

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

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



github.com/MoatLab/FEMU

CXS-SSDs: TB-scale, Memory-Like Storage

Limitations of Existing CXL-SSD Emulators

Motivation

- Memory wall limits scalability
- CXL enables memory-semantic storage
- CXL-SSDs show large latency asymmetry

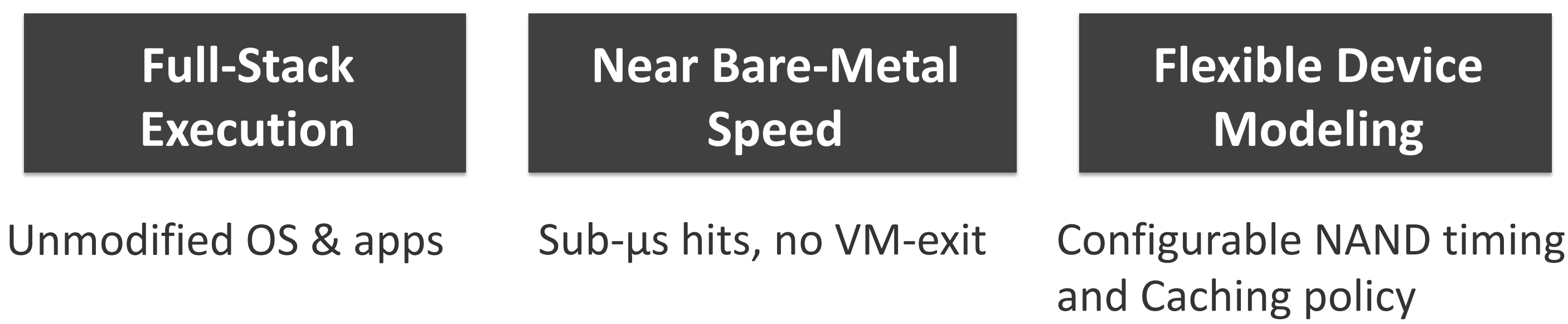
Problem Gap

- Hardware prototypes are opaque
- Simulators are too slow
- Functional emulators lack fidelity

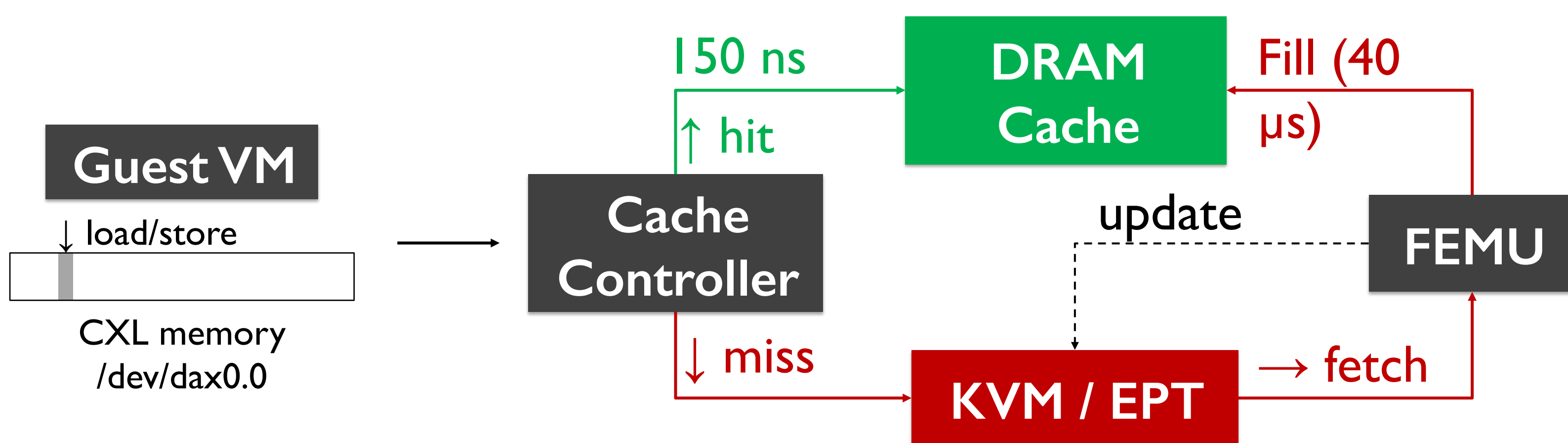
System	Validation	Performance	Full-stack	Transparency	Extensibility
MQSIM-CXL	no	low	no	-	yes
ESF	no	low	no	-	yes
CXL-SSD-Sim	no	very low	no	-	no
CXL-DMSim	no	very low	no	-	yes
OpenCXD	-	low	yes	yes	-
QEMU	-	low	yes	yes	-
Cylon	yes	high	yes	yes	yes

Cylon: High-Performance, High-Configurable CXL-SSD emulation platform

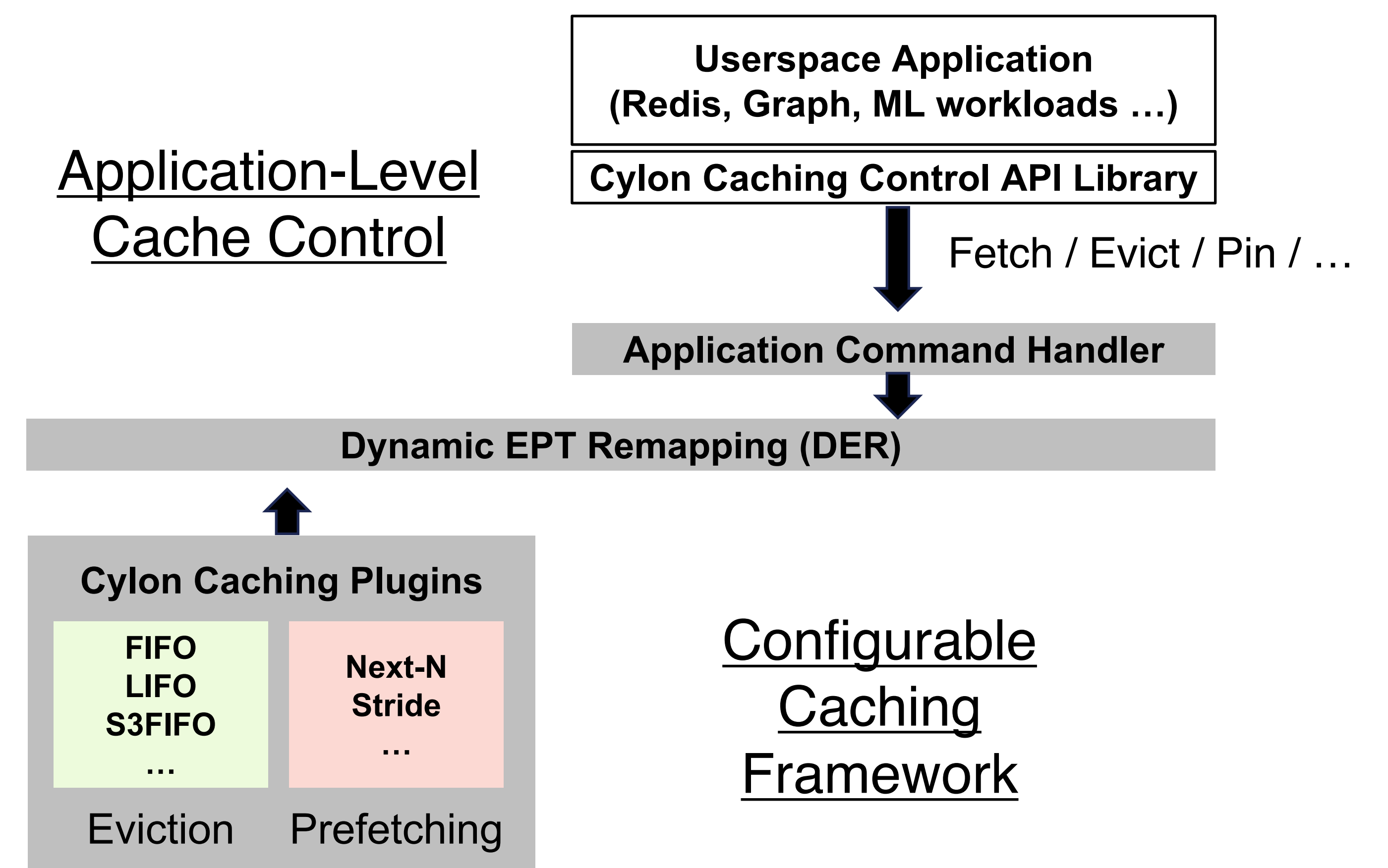
Cylon Key Contributions



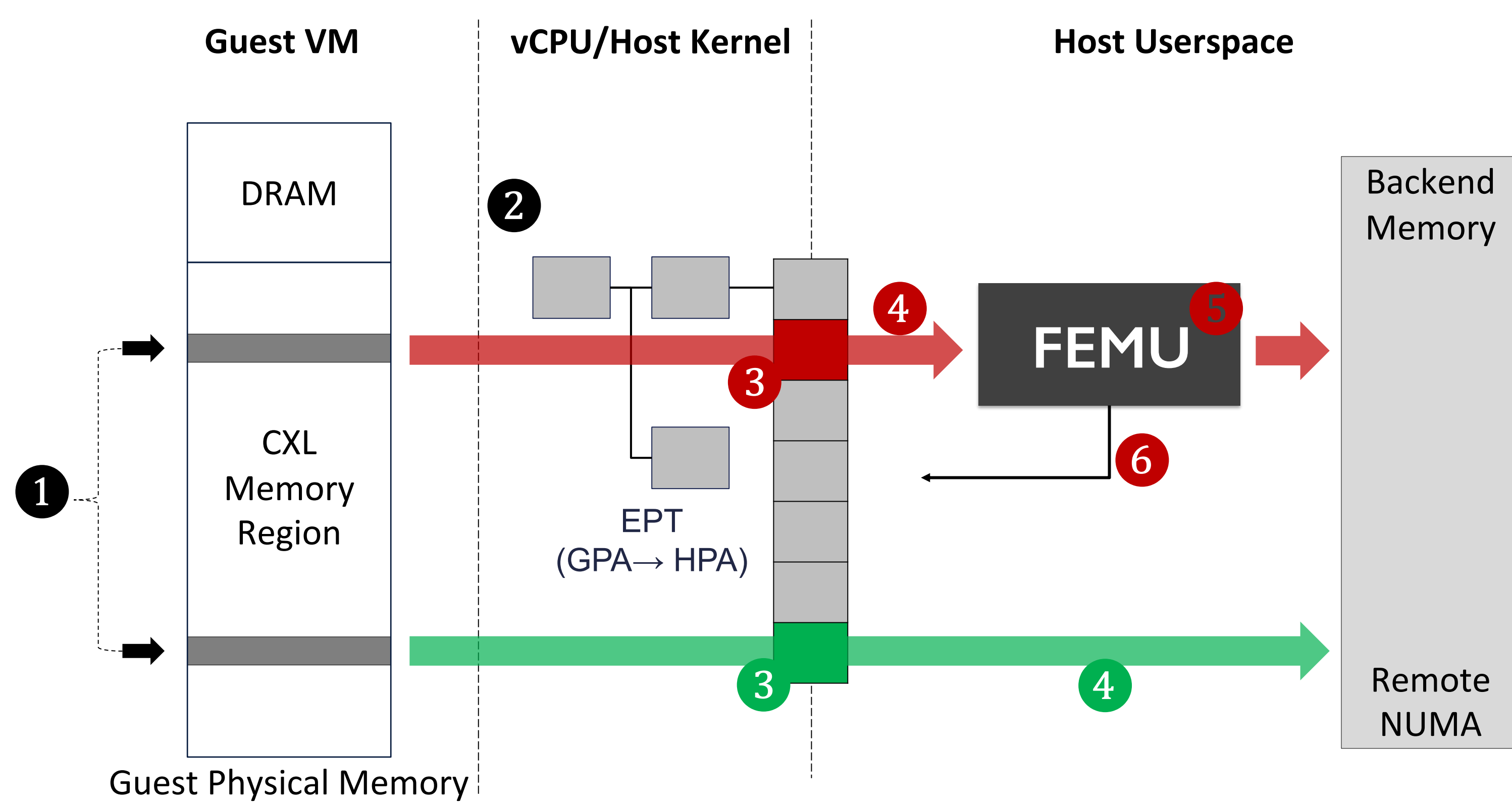
Cylon Architecture



Cylon Key Flexibility Features



Cylon Key Optimization: Dynamic EPT Remapping (DER) and Shared EPT



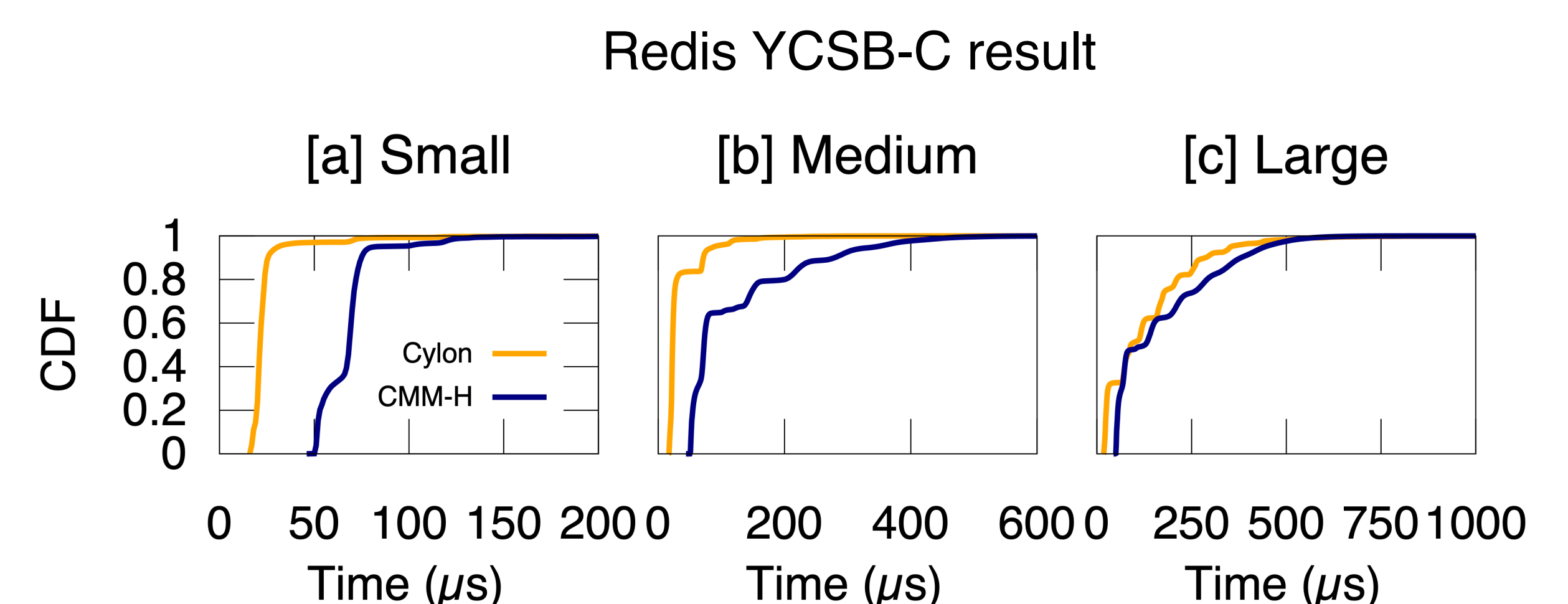
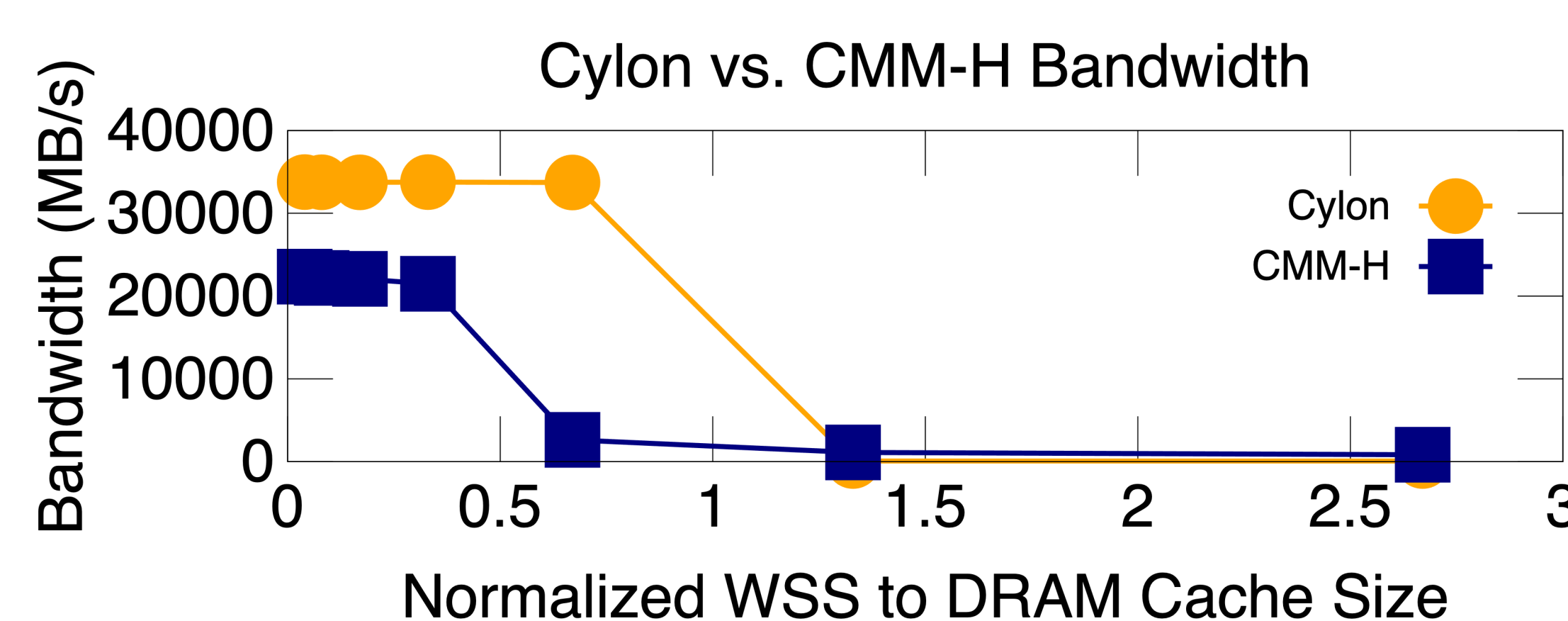
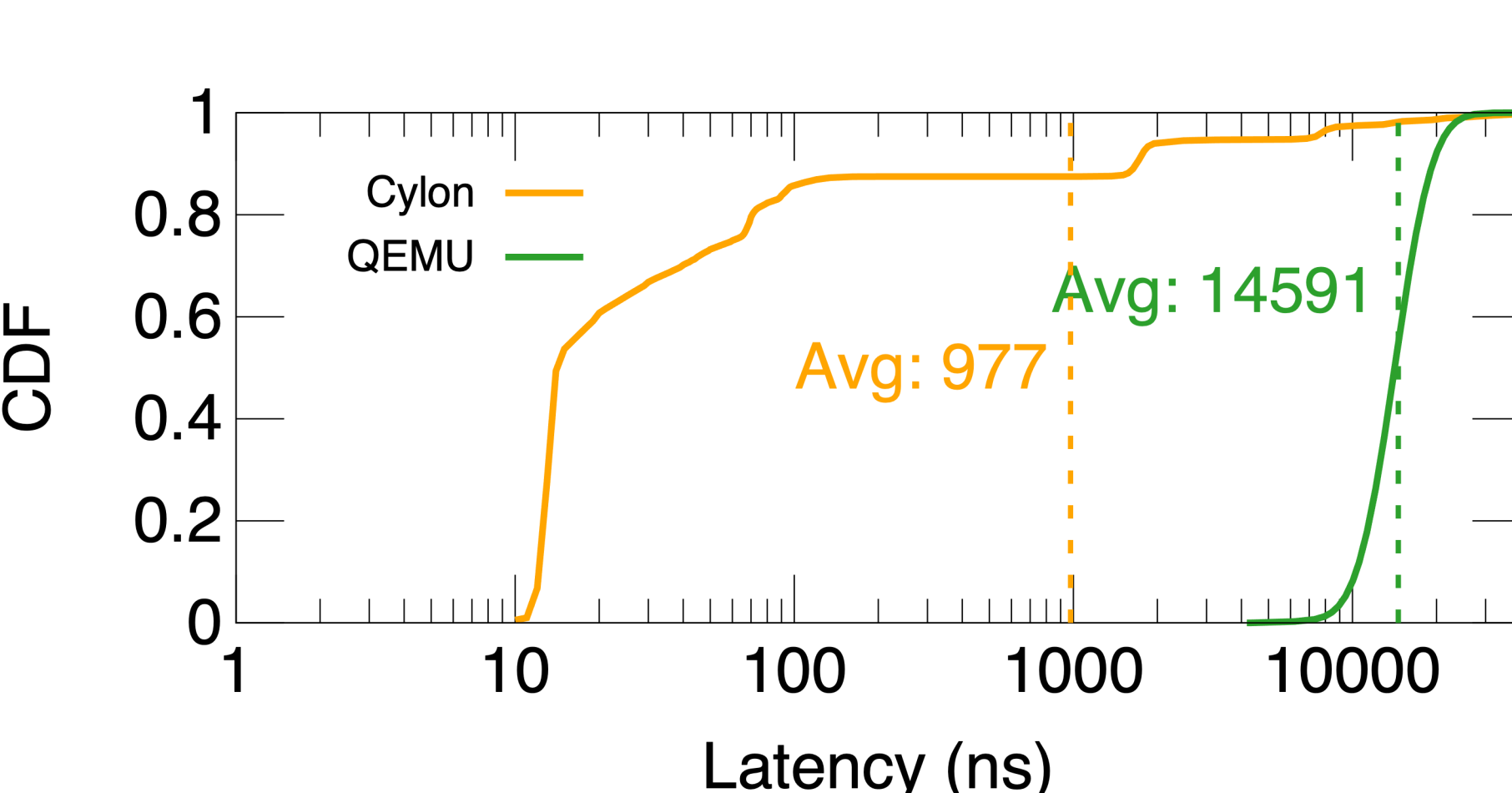
Cache Miss

- 1 Guest accesses CXL-SSD memory
- 2 MMU walks guest page table
- 3 EPTE marked as a MMIO-page
- 4 VM-exit triggered
- 5 FEMU performs cache management and NAND emulation
- 6 Upon cache entry evict/fetch, mark EPTE trap/direct

Cache Hit

- 1 Guest accesses CXL-SSD memory
- 2 MMU walks guest page table
- 3 EPTE points to the physical address of backend memory
- 4 Guest CPU accesses backend memory without VM-exit

Evaluation Results



Cylon is Fast

Cylon achieves realistic latency distribution by overcoming MMIO cost

Cylon mimics real CXL-SSD behaviors

Cylon reproduces the performance characteristics of real CXL-SSD devices

Cylon supports Full-stack Execution

Cylon runs unmodified applications and tracks real-hardware trends

