

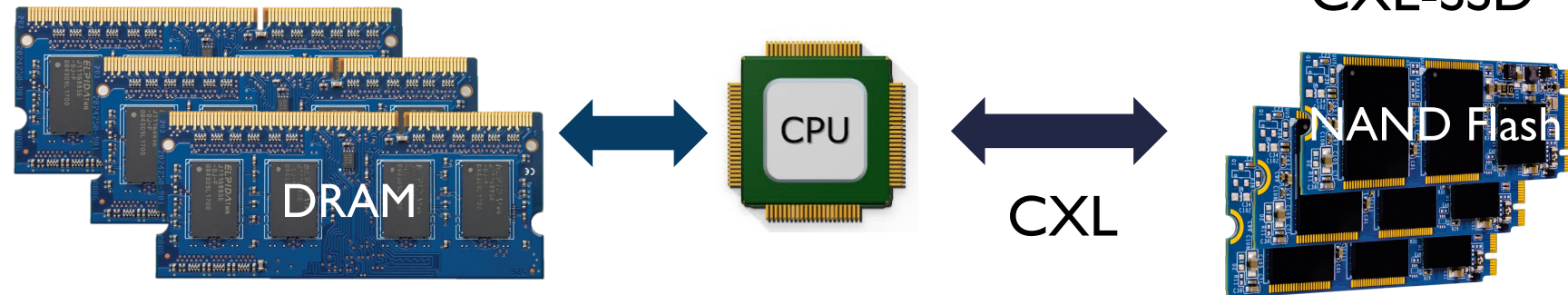
Cylon: Fast and Accurate Full-System Emulation of CXL-SSDs

Dongha Yoon*, *Hansen Idden**, Jinshu Liu, Berkay Inceisci, Sam H. Noh, Huaicheng Li



Convergence of Memory and Storage

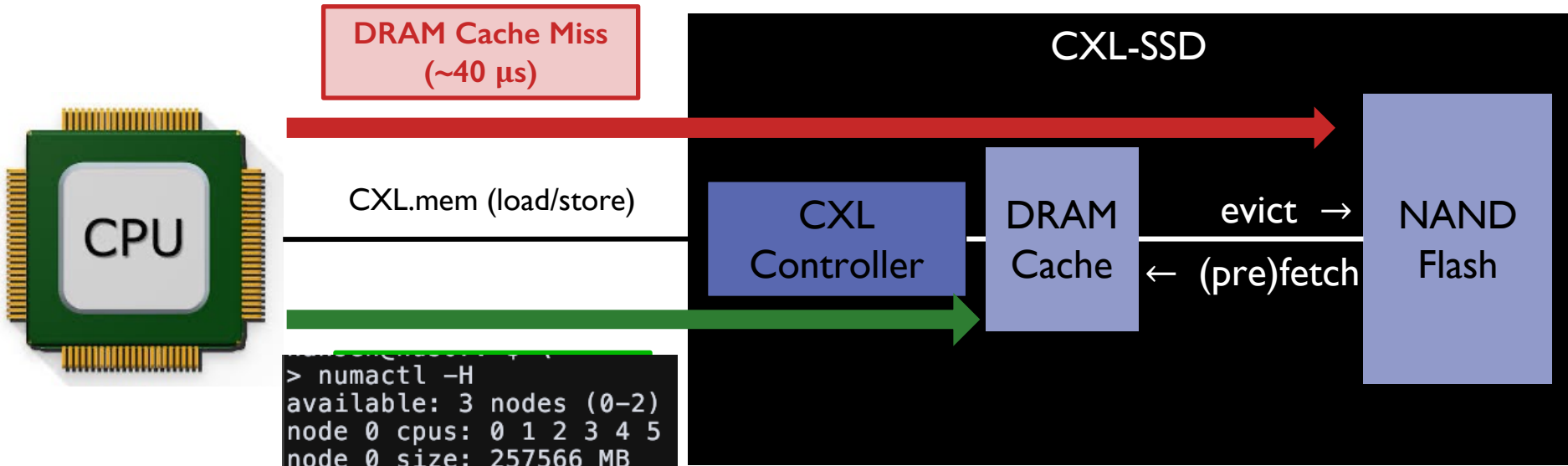
Memory-intensive applications exploding in size faster than DRAM can scale



Expensive (\$11-15/GB), Limited Size
Low memory access latency (~100ns)

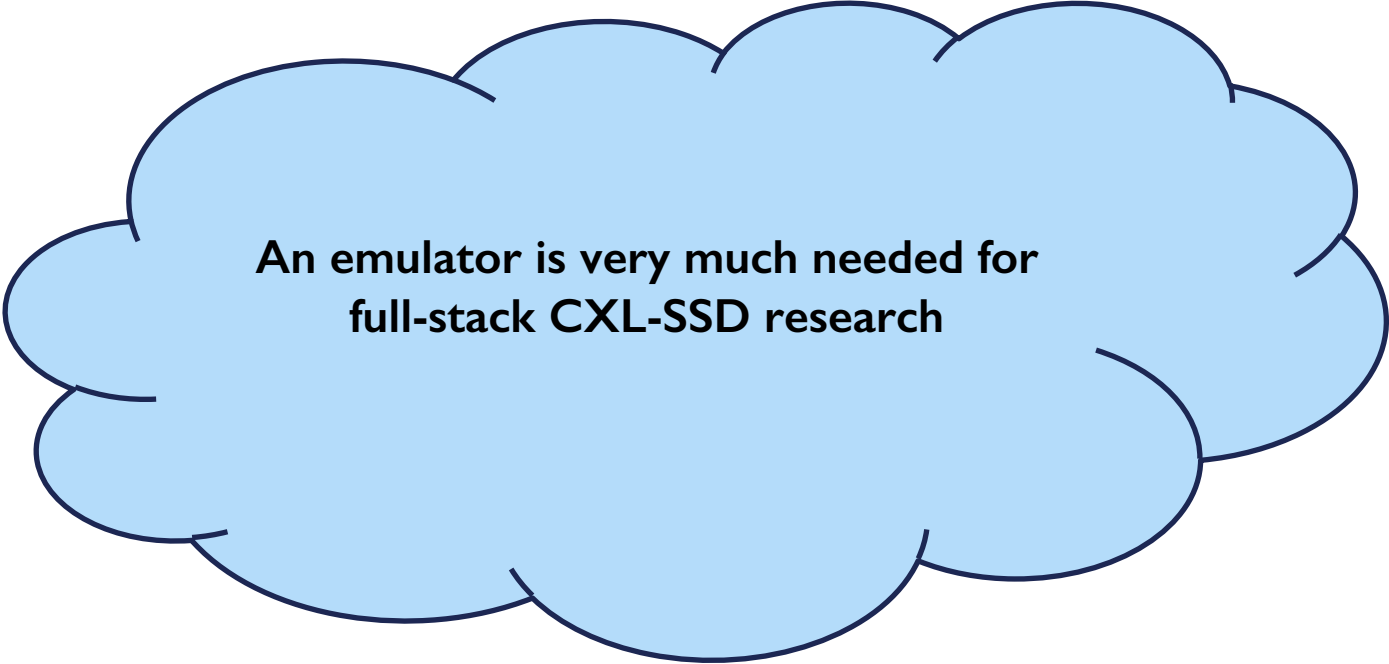
Cheap, Multi-TB capacity
Slow (tens of μ s)

CXL-SSDs promise DRAM-like programmability at storage-like cost



```
> numactl -H
available: 3 nodes (0-2)
node 0 cpus: 0 1 2 3 4 5
node 0 size: 257566 MB
node 0 free: 253663 MB
node 1 cpus: 64 65 66 67
node 1 size: 258016 MB
node 1 free: 256681 MB
node 2 cpus:
node 2 size: 901151 MB
node 2 free: 900888 MB
```

50x gap! This makes cache management absolutely critical



**An emulator is very much needed for
full-stack CXL-SSD research**

**Full-Stack
Execution**

Unmodified OS & apps

**Near Bare-Metal
Speed**

Sub- μ s hits, no VM-exit

**Flexible Device
Modeling**

Configurable NAND timing
and Caching policy



There is a lack of research platform that delivers all three requirements

Simulators (MQSim-CXL)
No Full-stack

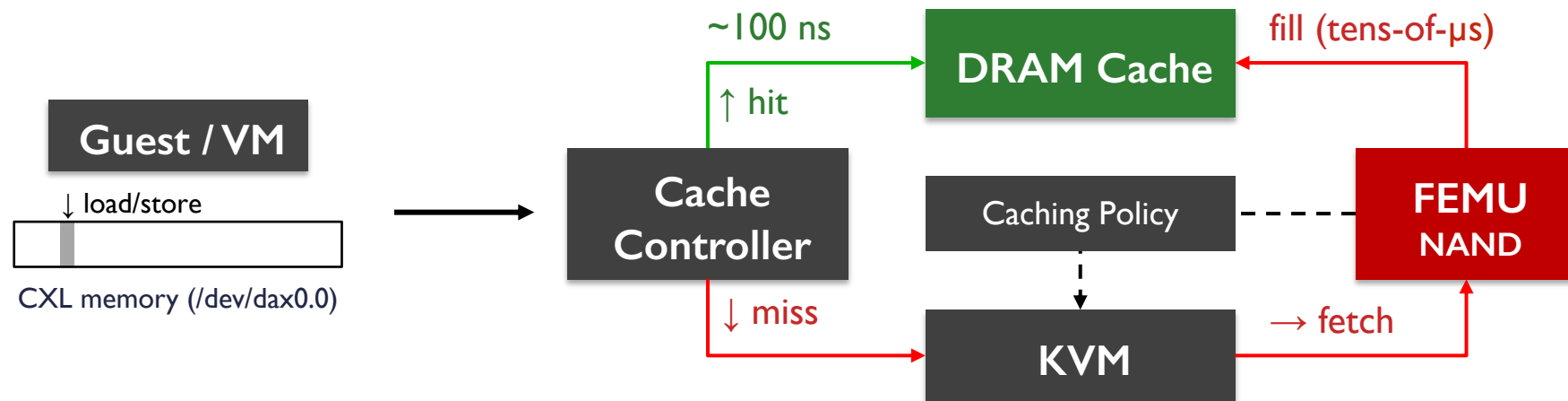
Gem5, QEMU-CXL
Too slow

Prototypes (CMM-H)
Opaque, scarce

We need an open, fast, and accurate emulator to enable CXL-SSD research

MQSim-CXL
ESF
CXL-SSD-Sim
CXL-DMSim
OpenCXD
QEMU
Cylon

Cylon is validated, fast, full-stack, transparent, and extensible



Cache Hit : No VM-exit, remote DRAM speed

Cache Miss: VM-exit → FEMU → NAND

Overview

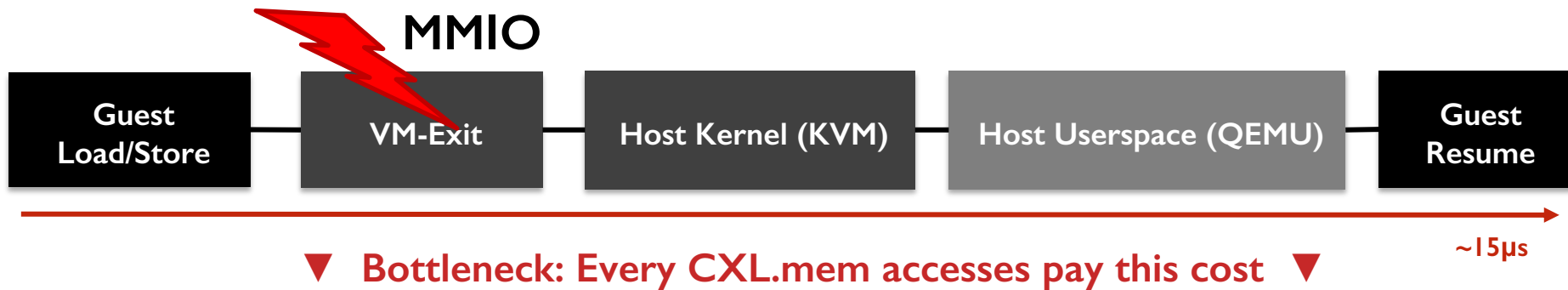
Cylon Hypervisor Design

Extensibility

Evaluation

Cylon build on FEMU for accurate timing

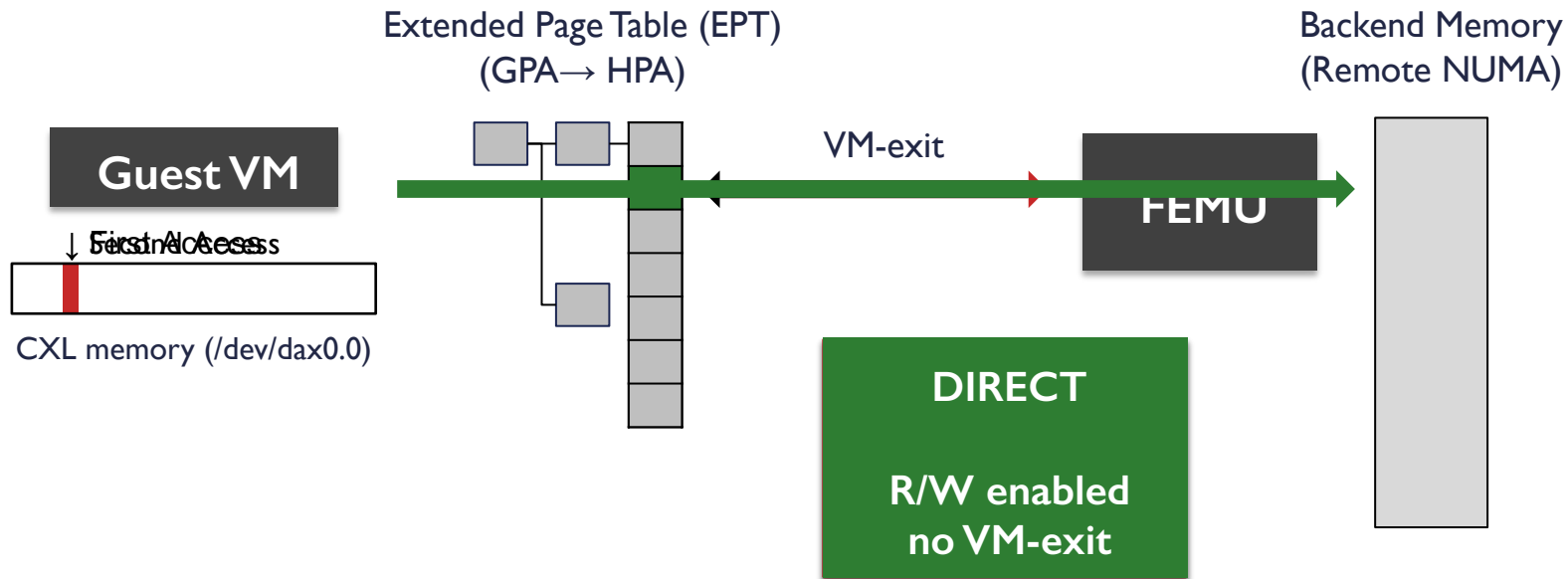
QEMU-CXL Emulation Path



Acceptable for NAND flash emulation, but too high for CXL-attached DRAM

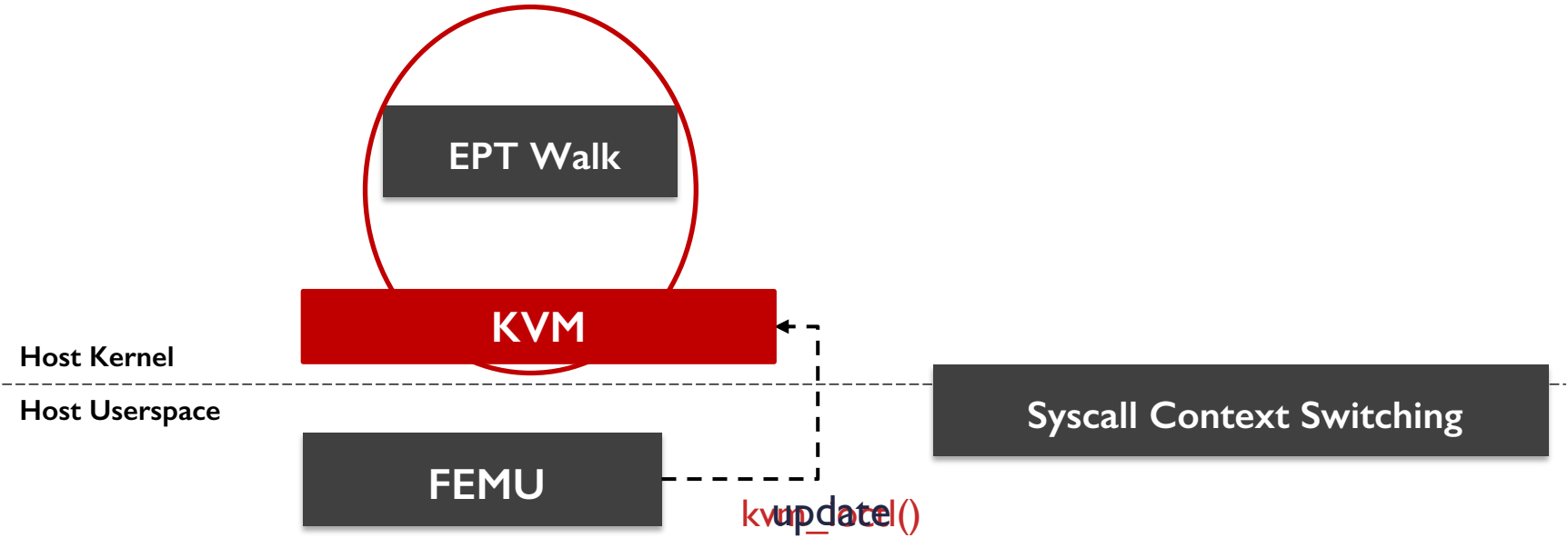
We need to optimize QEMU/KVM to reproduce sub-μs cache hit latency

Dynamically toggling EPT entry between TRAP and DIRECT state



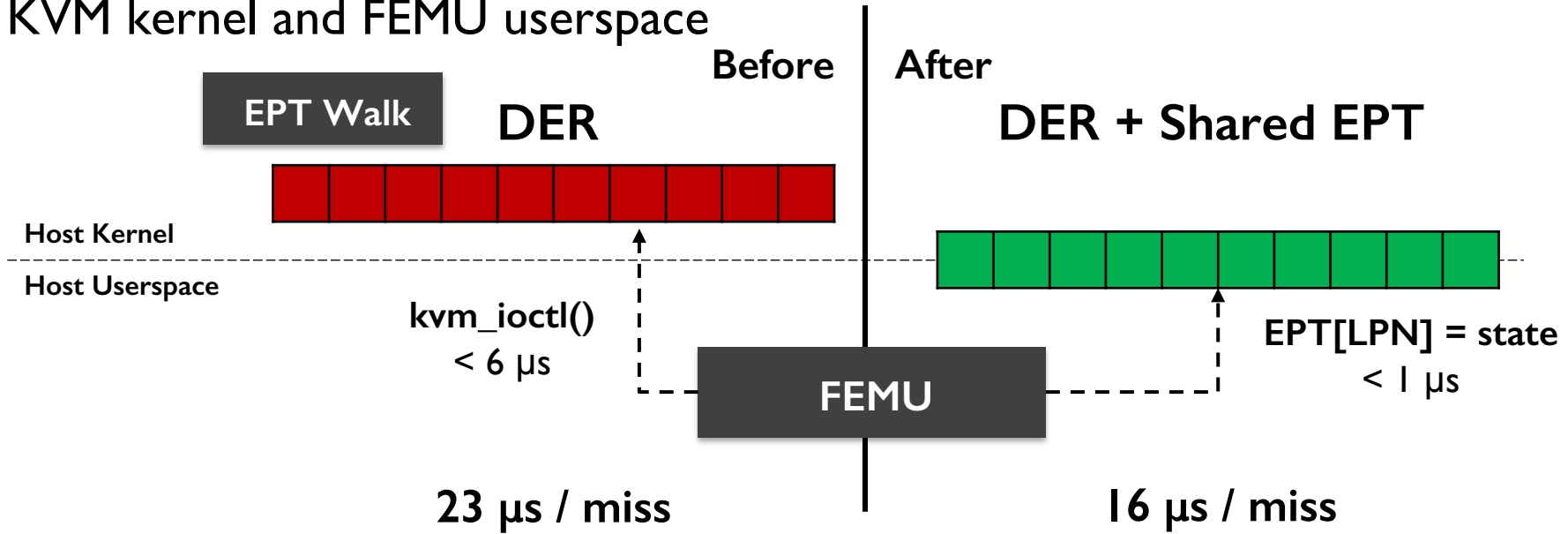
An EPT entry bit-flip per page eliminates all VM-exit overhead on cache hits

Challenge #2: The ioctl() Bottleneck

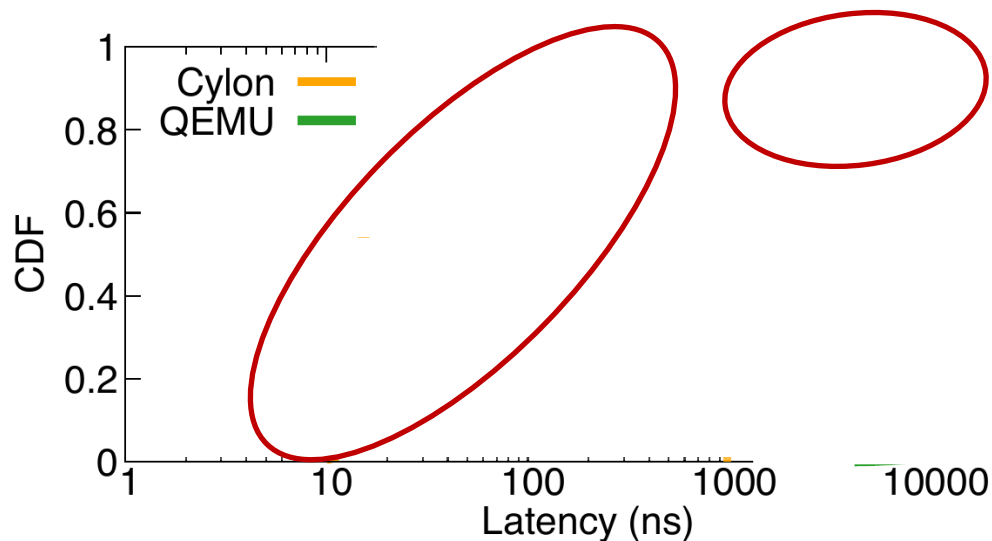


ioctl() calls for every EPT updates make Cylon less accurate and unpredictable

Pre-allocates all EPT entries in a single contiguous region mapped into both KVM kernel and FEMU userspace



29% lower VM-exit overhead, no syscall, O(1) EPT walk



8GB working set > 4.8GB cache
(exercises both hits and misses)

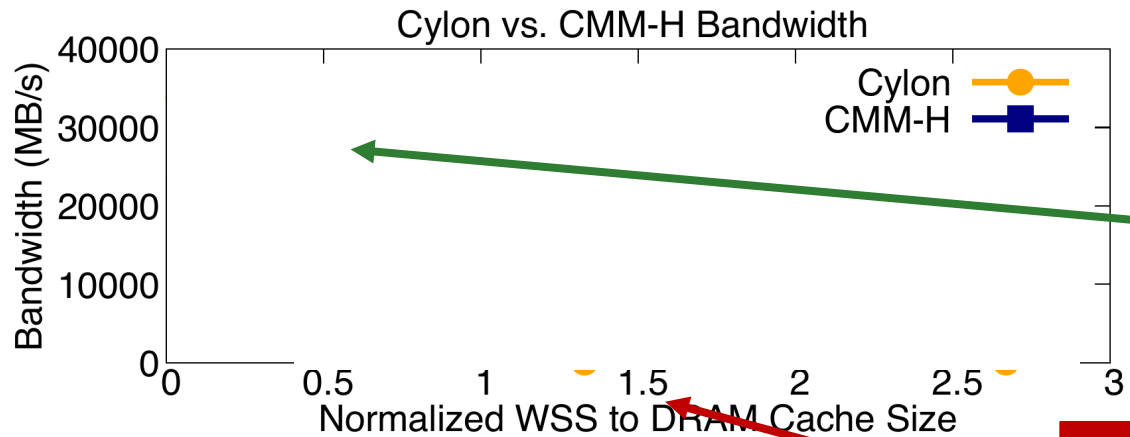
QEMU (green):

- Single inflated mode at 14.6μs

Cylon (orange):

- Clear bimodal distribution

First emulator to reproduce realistic CXL-SSD latency distributions



Intel MLC bandwidth test

WSS (working set size) normalized to DRAM cache size

Phase 1: $WSS \leq \text{cache size}$

- Near-memory bandwidth
- Cylon sustains 32 GB/s

Phase 2: $WSS > \text{cache size}$

- Sharp drop to NAND-bound
- Both converge to same throughput

CMM-H drops earlier due to opaque controller overhead (FPGA artifacts)

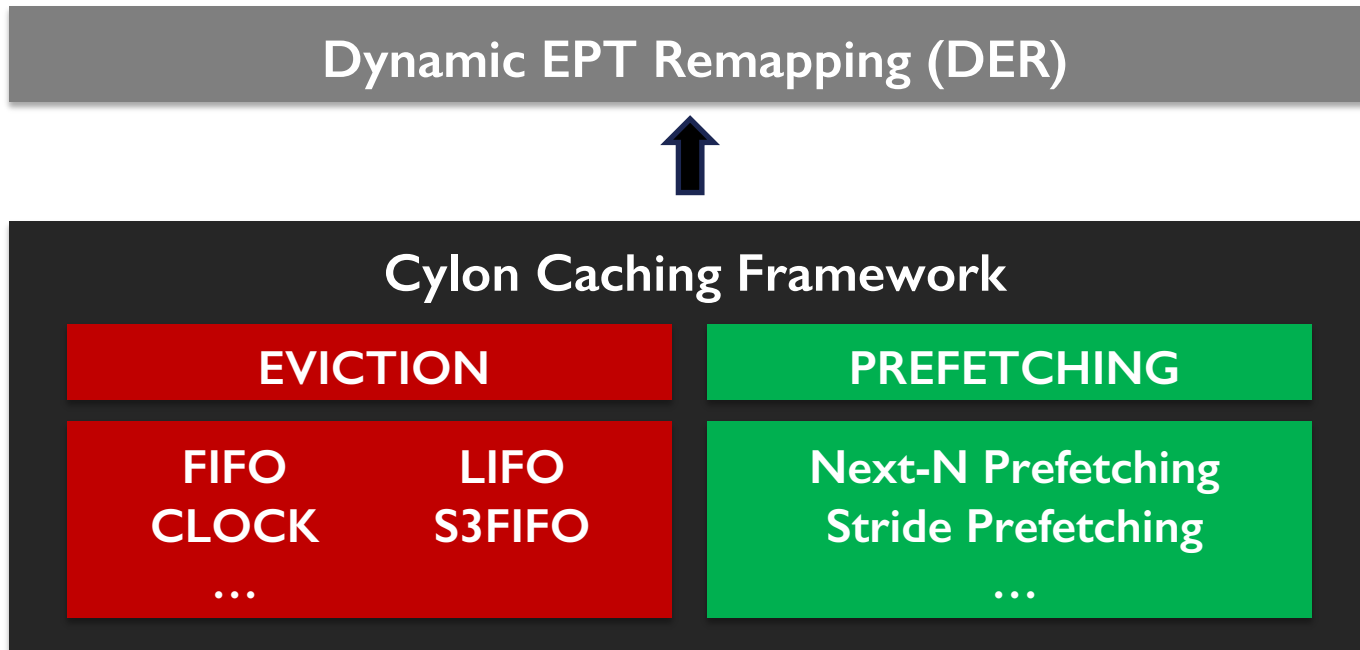
Cylon reproduces the fundamental DRAM-to-NAND transition

Overview

Cylon Hypervisor Design

Extensibility

Evaluation



Cylon enables pluggable eviction and prefetching policies to explore CXL-SSD cache management design space

Redis

Graph (GAPBS)

ML Pipeline



Cylon API Commands

prefetch

pin

evict

set_policy

query_stats



Cylon CXL-SSD (shared-memory ring queue)

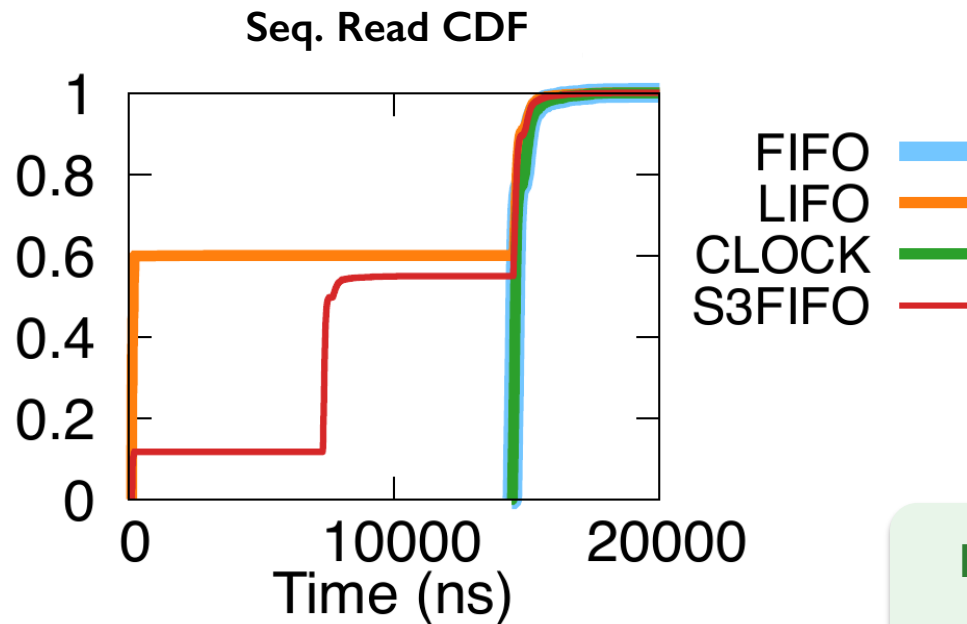
Cylon enables detailed memory access behavior profiling

Overview

Cylon Hypervisor Design

Extensibility

Evaluation

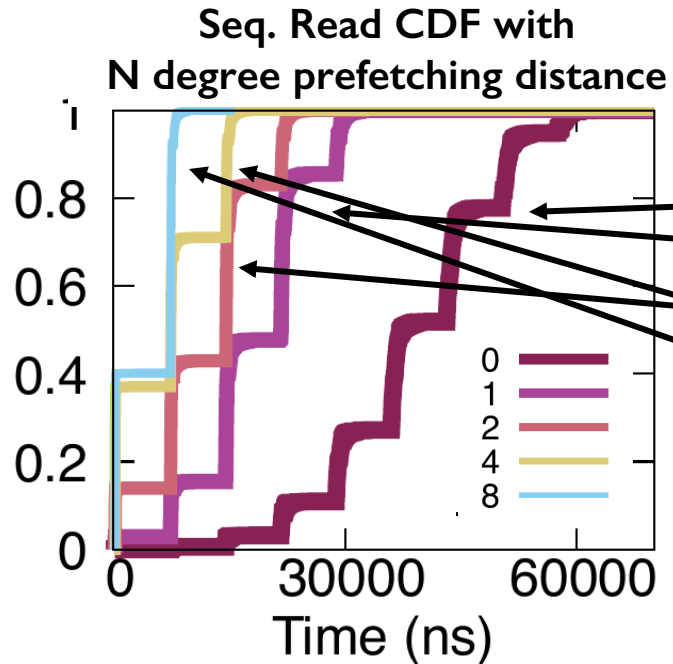


Cache Hit Rate Stride-1024

Policy	Hit Rate
--------	----------

LIFO → 15% higher hit rate than FIFO/CLOCK

Cylon enables rapid eviction-policy comparison



Cache Hit Rate Stride-4096 (S3FIFO)

Degree	Hit Rate
--------	----------

18% → 86% just by tuning prefetch degree

Cylon shows increasing prefetch degree shifts accesses from NAND to DRAM

Redis YCSB-C (100% reads)

(a) Small WSS (0.33× cache):

All in DRAM, fast hits

Cylon: hundreds of ns

(b) Medium WSS (1.06× cache):

Mix of hits + misses

Gap narrows

(c) Large WSS (2.10× cache):

Mostly NAND, both converge

to tens–hundreds of μ s

The gap between Cylon and CMM-H is due to FPGA overhead in CMM-H

Cylon can replicate cache-dominated to NAND-dominated performance

Cylon Performance Optimizations

Dynamic EPT Remapping (DER)

Eliminates VM-exits on cache hits by toggling EPTE flags

Shared EPT Memory

O(1) constant-time cache residency updates without syscalls or page-table walks

Cylon Flexibility Features

Configurable Caching Framework

Pluggable eviction and prefetching policies

Application-Level Cache Control

Apps issue prefetch / pin / evict commands for HW-SW co-design

