



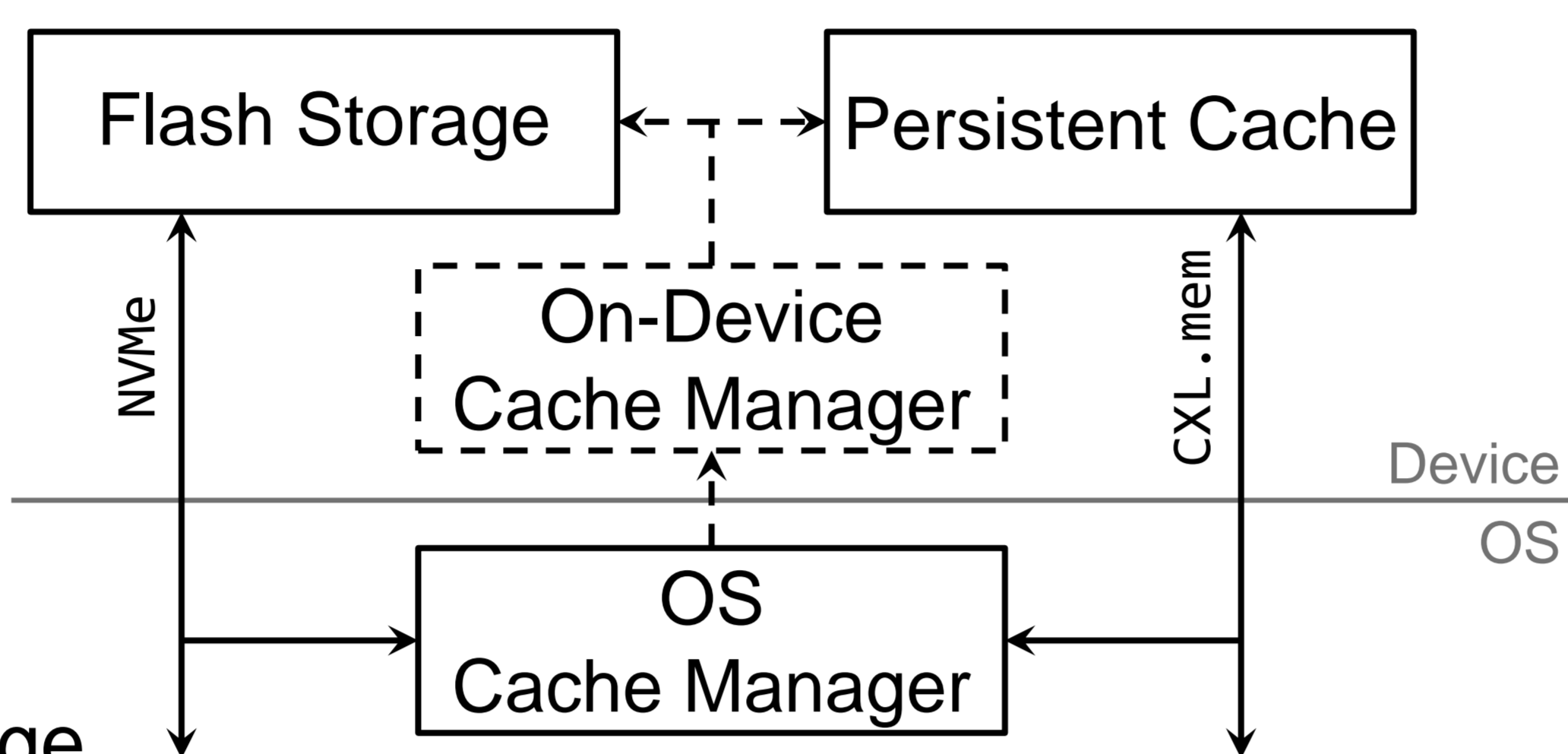
Paper

Fundamental OS Design Considerations for CXL-based Hybrid SSDs

Daniel Habicht, Yussuf Khalil, Lukas Werling, Thorsten Gröninger, and Frank Bellosa

Hybrid SSDs

- Dual-Paradigm
 - Sync load/store
 - Async block interface
- Small on-device cache
- Flash for permanent storage
- Commercial products on horizon (e.g., Samsung's CMM-H)



Direct Access (DAX)

- Bypass volatile page cache for direct load/store access to storage
- File system support (fsdax) allows applications to mmap storage
- CXL GPF for instant persistence
- Supported in Linux and Windows

Analysis of DAX in Linux

- Controlled through **per-inode DAX flag**
- Assumes “uniform” PMEM device (e.g., Optane PMM)
 - low-latency load/store access required at all times
- On-device caches not considered → OS out of loop

Linux DAX for hybrid SSDs

stalls CPU on cache miss

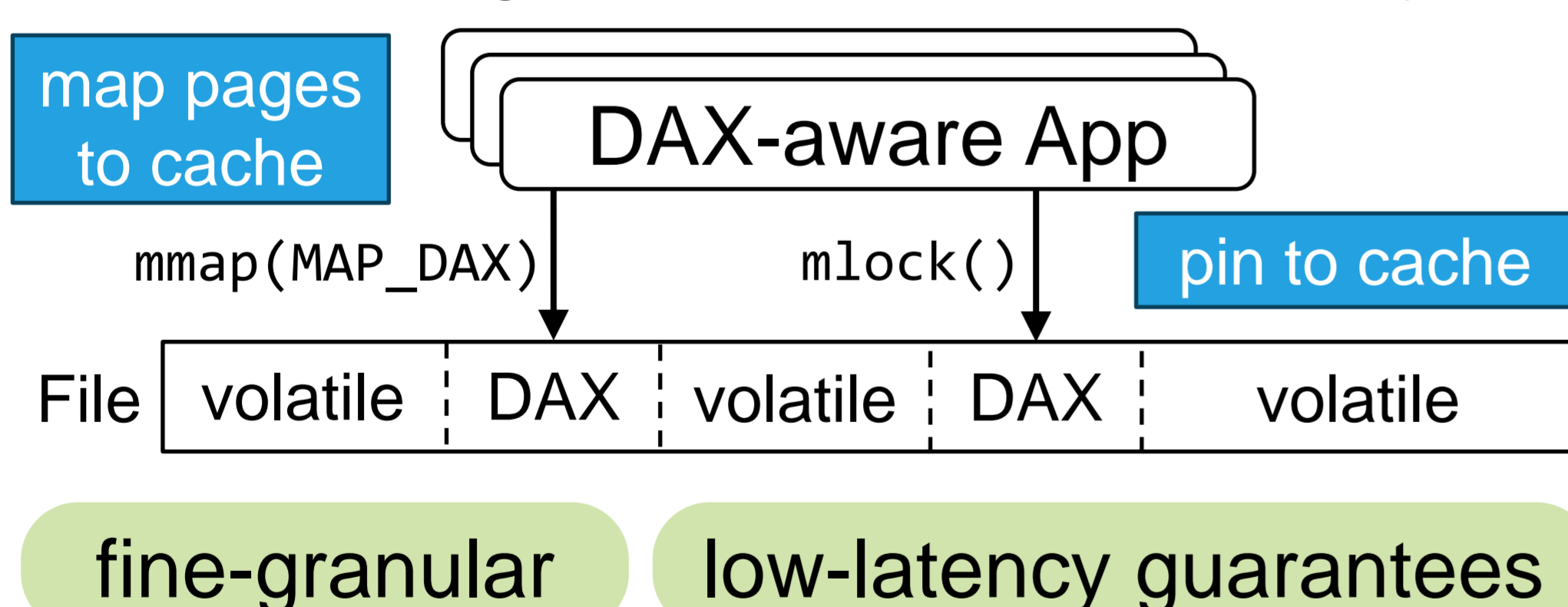
pressure on cache due to DAX granularity

no performance isolation

Our Solution: OS-Centric Hybrid SSD Management

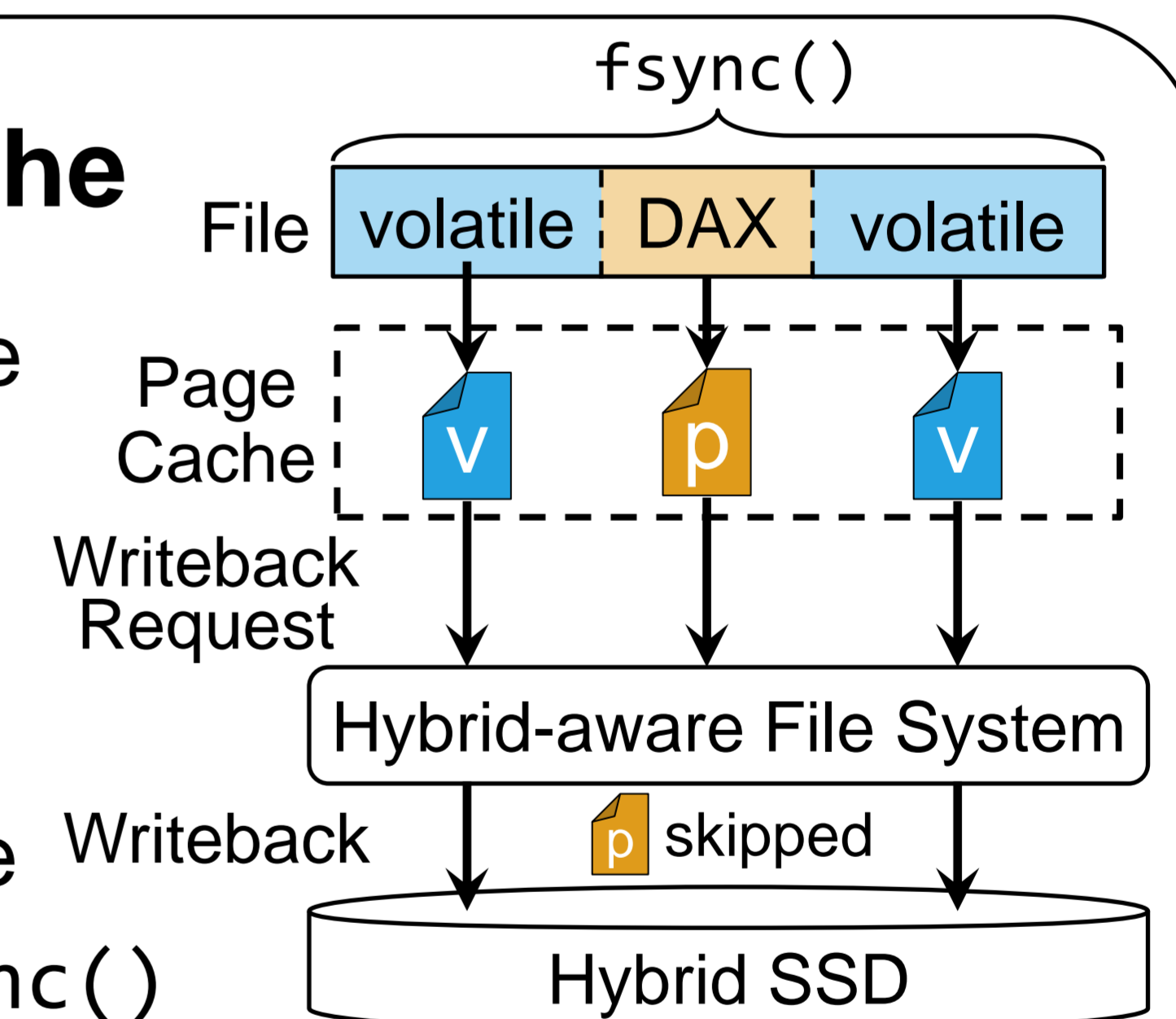
User-Space Interface

- DAX mappings on page granularity
- Pinned pages for predictable latency



Persistence-Aware Page Cache

- DAX VMAs tracked in per-file rb tree
 - check tree when allocating page
- Page cache reflects cache state
 - page fault on cache miss
- Use persistence guarantees of page
 - skip synchronous writeback on `fsync()`

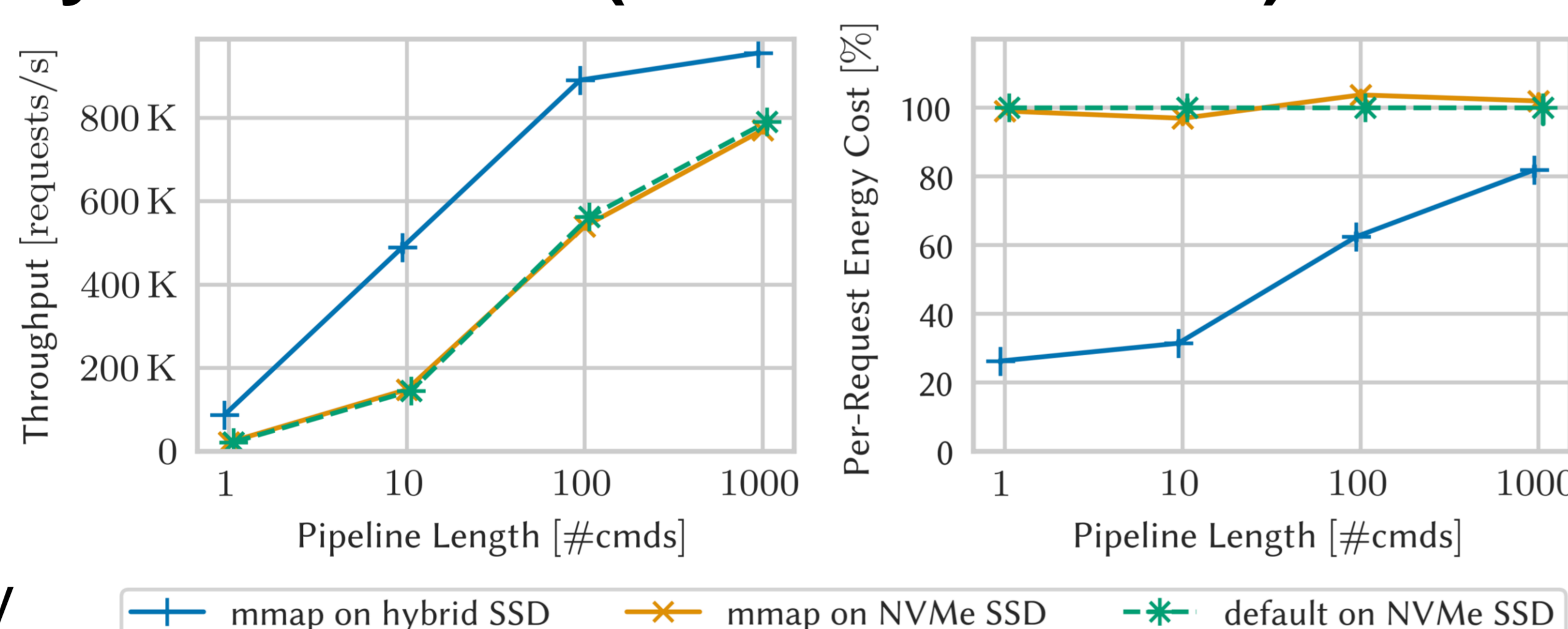


asynchronous cache miss handling

lightweight `fsync()`

Evaluation using Key-Value Store (emulated device)

- Impractical overhead for AOF (always sync)
- Emulate hybrid SSD with SSD + CXL mem
- With hybrid SSDs up to $4.1 \times$ faster and 74 % lower energy



Future Work

- Transparent cache usage
- Explore HW design space
- Reevaluate on real-world HW
- Evaluate hybrid SSDs in consumer context