SOSP 2024

VPRI: Efficient I/O Page Fault Handling via Software-Hardware Co-Design for IaaS Clouds

Kaijie Guo, Dingji Li, Ben Luo, Yibin Shen, Kaihuan Peng, Ning Luo, Shengdong Dai, Chen Liang, Jianming Song, Hang Yang, Xiantao Zhang, Zeyu Mi

Shared by **Zheng Yang**
2025-06-24

Device pass-through kills memory paging

Pass-through: good I/O performance

But bad for dynamic memory utilization

Software solution: not suitable for IaaS cloud

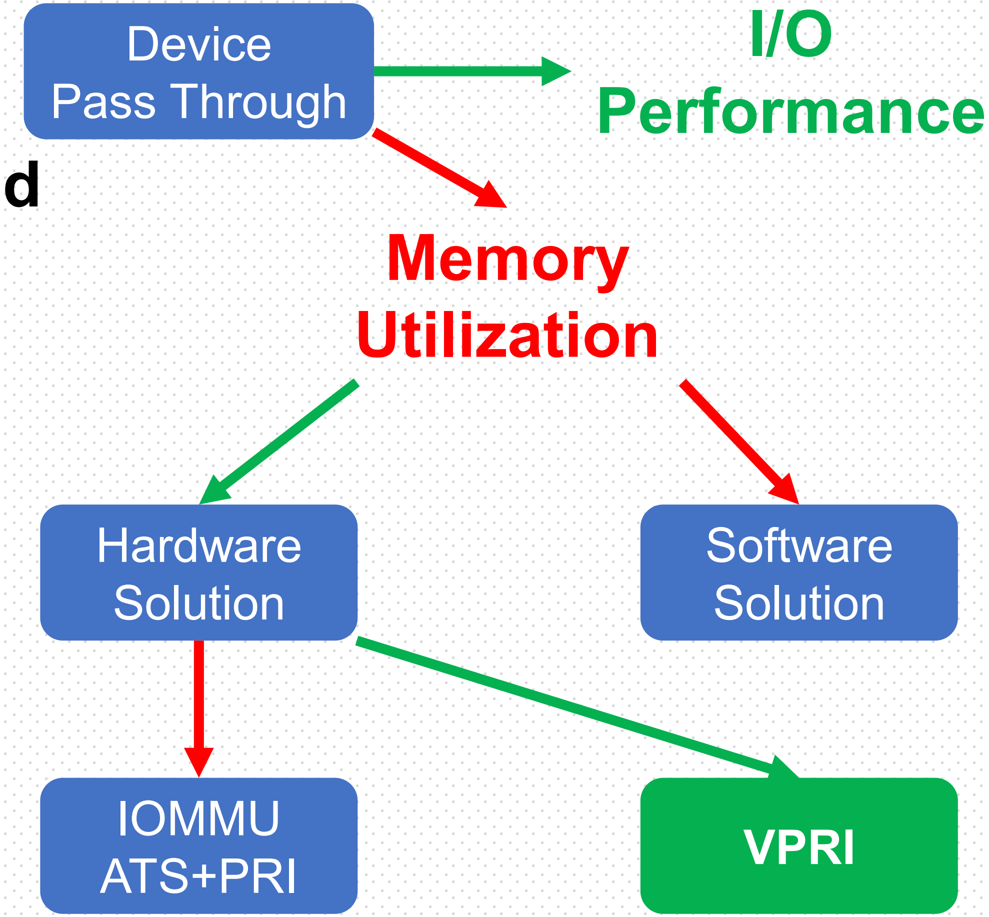
- Para-virtualization
- Guest OS modification and/or performance setback

I/O Page Fault (IOPF): difficult to popularize

- ATS: Address Translation Service
- PRI: Page Request Interface

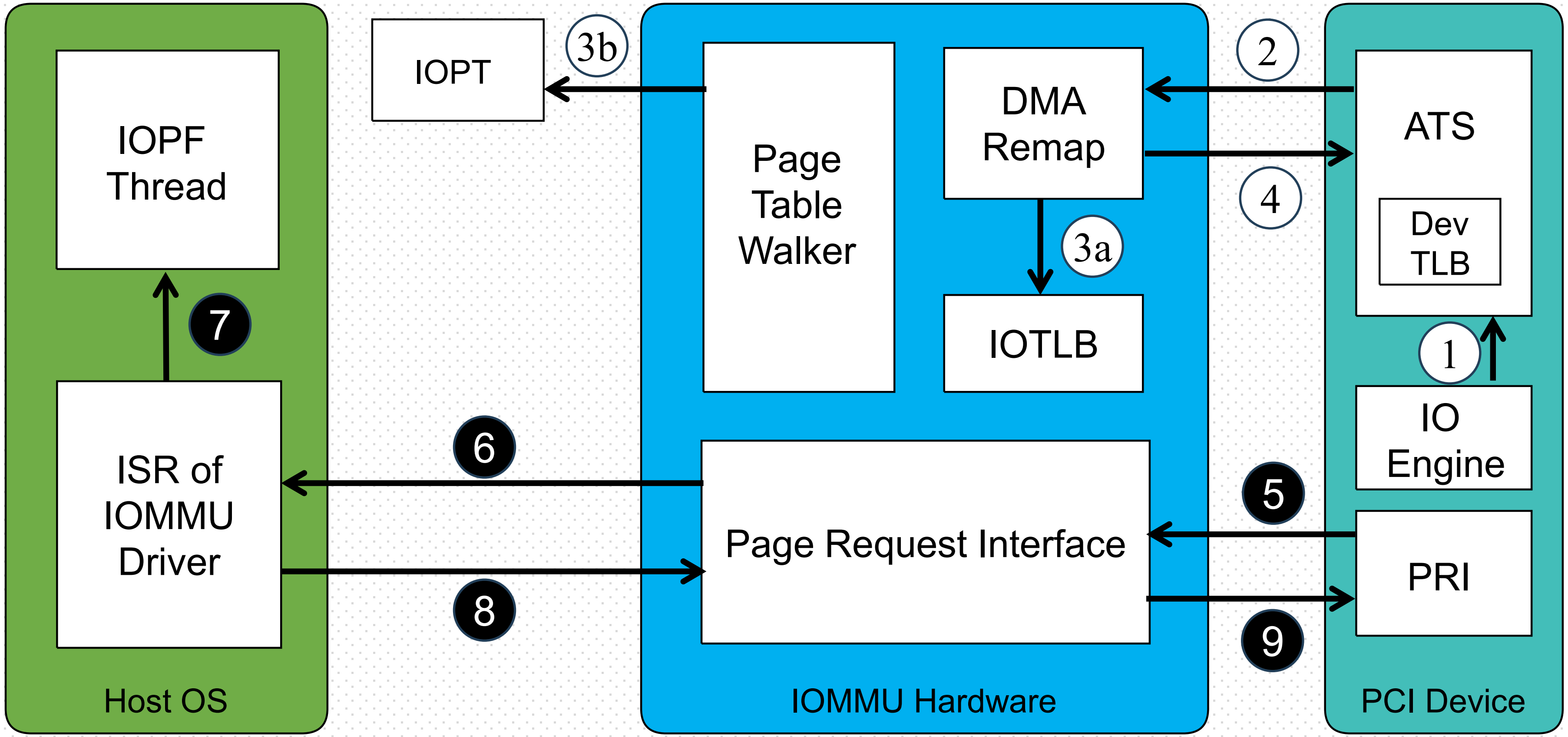
VPRI: Virtualized Page Request Interface

- **Low cost/complexity**
- **Works with existing platforms**
- **Up to 99% reduced IOPFs**



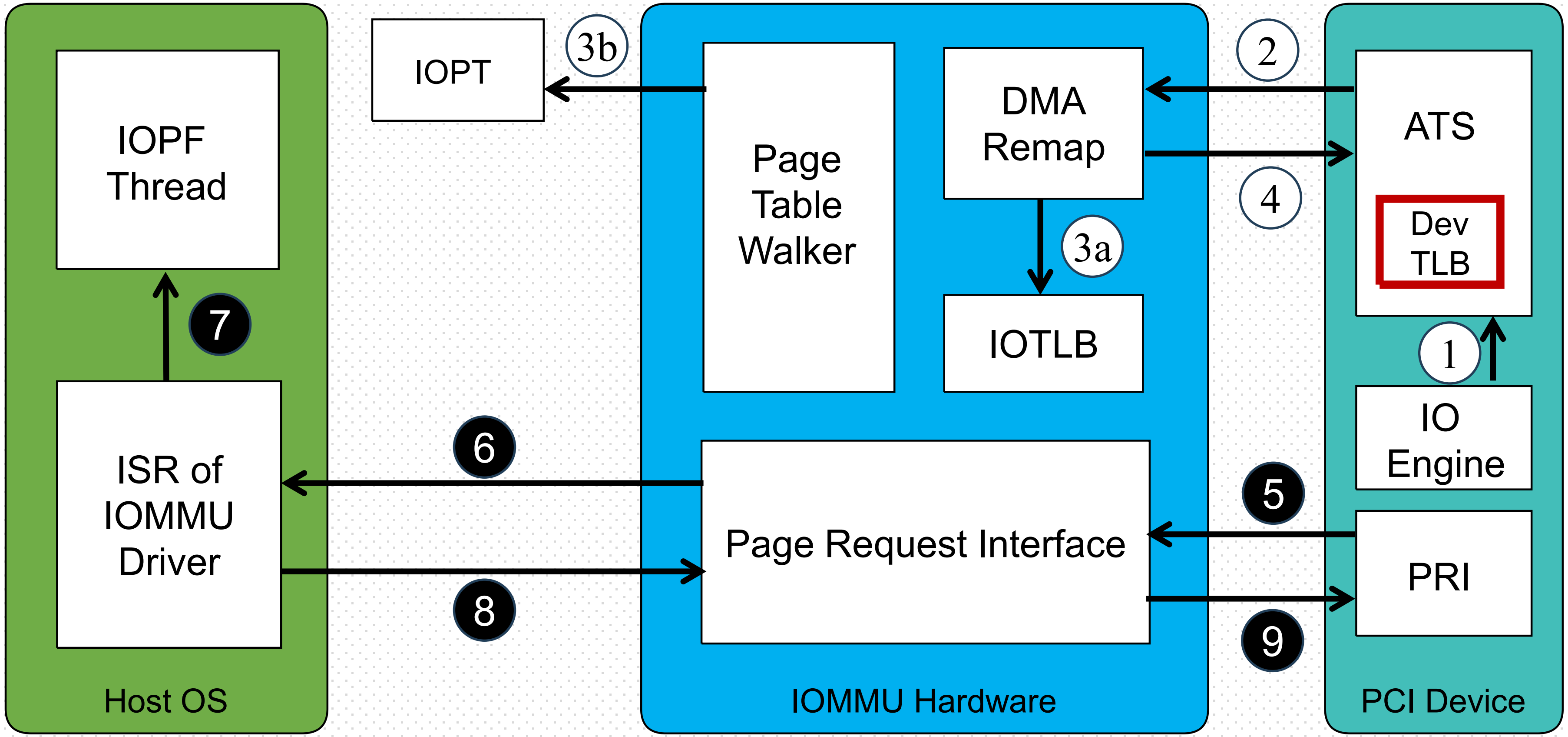
IOPF: IOMMU ATS+PRI

○ Address Translation Service (ATS)   ● Page Request Interface (PRI)



Limitations of IOPF: Standard IOMMU ATS+PRI

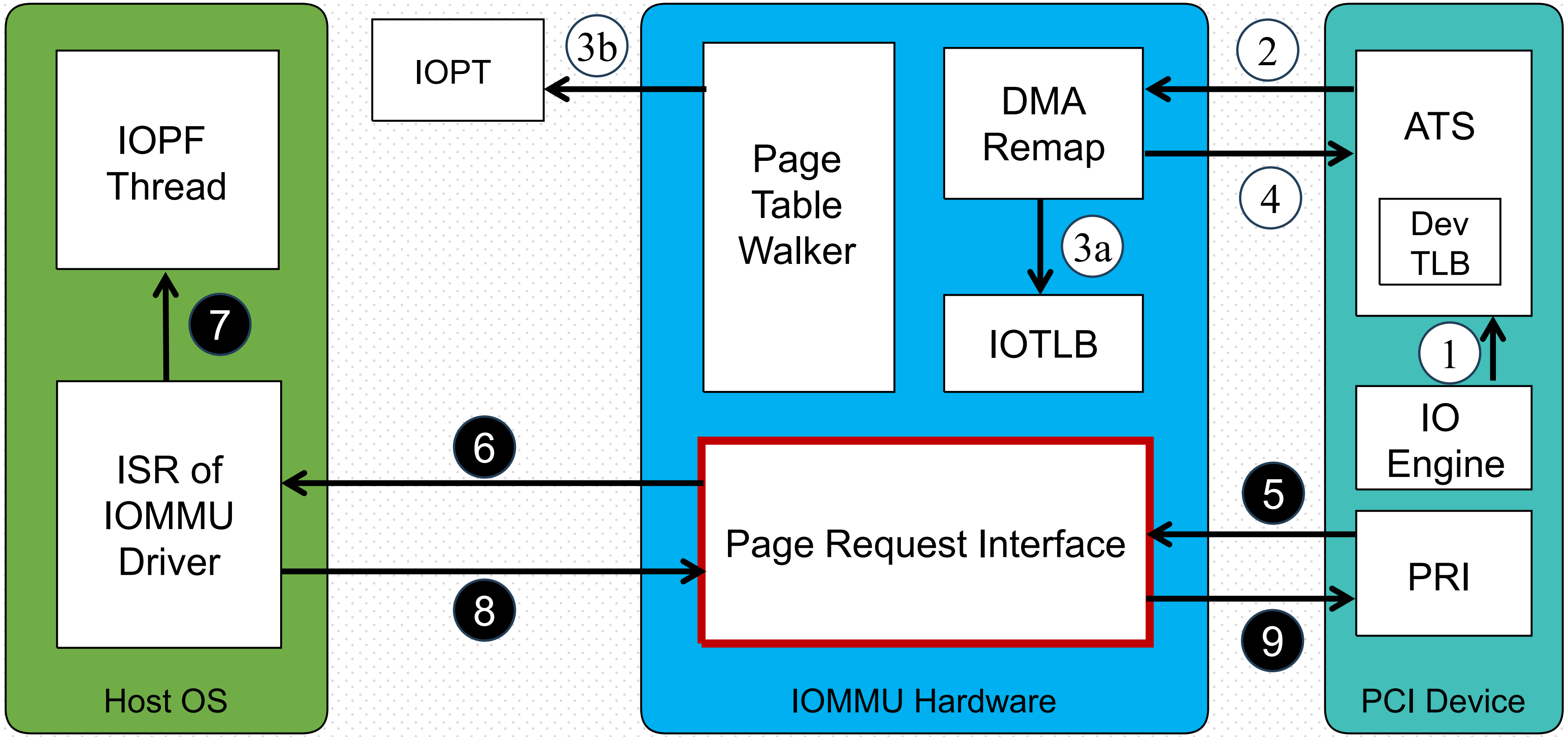
1. Device TLB



Limitations of IOPF: Standard IOMMU ATS+PRI

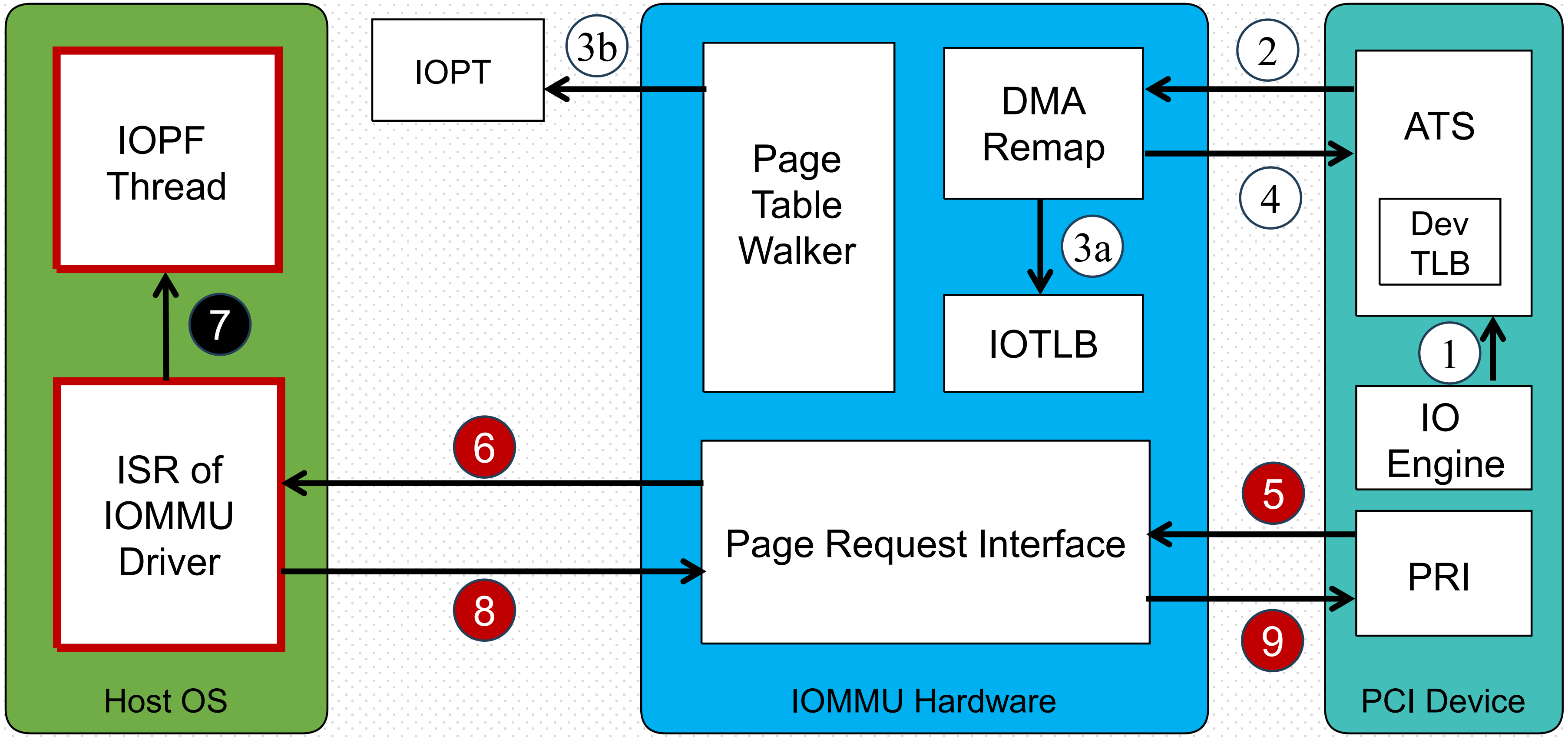
1. Device TLB

2. Compatibility

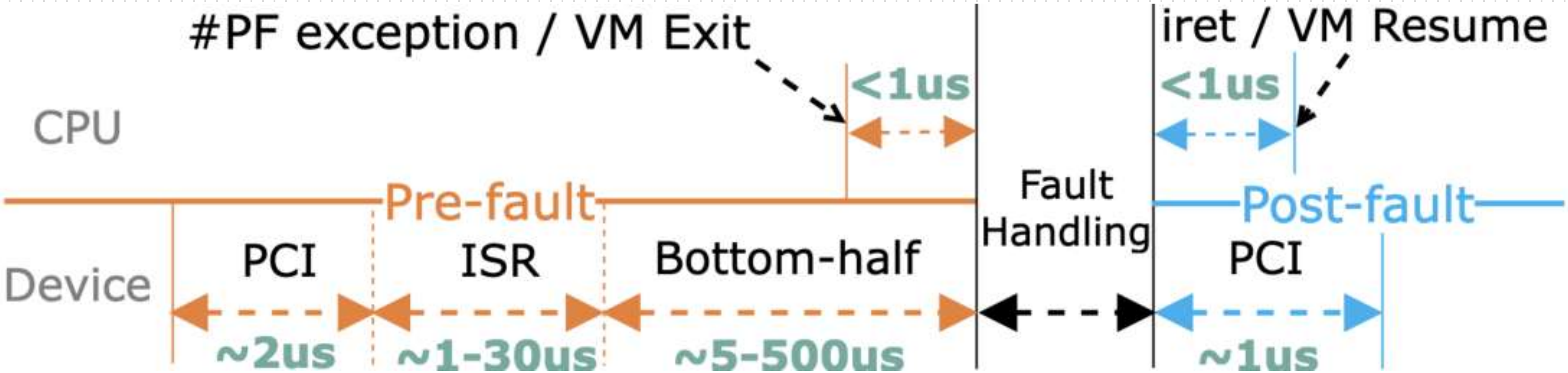


Limitations of IOPF: Standard IOMMU ATS+PRI

1. Device TLB 2. Compatibility 3. Performance



Breakdown of I/O fault v.s. CPU fault



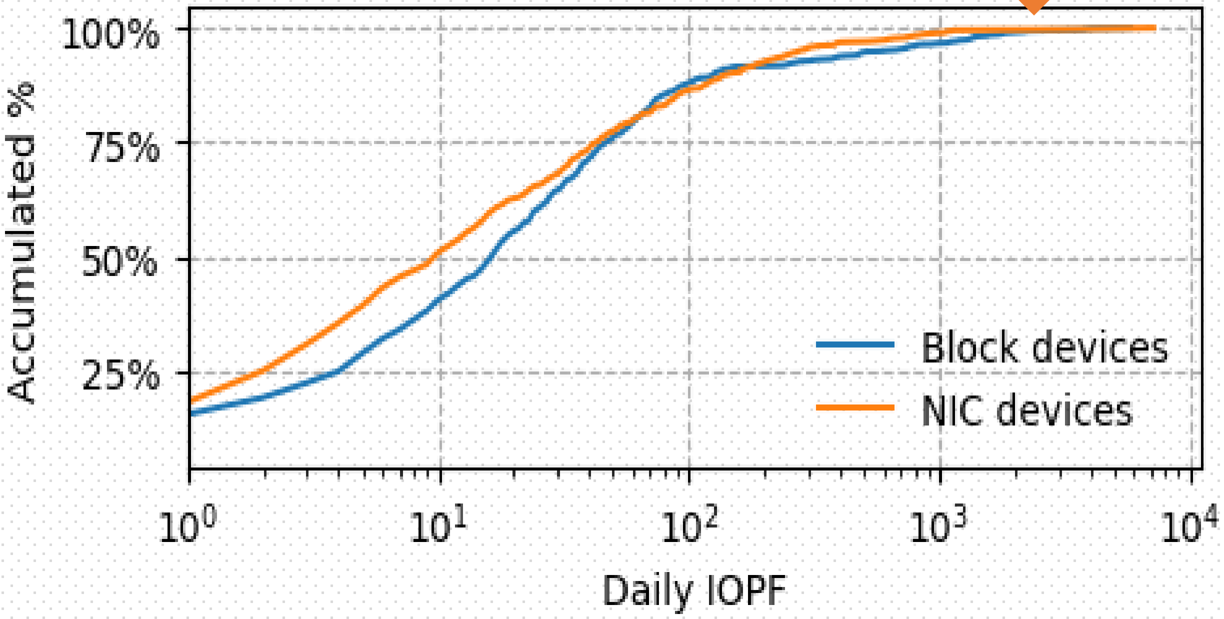
Pre-fault Latency:

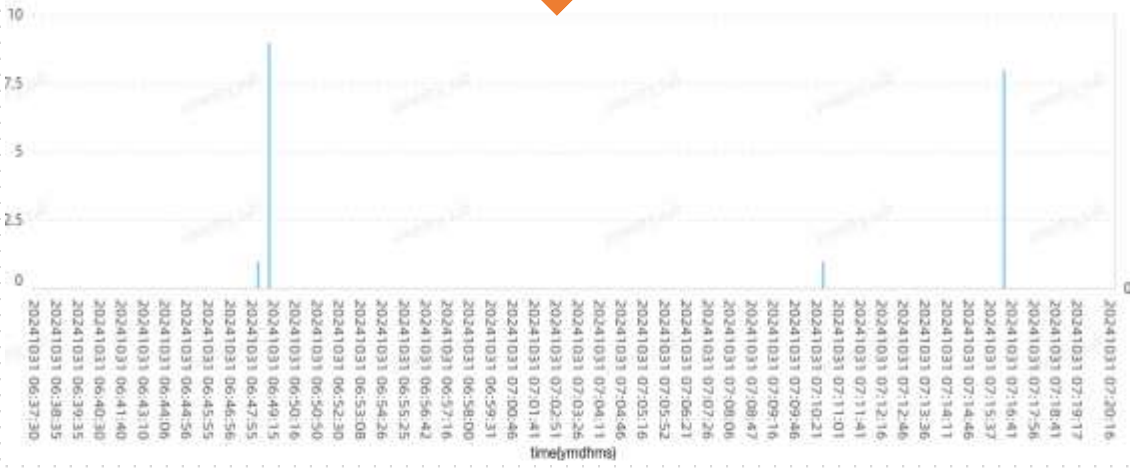
- CPU: < 1us
- I/O: ~10-500 us

Observation of IOPF counts in production

- Block : NIC = 2 : 1
- Variation across VM/devices
- Max count ~3000 per day

- Burst with workloads
- Peak burst IOPF rate > 20/s



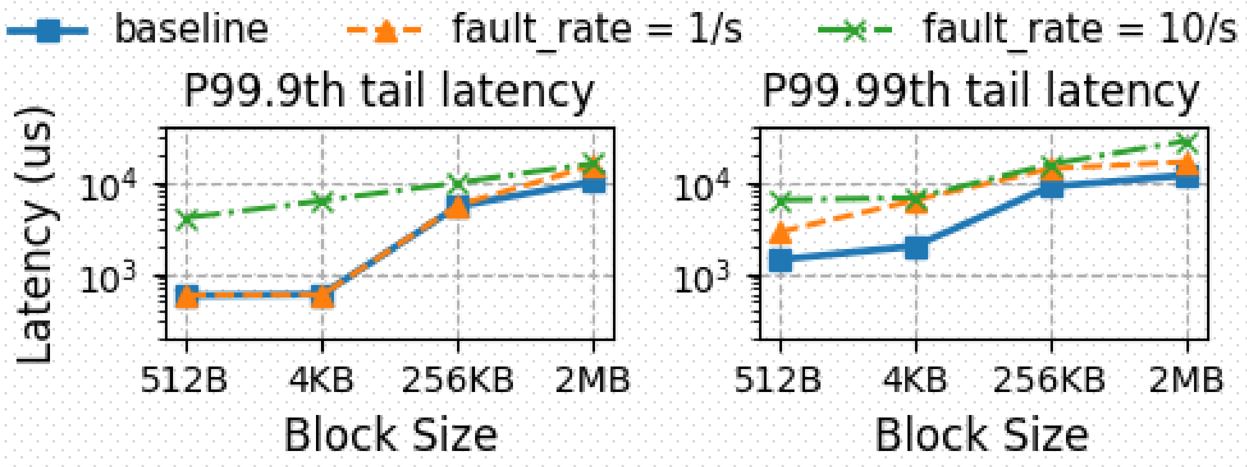


IOPF of a device in 40 minutes

Queue blocked during IOPF handling

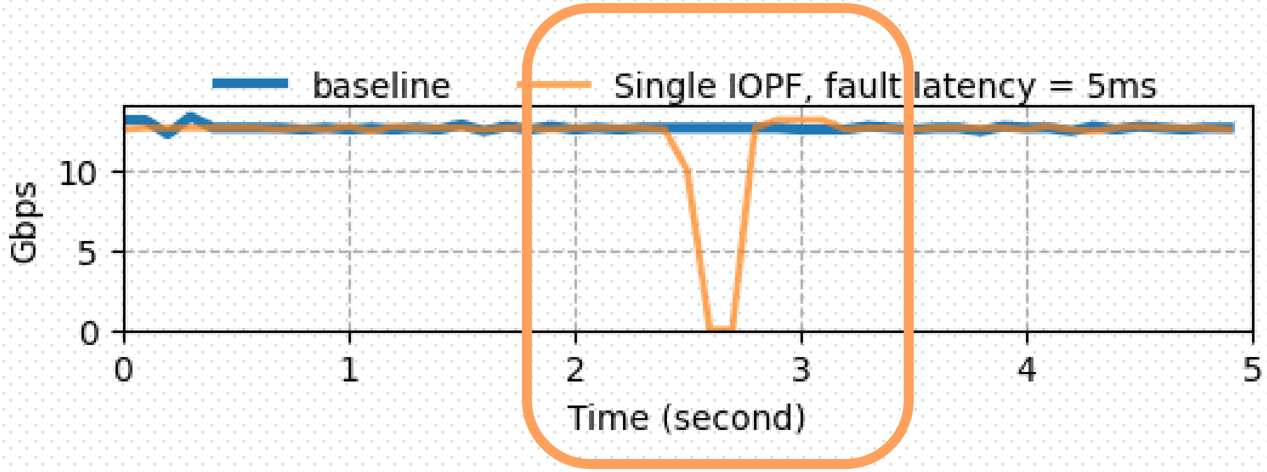
Block devices:

- Hike of long tail I/O
- P99.9TH: hike @ fault rate of 10/s
- P99.99th: hike @ fault rate of 1/s



NIC devices:

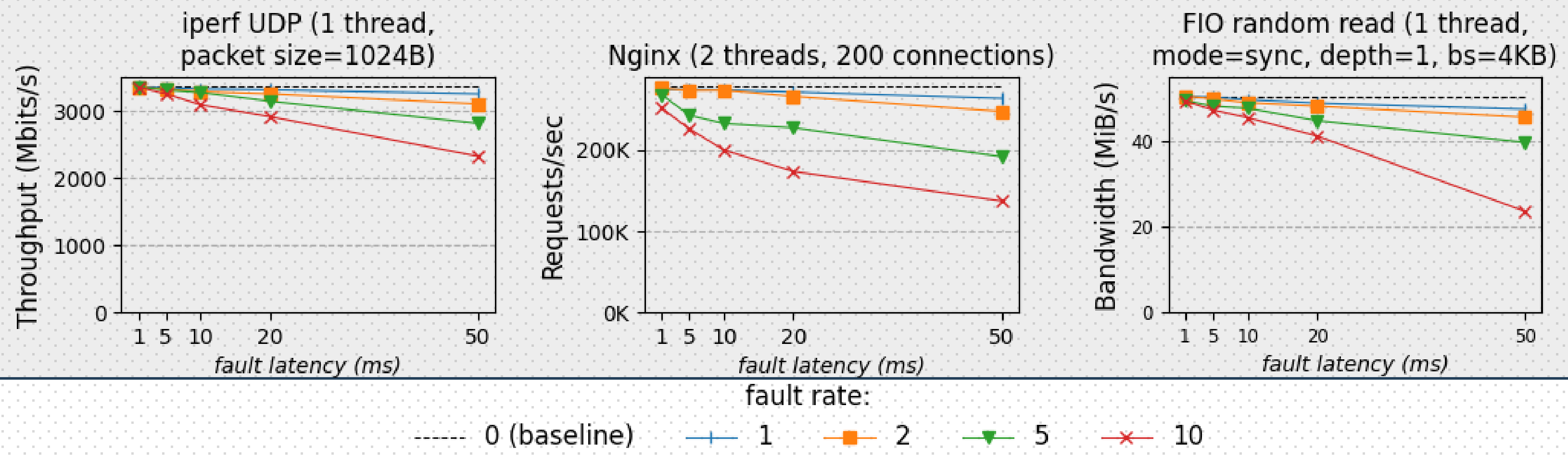
- Packet drop
- Retransmission
- Service interruption



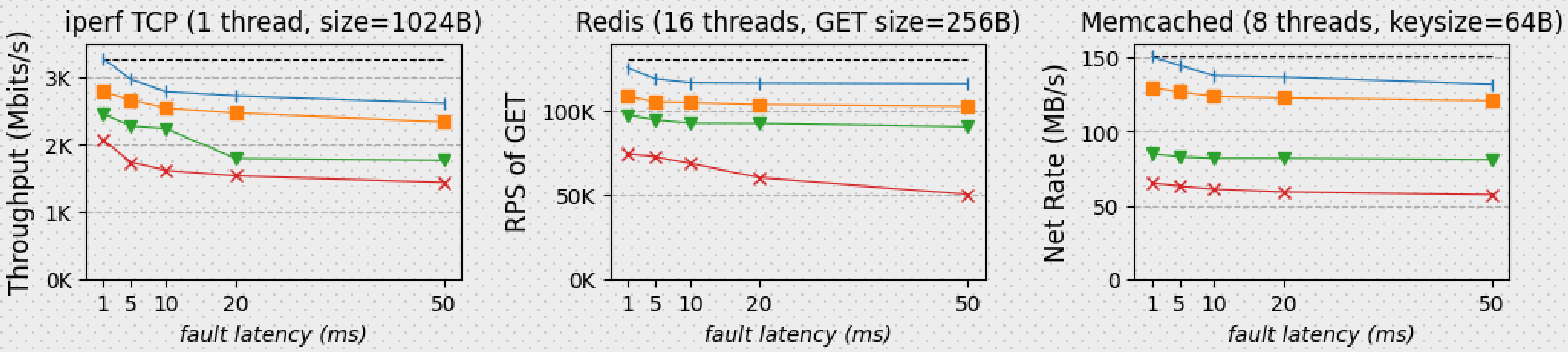
300 ms!

Evaluation: impact to workloads

Non-TCP Workloads



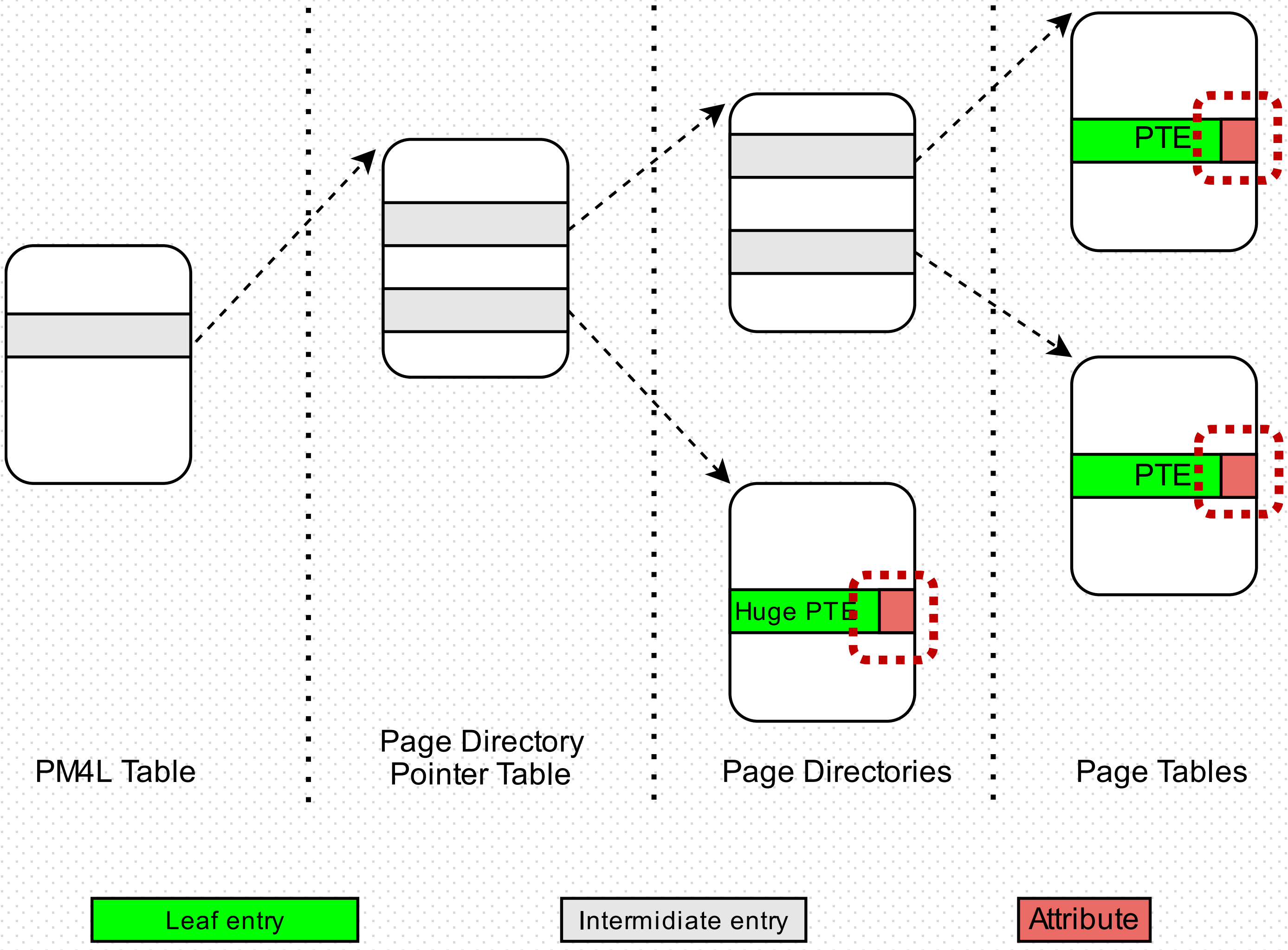
TCP Workloads



Design Goal

Problem	Design Goal	Solution
Fault detection needs ATS	No ATS/MMU in detecting faults	???
Fault reporting needs PRI	No PRI in reporting faults	???
Performance impact of IOPF	Reduce the rate	???

Challenge #1: fault detection



- Page tables are huge
- But attributes are small (2 bits)

- Page walk is costly
- But fault detection can be simple

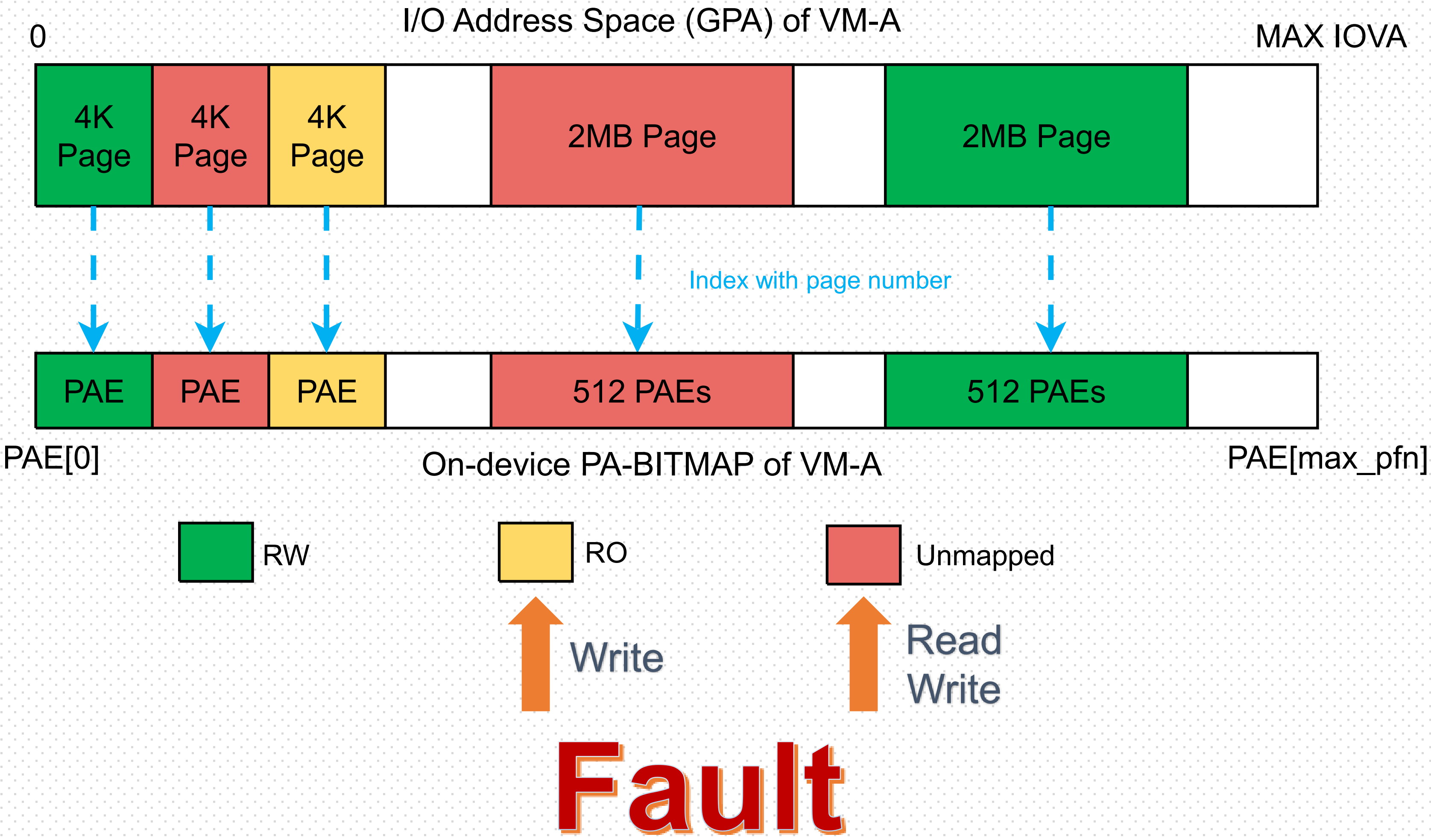
PA-BITMAP: Coherent on-device page attributes bitmap

Page Attribute Entry (PAE):

❑2 bits per PAE

❑4KB GPA/IOVA per PAE

❑On-device memory



I/O Address Space (GPA) of VM-A

0

MAX IOVA

4K Page

4K Page

4K Page

2MB Page

2MB Page

Index with page number

PAE

PAE

PAE

512 PAEs

512 PAEs

PAE[0]

On-device PA-BITMAP of VM-A

PAE[max_pfn]

🟩 RW

🟨 RO

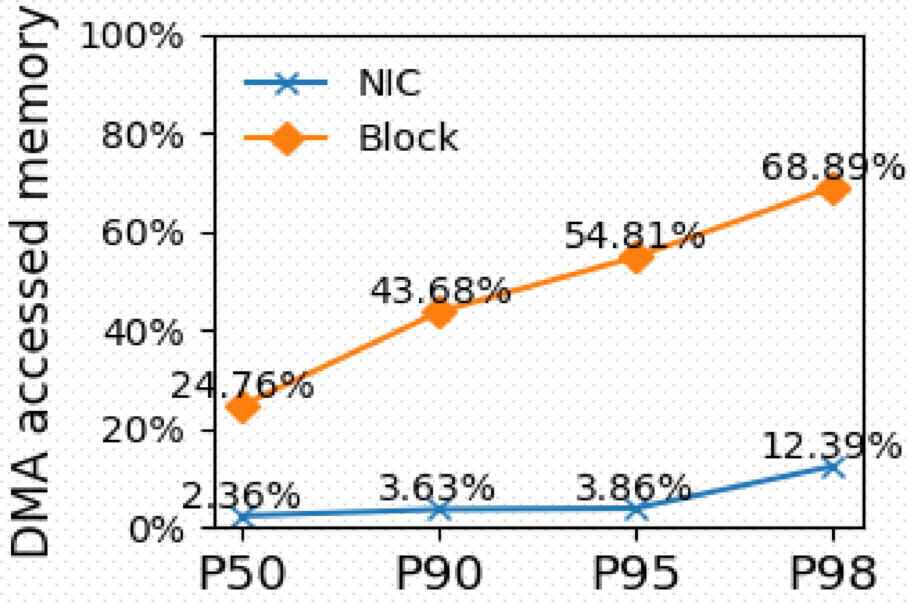
🟥 Unmapped

Write

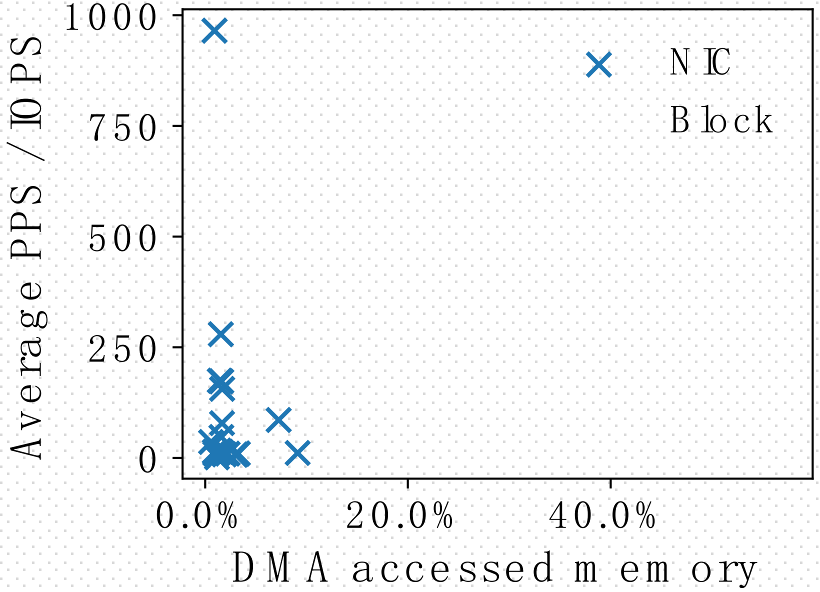
Read
Write

Fault

Challenge #2: reduce IOPF rates



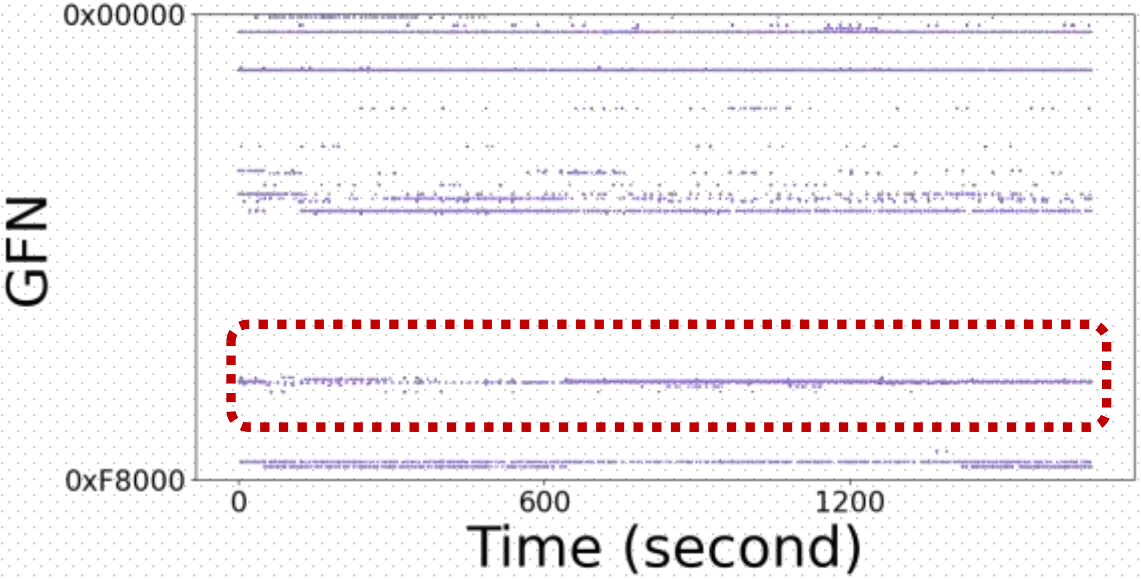
Footprints distribution in 30 minutes

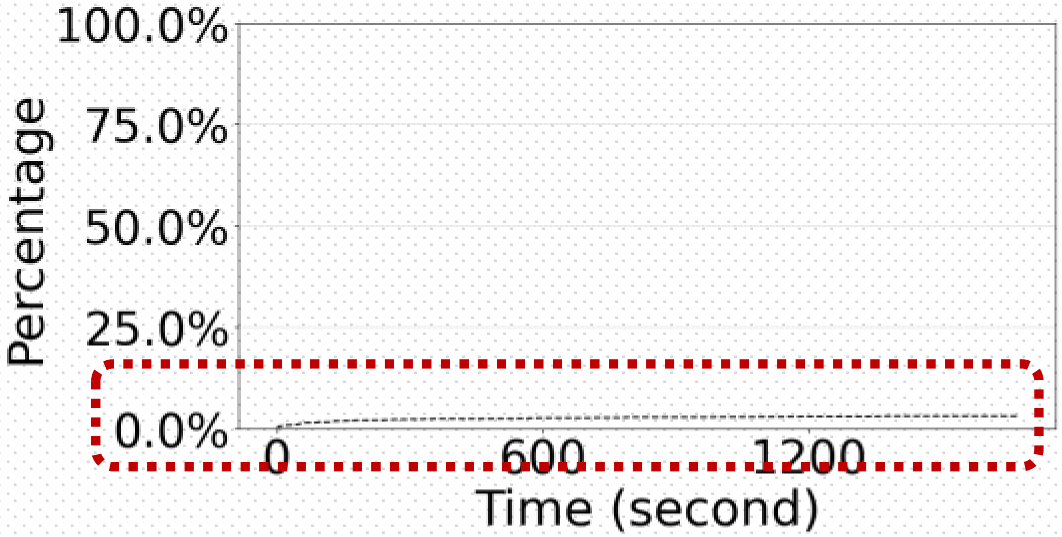


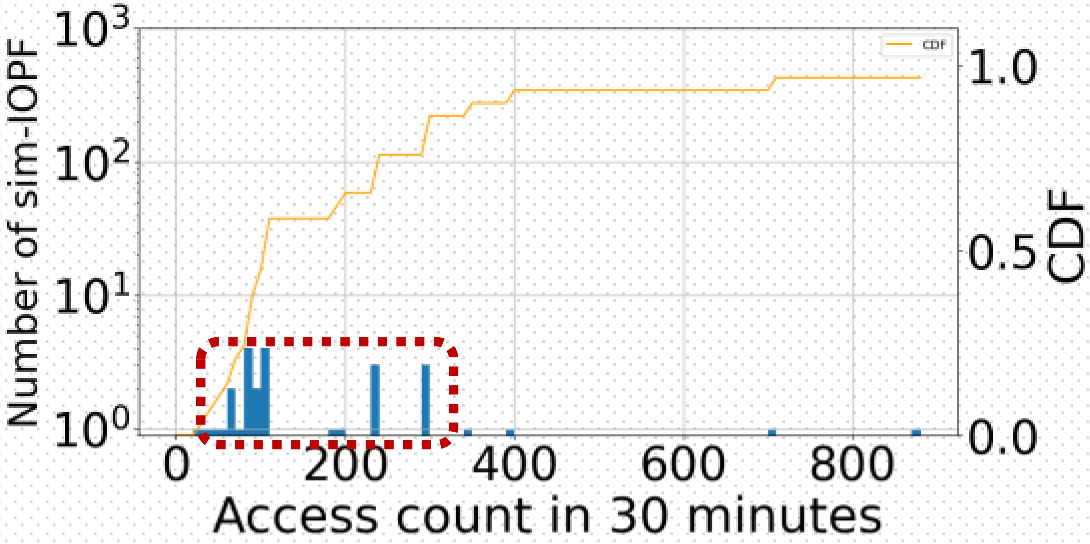
Footprints distribution in 30 minutes

Challenge #2: NIC characterization

- (Almost) bounded buffer
- Very strong temporal locality
- High return by pinning with LRU

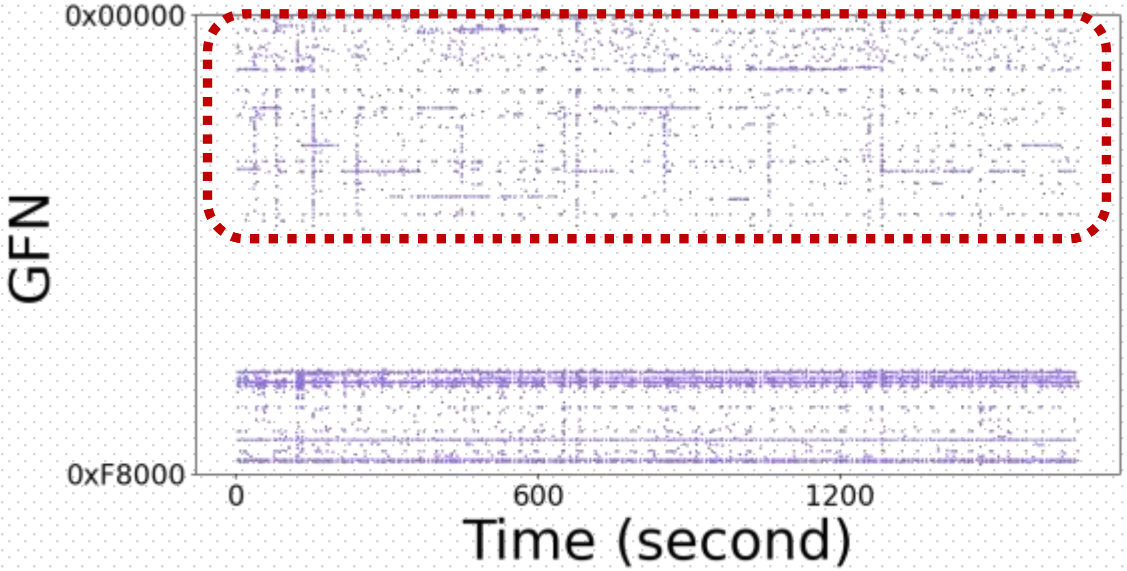


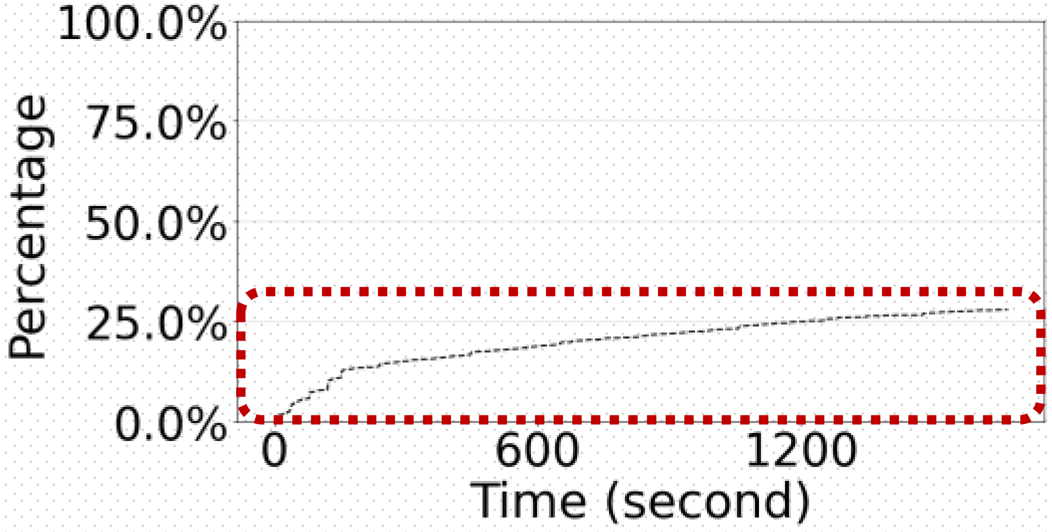


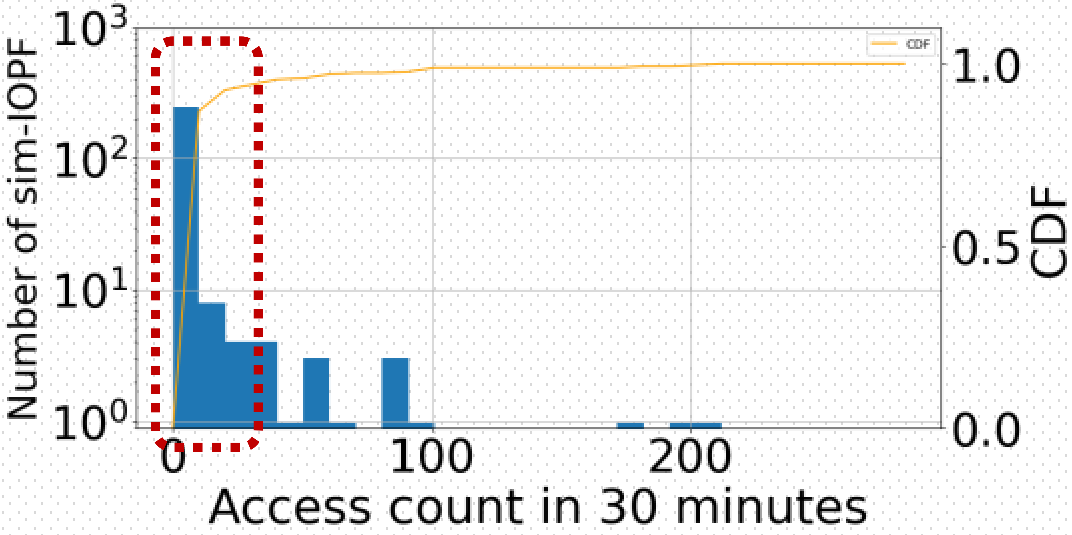


Challenge #2: Storage characterization

- Unbounded buffer (page cache)
- Weaker temporal locality
- >70% page will be visited > twice
- Long access distance







Key Takeaways

PA-BITMAP:

Break dependency on ATS

Customized PCIe interface:

Break dependency on PRI

DMA access tracking:

Device level

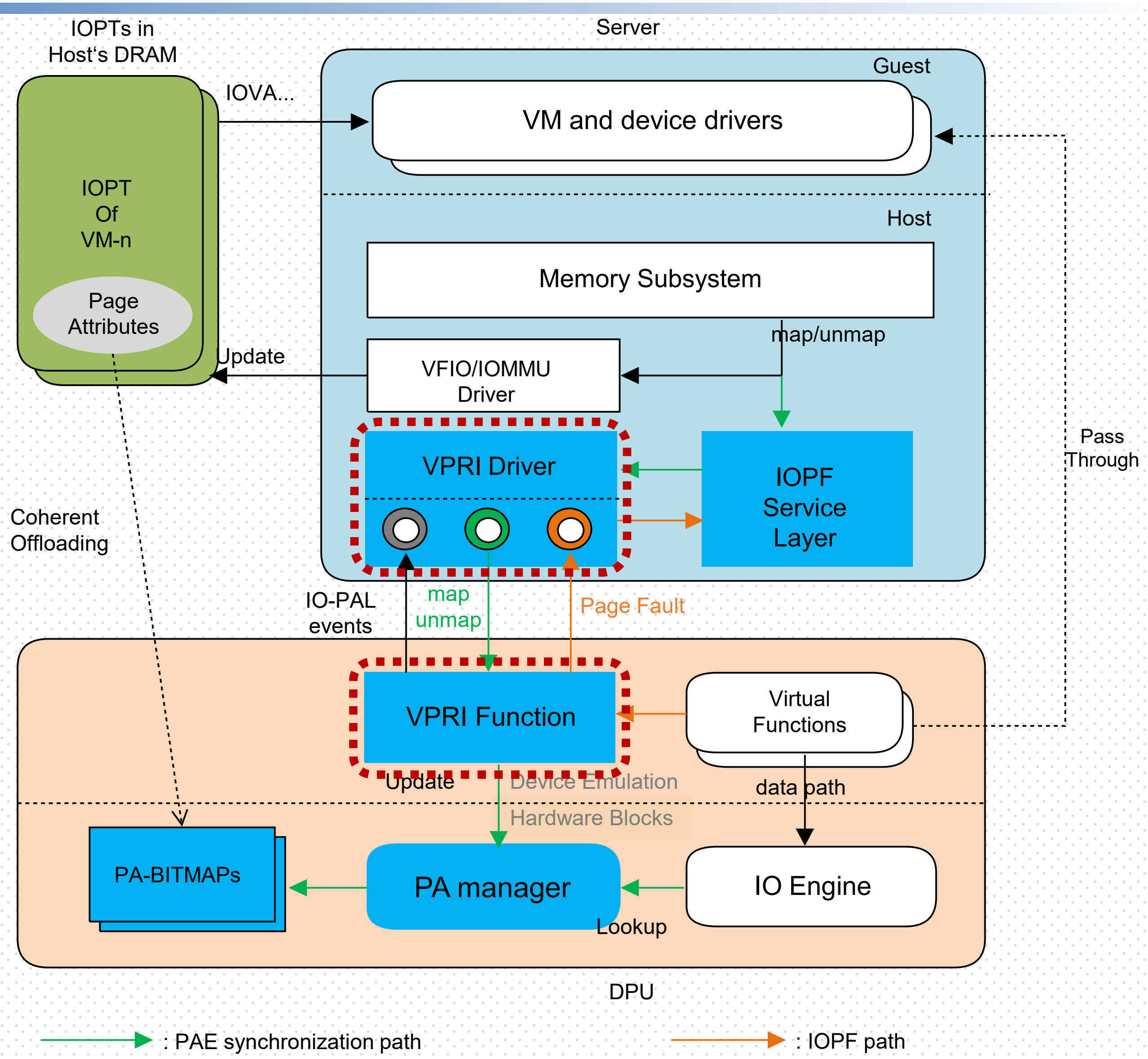
Gap in today’s hardware

Software pinning policy:

Minimum pin ratio

Maximum IOPF reduction ratio

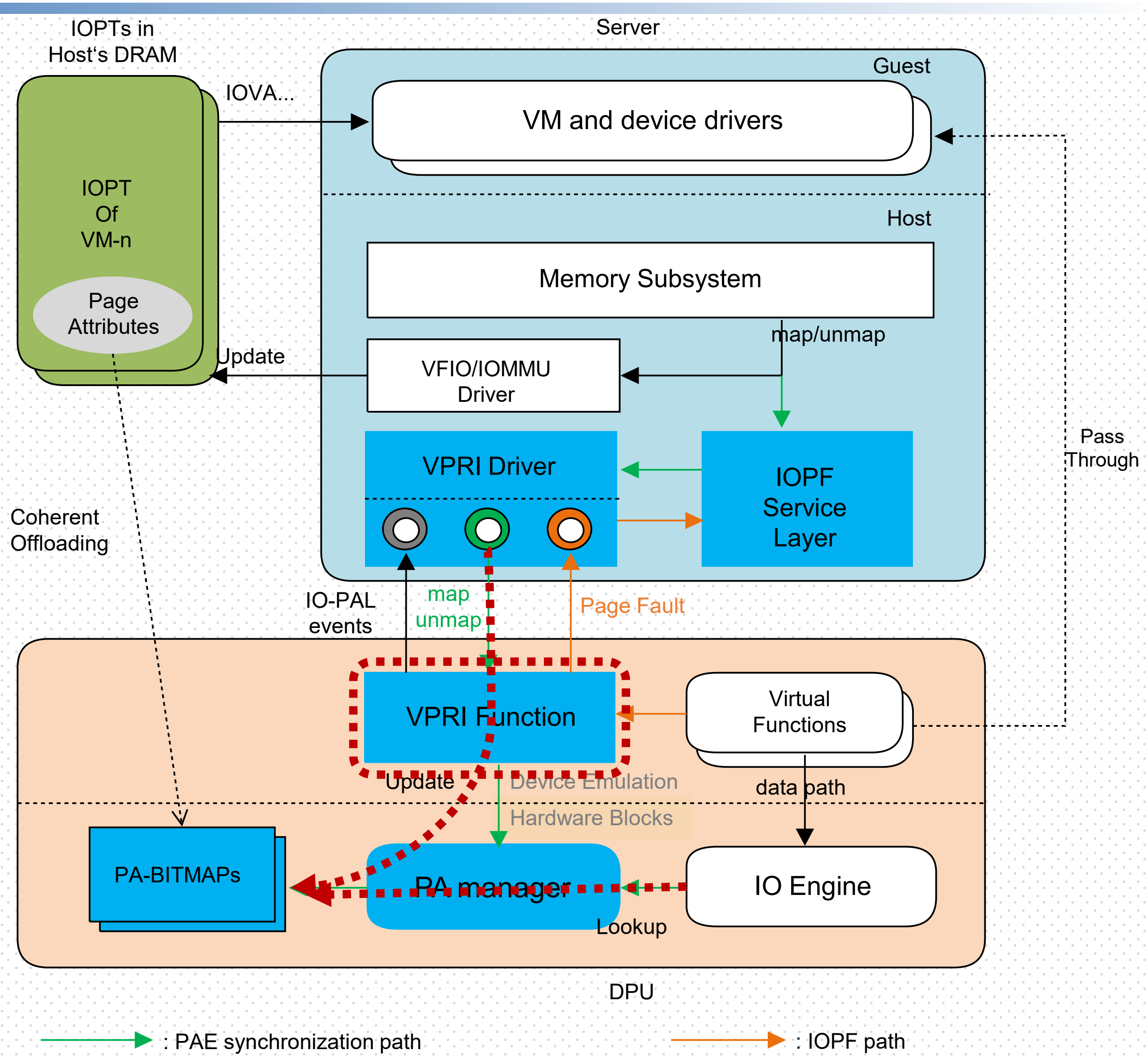
VPRI Hardware Overview



VPRI Hardware Overview

Fault detection:

- On-device PA-BITMAP
- Free from ATS



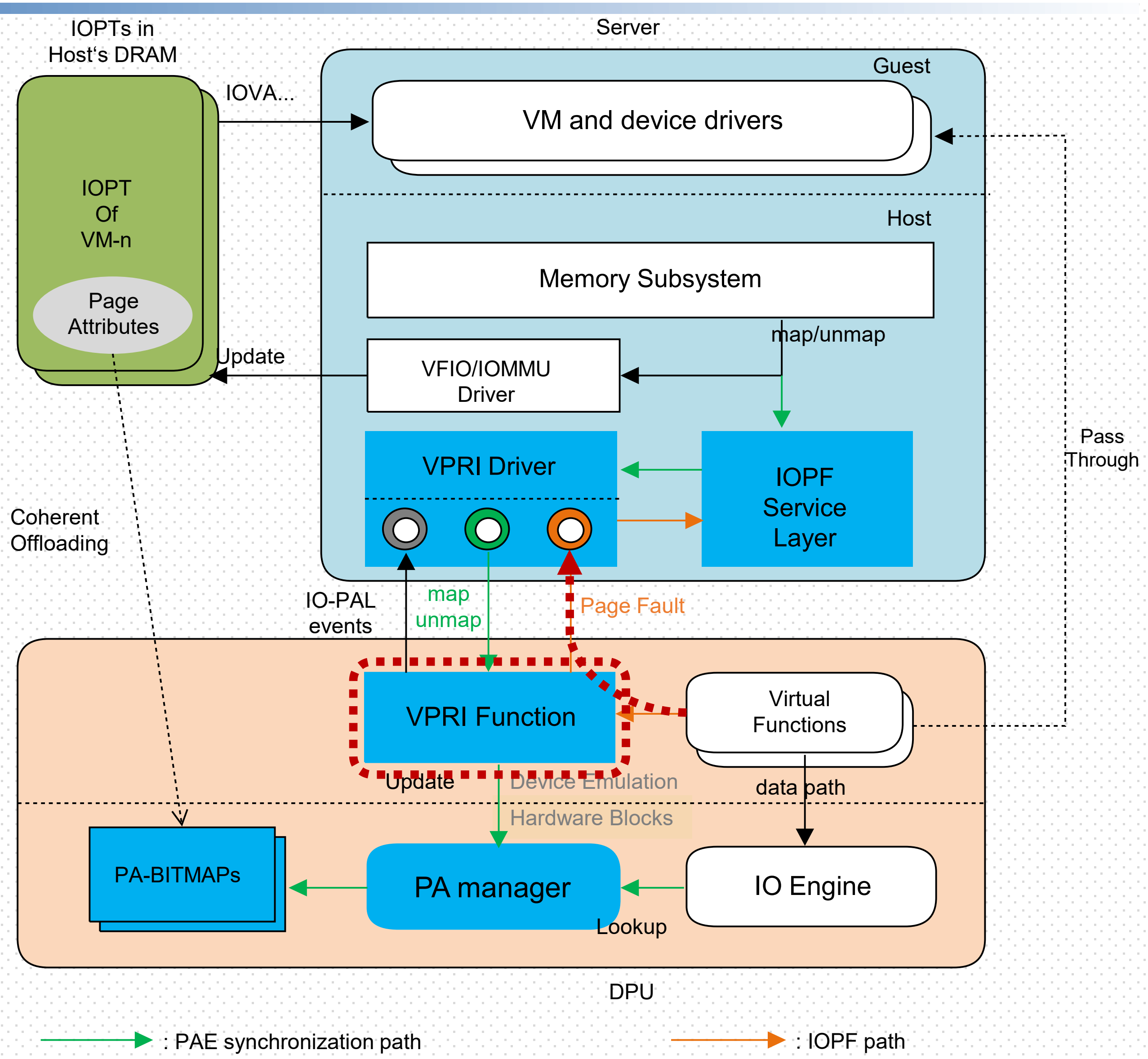
VPRI Hardware Overview

Fault detection:

- On-device PA-BITMAP
- Free from ATS

Fault reporting:

- Sideband channel
- Free from PRI



VPRI Hardware Overview

Fault detection:

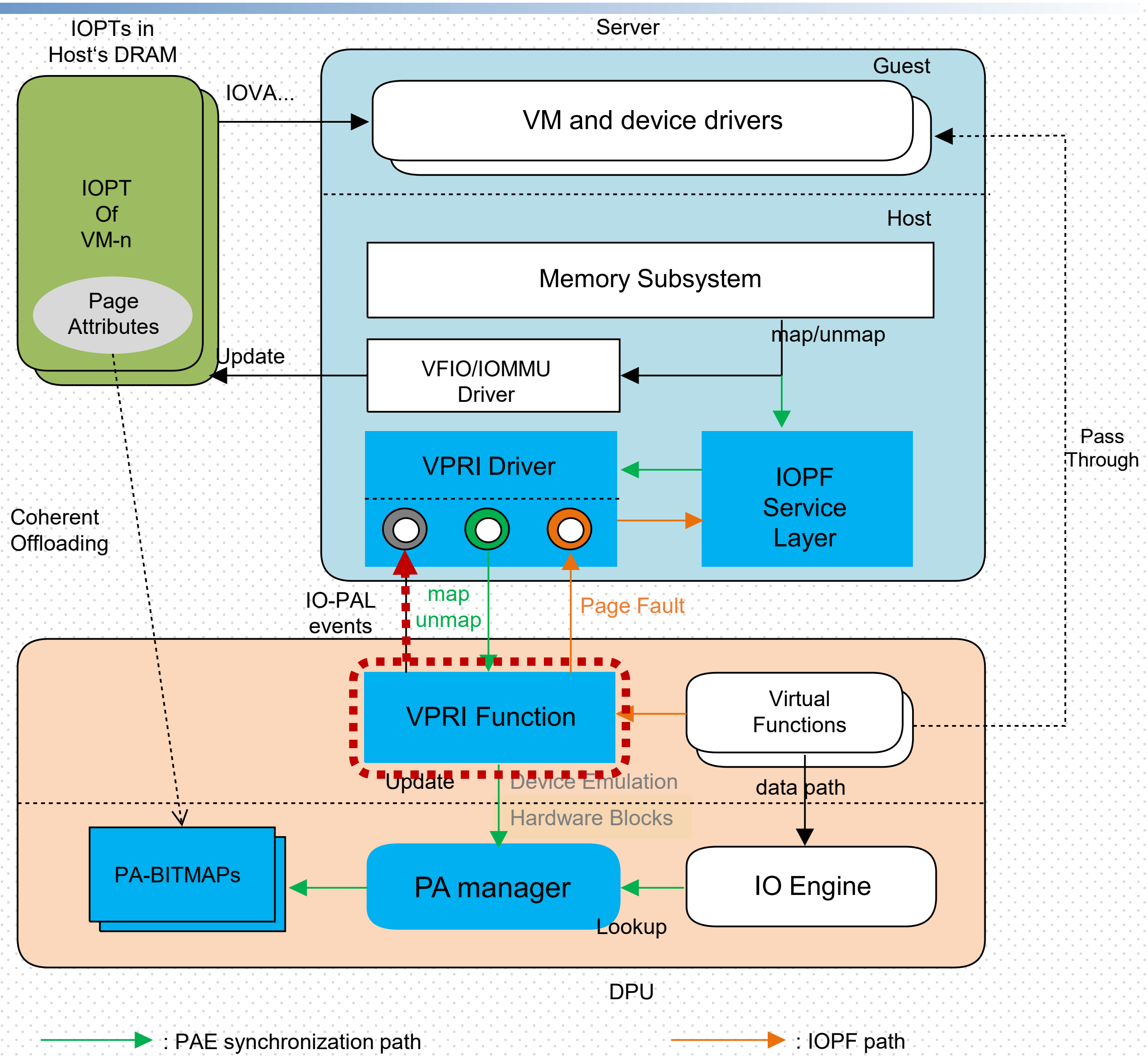
- On-device PA-BITMAP
- Free from ATS

Fault reporting:

- Sideband channel
- Free from PRI

Performance opt:

- NIC: > 95% reduction
- Block: > 50% reduction



VPRI Hardware Overview

Fault detection:

- On-device PA-BITMAP
- Free from ATS

Fault reporting:

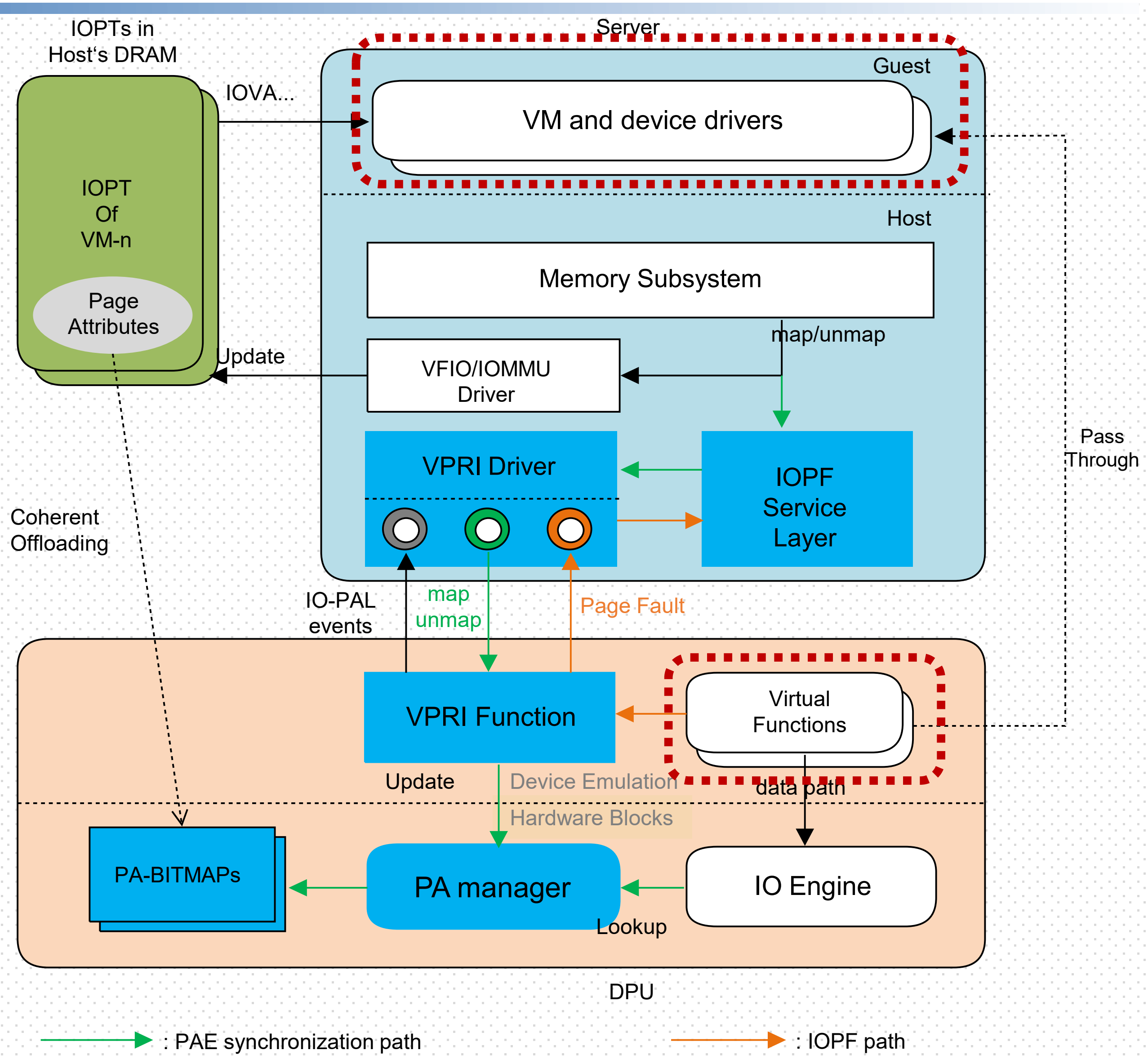
- Sideband channel
- Free from PRI

Performance opt:

- NIC: > 95% reduction
- Block: > 50% reduction

Zero changes to VM:

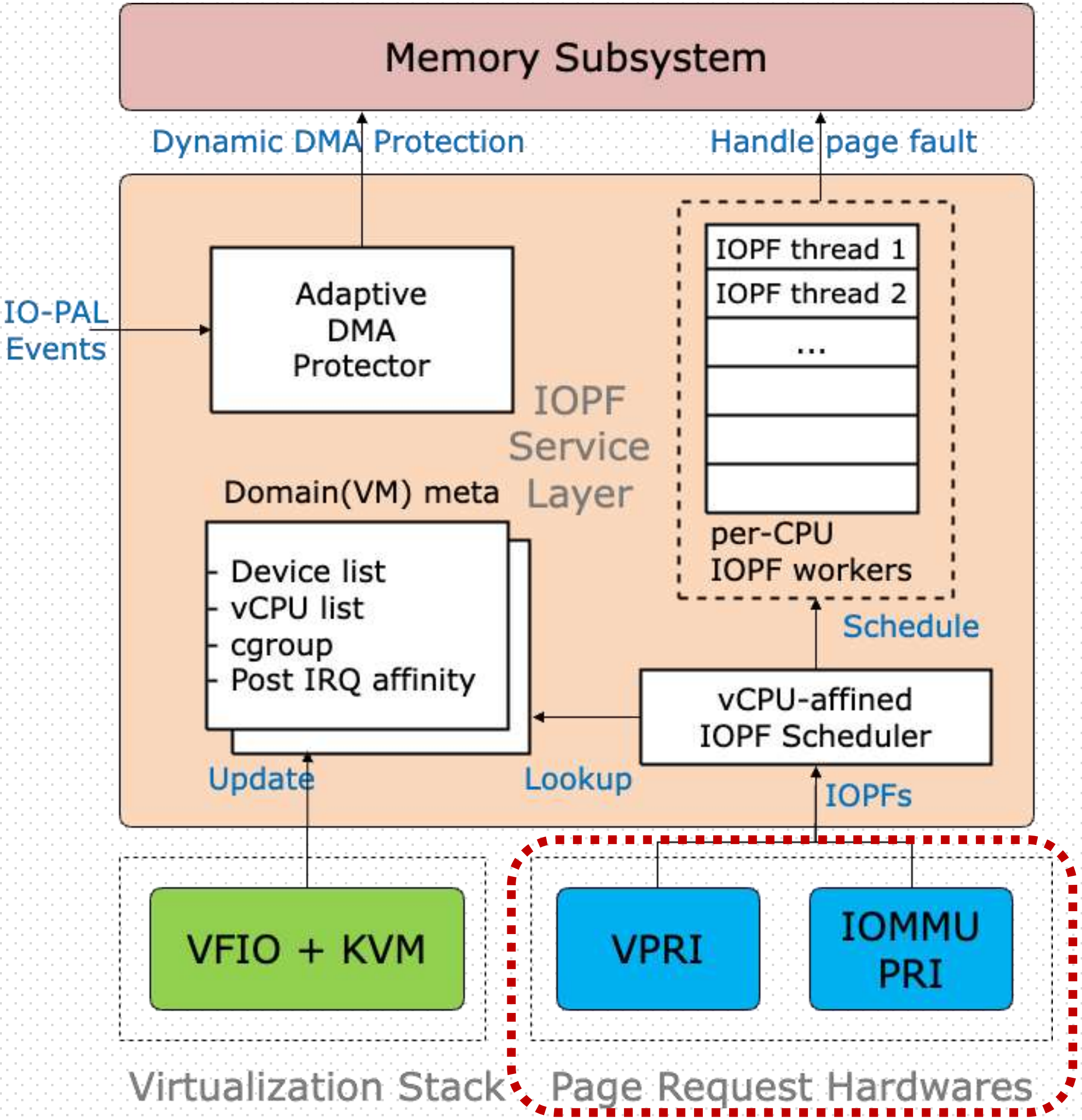
- Device
- Driver



VPRI Software Overview

IOPF HW Abstraction Layer:

- VPRI driver
- IOMMU PRI driver



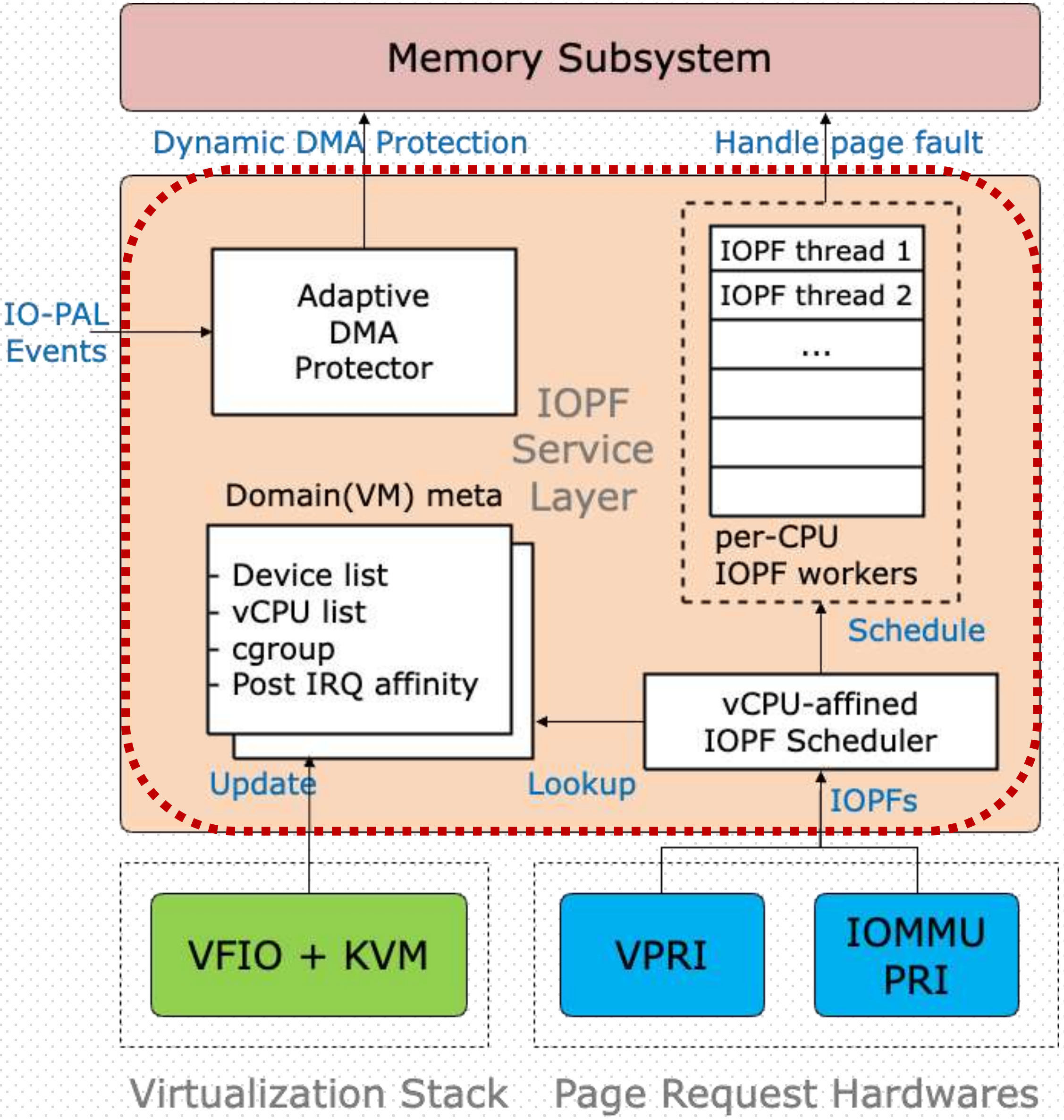
VPRI Software Overview

IOPF HW Abstraction Layer:

- VPRI driver
- IOMMU PRI driver

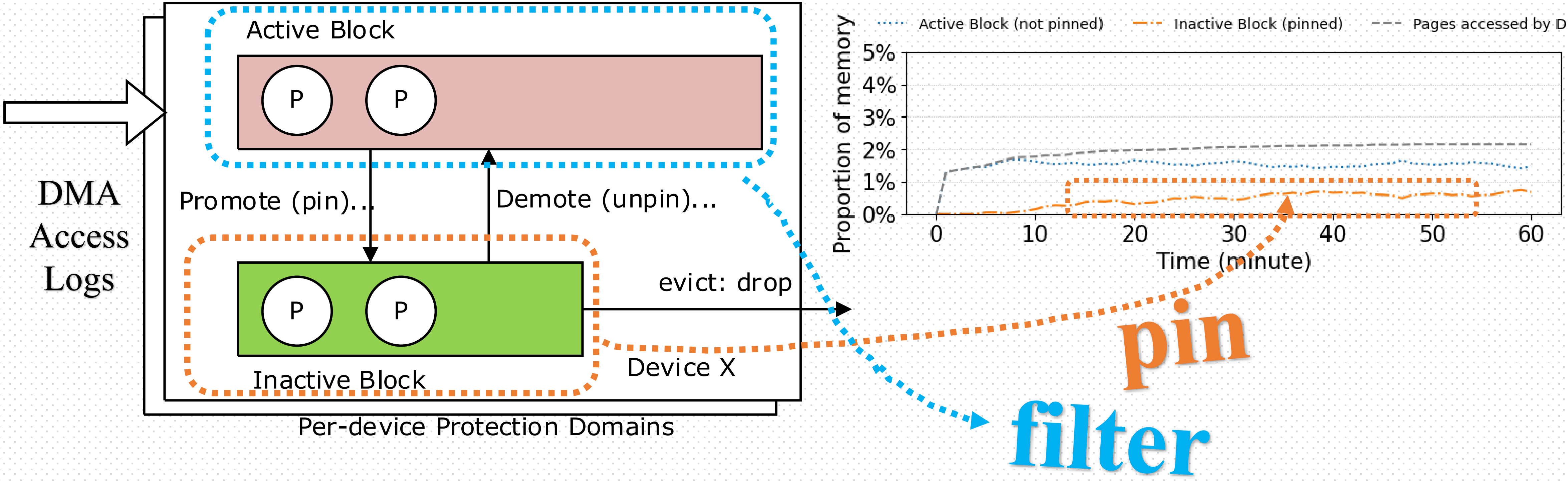
IOPF Service Layer:

- Device-domain mapping
- vCPU-affined IOPF scheduling
- Fault handling
- Adaptive DMA Protector



ADP: Adaptive DMA Protector

- **Active DMA pages are not likely to be swapped out.**



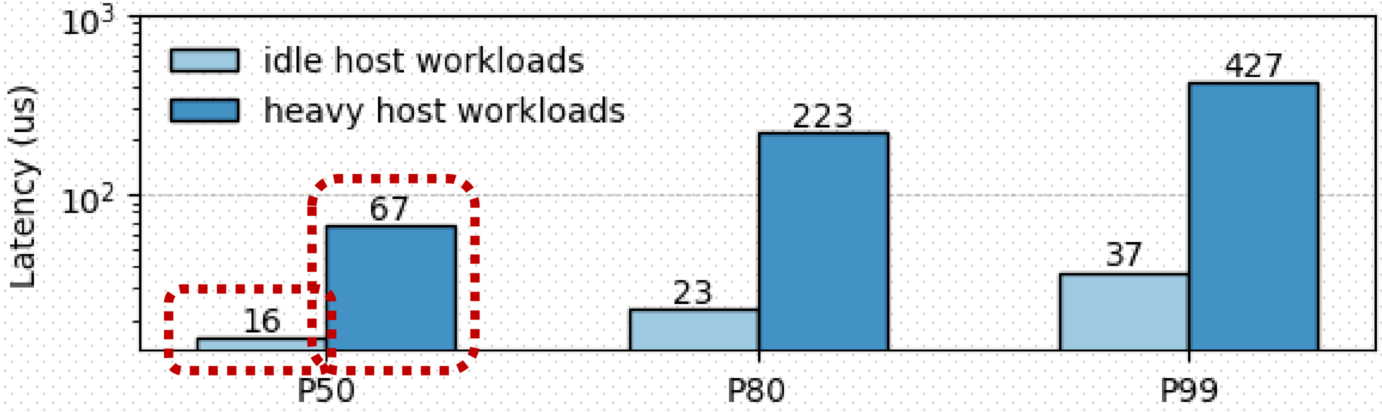
- **(Temporarily) Inactive DMA pages are more prone to IOPF**

Evaluation: Hardware

Bitmap

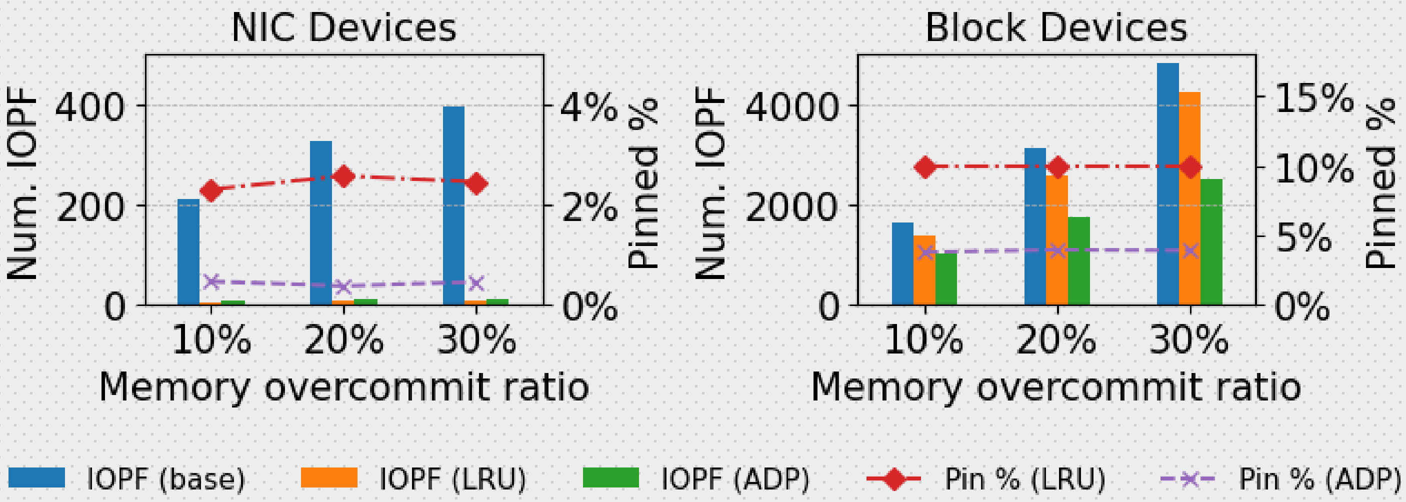
Metric	On-device lookup (ns)	Host update (ns)
Avg.	69	1,742
P99th	92	2,341

Page Fault RTT

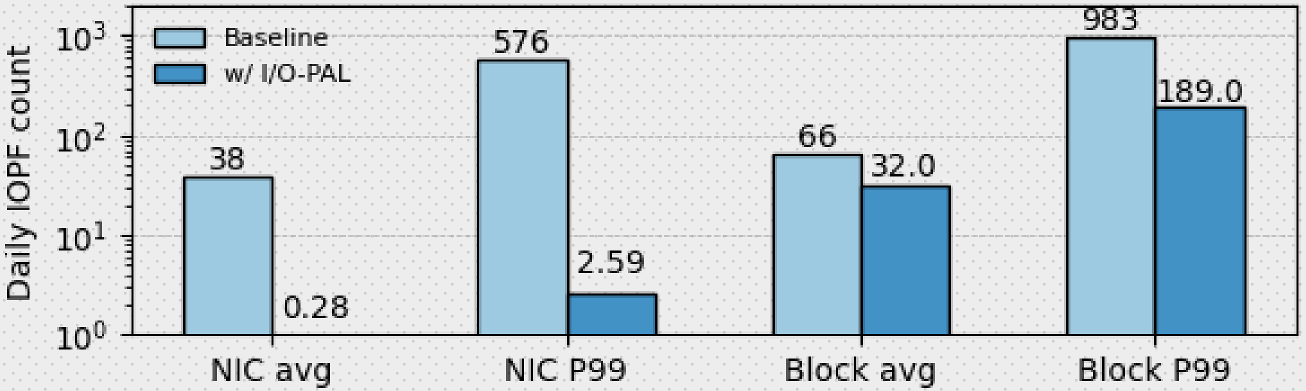


Evaluation: IOPF Reduction

- **Memory overcommit:**
  - 10%-30%
- **Mixed workloads:**
  - Network
  - Storage
- **Result (cmp w/ LRU):**
  - NIC: 6.28x
  - Block: 10.56x



- **In Production**
  - NIC: > 99% reduction
  - Block > 50% reduction
  - 5.2% pinning per VM



Summary

- ❑Deployed in Alibaba Cloud
  - ❖Negligible HW cost
  - ❖Compatible with all x86 platforms
  - ❖Landed in production in just 6 months
- ❑Facilitates memory overcommitment
- ❑Other use cases:
  - ❖Post-copy opt.
  - ❖Page migration
  - ❖Fast boot
  - ❖Etc.

SOSP 2024

VPRI: Efficient I/O Page Fault Handling via Software-Hardware Co-Design for IaaS Clouds

Kaijie Guo, Dingji Li, Ben Luo, Yibin Shen, Kaihuan Peng, Ning Luo, Shengdong Dai, Chen Liang, Jianming Song, Hang Yang, Xiantao Zhang, Zeyu Mi

Thank you for you attention!