SSICLOPS

Persistence in Networking: Redesigning Stack, API and Networks

Michio Honda (NEC)
Lars Eggert (NetApp)
Douglas Santry (NetApp)

Munich Internet Research Retreat

May 22th 2017

Published at HotNets 2016
https://goo.gl/Ssreei

Motivation

SSICLOPS

- Non-Volatile Main Memories (NVMMs)
  - Persistent
  - Byte-addressable
  - Low latency
    - 10s-1000s of ns

https://www.hpe.com/us/en/servers/persistent-memory.html

- Shift from ***block***- to ***byte***-granularity persistency
  - OS abstractions
    - ***Direct*** access to mmap()-ed ***files***
  - Data structures
    - Filesystems and databases

What are implications for networking?

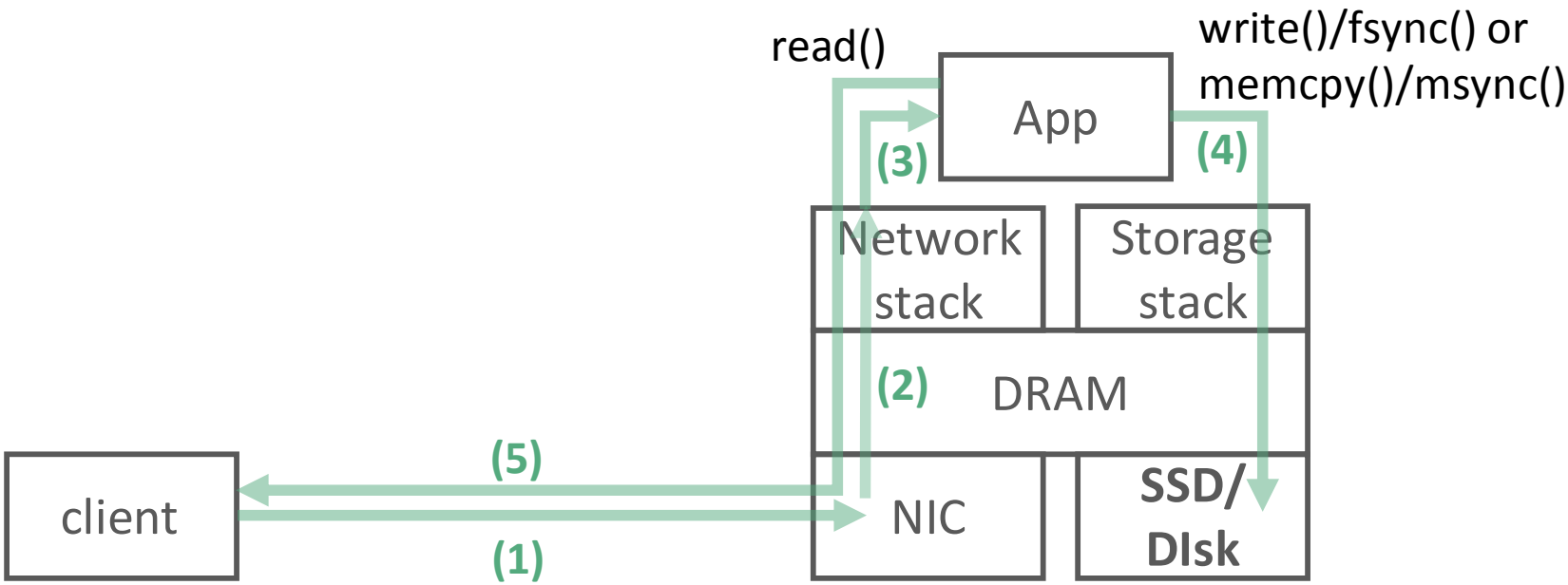
Case Study: Careful Data Transfer

- Server persists client's request prior to acknowledgment
- e.g., 1KB commit:

Case Study: Careful Data Transfer

SSICLOPS

- Server persists client's request prior to acknowledgment
- e.g., 1KB commit:

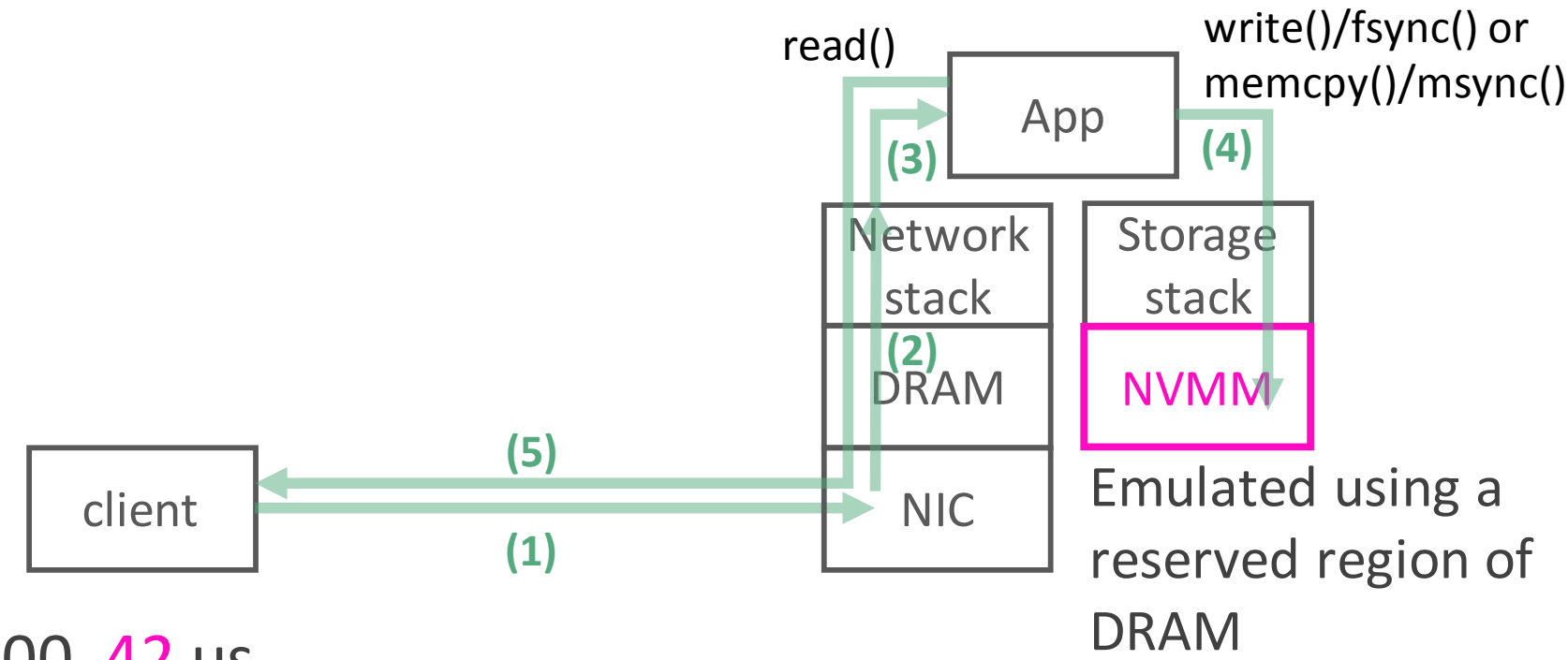


- 2030 us
  - Networking (w/o (4)) takes 40 us

Case Study: Careful Data Transfer

SSICLOPS

- Server persists client's request prior to acknowledgment
- e.g., 1KB commit:

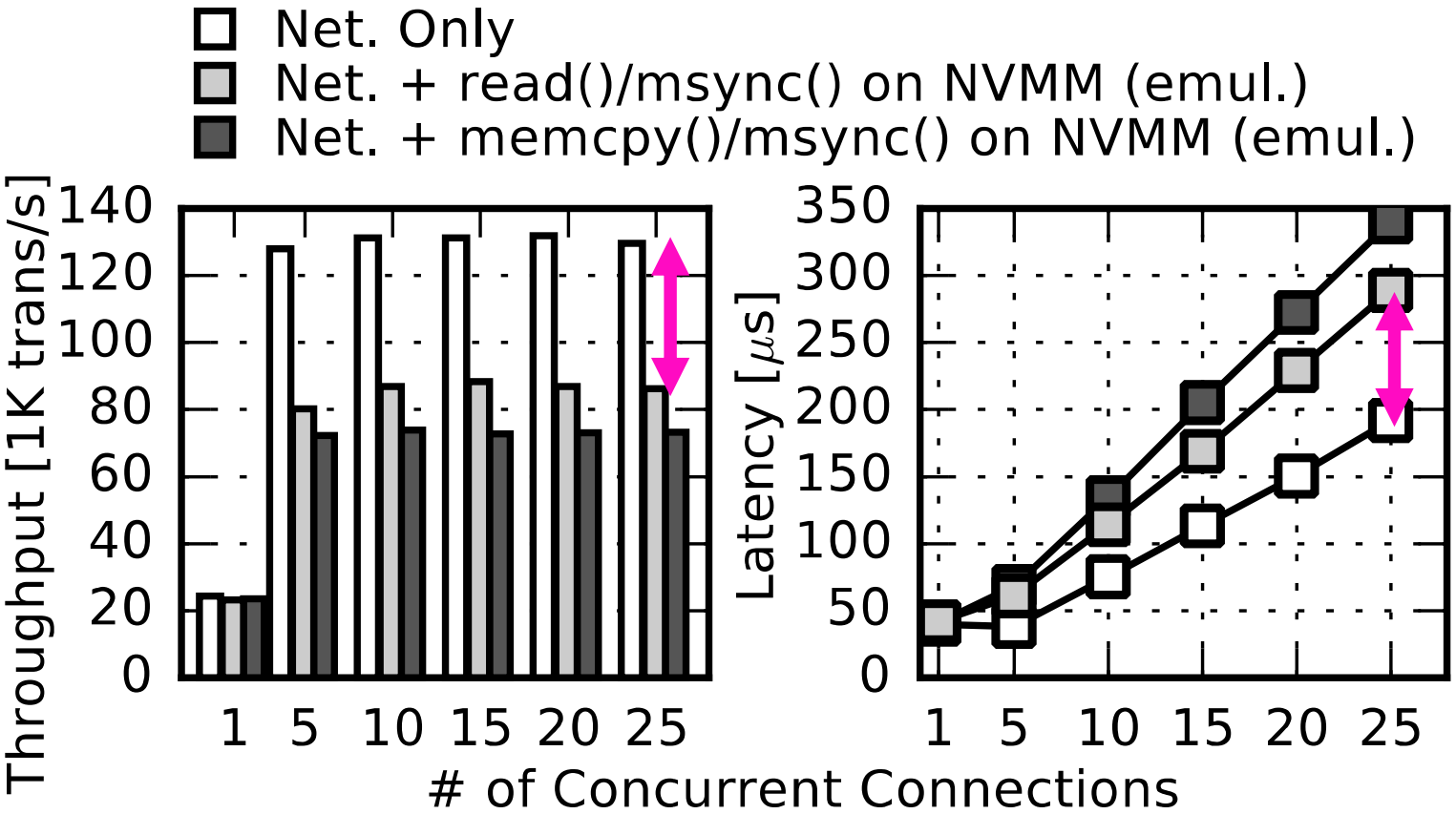


- ~~2000~~ 42 us
  - Networking takes 40 us
- This 2 us is not small

Case Study: Careful Data Transfer

SSICLOPS

- Parallel requests are serialized on each core

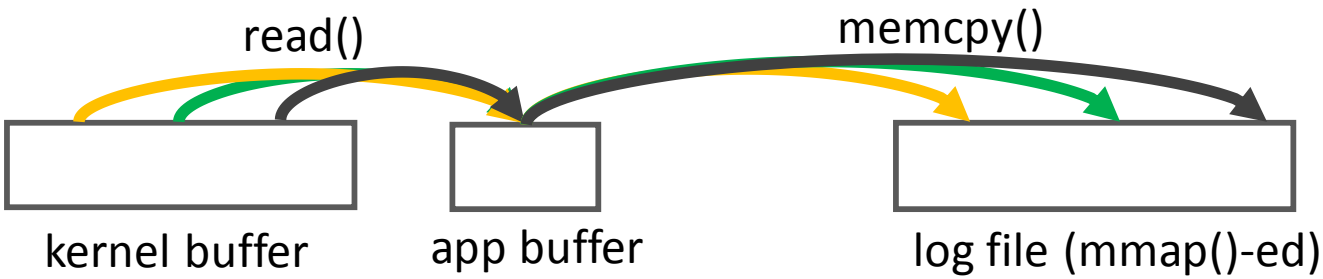


33 % throughput decrease, 50 % latency increase

Data Copies Matter

SSICLOPS

- Cache Misses
  - Persisting data (e.g., to a log) always happens to a different destination



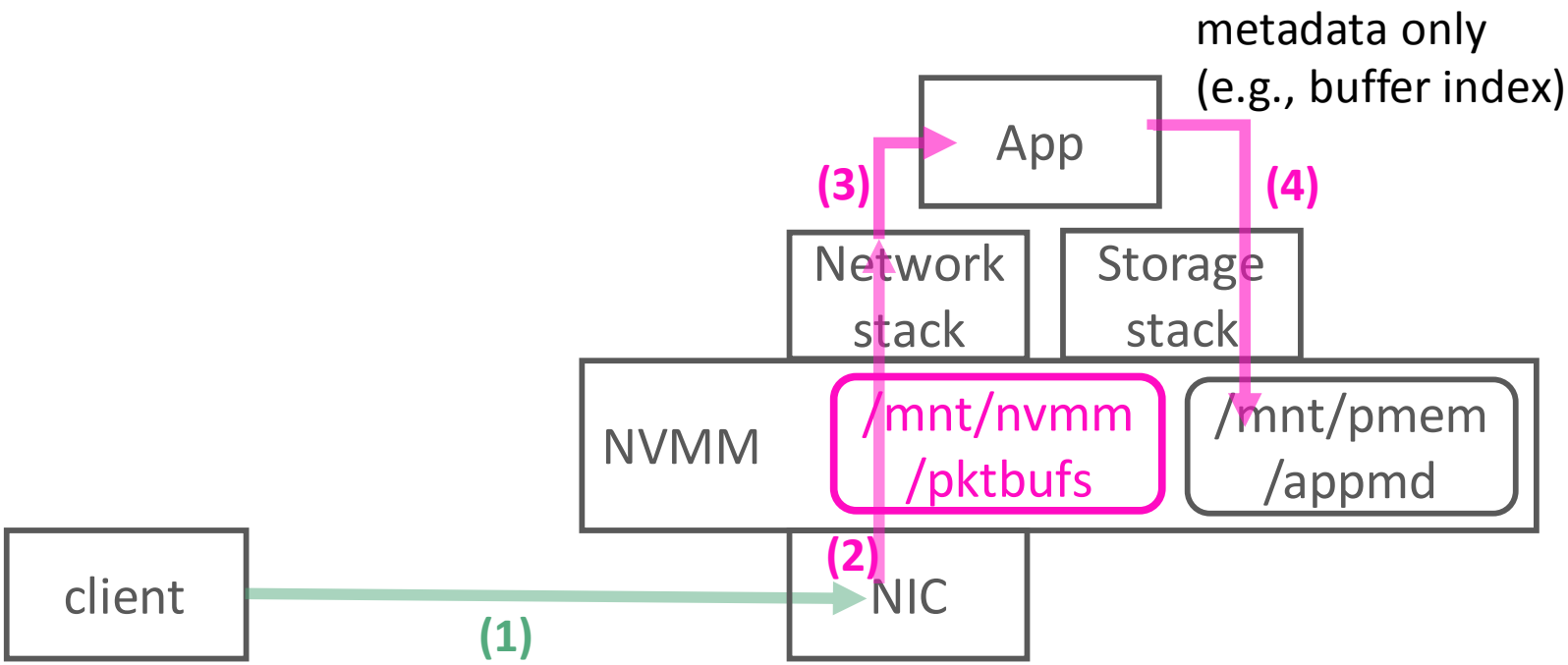
	Overall cache misses	Largest Contributor
Networking only	0.0004 %	net_rx_action() (84%)
Networking + NVMM (read() + memcpy() + msync())	**4.4121** %	**memcpy() (98%)**

We must avoid data copy!

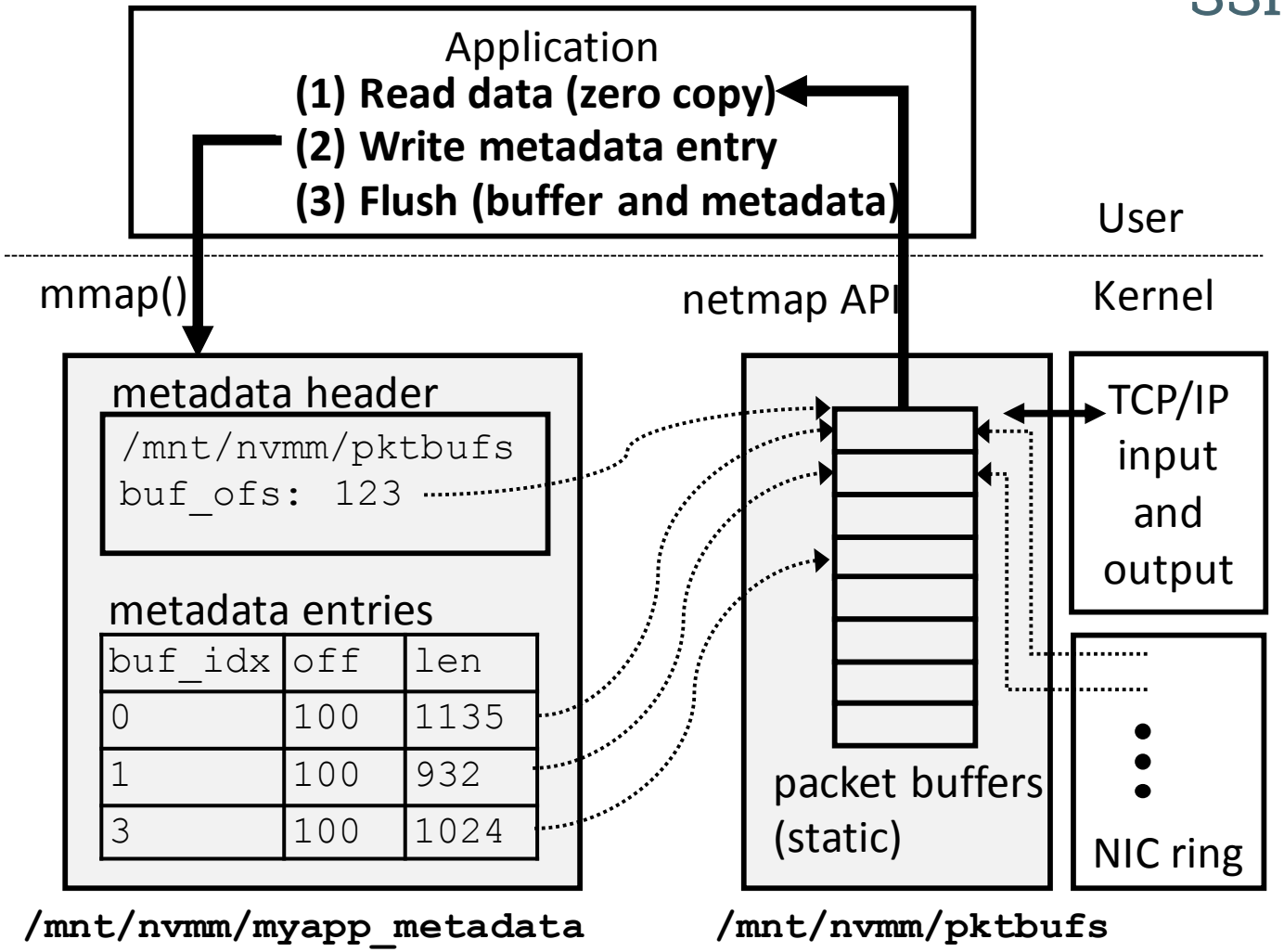
Packet Store (PASTE) Overview

SSICLOPS

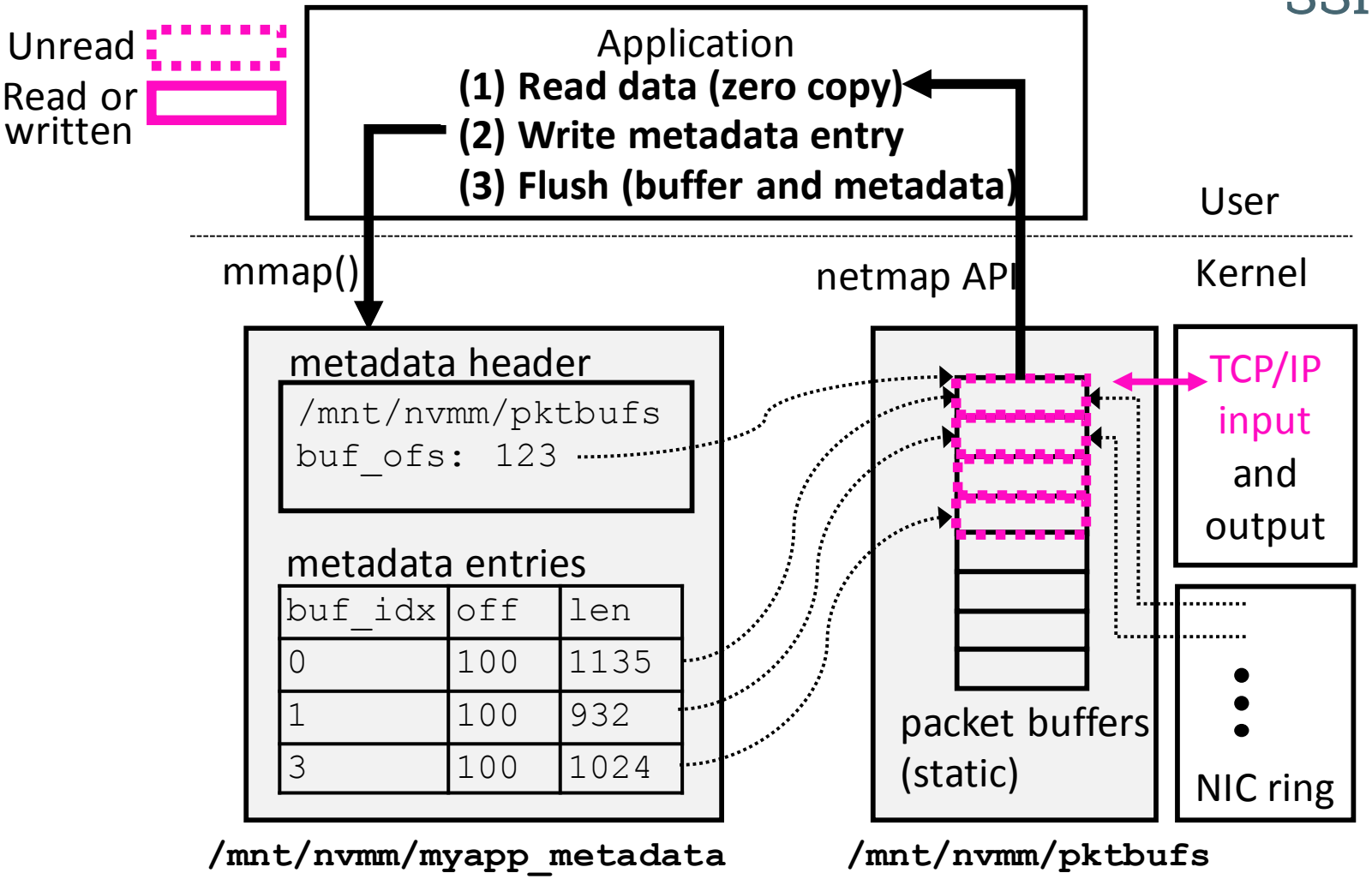
- Static packet buffers on a named NVMM region
  - DMA to NVMM
- Zero-copy APIs



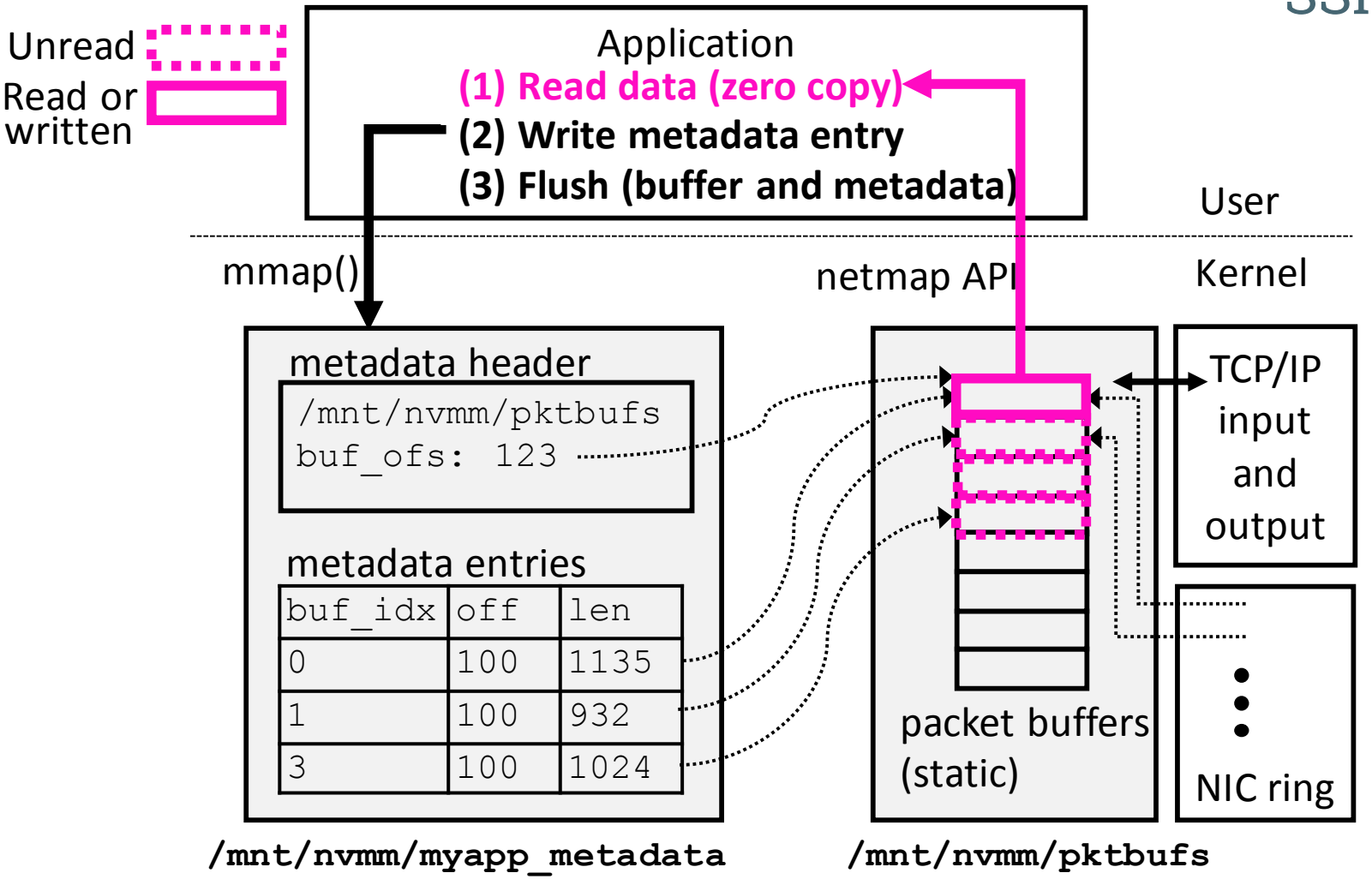
Fast Persistent Write with PASTE



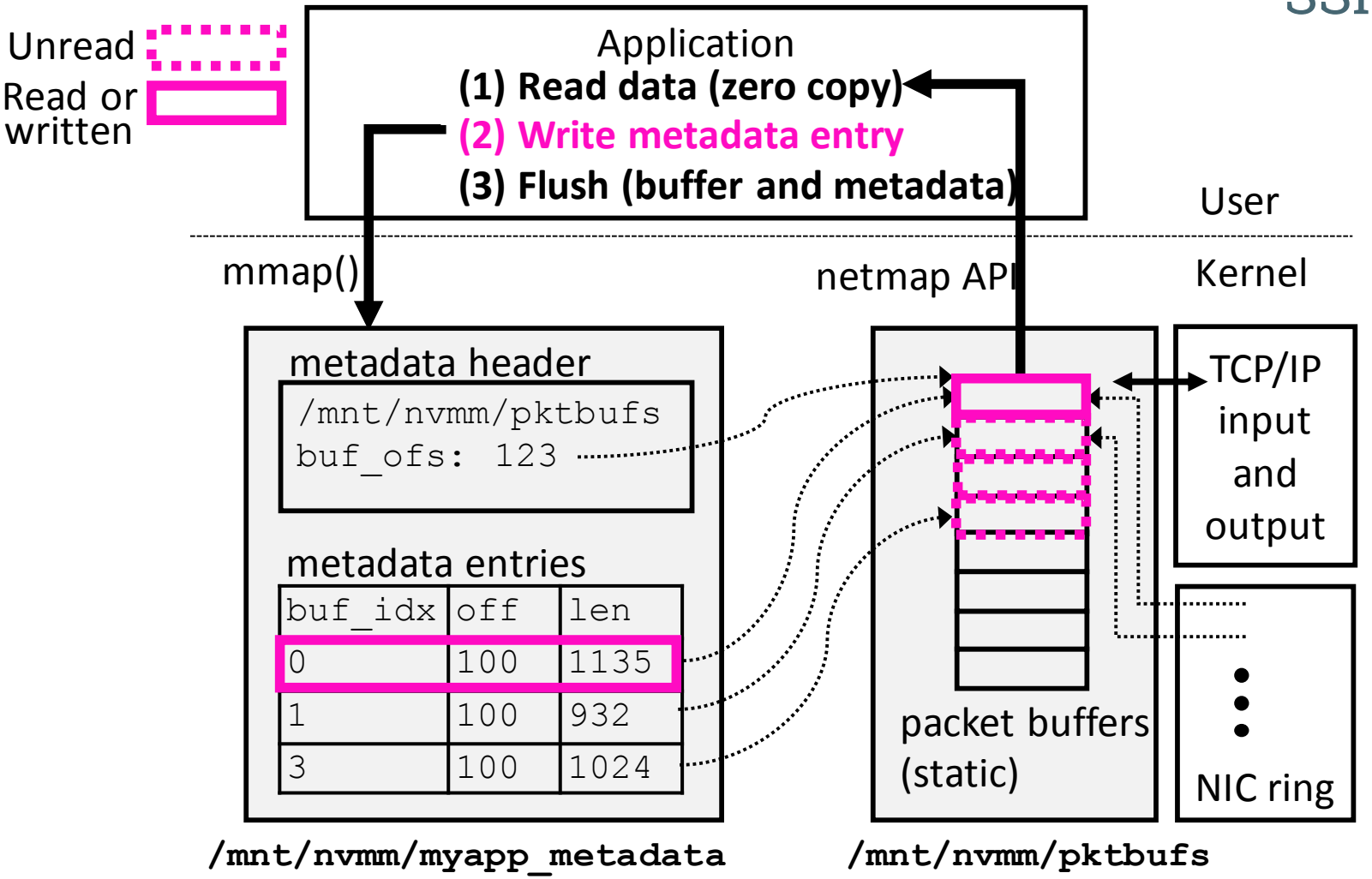
Fast Persistent Write with PASTE



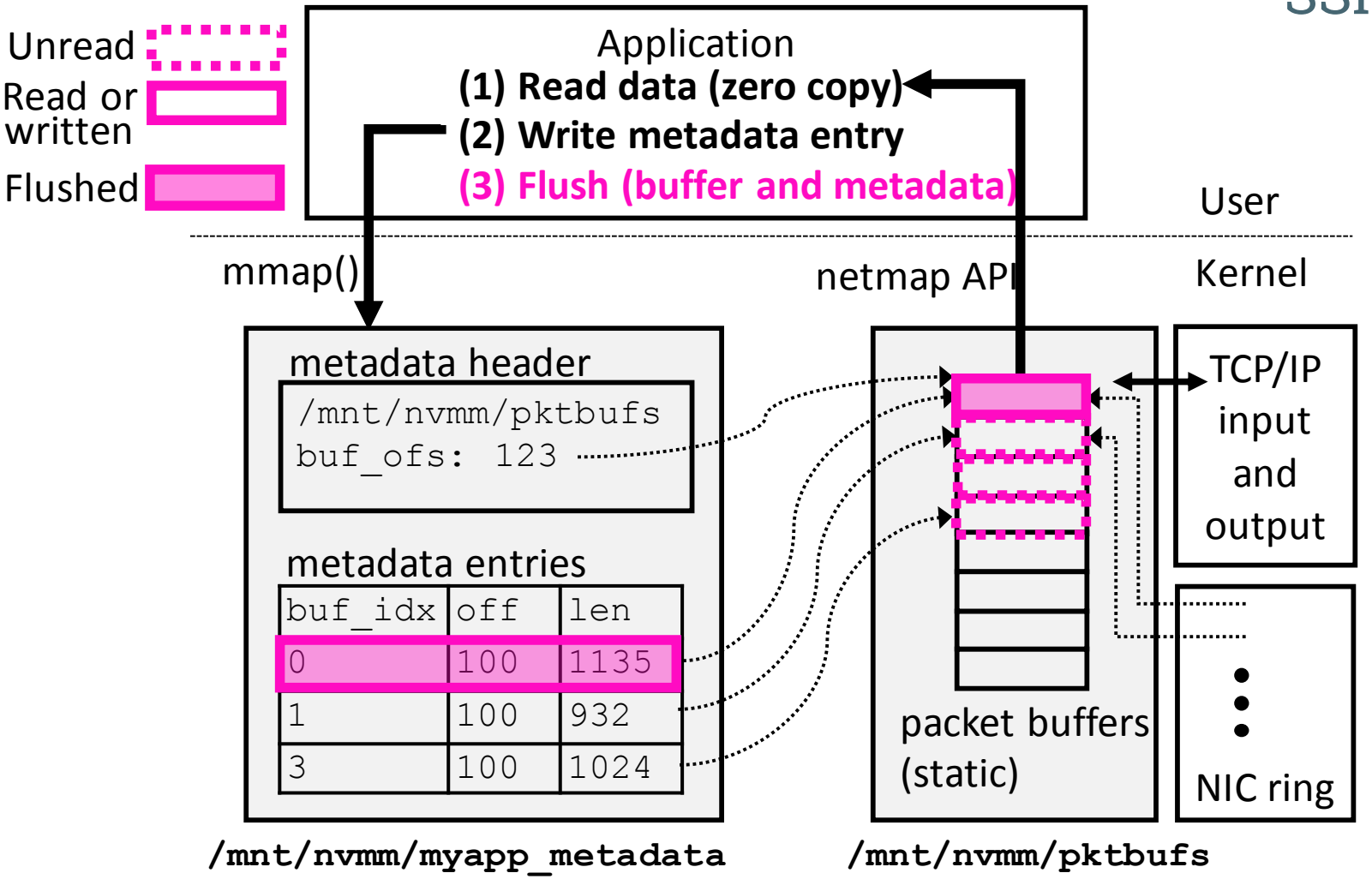
Fast Persistent Write with PASTE



Fast Persistent Write with PASTE

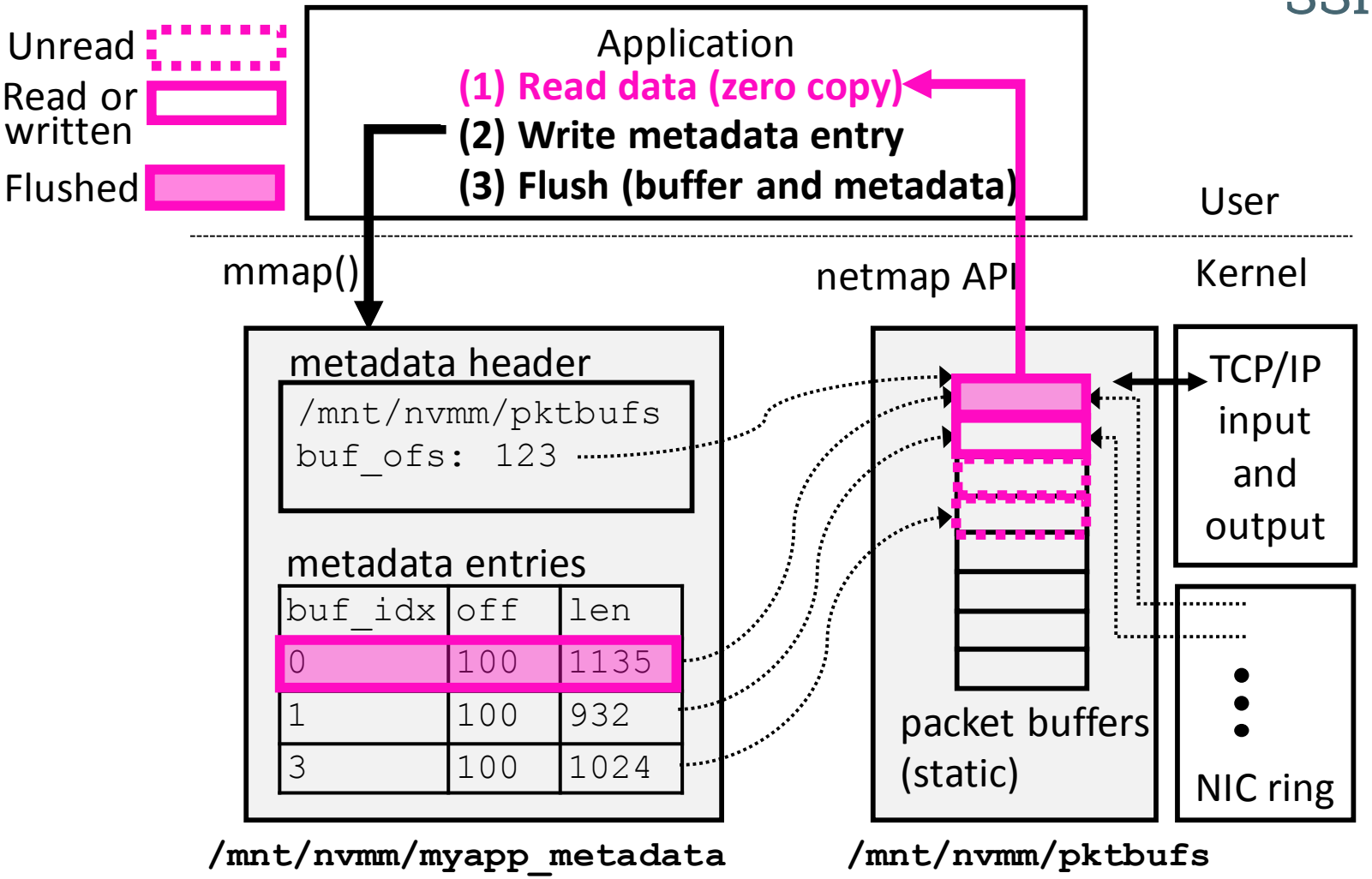


Fast Persistent Write with PASTE



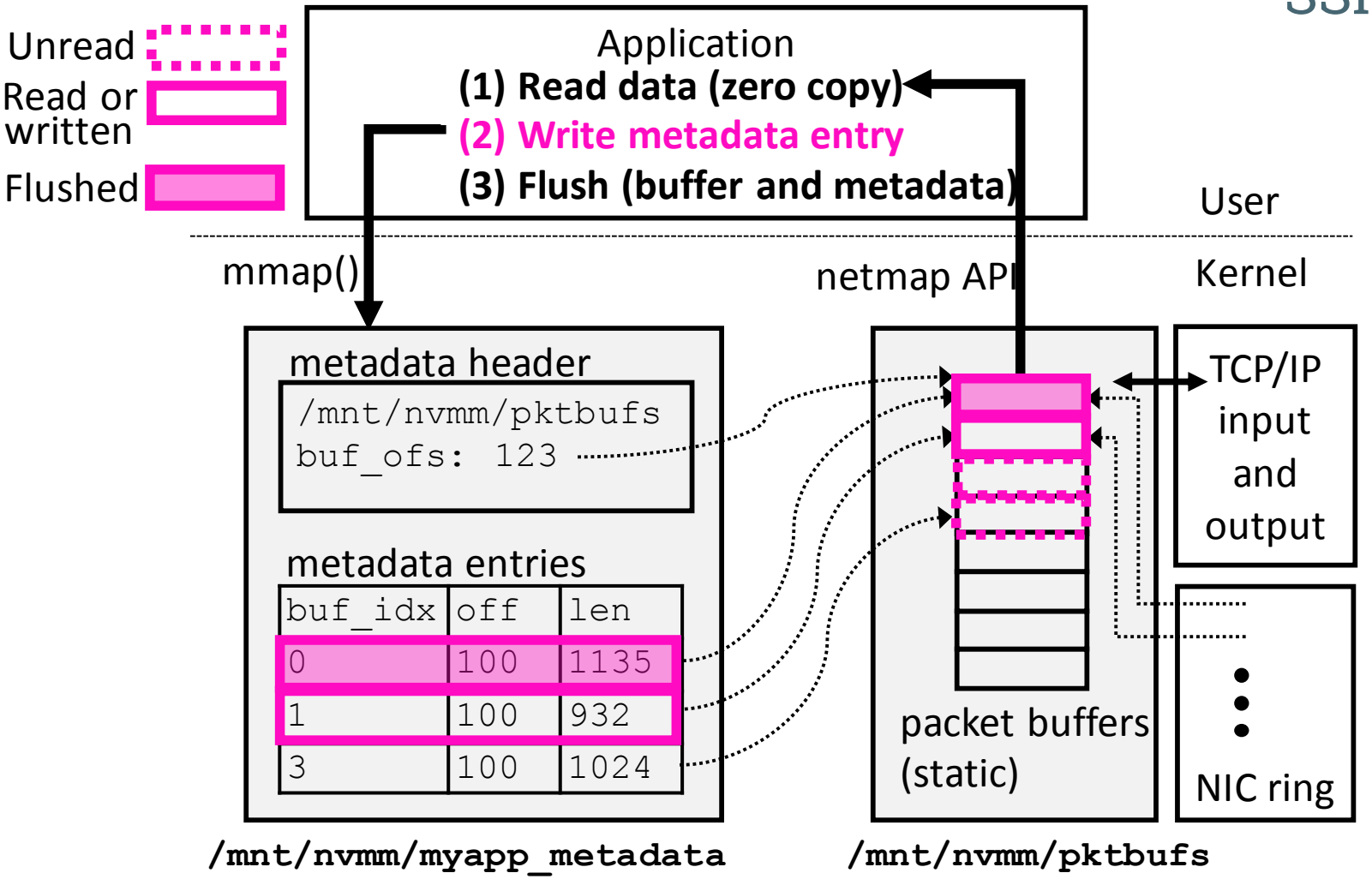
DMA is performed to L3 cache (DDIO)

Fast Persistent Write with PASTE



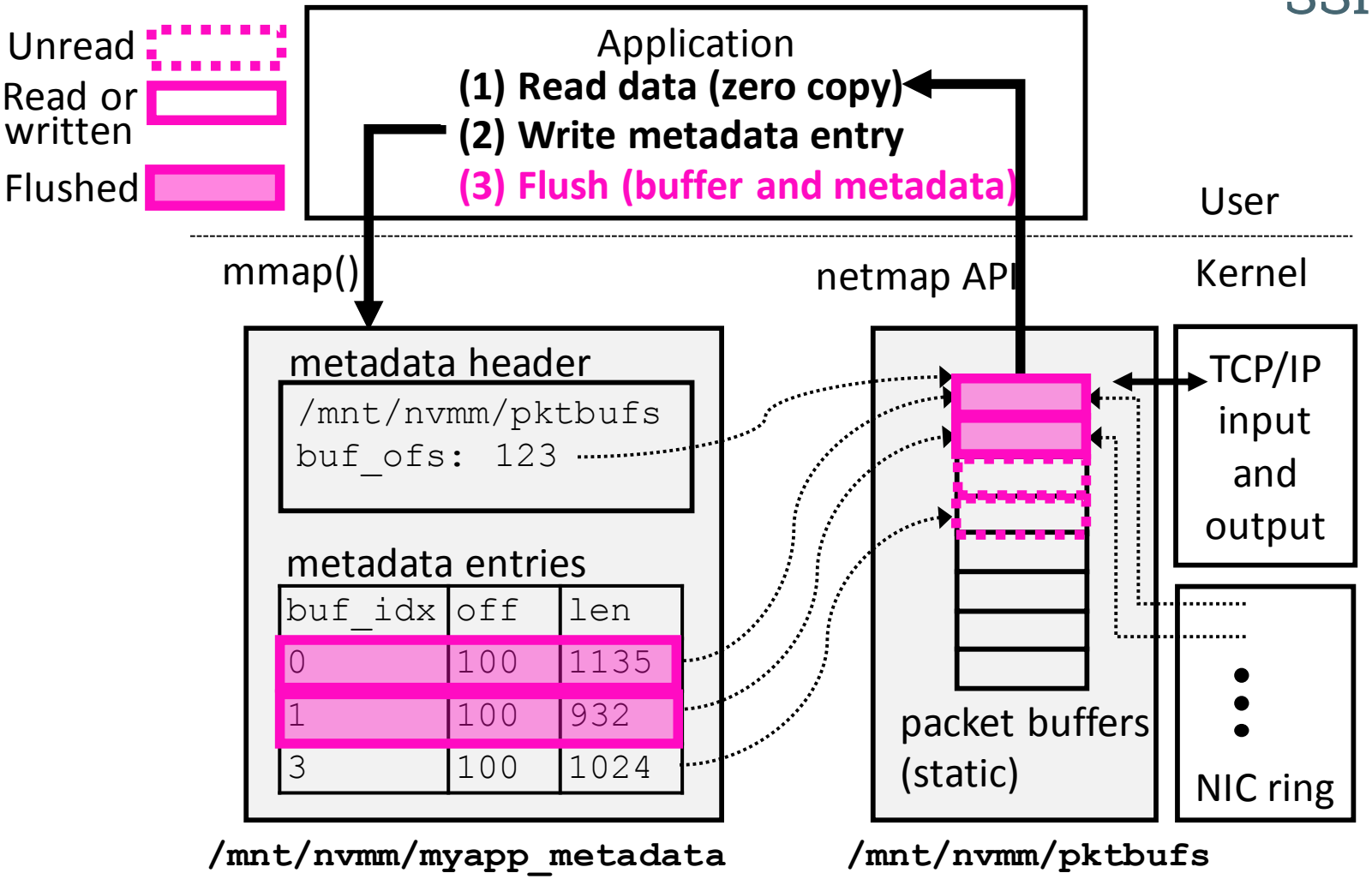
DMA is performed to L3 cache (DDIO)

Fast Persistent Write with PASTE



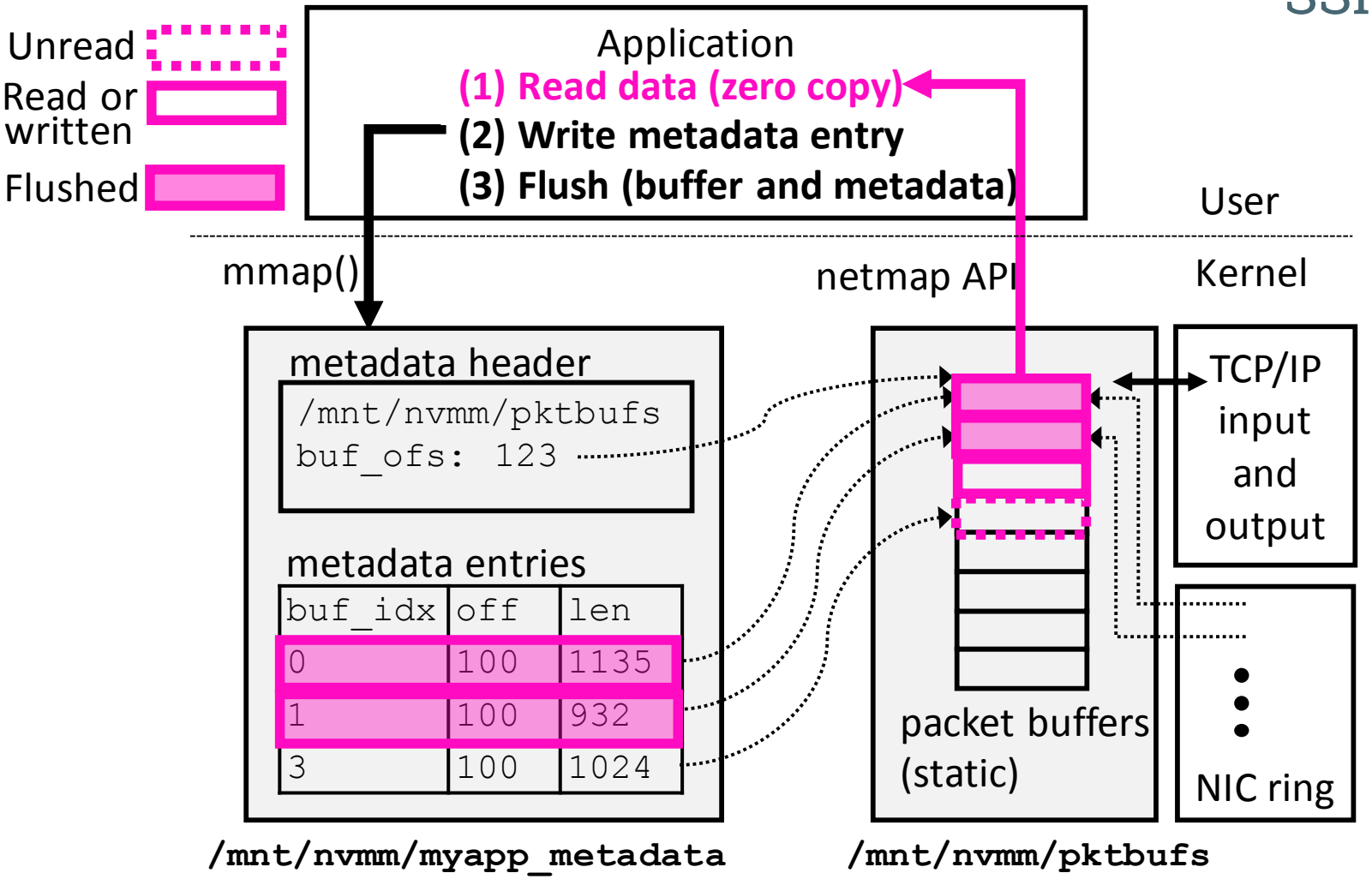
DMA is performed to L3 cache (DDIO)

Fast Persistent Write with PASTE



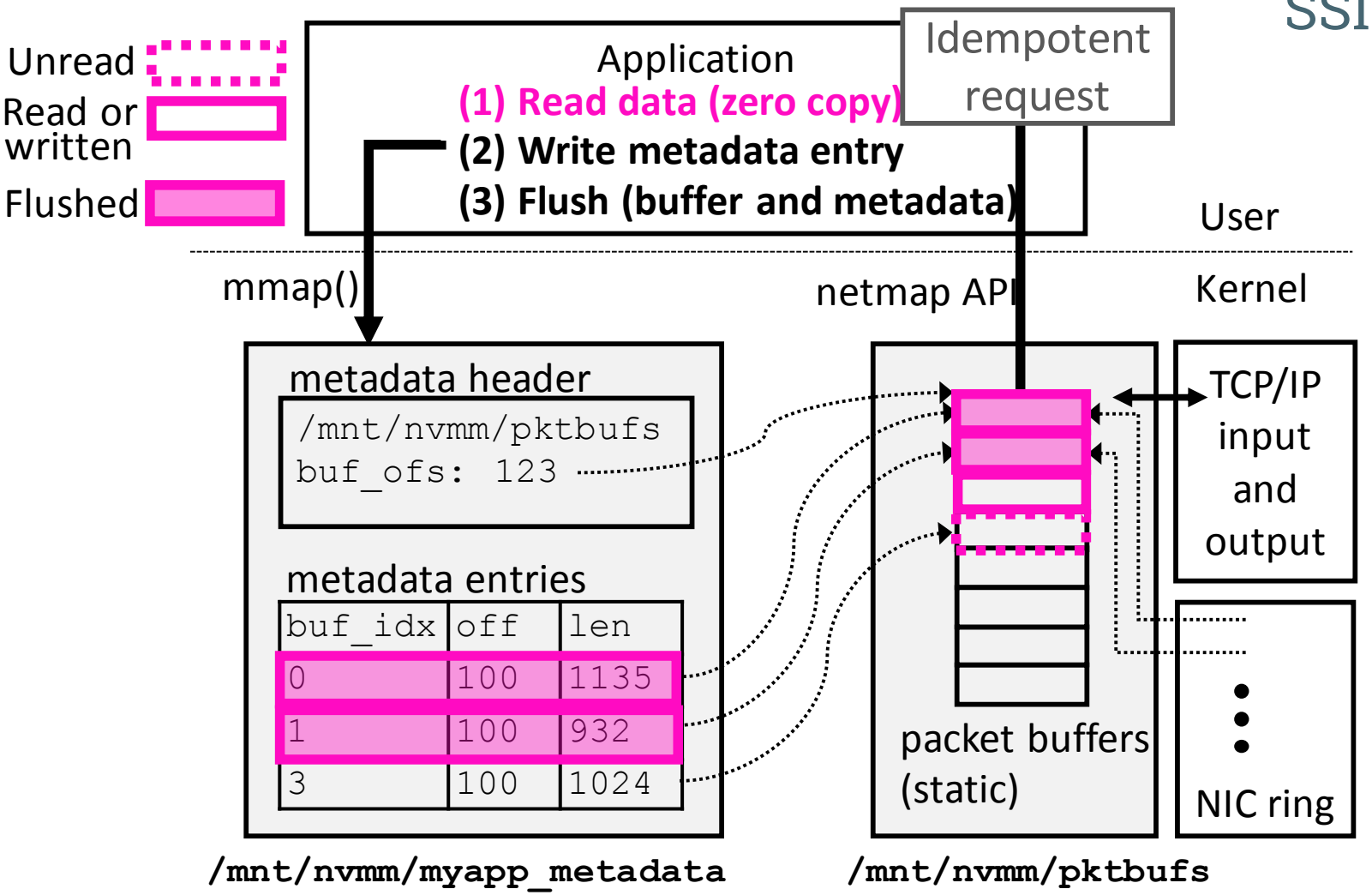
DMA is performed to L3 cache (DDIO)

Fast Persistent Write with PASTE



DMA is performed to L3 cache (DDIO)

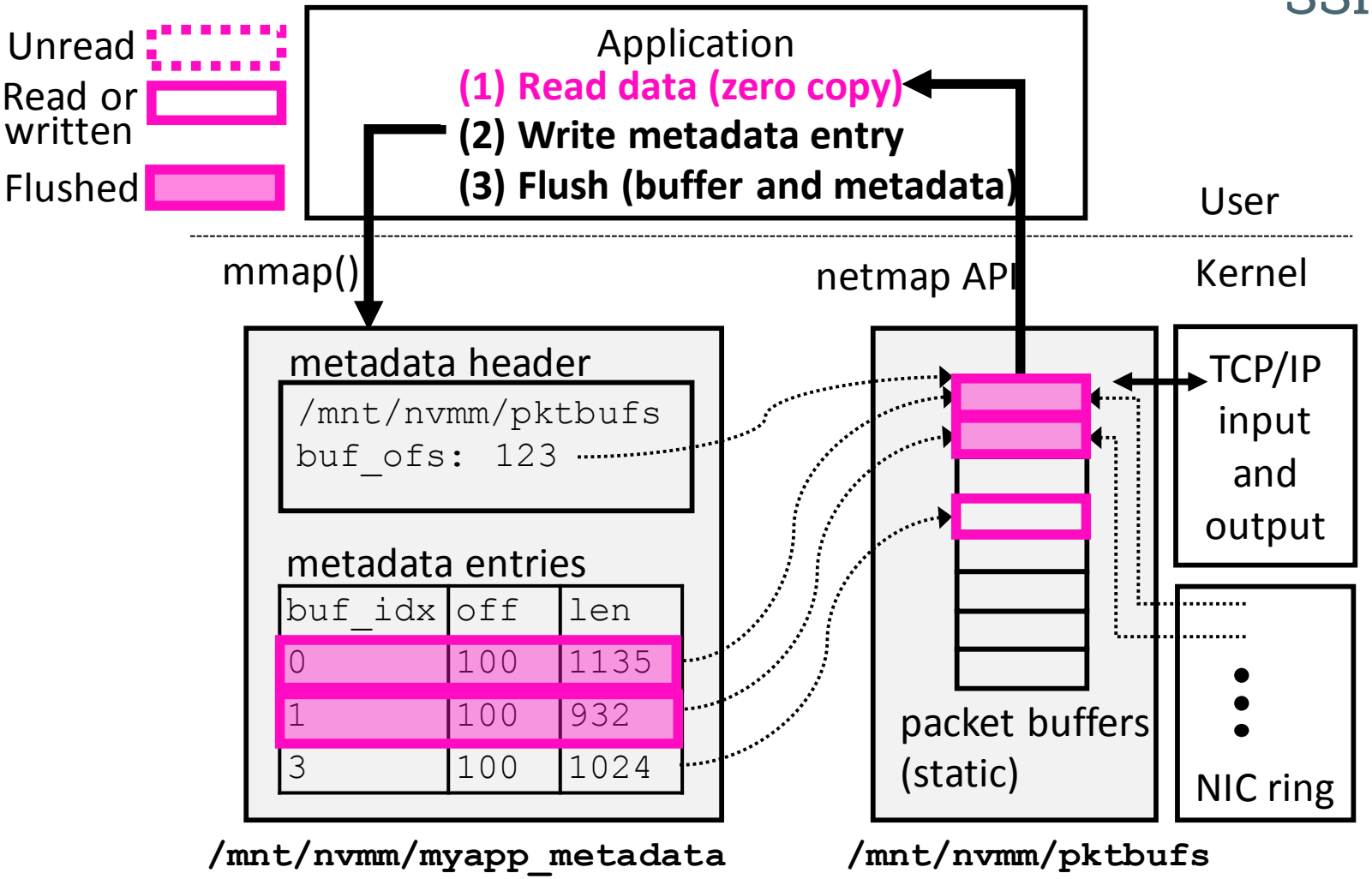
Fast and Selective Persistent Write with PASTE



DMA is performed to L3 cache (DDIO)

Unnecessary data is not flushed to DIMM

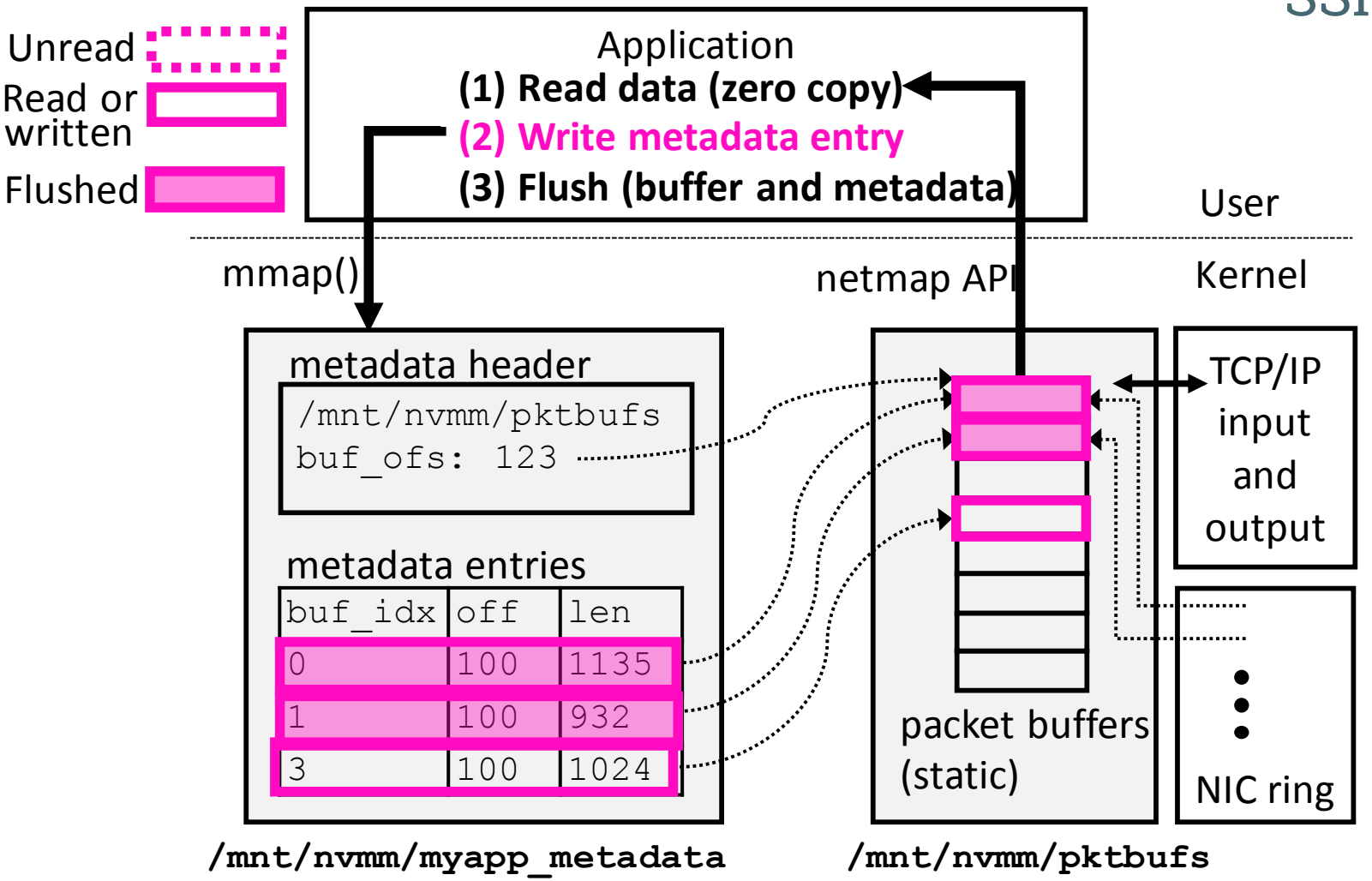
Fast and Selective Persistent Write with PASTE



DMA is performed to L3 cache (DDIO)

Unnecessary data is not flushed to DIMM

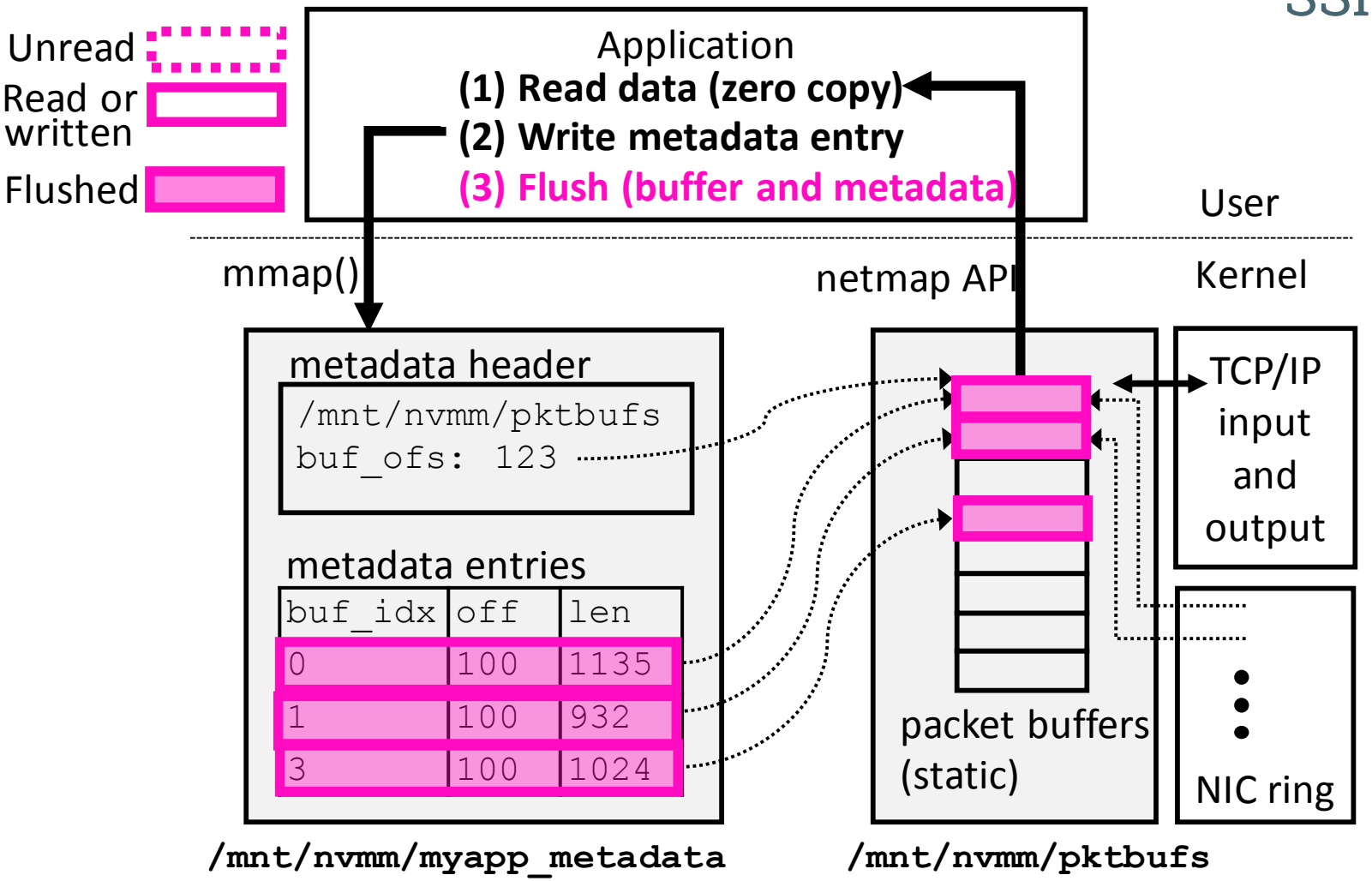
Fast and Selective Persistent Write with PASTE



DMA is performed to L3 cache (DDIO)

Unnecessary data is not flushed to DIMM

Fast and Selective Persistent Write with PASTE



DMA is performed to L3 cache (DDIO)

Unnecessary data is not flushed to DIMM

Implementation

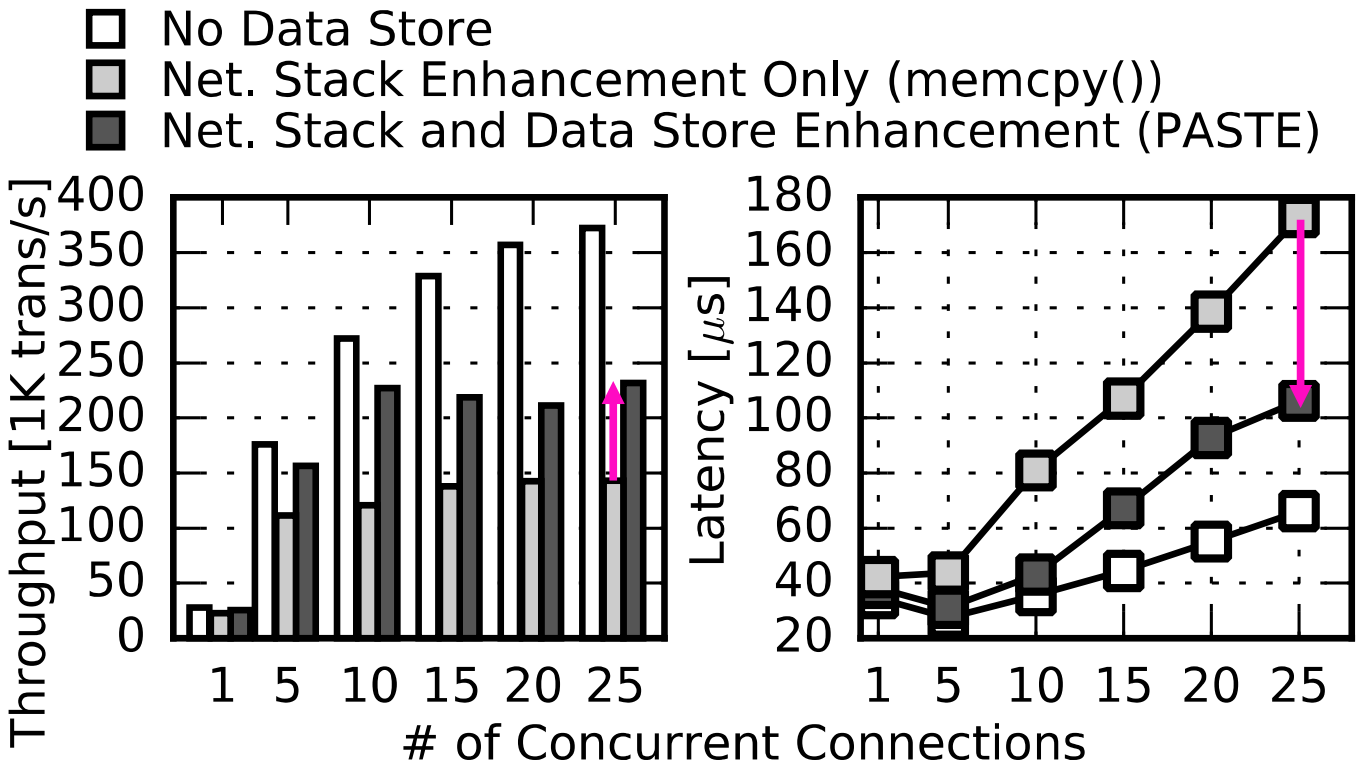
SSICLOPS

- Extension to netmap memory allocator
  - Exploit Linux NVDIMM stack
  - Claim packet buffers from a given file backed by NVMM
    - e.g., pkt-gen -i eth1@/mnt/pmem/bufs -f rx

© 2016 NetApp, Inc. All rights reserved.

Preliminary Results

- Implementation
  - Extend the netmap framework
    - Stackmap for TCP/IP



10-88 % throughput increase, 9-46 % latency reduction

Related Work

SSICLOPS

- Enhanced network stacks
  - MegaPipe (OSDI'12), Stackmap (ATC'16), Fastsocket (ASPLOS'16)
  - IX and Arrakis (OSDI'14), mTCP (NSDI'13), Sandstorm (SIGCOMM'14), MICA (NSDI'14)

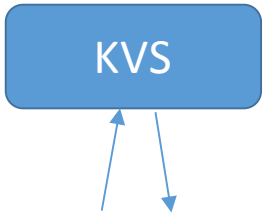
No NVMM aware

- NVMM filesystems
  - BPFS (SOSP'09), NOVA (FAST'15)
- NVMM databases
  - NVWAL (ASPLOS'15), REWIND (VLDB'15), NV-Tree (FAST'15)

No networking aware

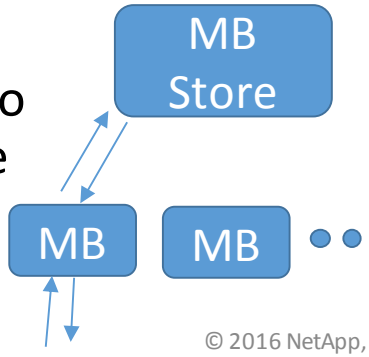
Use cases

- Persistent Key Value Store (e.g., Redis)

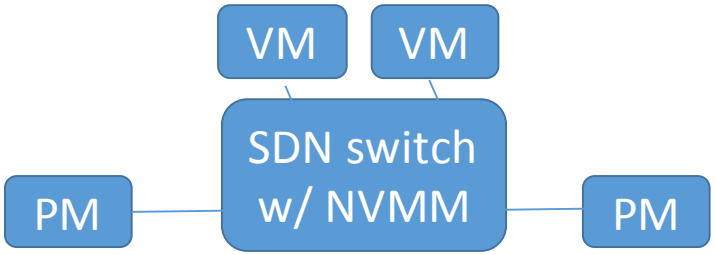


- Networked NVMM filesystem
  - Merge packet metadata and inodes

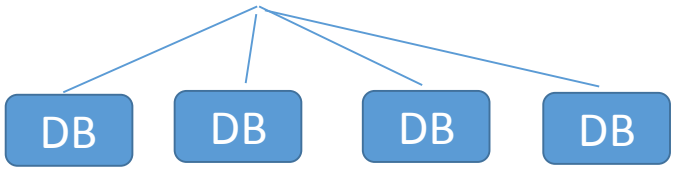
- Fault-tolerant, Distributed Middlebox (e.g., Yoda [EuroSys16]))
  - PASTE enables finger-grained state updates
  - Yoda was able to store only some vantage points



- Software-Defined Monitoring
  - e.g., new SDN switch action: Logging (OpenSketch [NSDI'13] Mille-Feuille [HotNets'16])
  - Operators mirror traffic for monitoring
  - PASTE enables realtime (local) logging



- Datacenter-Wide DB
  - Fast I/O and replication
  - Use PTP for consistency



© 2016 NetApp, Inc. All rights reserved.

Conclusion

SSICLOPS

PASTE: Fast data persistence with named packet buffers on NVMM and zero-copy API

- Implications
  - Network stacks are now a bottleneck for durably storing data
  - Improving network and storage stacks in isolation is not enough
  - We need new stacks design for end systems and middleboxes