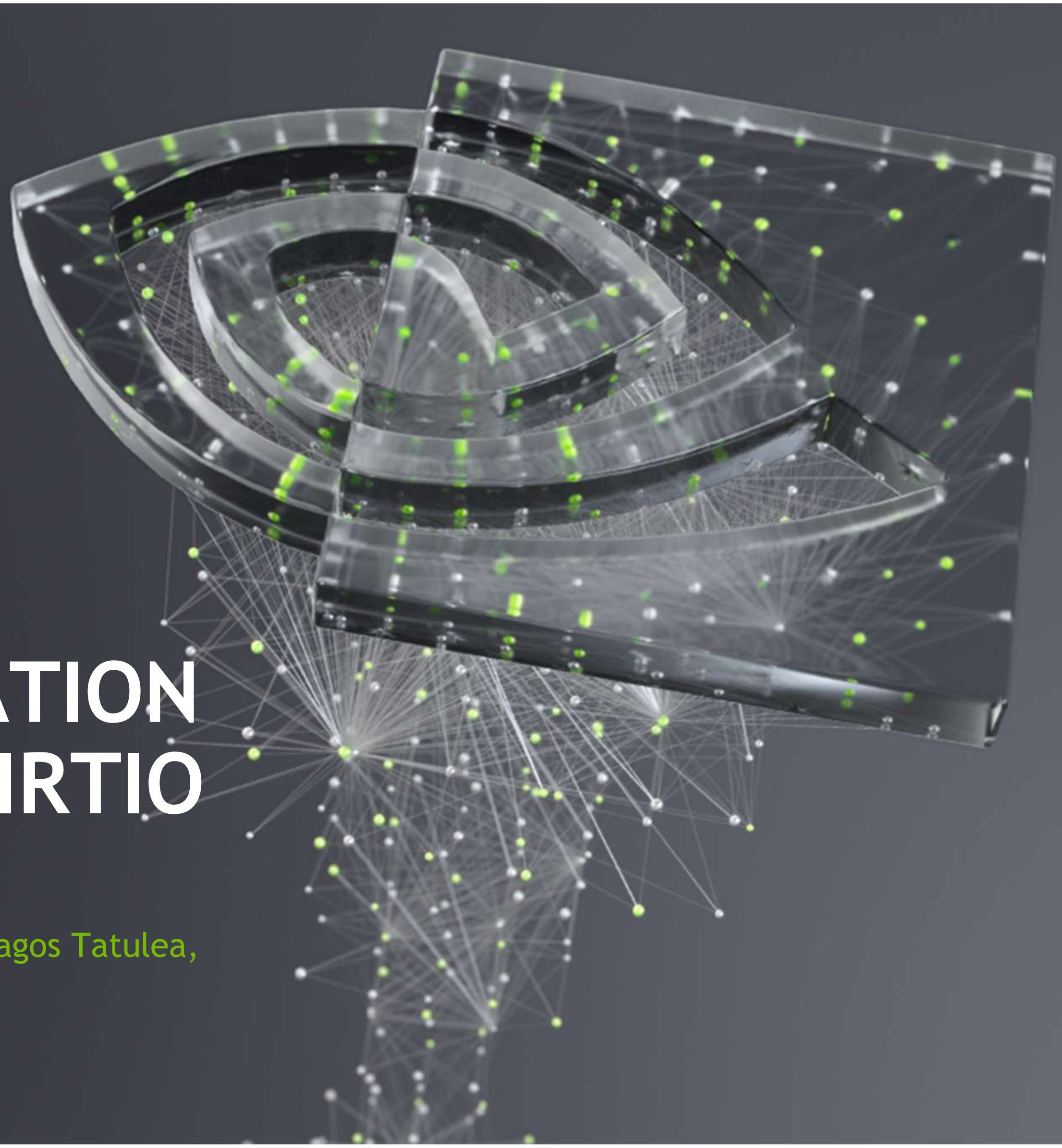
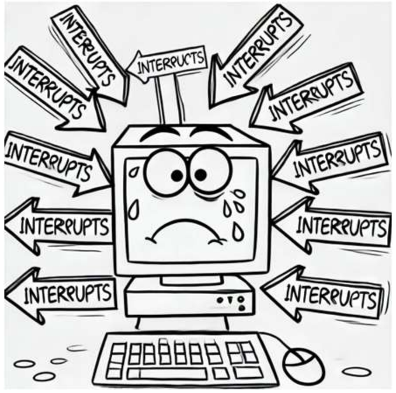


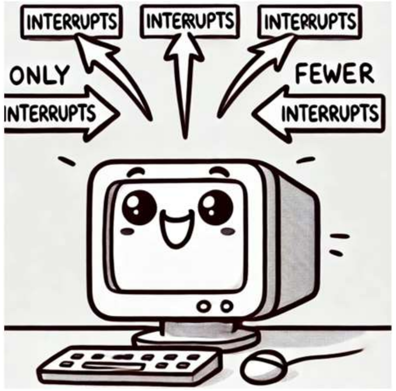
INTERRUPT MODERATION APPROACHES FOR VIRTIO DEVICES

Parav Pandit, Kailiang Zhou, Jun Deng, Lijun Yu, Dragos Tatulea,
July 19, 2024

AGENDA

Interrupt moderation





1. How to reduce interrupts for the vdpa device?

2. How to reduce interrupt for the virtio PCI device?

3. How to moderate them adaptatively?

Solved problem in 2017 using netdim!

Thanks to Tal Gilboa, Andy Gospodarek

BACKGROUND-1

Interrupt moderation options in virtio specification

1. TXQ, RXQ interrupt moderation (VIRTIO_F_RING_EVENT_IDX)
    1. Present from v1.0 from 2016
    2. Known as event suppression
    3. Designed for software backend who has < 10nsec latency to guest VM memory
    4. Pros:
        1. Guest VM can directly control the device
        2. Simple scheme but hard for PCI hardware devices to do it efficiently
    5. Cons:
        1. Based on queue producer/consumer – requires reading on every q wrap around
        2. Driver does not consider (a) throughput, (b) latency or (b) interrupt rate to program queue indices.
        3. PCI hardware device expected to read them stalling the receive path
        4. Ring index located at sparse location (away from available index aka producer index)
        5. Disables drivers' notifications too!
        6. Racy between hardware reading, software writing from a shared memory

BACKGROUND-2

Interrupt moderation options in virtio specification

1. TXQ, RXQ interrupt moderation (VIRTIO_NET_F_VQ_NOTF_COAL)
    1. Introduced in v1.3 in 2023
    2. Known as notification coalescing
    3. Pros:
        1. Based on packet count and timer based
        2. Usable by ethtool, per queue.
        3. Used for netdim adaptive moderation in kernel 6.x but only for rxq.
        4. Guest VM can directly control the device
        5. Efficient for PCI hardware virtio devices and software devices.
    4. Cons:
        1. Almost no guest VM kernels have it in 2024.
        2. Parameters modification was protected by global rtnl lock… Optimized finally!
        3. No efficient way to disable coalescing, expected to solve in version 1.4.

PROBLEM STATEMENT

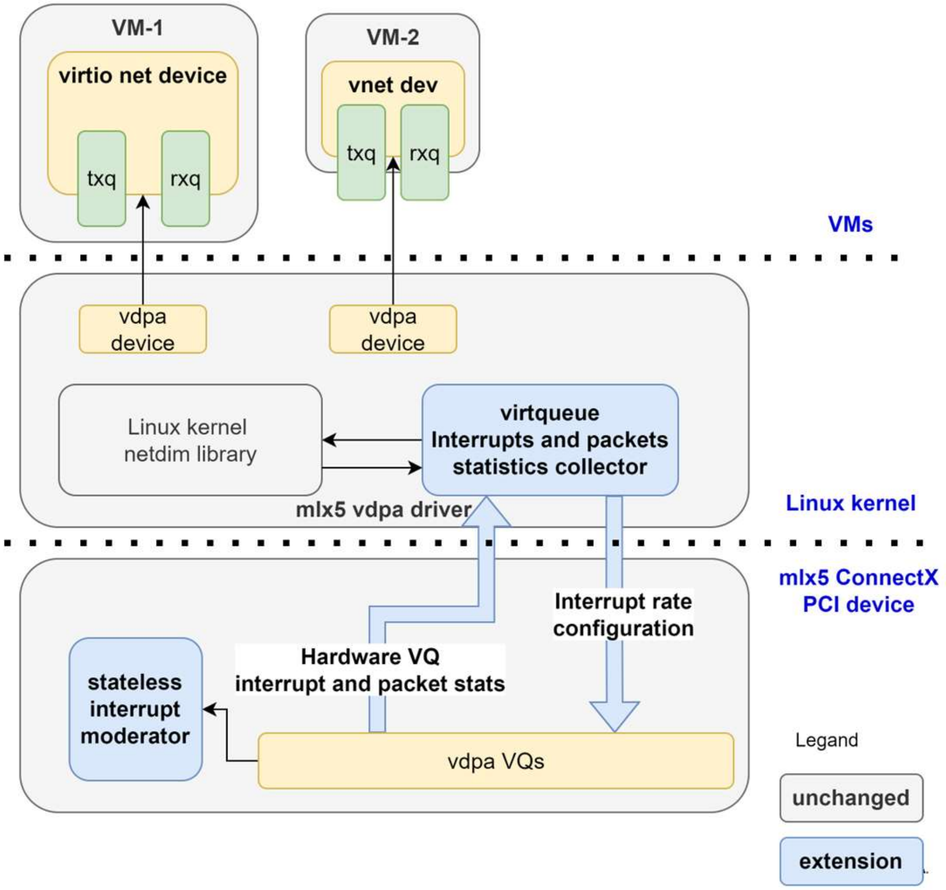
1. How to moderate interrupts for guest VMs kernels from 2016 to 2026?
    1. Without modifying the guest VM drivers?
    2. Without offering VIRTIO_NET_F_VQ_NOTF_COAL?
    3. When VIRTIO_NET_F_VQ_NOTF_COAL offered but not used for TXQ?
    4. Without offering VIRTIO_F_RING_EVENT_IDX?

Solved problem in 2017 using netdim!

Thanks to Tal Gilboa, Andy Gospodarek

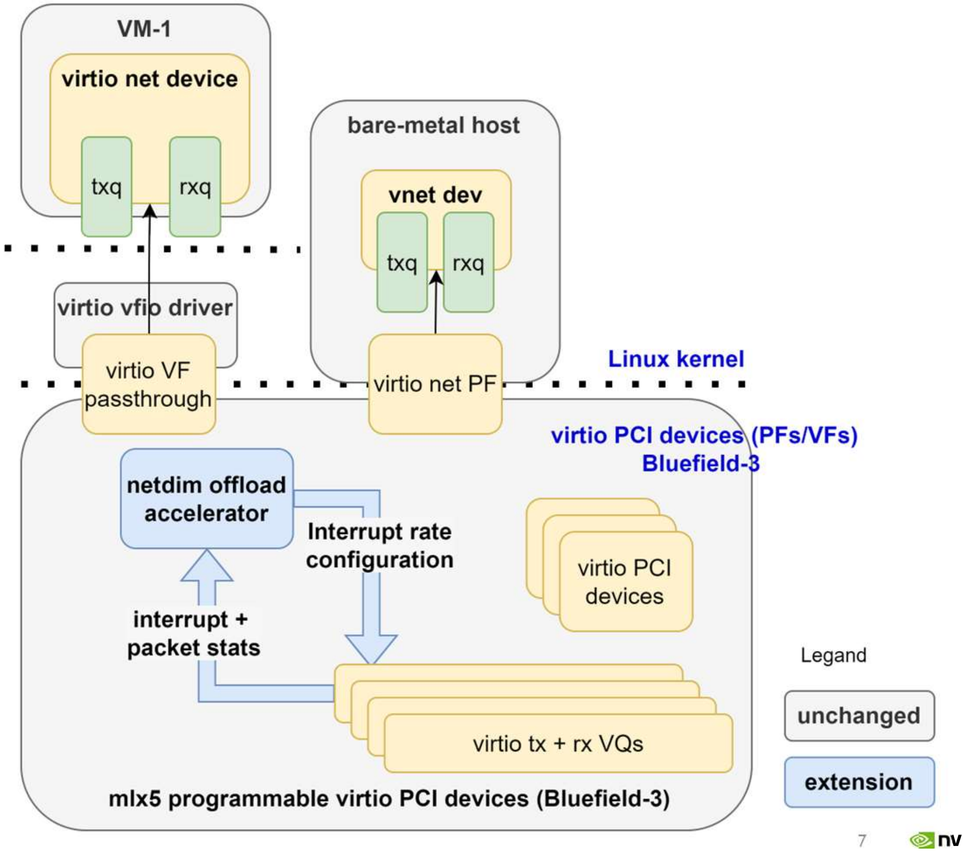
NETDIM ONLOAD FOR VDPA DEVICES

- Pros:
  - Utilize Hypervisor software dynamic interrupt moderation infrastructure
  - Stateless interface to firmware
  - New interrupt and statistics collector feeds the data to Linux kernel.
  - Takes decisions from Linux kernel netdim and moderates interrupt rate in the device
  - Period collector and configuration in mlx5 vdpa driver per vdpa device
  - Useful till hundreds of devices.
- Cons:
  - 100 interrupts/sec increase per vdpa device in hypervisor, @10msec periodic query.
  - Not scalable beyond a threshold.
  - Usable only for the VFs (bare-metal systems cannot use)


VM-1
virtio net device
txq
rxq
VM-2
vnet dev
txq
rxq
VMs
vdpa device
vdpa device
Linux kernel netdim library
virtqueue Interrupts and packets statistics collector
mlx5 vdpa driver
Linux kernel
mlx5 ConnectX PCI device
Interrupt rate configuration
Hardware VQ interrupt and packet stats
stateless interrupt moderator
vdpa VQs
Legand
unchanged
extension

NETDIM OFFLOAD FOR VIRTIO PCI DEVICES

- Virtio PCI devices are passthrough the guest VM
- Hypervisor passthrough driver is unaware of virtio semantics
- Utilize programmable virtio PCI devices
- NETDIM offloaded to virtio PCI devices
- **Pros:**
  - VMs and bare-metal hosts interrupts moderated
  - Better scalability as computation offloaded to the devices which has support for it.
  - Zero interrupts on hypervisors for moderation tasks
  - Ability to apply heuristics for latency gains

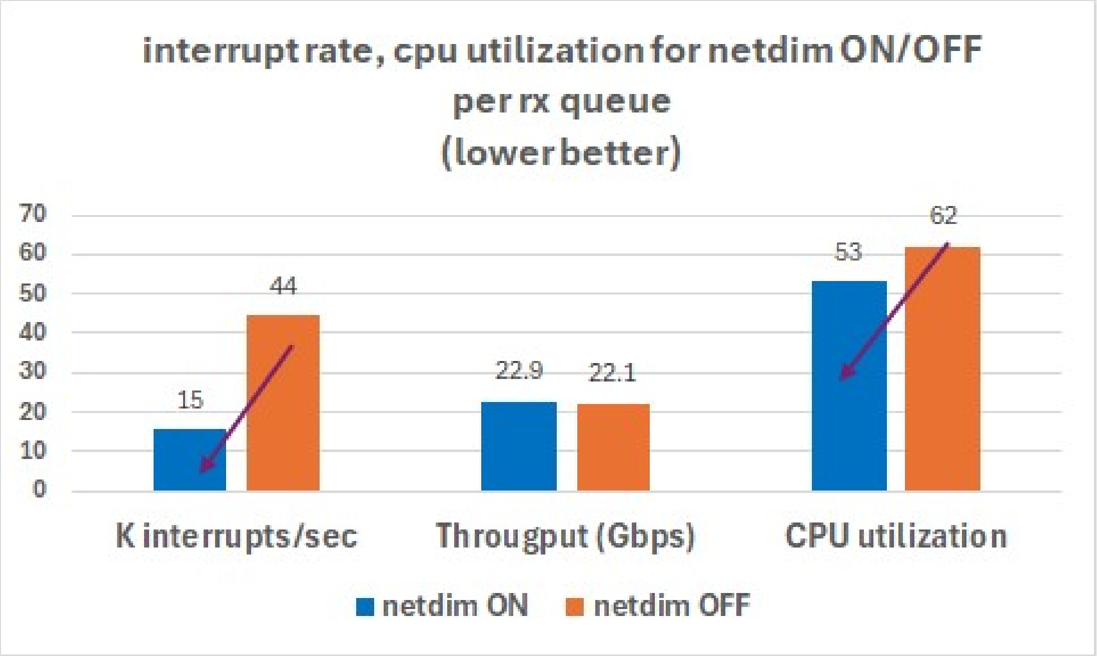


PERFORMANCE RUNS

Micro benchmark

- Virtio netdevice configuration:
  - q depth = 1K
  - 2 rxqueues per device
  - MTU = 9K
  - Application: iperf
  - Device: Bluefield-3 virtio PCI devices

- Results:
  - Throughput maintained at 26.x Gbps
  - CPU utilization reduced by 9%
  - Interrupts rate reduces by 300%
  - Challenge: Latency cannot be reduced by 2usec timer limit of DIM.



LATENCY ADAPTIVE NETDIM

Powered by packet heuristics

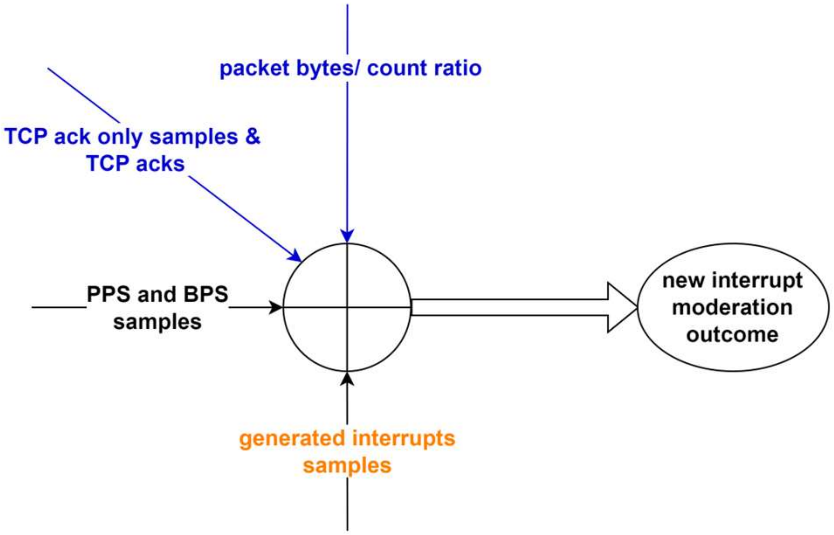
Challenge:

a. Latency cannot be reduced to lower than 2usec timer limit of DIM. (left most profile)

b. DIM timeout in range of 10% of RTT contributes to bigger congestion window.

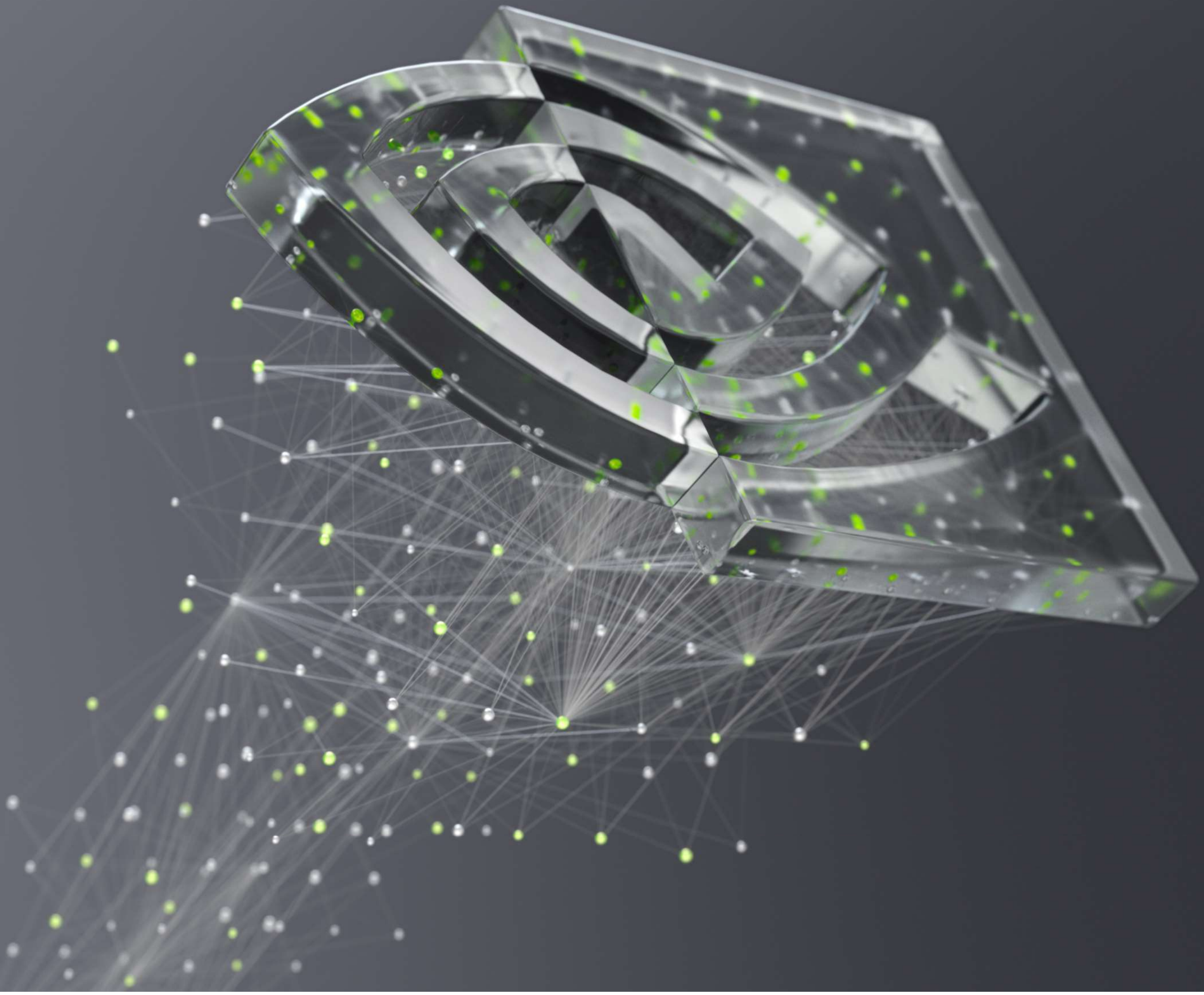
Solution approaches: Packet heuristics

1. Programmable PCI device runs a high pass filter on packets metadata.
    1. More than 50% TCP acks only packets or 20% TCP acks
    2. Short message burst: ratio of packet bytes to packets in a burst.
2. Latency sensitive short messages enjoys the gain of 2usec one way reduction.


packet bytes/ count ratio
TCP ack only samples & TCP acks
PPS and BPS samples
new interrupt moderation outcome
generated interrupts samples

SUMMARY

- Netdim library from hypervisor for vdpa device is useful at low to moderate scale of vdpa devices.
- Netdim offload to DPU based virtio pci devices benefits CPU utilization for guest VMs + bare-metal hosts.
- Latency enhancements do not have conclusive results due to jitter.



NVIDIA®