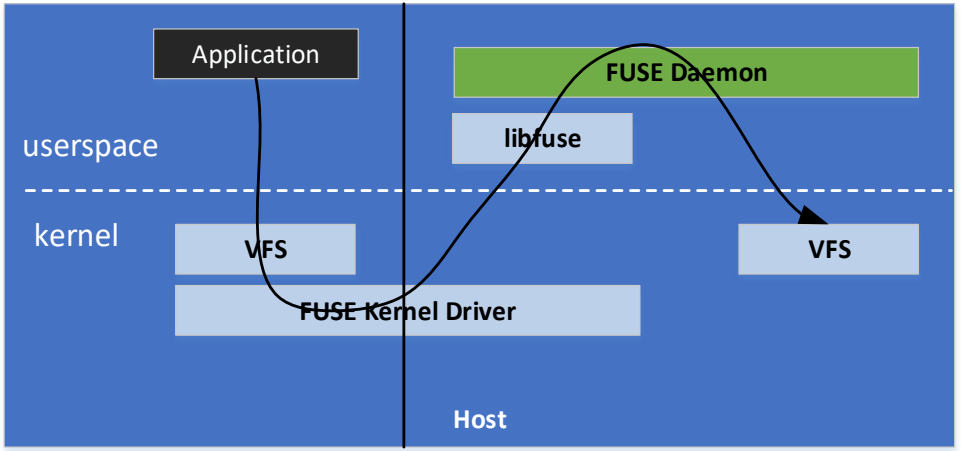


Introduction to FUSE

- FUSE (Filesystem in Userspace) is an interface for userspace programs to export a filesystem to the Linux kernel.
- The FUSE project consists of two components:

  - fuse kernel module and the libfuse userspace library.

  - libfuse provides the reference implementation for communicating with the FUSE kernel module.



Example usage of FUSE(passthrough)

Virtio-fs

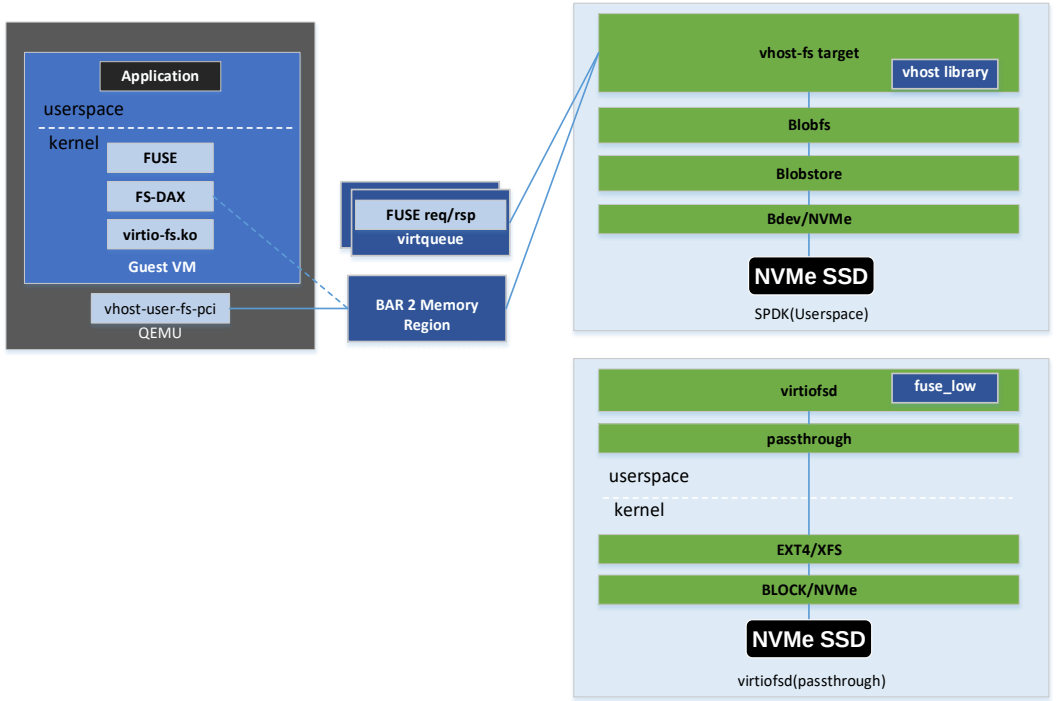
- virtio-fs is a shared file system that lets virtual machines access a directory tree on the host. Unlike existing approaches, it is designed to offer local file system semantics and performance. This is especially useful for lightweight VMs and container workloads, where shared volumes are a requirement.
- virtio-fs was started at Red Hat and is being developed in the Linux, QEMU, FUSE, and Kata Containers communities that are affected by code changes.
- virtio-fs uses FUSE as the foundation. A VIRTIO device carries FUSE messages and provides extensions for advanced features not available in traditional FUSE.
- DAX support via virtio-pci BAR from host huge memory.

SPDK Vhost-fs Target vs. Virtiofsd

- Eliminate userspace/kernel space context switch by providing a user space file system
- IO thread model

  - SPDK uses one poller to poll all the virtqueues while virtiofsd uses one thread per queue

- Page cache in Host can be shared for virtiofsd
- Easy to add new features in userspace



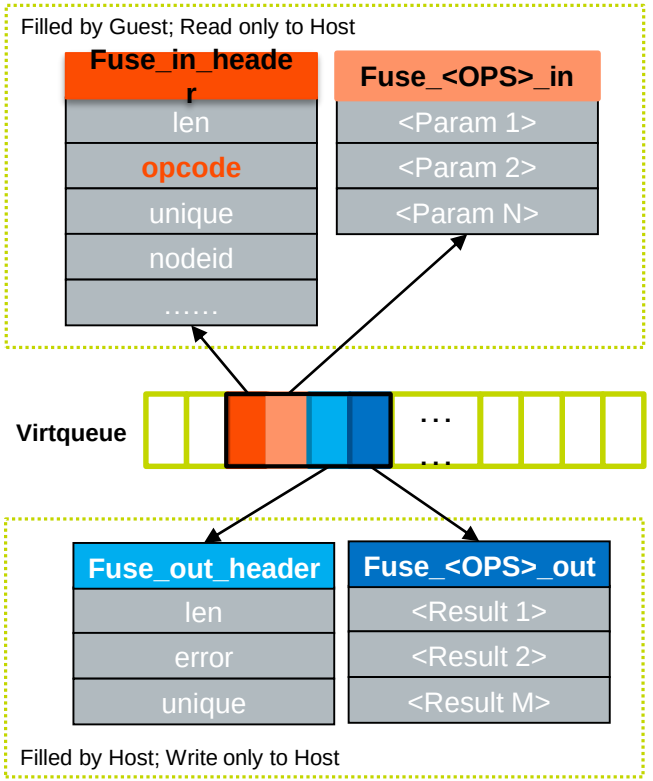
SPDK Blobfs APIs vs. FUSE

- Open, read, write, close, delete, rename, sync interface to provide POSIX similar APIs
- Asynchronous APIs provided
- Random write support ?
- Memory mapped IO support ?
- Directory semantic support ?

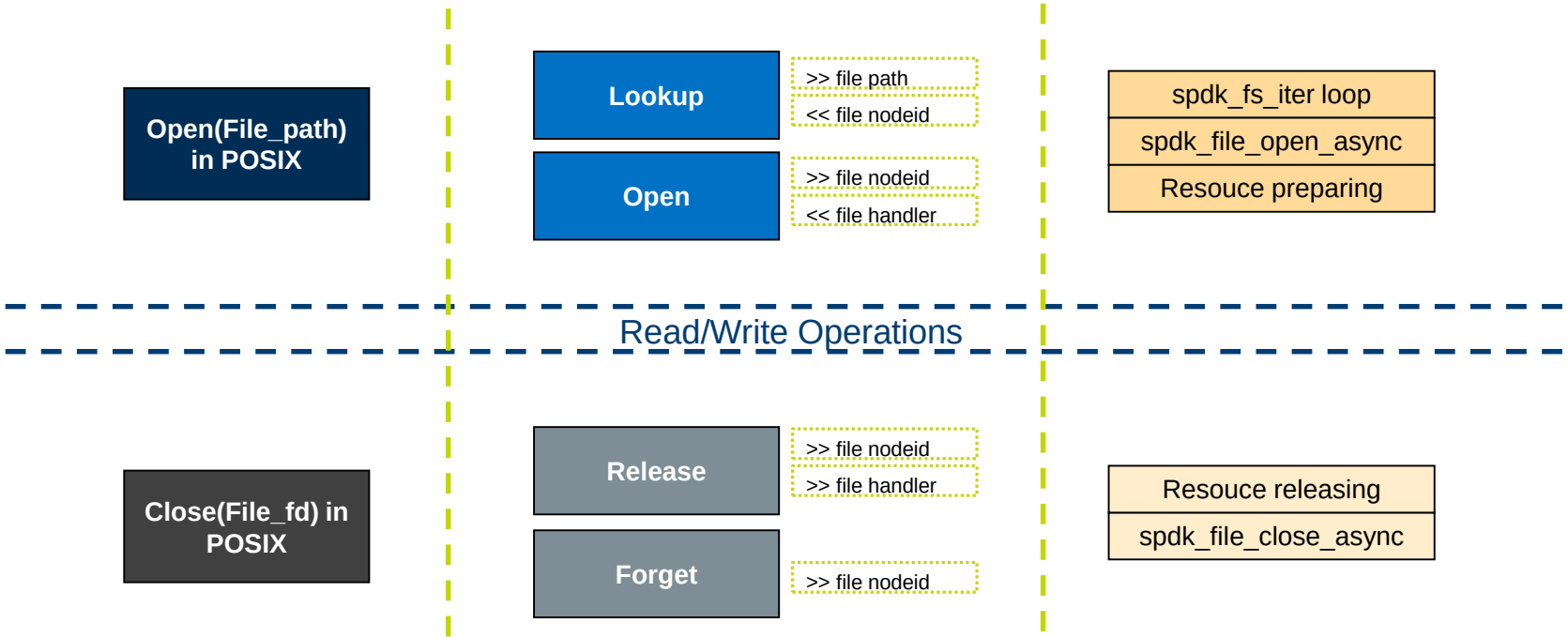
FUSE Command	Blobfs API
Lookup	spdk_fs_iter_first, spdk_fs_iter_next
Getattr	spdk_fs_file_stat_async
Open	spdk_fs_open_file_async
Release	spdk_file_close_async
Create	spdk_fs_create_file_async
Delete	spdk_fs_delete_file_async
Read	spdk_file_readv_async
Write	spdk_file_writev_async
Rename	spdk_fs_rename_file_async
Flush	spdk_file_sync_async

Operation Mapping of FUSE in Virtqueue

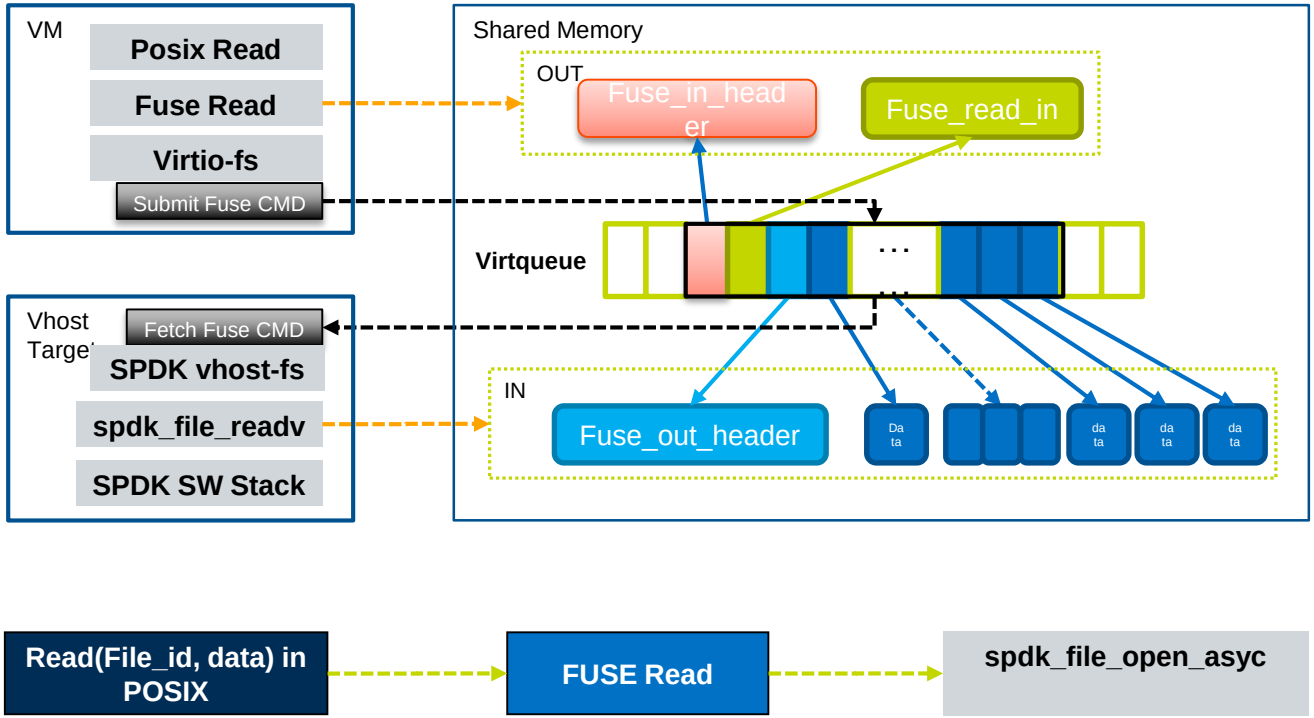
- General FUSE command has 2 parts: request and response.
- General FUSE request is consisted with IN header and operation specific IN parameters.
- General FUSE response is consisted with OUT header and operation specific OUT results.



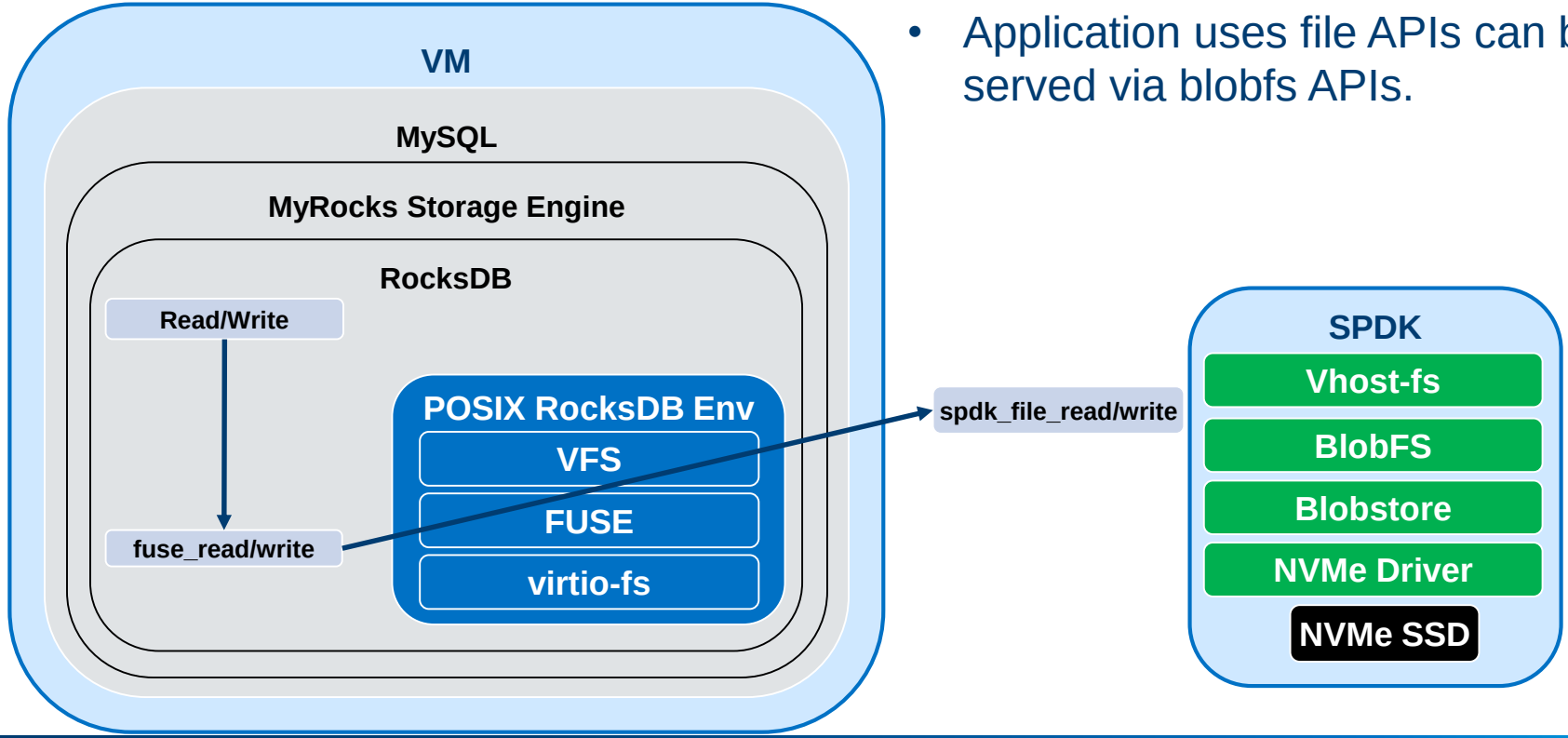
Open and Close Operations in FUSE and SPDK



Implementation Details with Read/Write



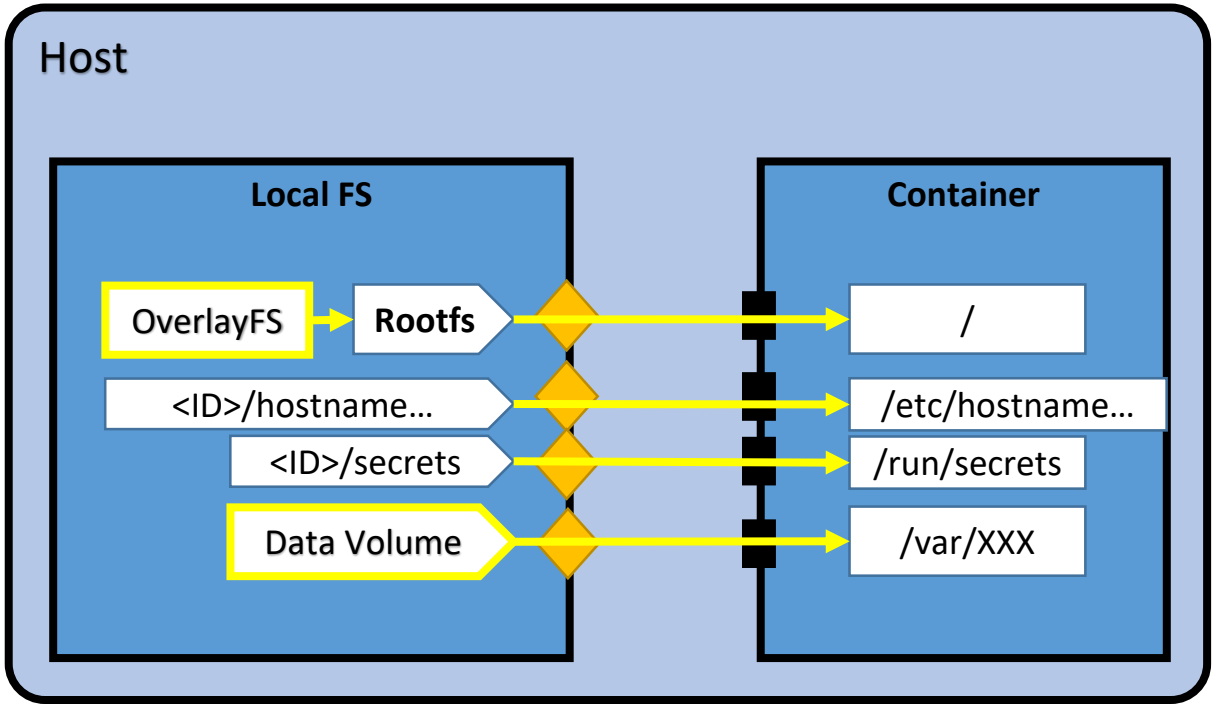
Application Acceleration in VM



- Application uses file APIs can be served via blobfs APIs.

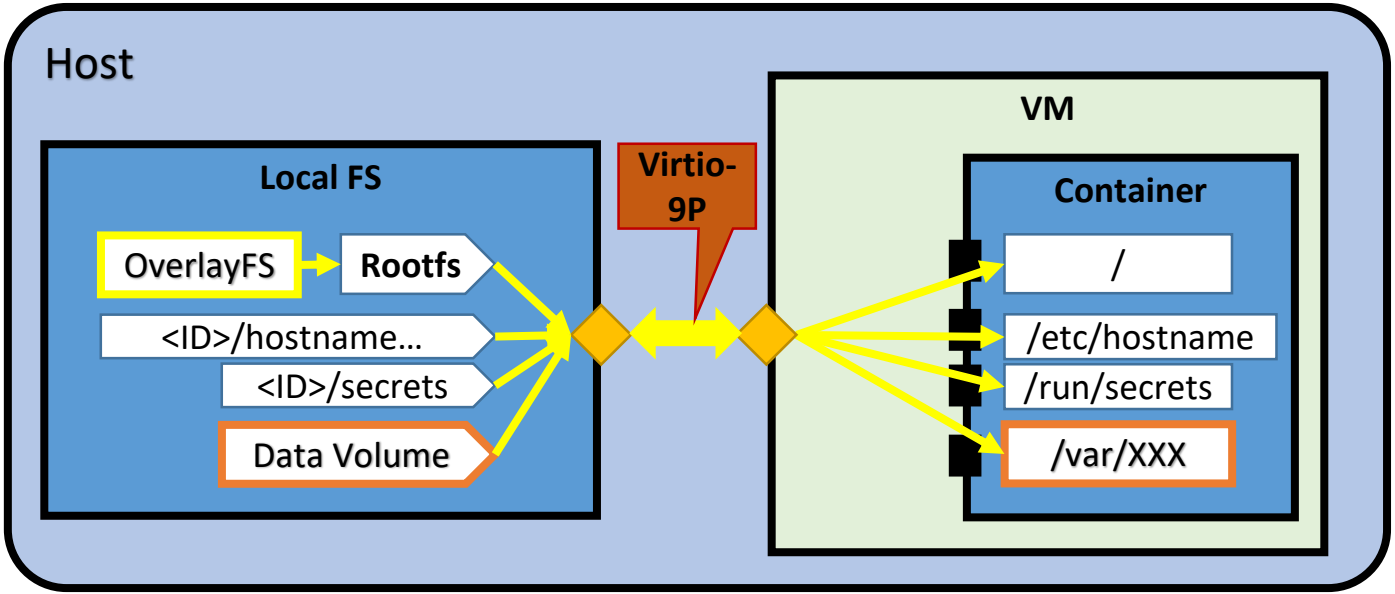
Brief View on Container Storage

- Isolation
- Layered rootfs
- Kinds of identification files
- Data volume for persistence.



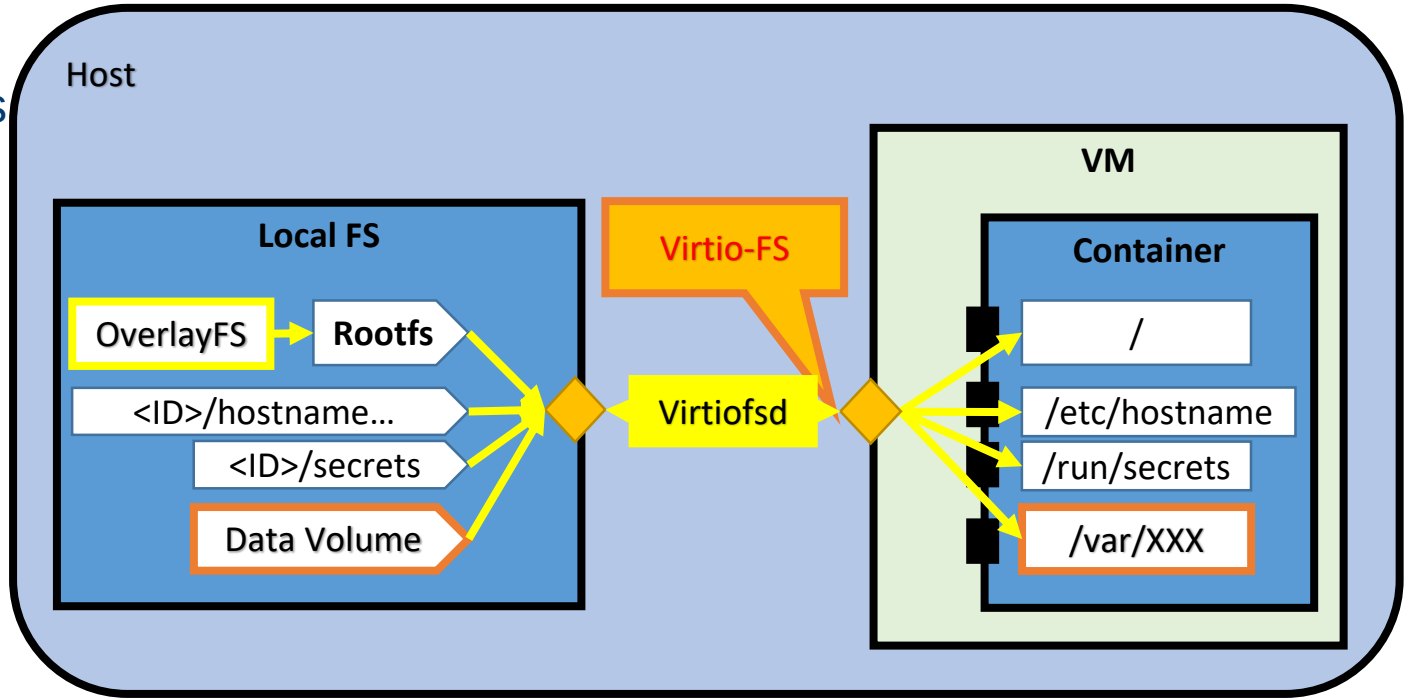
Brief View on Kata Container Storage

- VM gives better isolation for container
- Virtio-9P has been used as the transmission path between Host and Container



VirtioFS in Kata Container Storage

- Offer local file system semantics and performance
- Virtiofsd daemon handles VM request
- Virtiofsd daemon performs IO with file system calls

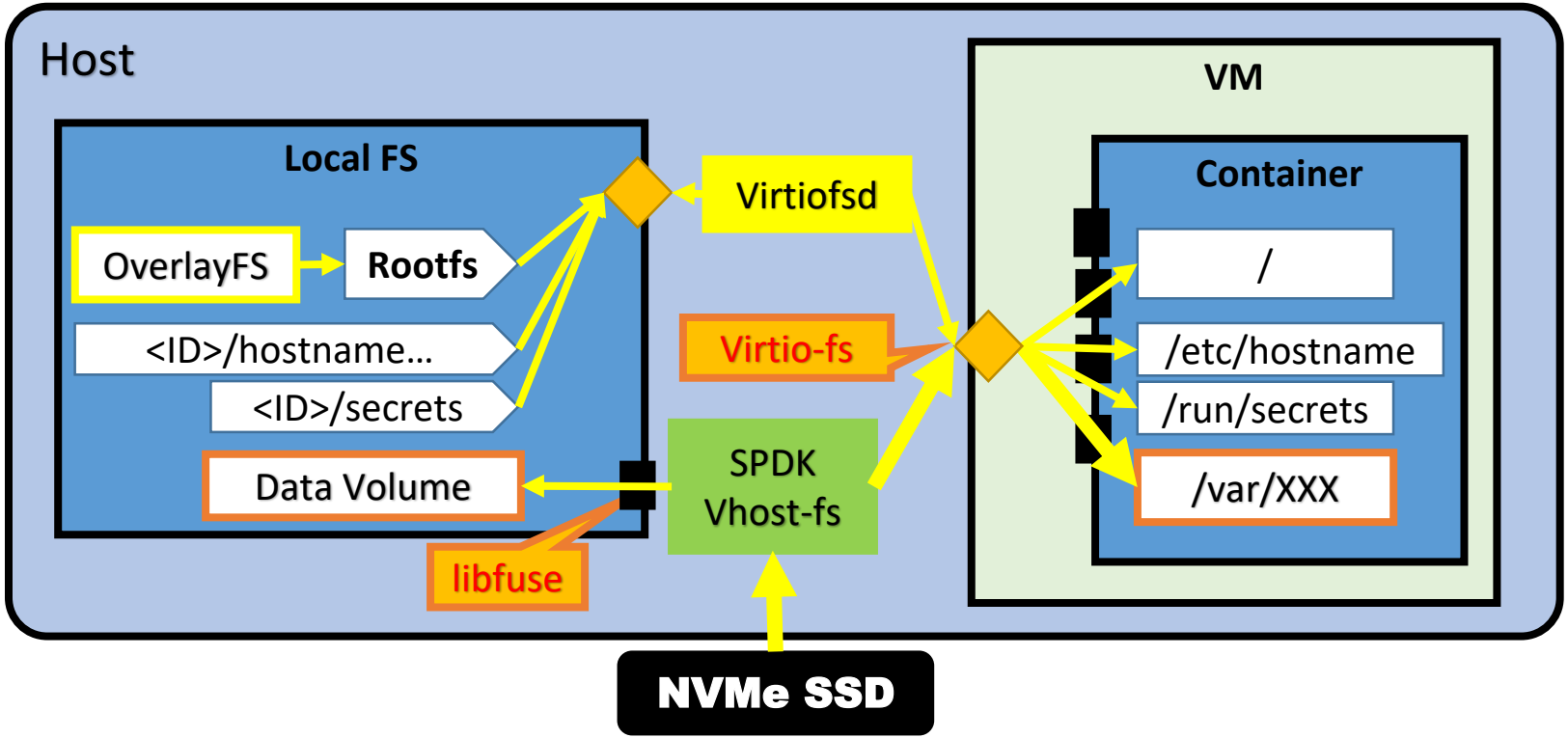


Kata-container

- The challenge when using with Kata-container
  - Shared file system is required for Kata-container
  - Overlay file system for container image
  - No directory view from Host side when using SPDK vhost-fs

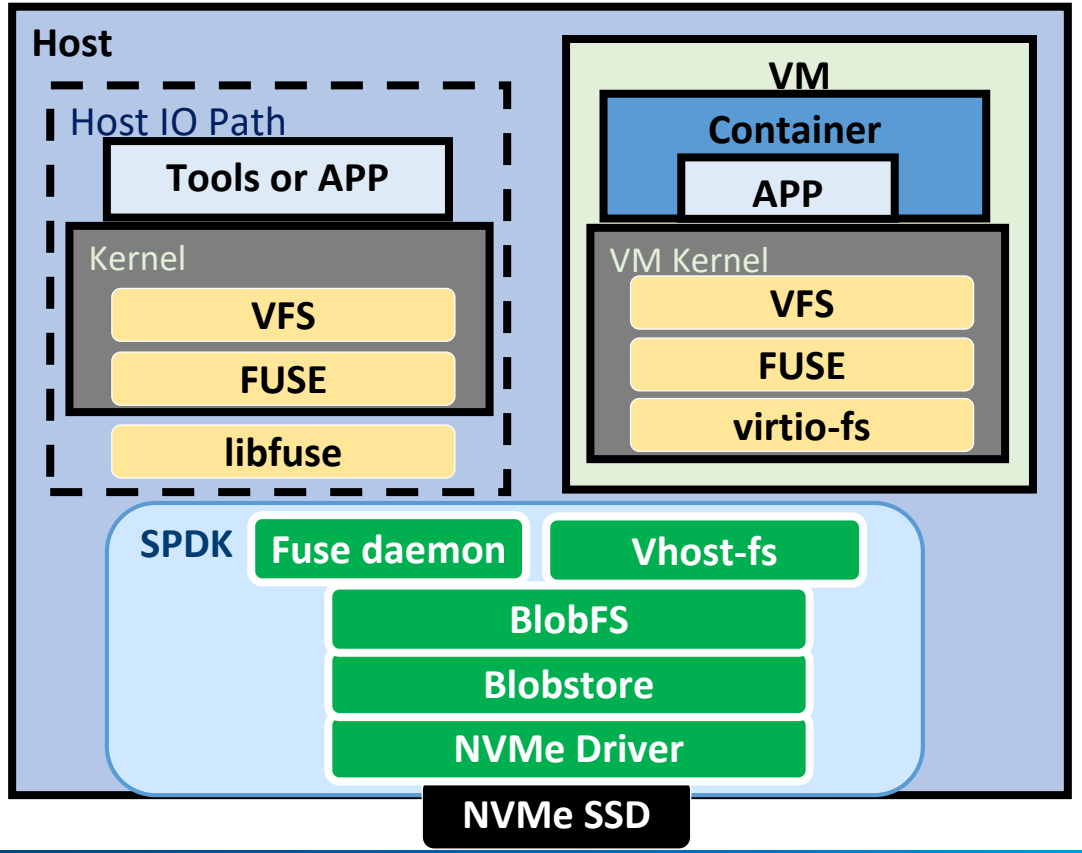
- How to use SPDK vhost-fs with Kata-container
  - Data volume can be used for shared data between different containers

SPDK vhost-fs in Kata Container Storage



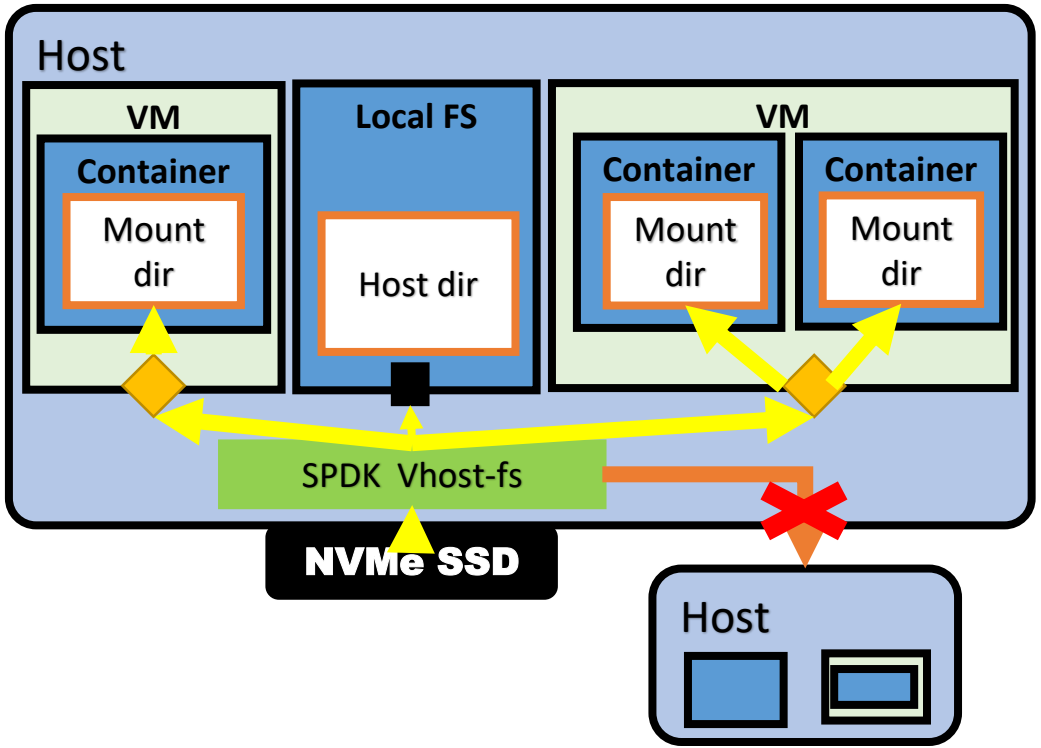
Software stack of vhost-fs for Kata container

- Vhost-fs for VM/container
- SPDK Fuse daemon for host



Sharing limitations for SPDK vhost-fs

- Sharing between Container and host
- Sharing between containers in different VM
- Sharing between containers in one VM
- How to sharing between containers in different host



SPDK Vhost Live Recovery

Background

- Baidu submit a patch that support SPDK vhost-blk live recovery.
- SPDK help push the patches to DPDK upstream as SPDK will abandon the internal rte_vhost lib（SPDK version >= 19.07）.
- SPDK add the packed ring support.

Baidu & SPDK

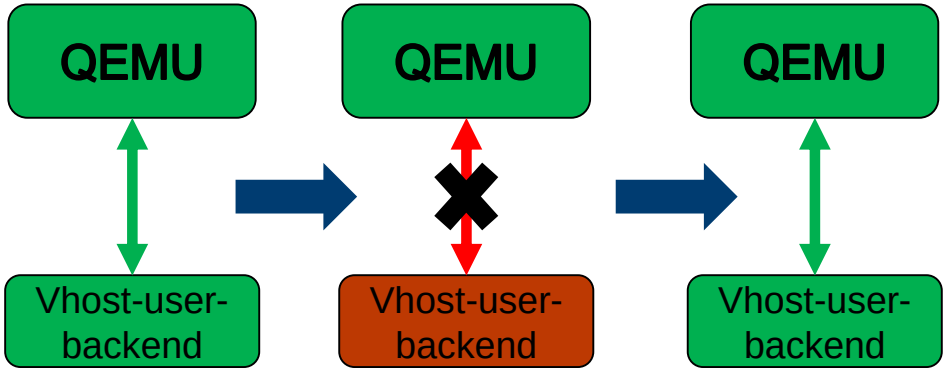
What is live recovery

Process status:

- VM is OK.
- I/Os are not lost

Requirements:

- Upgrade the Vhost backend



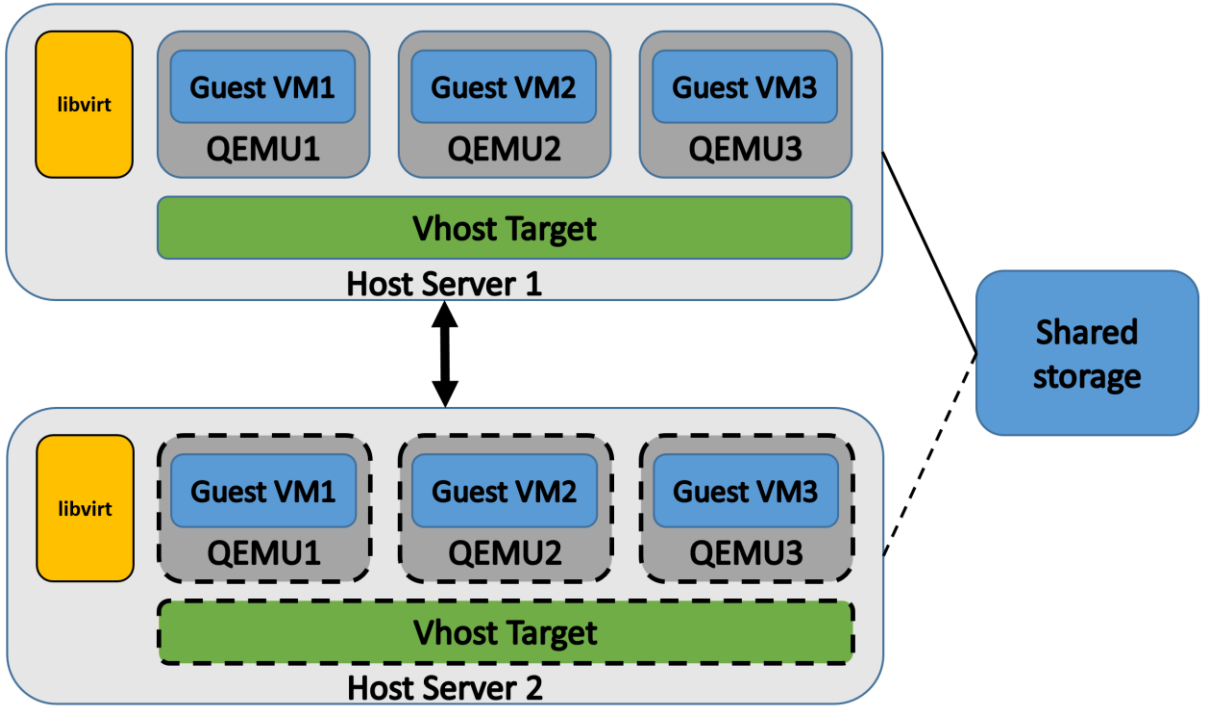
What is the previous solution

Previous solution:

- Live migration

Disadvantages:

- ❖ shared device
- ❖ shared storage



The SPDK solution

Implementation:

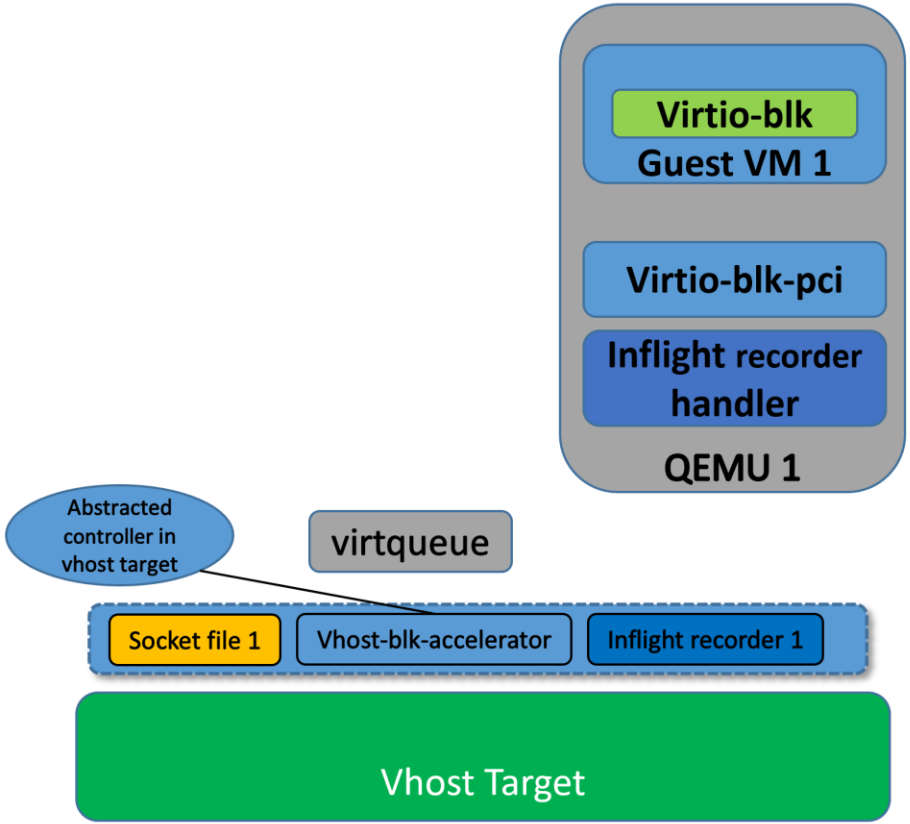
- Inflight recorder

Advantages:

✓ better and faster for upgrading

✓ less limitations

✓ no performance impact



How it works

Process:

1, allocate inflight buffer in vhost_tgt and send descriptor to QEMU

2, log all the outstanding requests into the buffer.

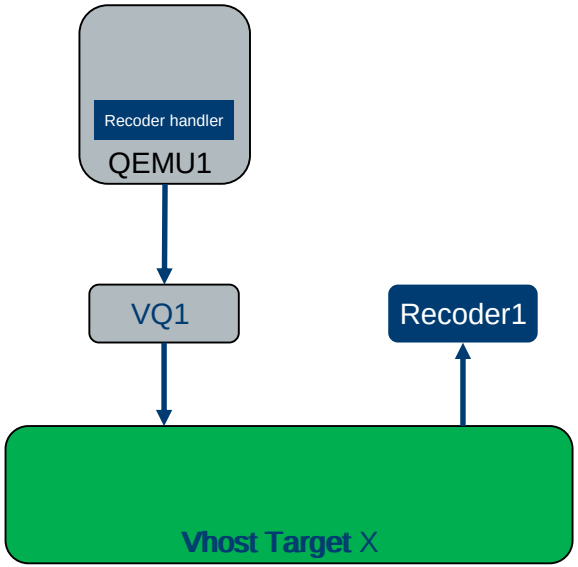
3, kill the vhost_tgt at any time.

4, after upgrading reconnect QEMU.

5, QEMU send descriptor to vhost_tgt

6, process outstanding reqs in inflight buffer

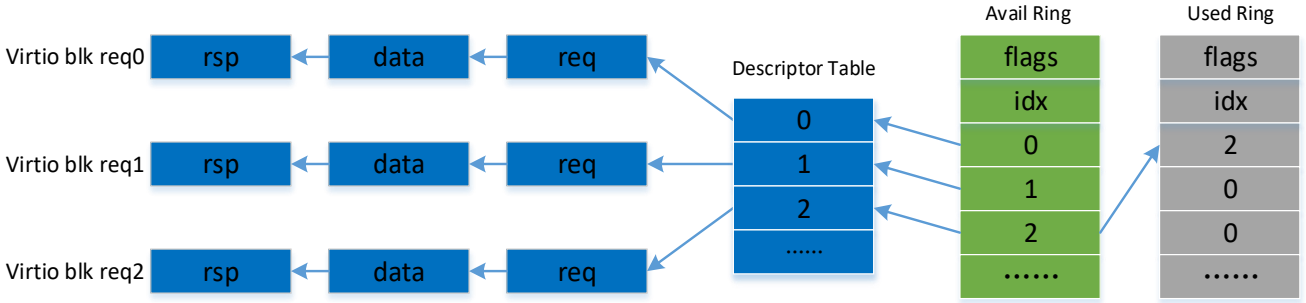
7, process coming requests.



Why we need inflight recorder

Reason:

- reqs may be not completed in order.



Patches

- SPDK Vhost-fs:

QEMU: https://gitlab.com/virtio-fs/qemu.git

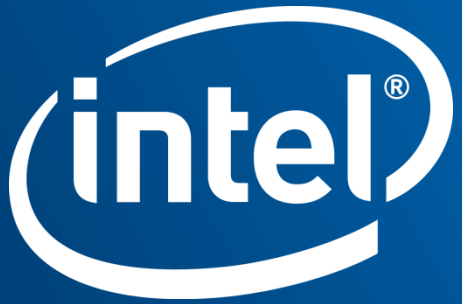
Linux: https://gitlab.com/virtio-fs/linux.git

SPDK: https://review.gerrithub.io/c/spdk/spdk/+/449162

- SPDK Vhost Live Recovery:

SPDK: https://review.gerrithub.io/c/spdk/spdk/+/455192

DPDK: https://patches.dpdk.org/patch/58207/


intel®