Hardware Accelerated VirtIO Networking for NFV

Varun Sethi, Yashpal Dutta

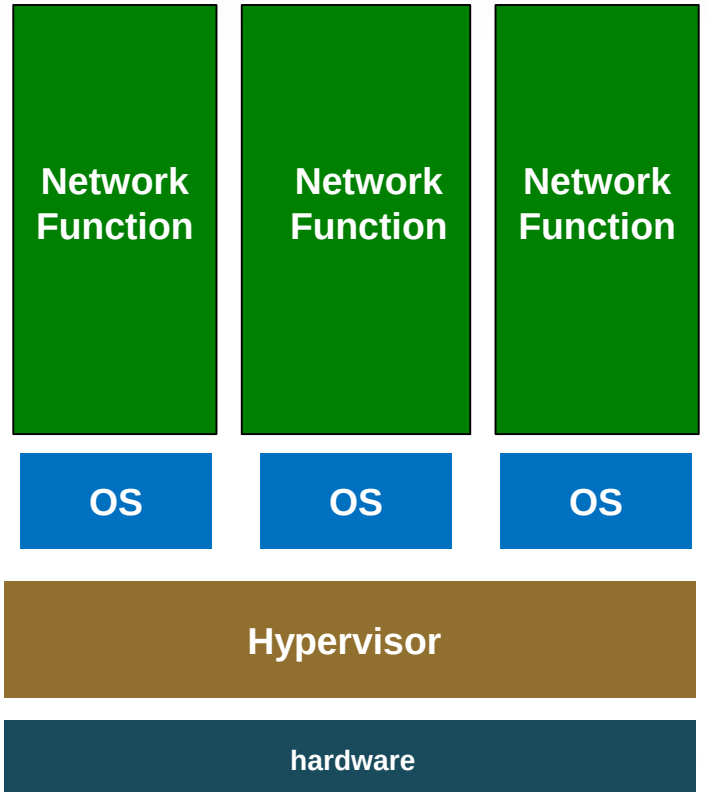
Agenda

- Network Function Virtualization
- I/O Virtualization Mechanisms
- Vhost-net Details
- NFV with virtio
  - Issues
  - Workaround
- Proposed Architecture

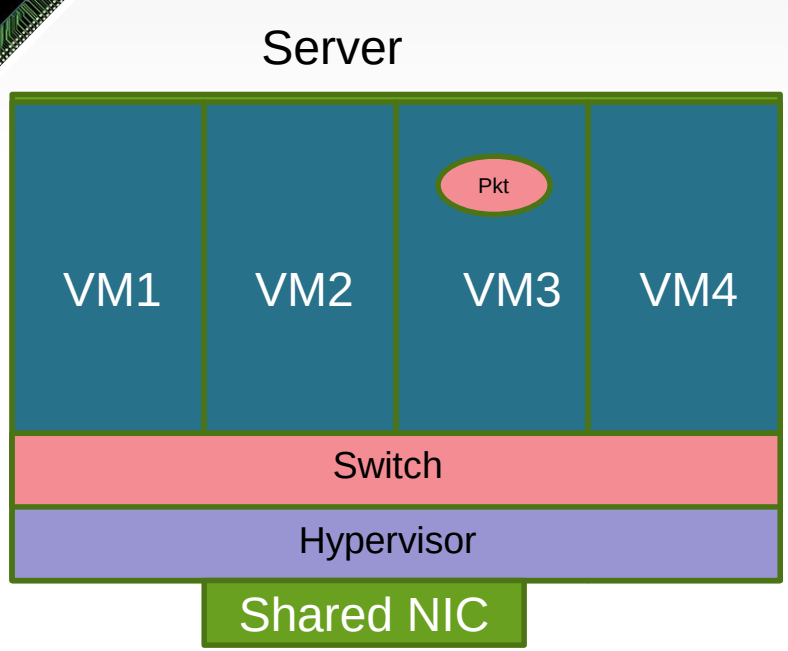
Network Function Virtualization

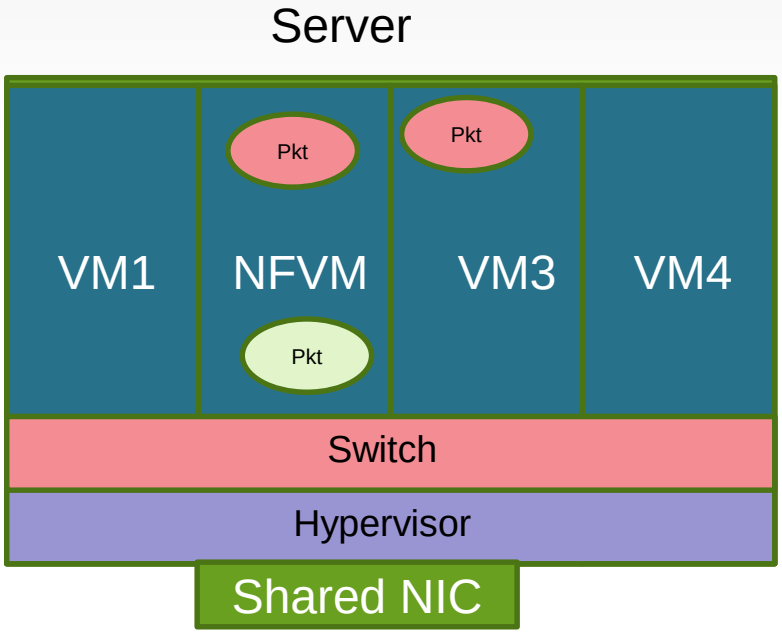
Network Function Virtualization

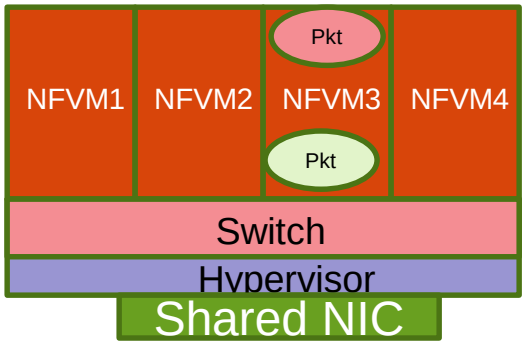
- Allows for dynamic network service launch
  - Reduces time for deploying new network services
- Same infrastructure can be used/shared for hosting multiple services
  - Reduced capital investment
  - Reduced cost of energy
- Offers Agility and Flexibility
  - Quickly scale up or down services based on the demand
  - Reduces time to deploy new networking services



NFV Deployment Scenario



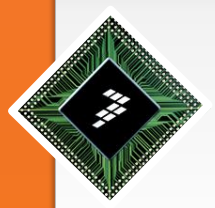




Network Appliance

Pkt

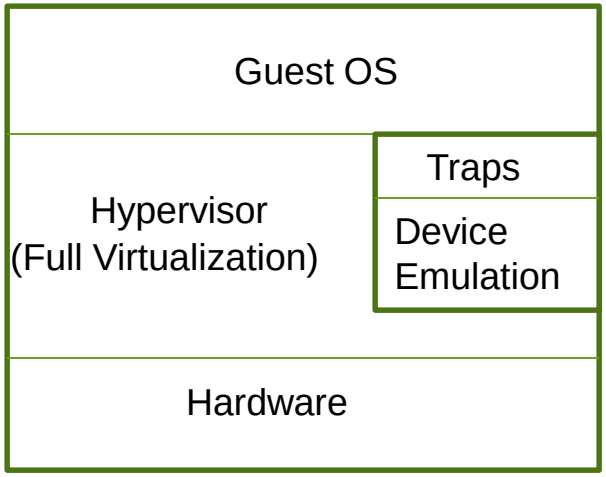
Pkt

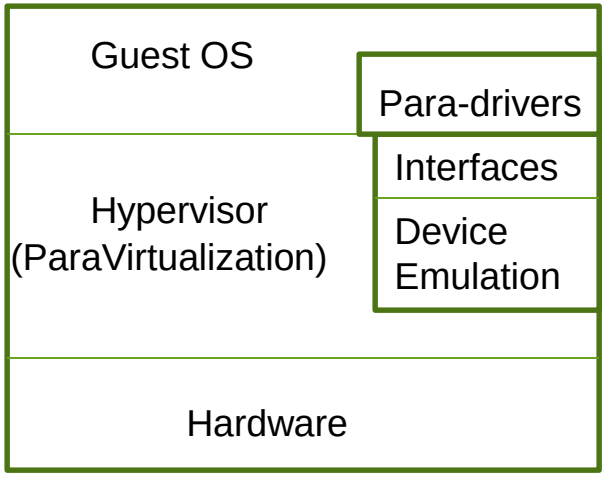


I/O Virtualization Mechanisms

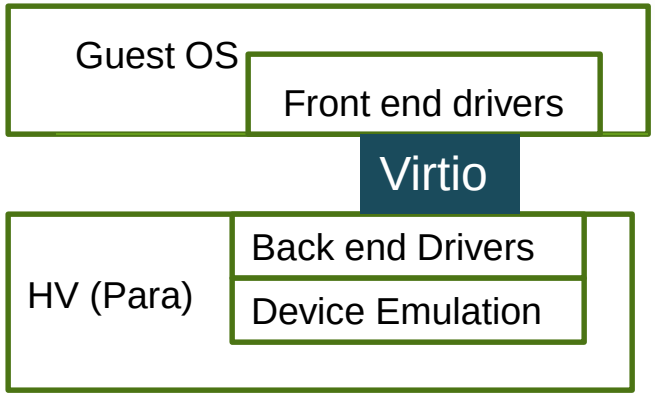
freescale™

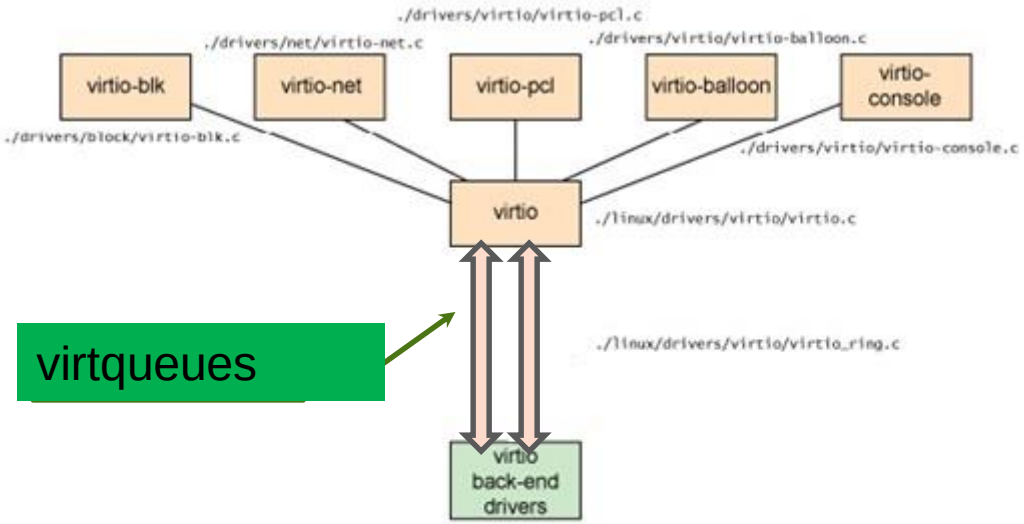
Full virtualization vs Paravirtualization





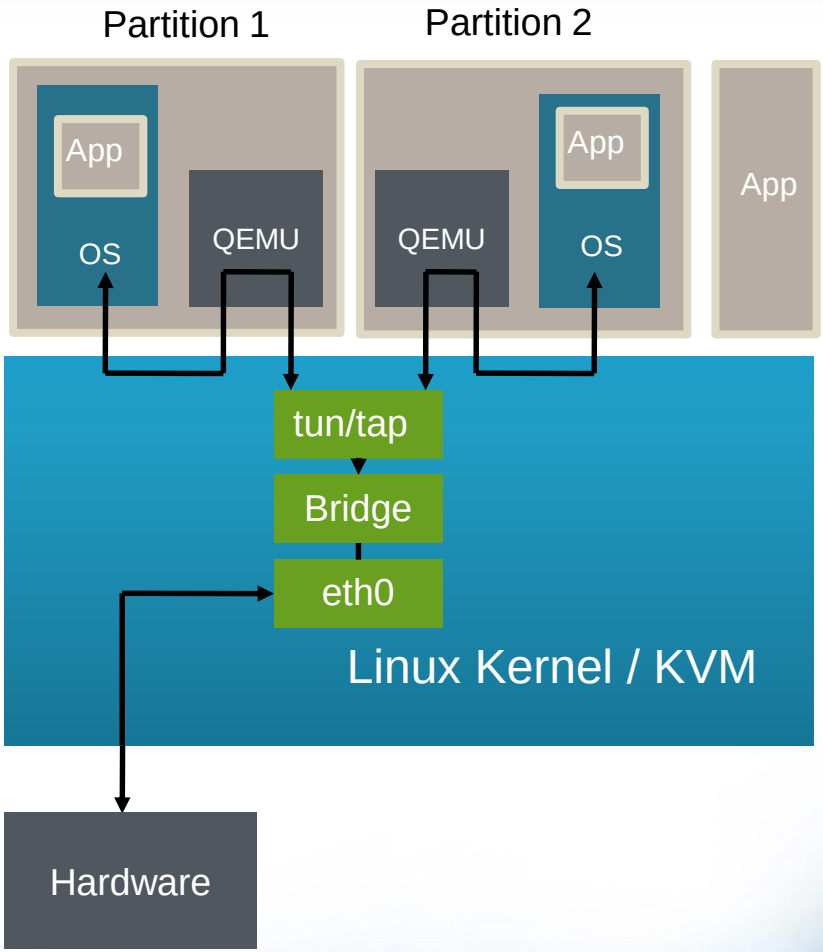
Virtio Architecture





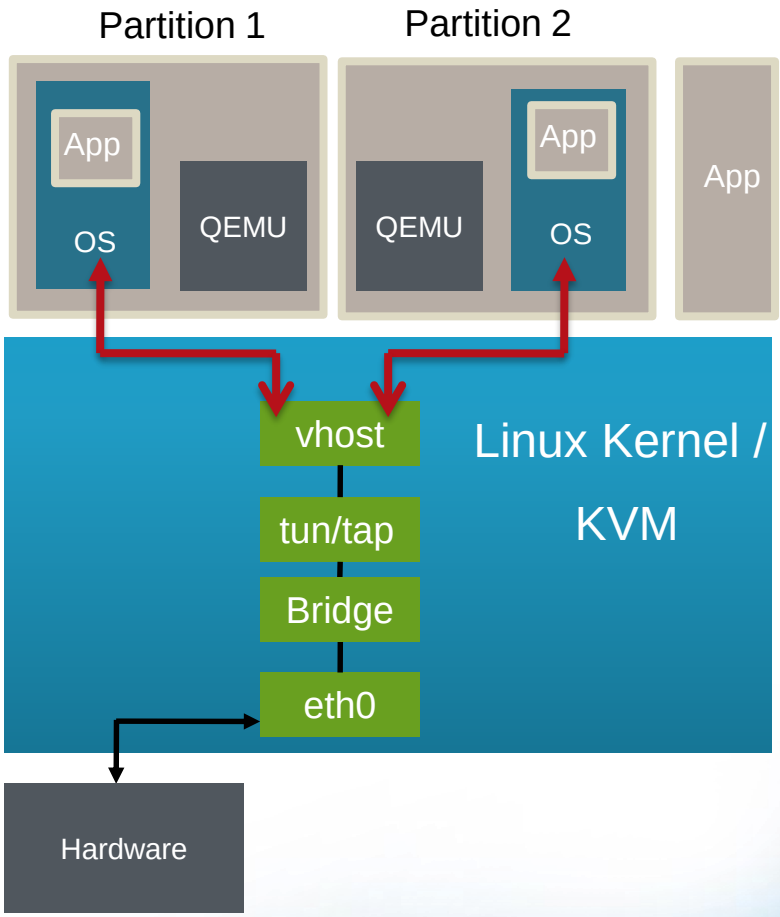
VirtIO Networking

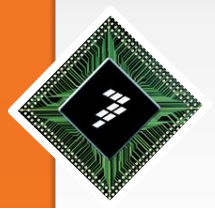
- Each guest sees a private VirtIO network device on PCI bus
- VirtIO network driver is needed in guest
- Standard tun/tap interface is used by QEMU to bridge to host network stack
- Backend driver in QEMU



Vhost Networking

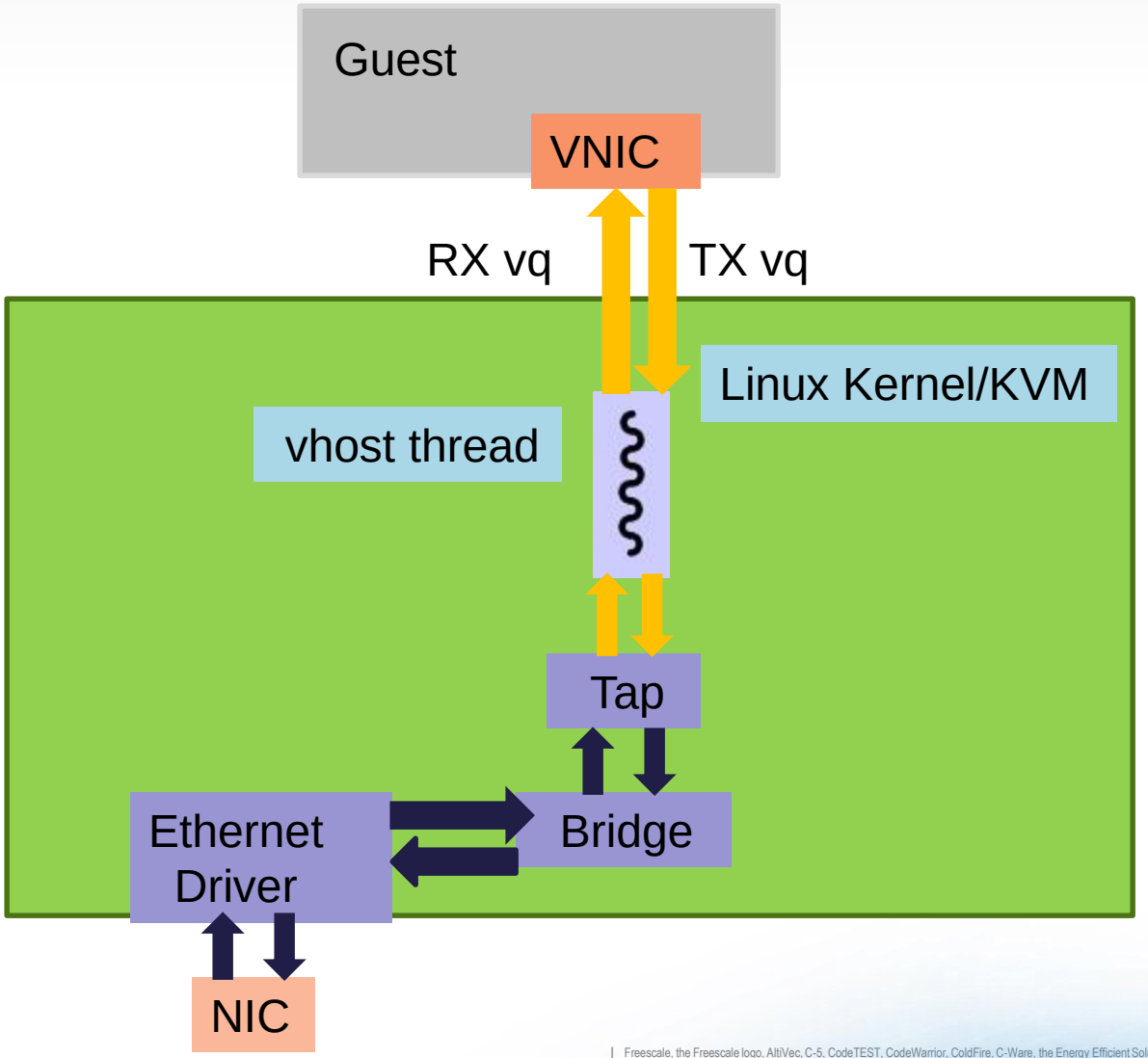
- Vhost-net is a module in Linux Kernel.
- Transfers buffers between host and guest.
- Bypasses QEMU for virtqueue operations



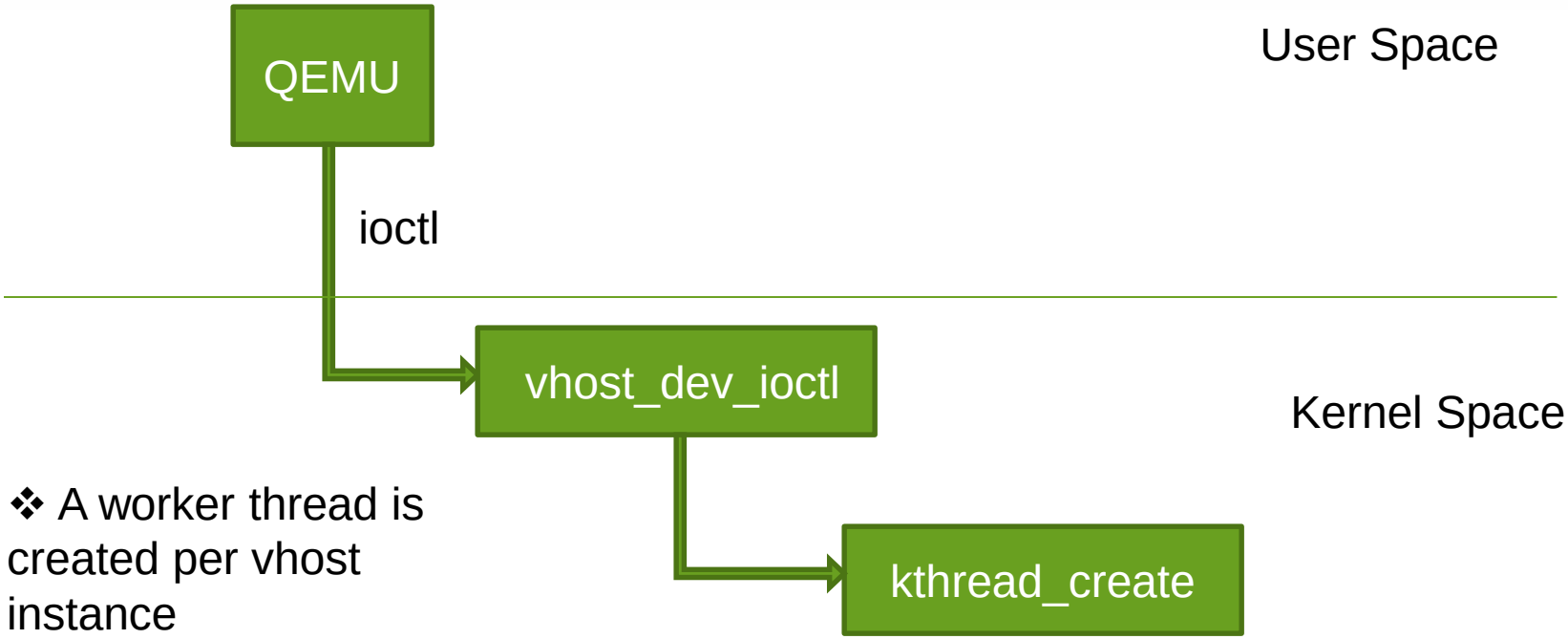


Vhost-net Details

Vhost-Net Flow Big Picture



Creation of vhost-worker thread

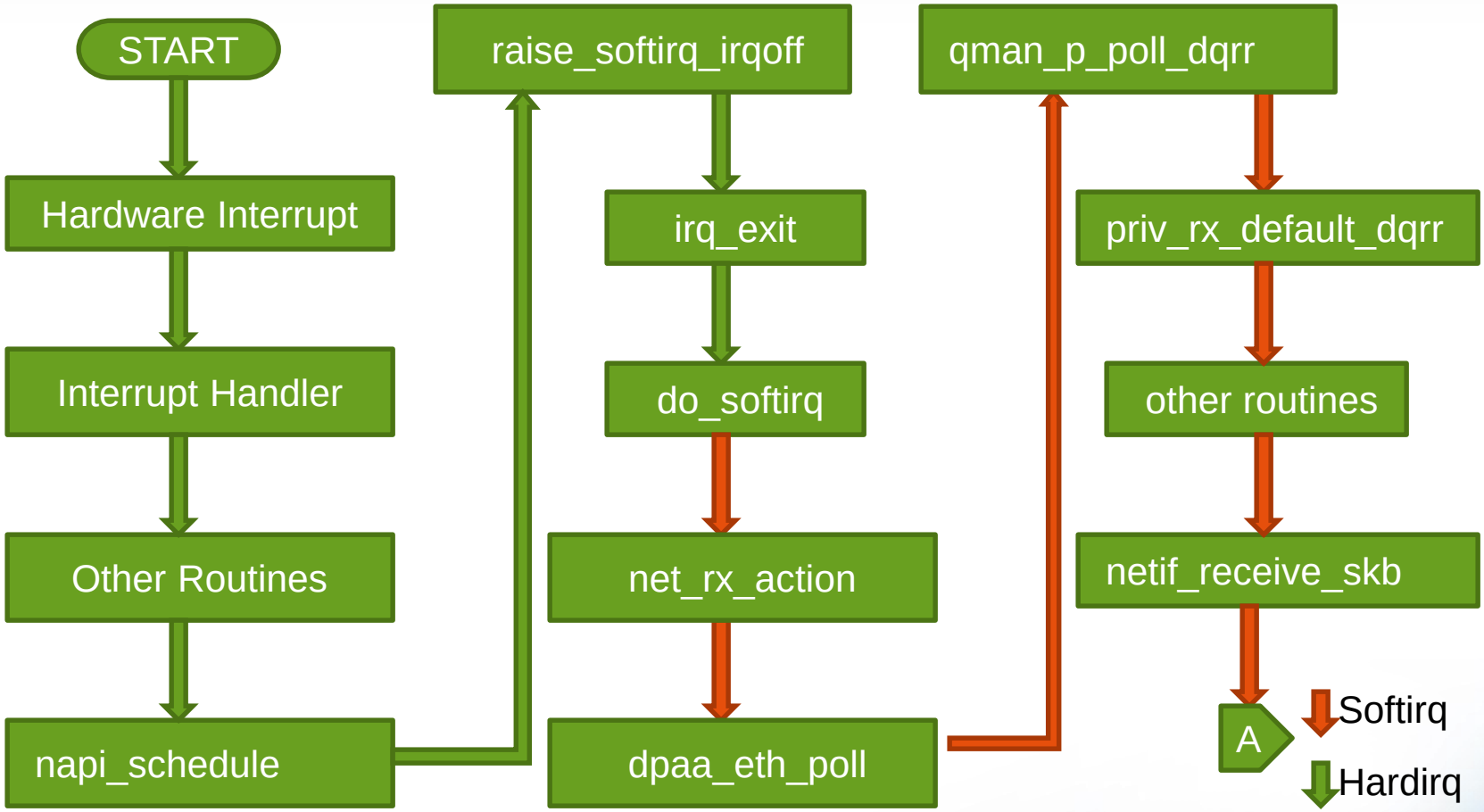


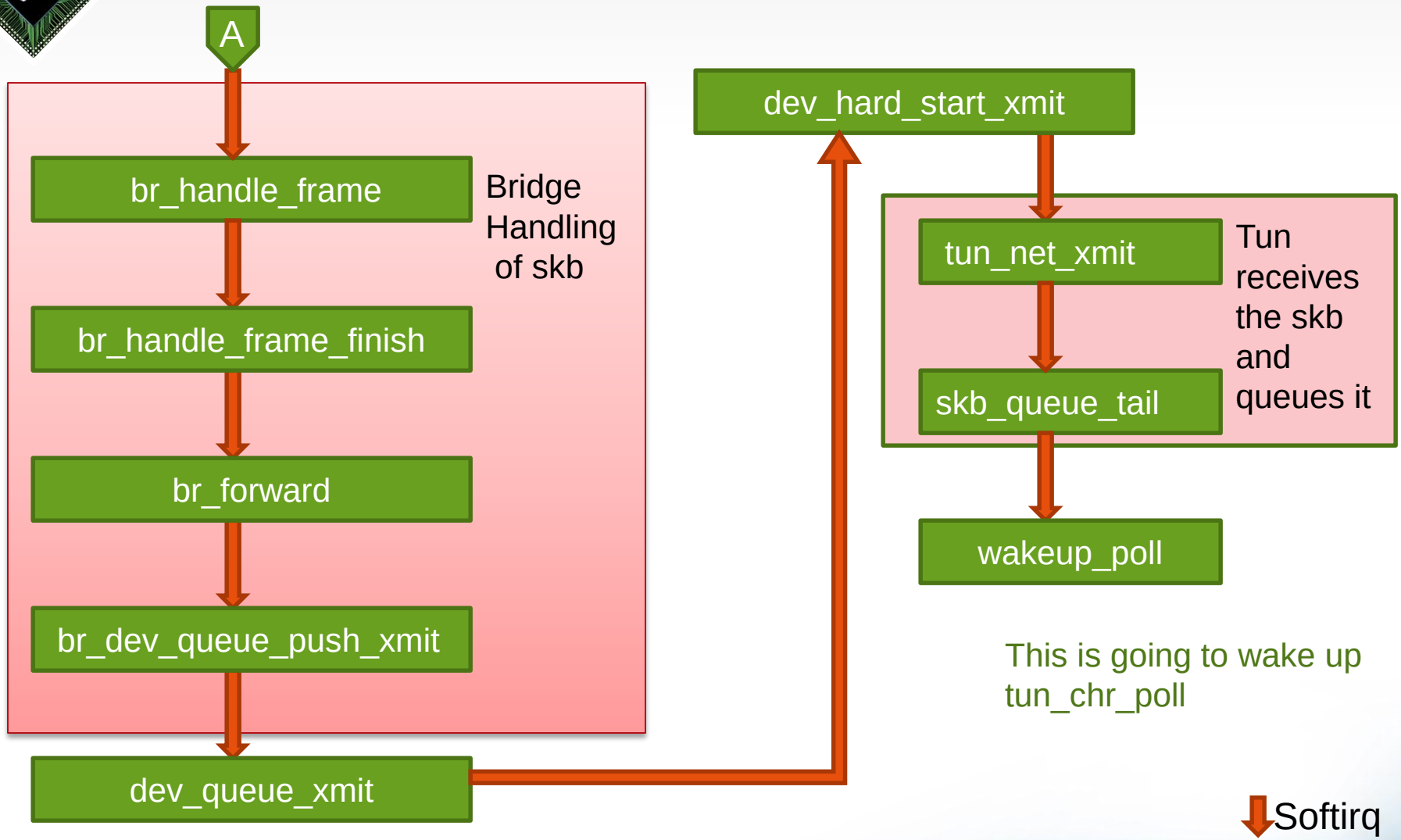
- A worker thread is created per vhost instance

- This thread is responsible for handling both Rx and Tx jobs for a vhost device

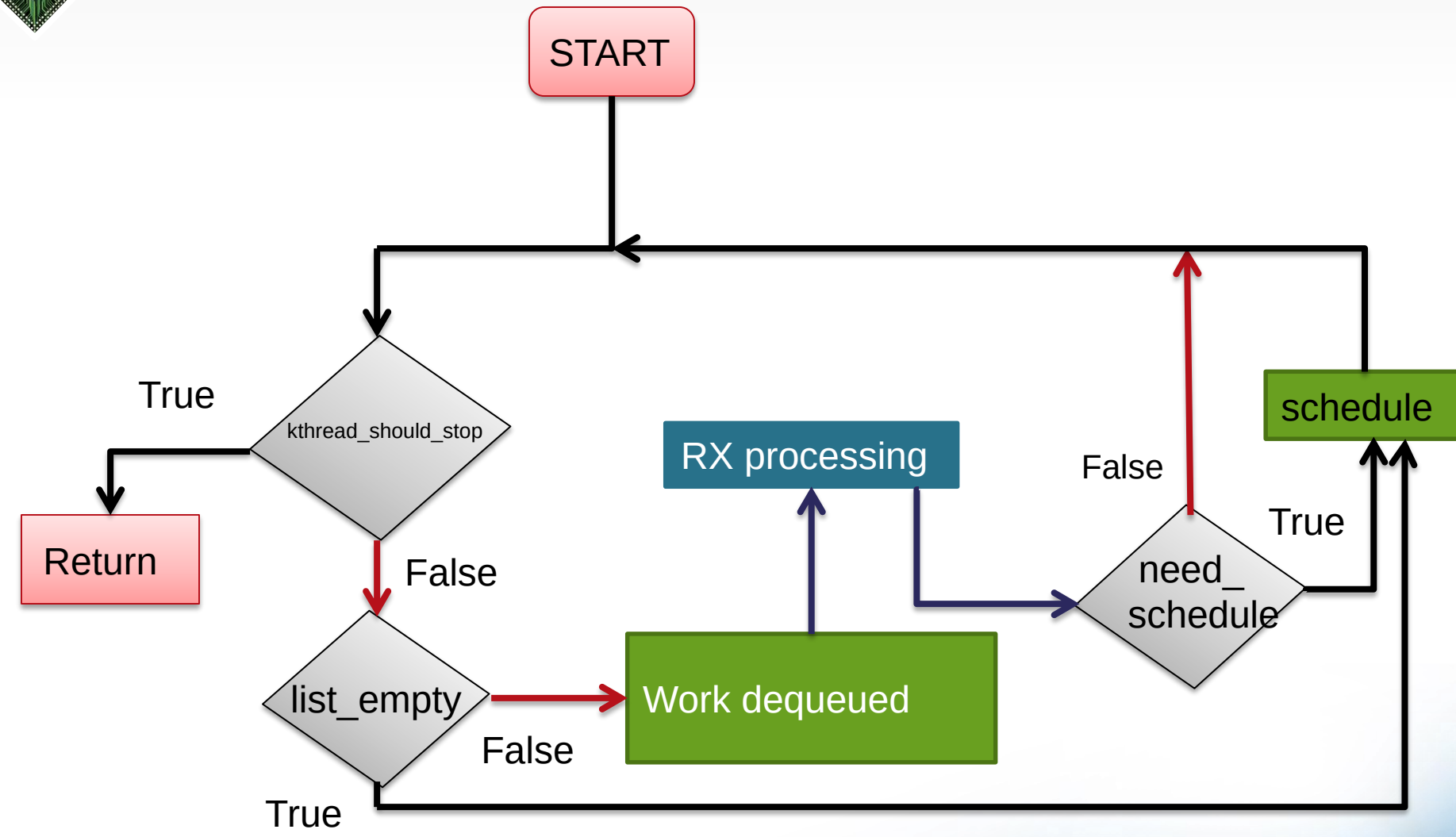
RX Packet Flow

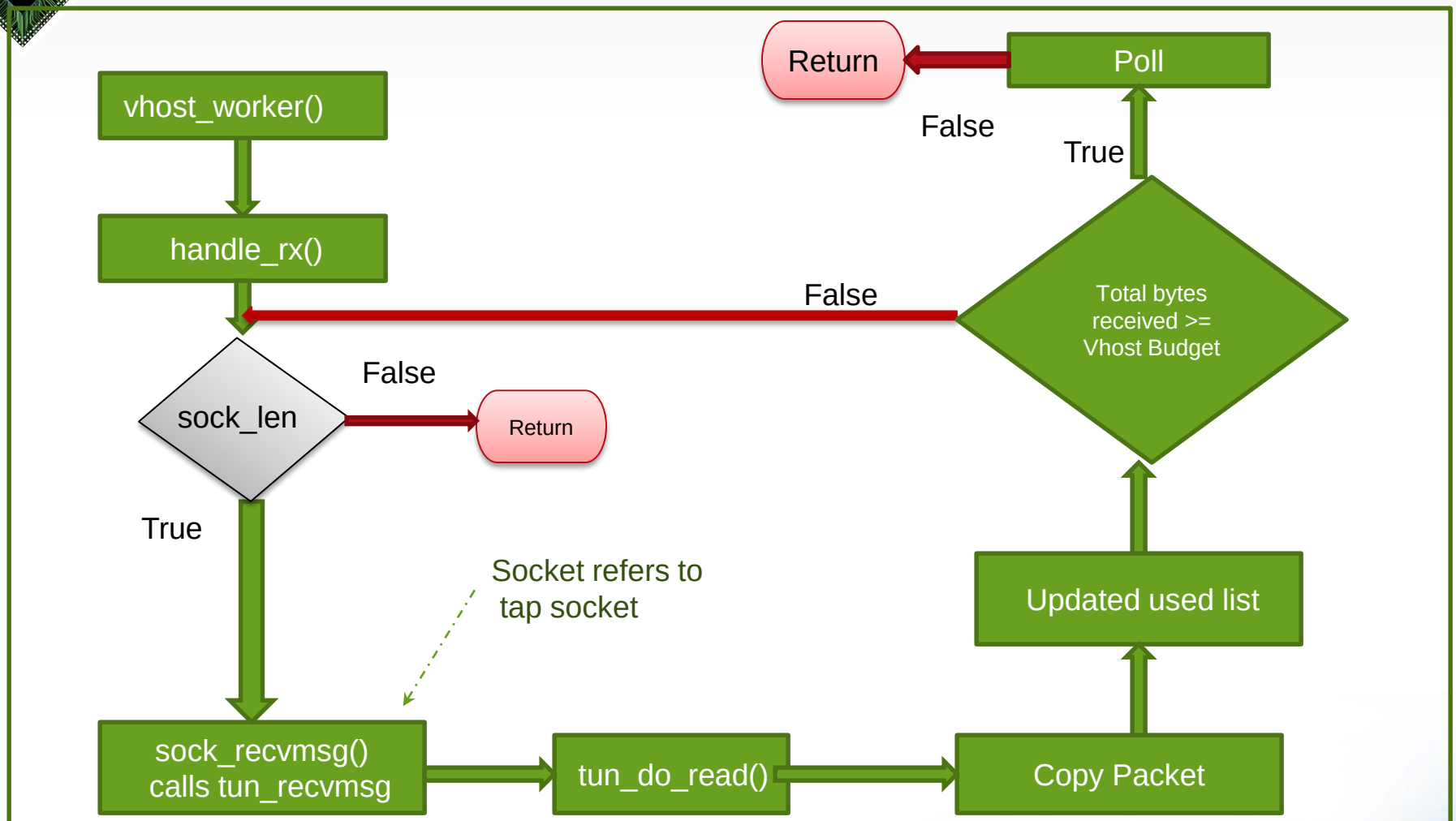
RX Packet Flow





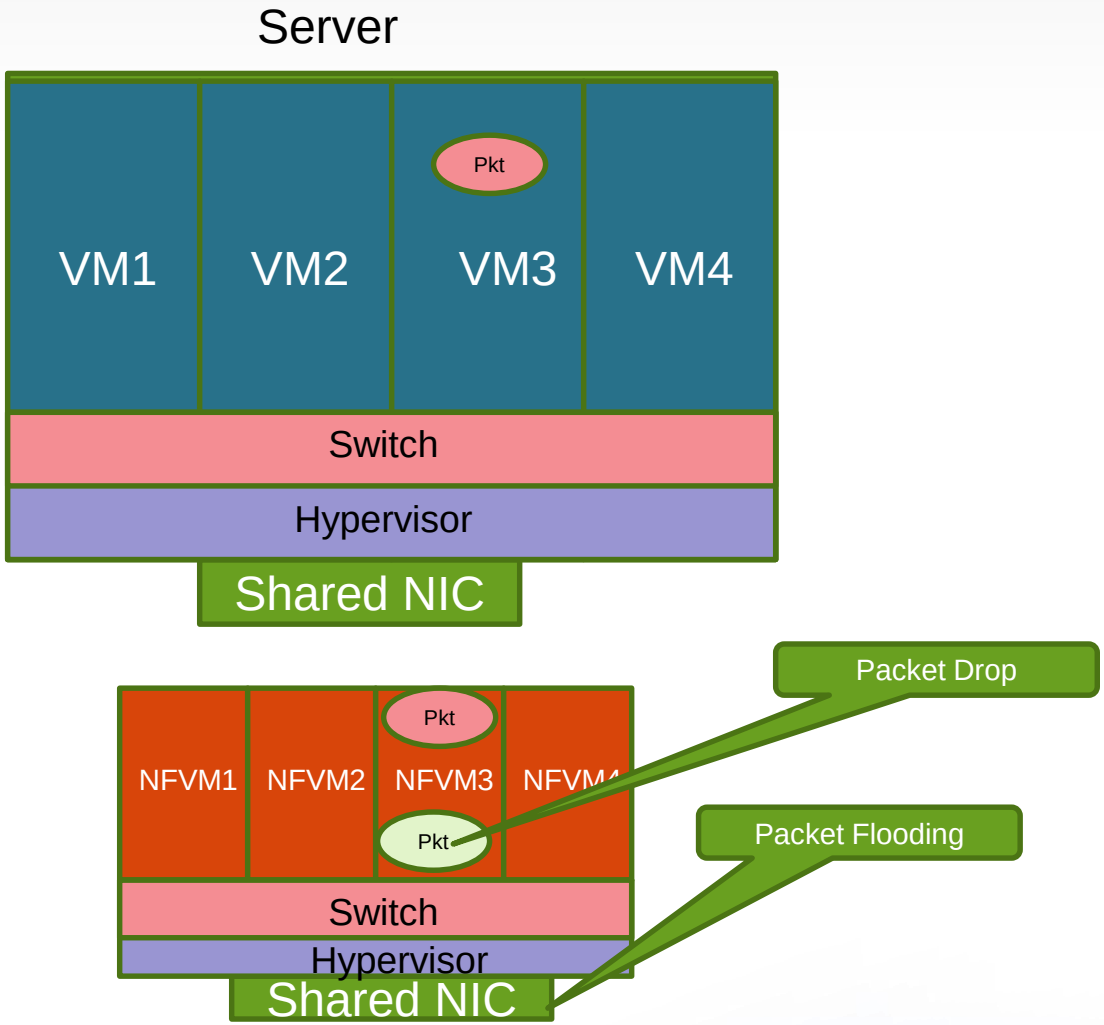
Vhost_worker flow



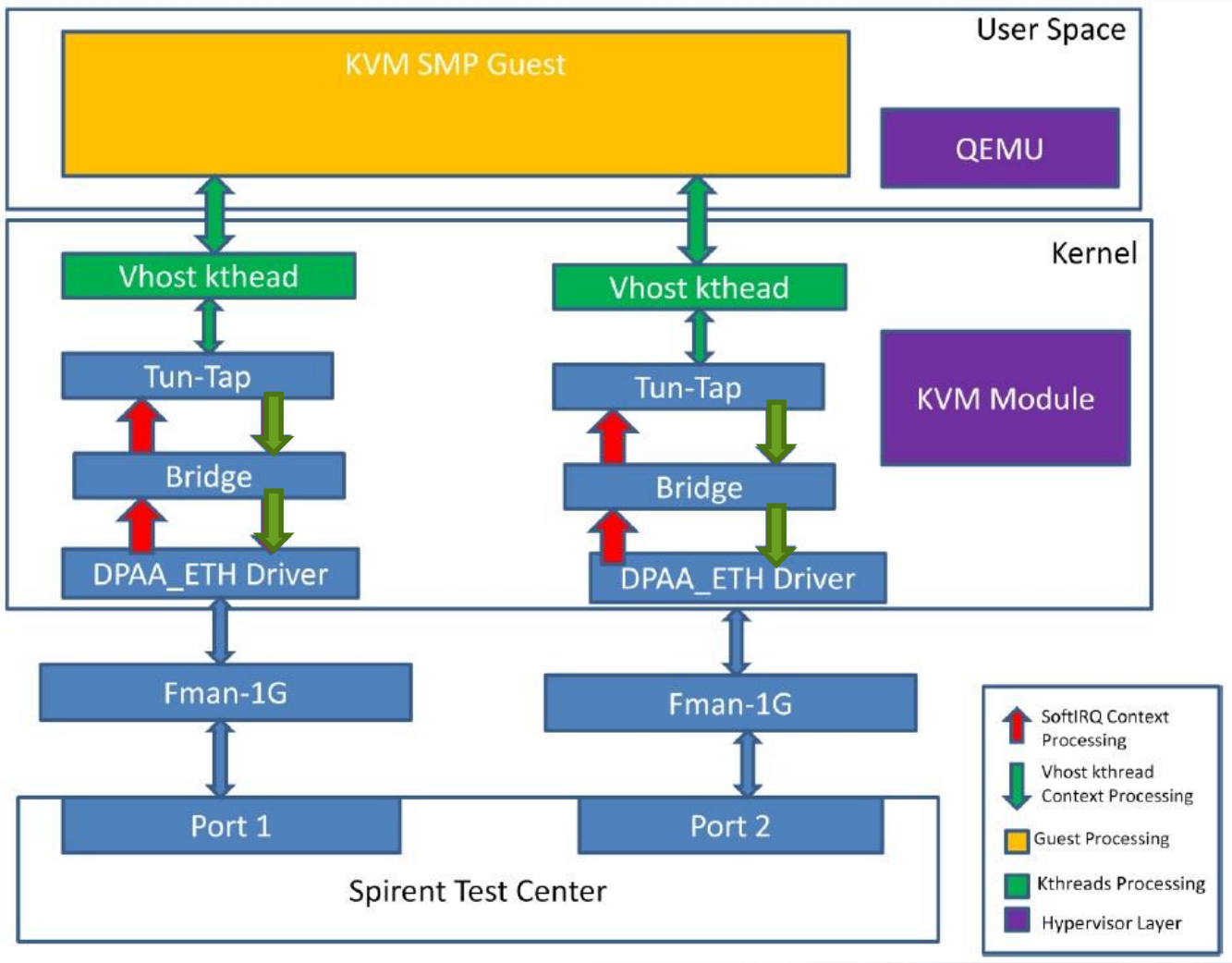


NFV with VirtIO

NFV with VirtIO



Setup for the Test



Existing Rx Flow

Vhost Thread

Fails to Schedule due to lower Priority than SoftIRQ

Tap

Bridge

NIC driver

Physical NIC

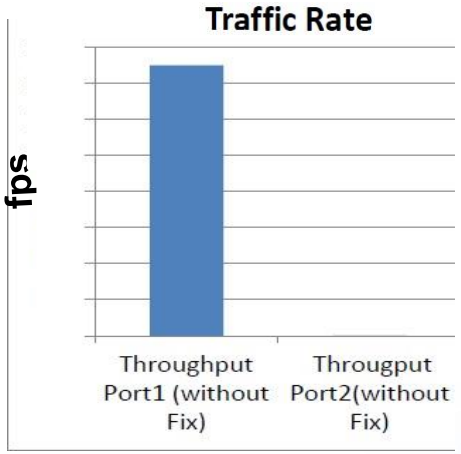
Packet Flooding

1 | 2 | … | 500

Tap Queue Full

Packets keep on dropping at tap level

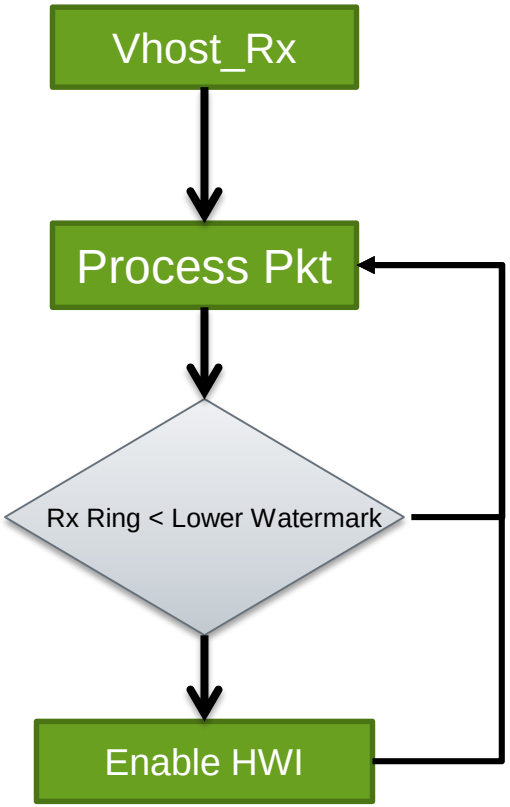
NAPI processing



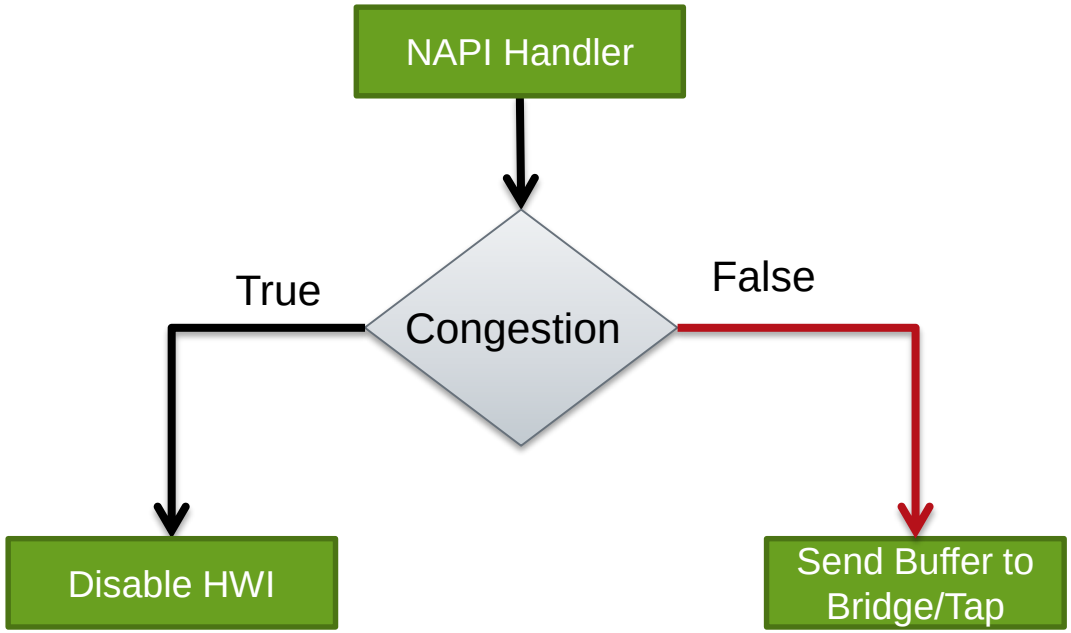
First Proposal for Hardware Assisted VirIO

Modified Flow

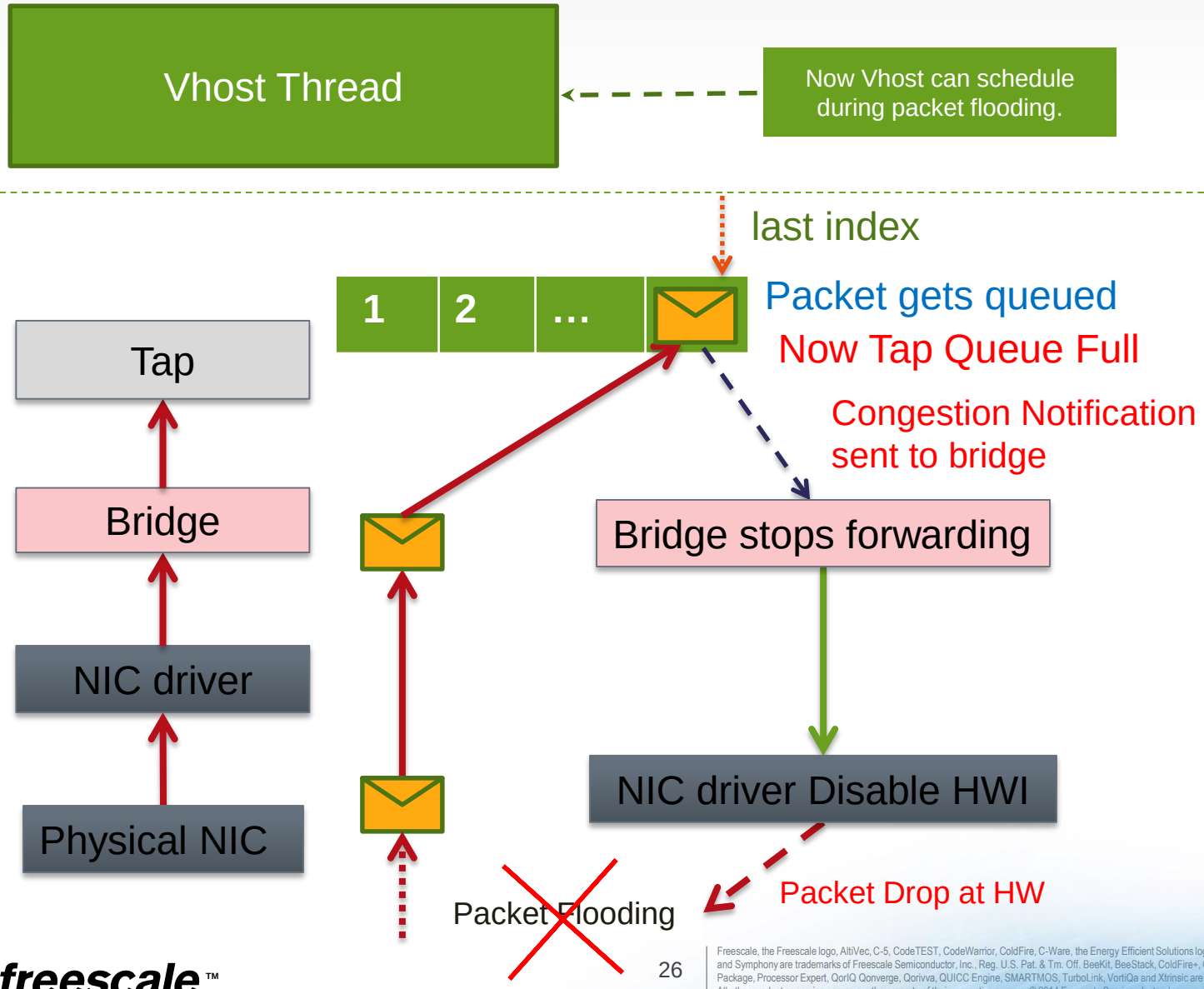
Vhost Rx Processing



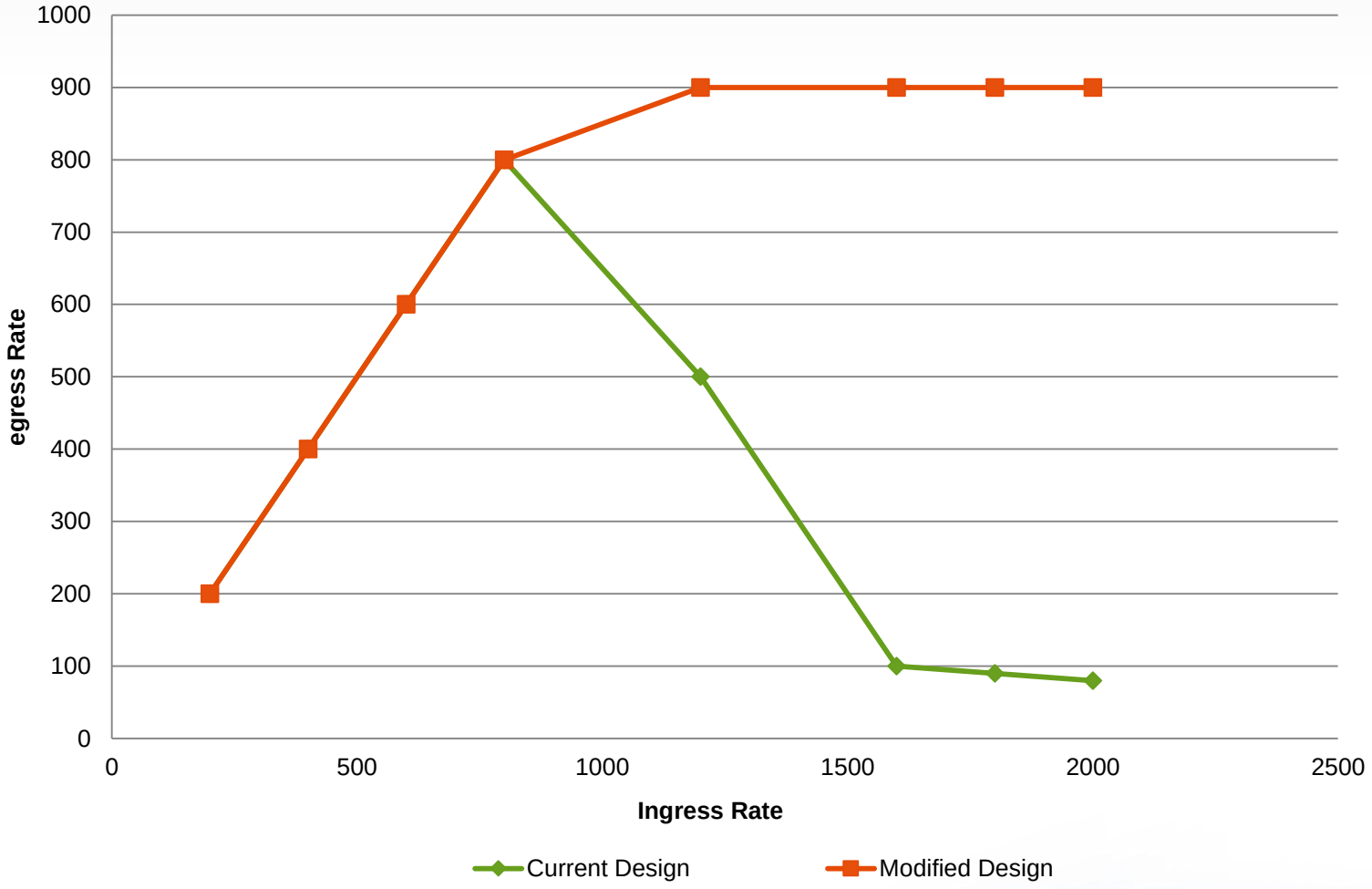
NAPI SoftIRQ Processing

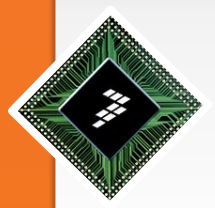


Modified Flow (Continued)



Performance Crawl Chart



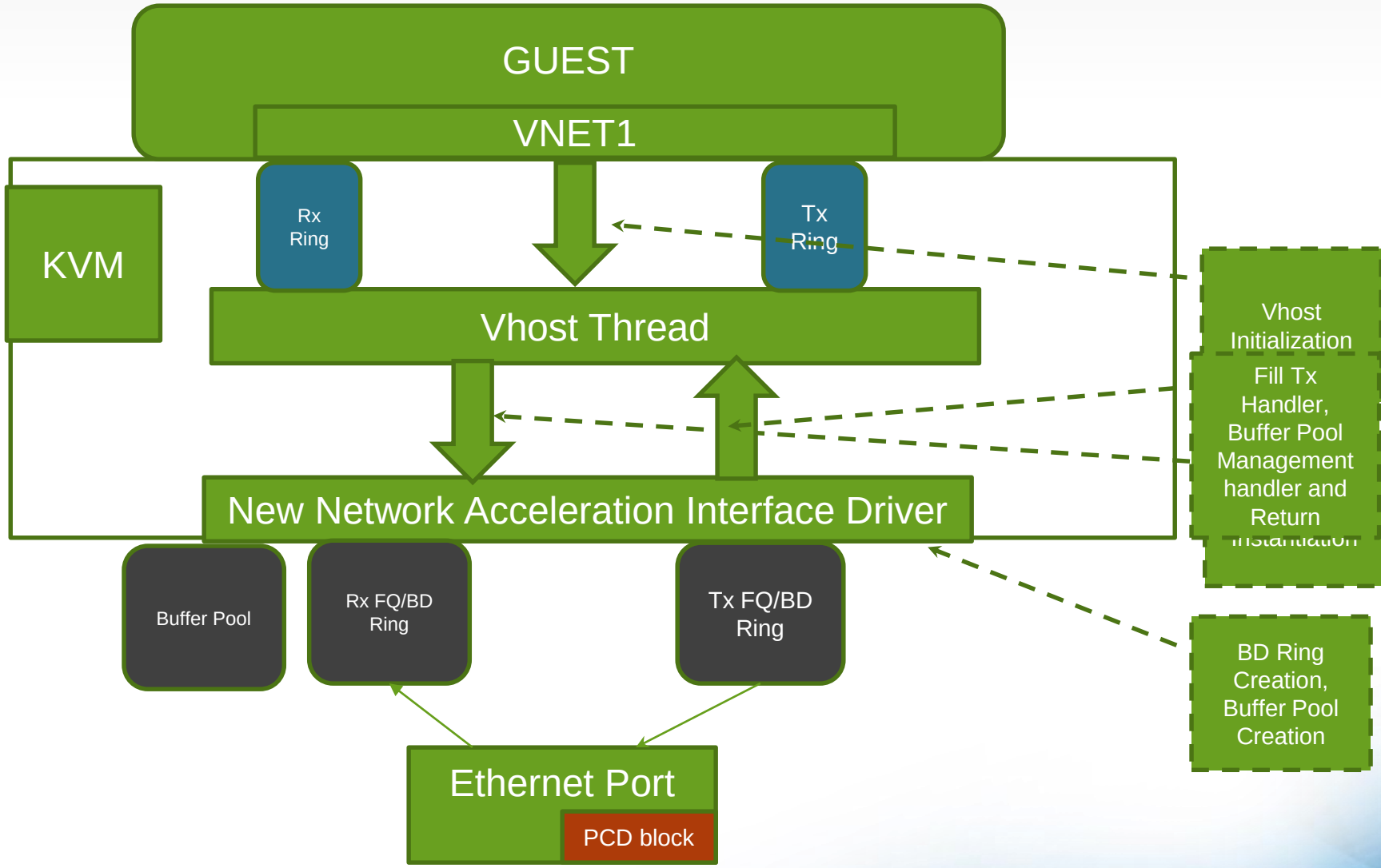


Modified Vhost and Driver Interface Proposal for Hardware Assisted VirtIO

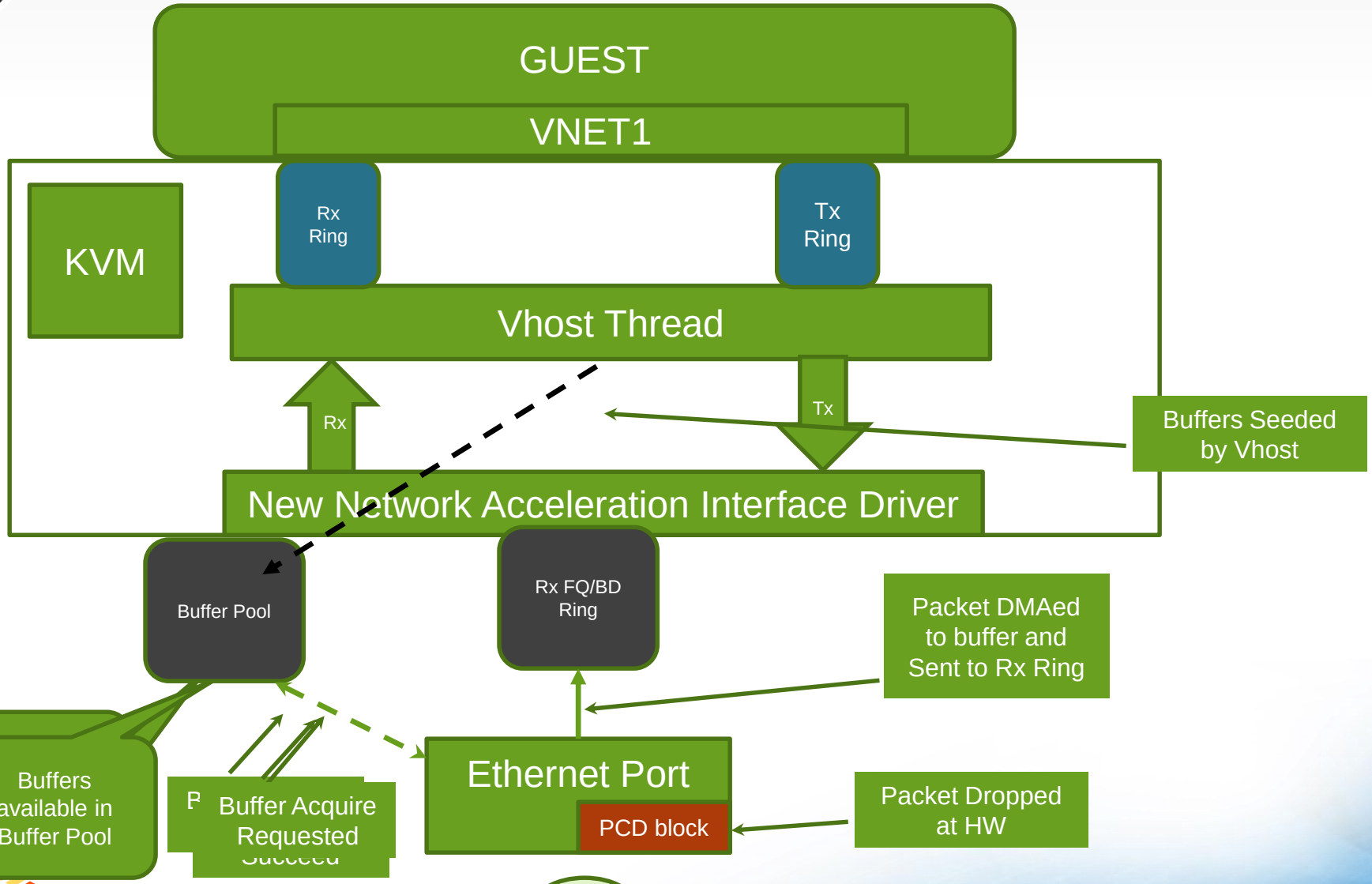
Introduction

- Dynamic vhost thread controlled NIC driver instance.
- Vhost thread registers with NIC driver:
  - Rx Function handle
  - Buffer depletion handler
- Driver provides following during registration to Vhost:
  - Tx Function Handler
  - Buffer pool management handle

Interface Initialization



Modified Rx Handling



Summary

- Vhost-net interfaces can get saturated at high traffic rates
- No mechanism to communicate Vhost-net interface saturation information to host
- With a more integrated guest/host Virtio infrastructure, it's possible to avoid system saturation and increase throughput
- Hardware acceleration can be leveraged for Vhost net interfaces


freescale™