Replicating Non-Volatile Main Memory

Yiying Zhang,

Jian Yang, Amirsaman Memaripour, Steven Swanson

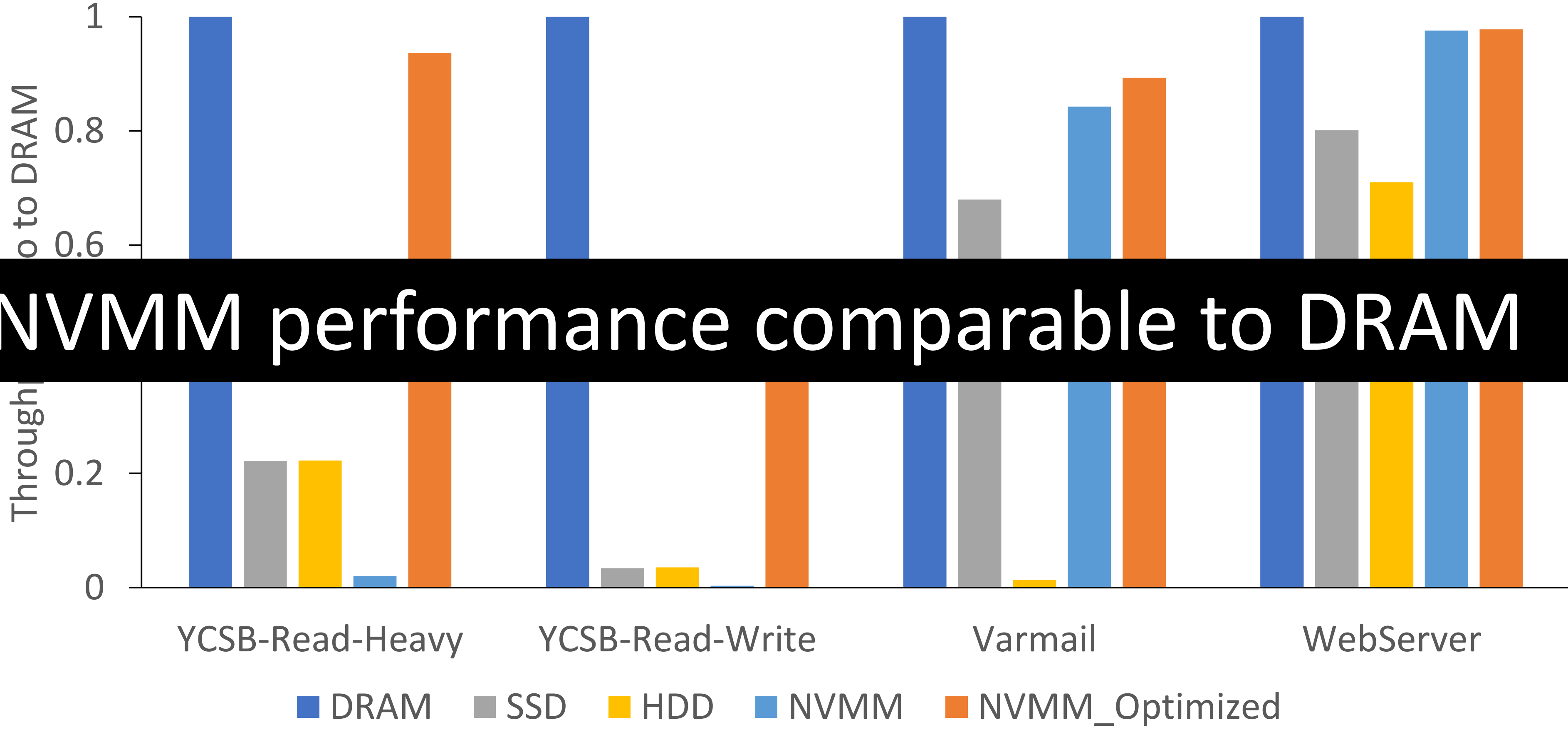
NVSL
Non-volatile Systems Laboratory

UCSDCSE
Computer Science and Engineering

UCSD

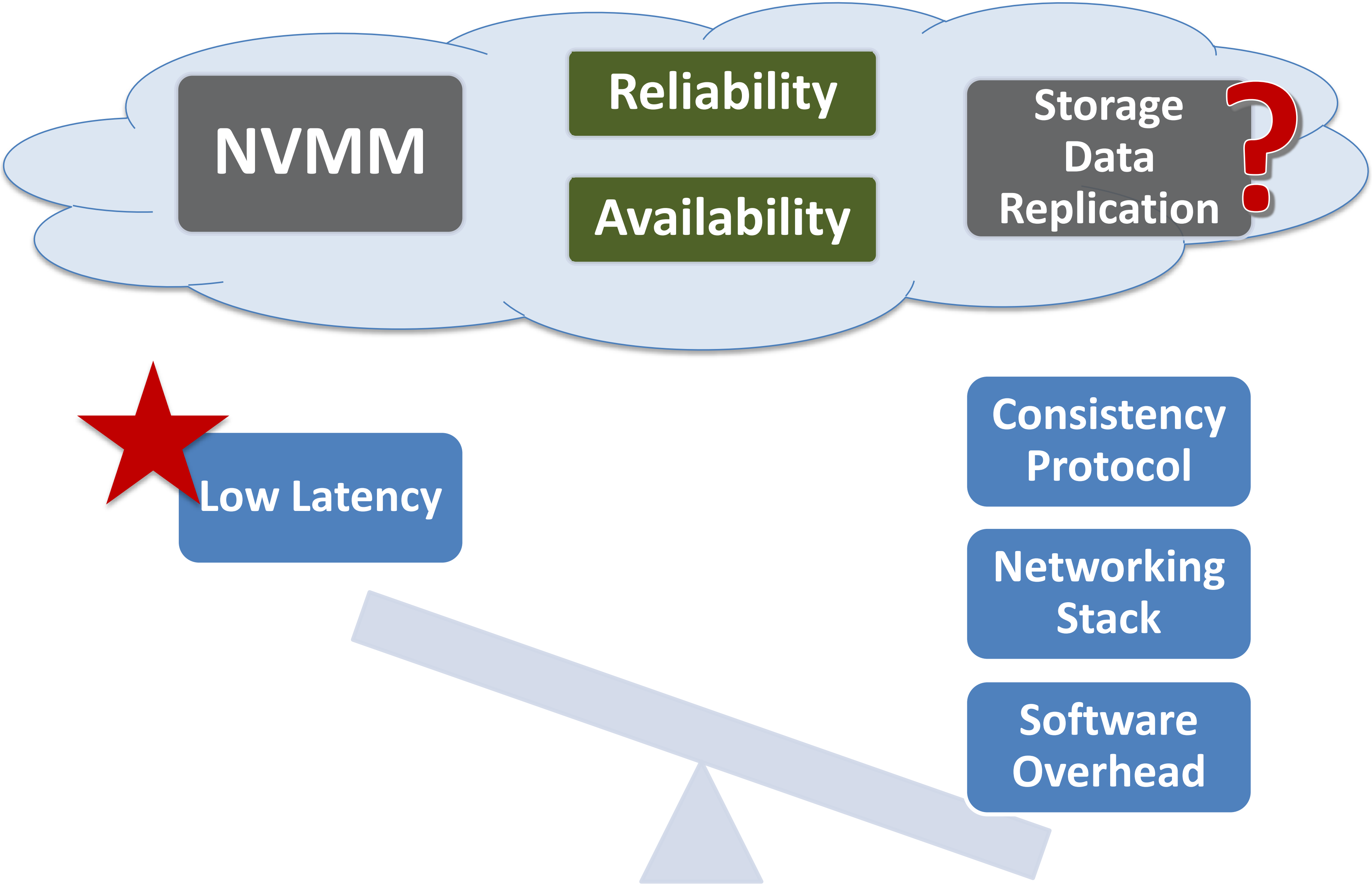
Non-Volatile Main Memory (NVMM)

- Next generation non-volatile memory
- NVMM: NVM as (persistent) main memory



NVMM performance comparable to DRAM

NVMM in Data Center



Mojim: Reliable and Highly-Available NVMM

- NVMM-to-NVMM fine-grained replication
- RDMA-based replication optimized for NVMM
- Two-tier architecture
- Flexible consistency, reliability, availability, costs

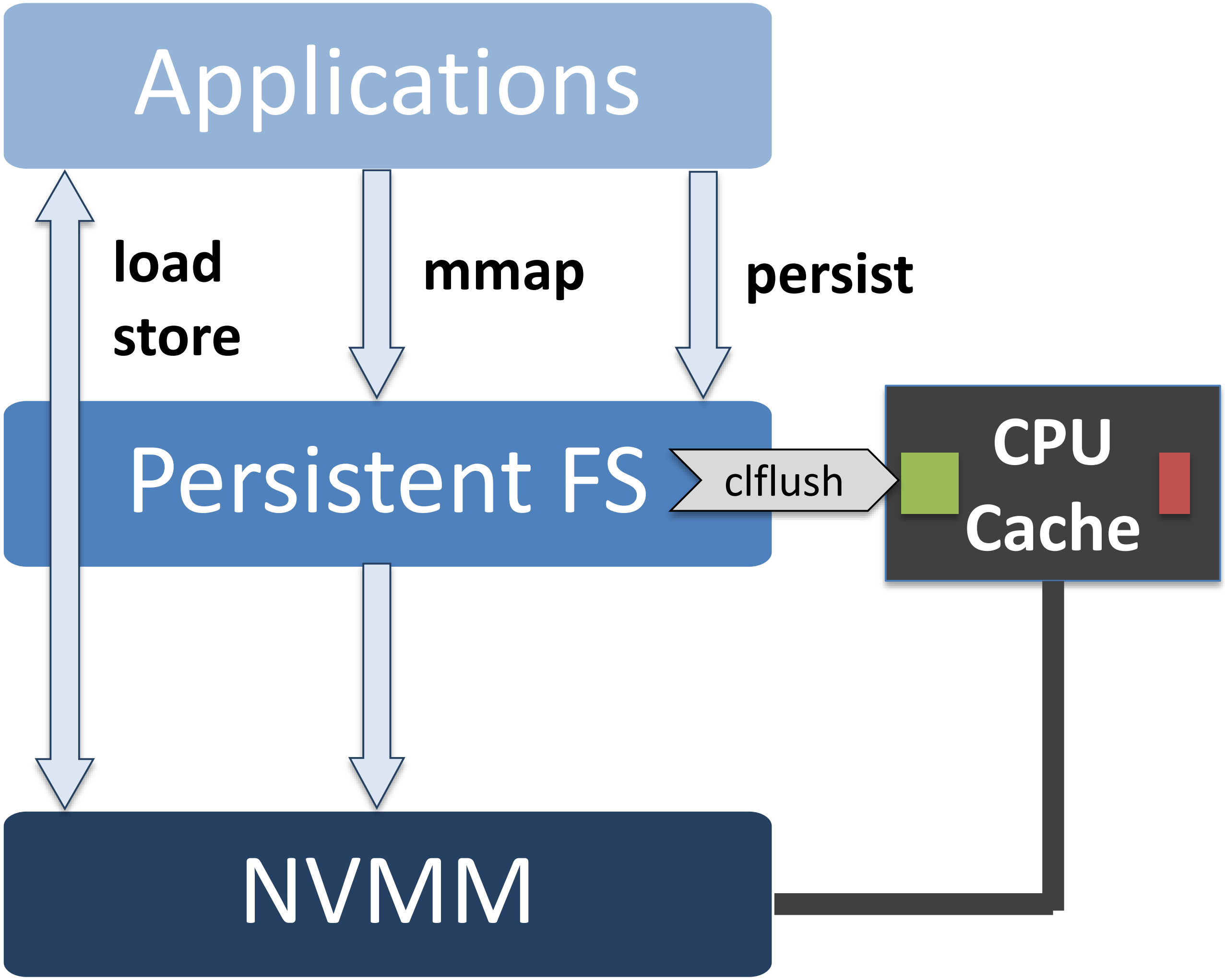
Mojim Results Highlight

- **29% – 73%** average latency of un-replicated
- **0.5x – 3.5x** bandwidth of un-replicated
- **3.4x – 4x** faster than MongoDB replication
- **And stronger consistency & reliability!**
- Instant fail over, 1.9 sec reconstruction

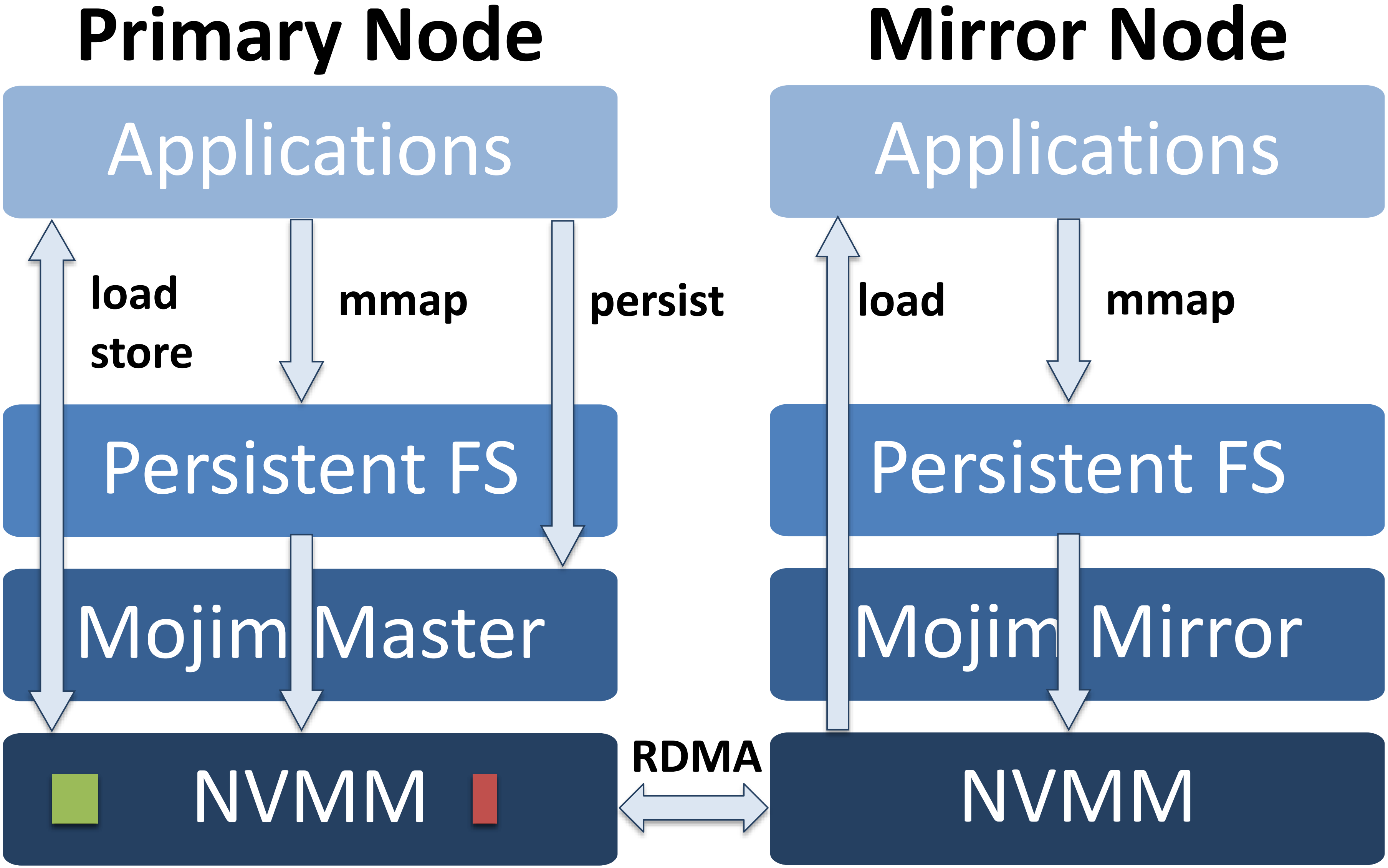
Outline

- Introduction
- Mojim Design and Architecture
- Mojim Implementation
- Evaluation Results
- Conclusion

Un-replicated NVMM

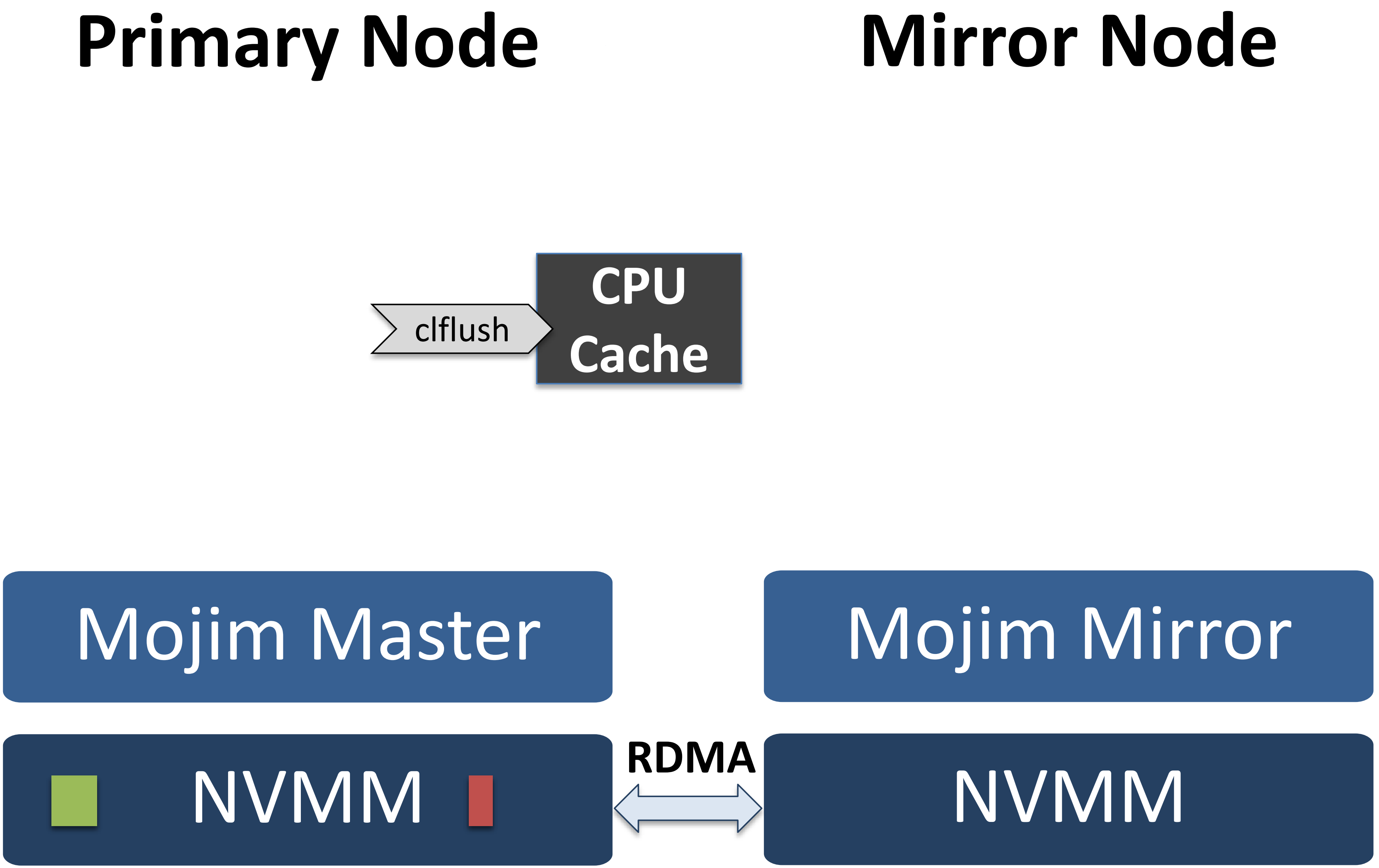


Mojim Architecture



Mojim Architecture

opt1



Primary Node

Mirror Node

clflush

CPU Cache

Mojim Master

Mojim Mirror

NVMM

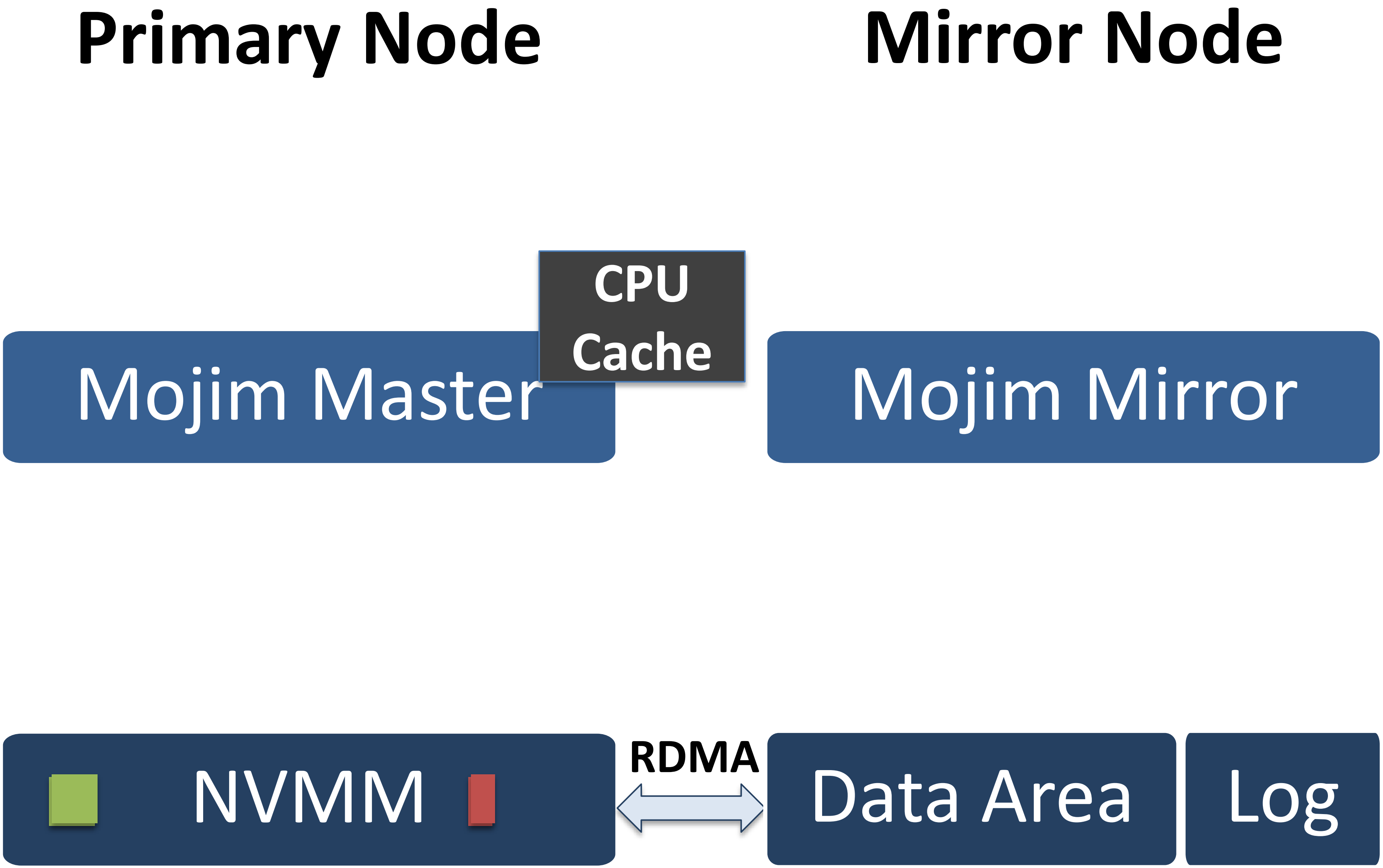
RDMA

NVMM

Mojim Architecture

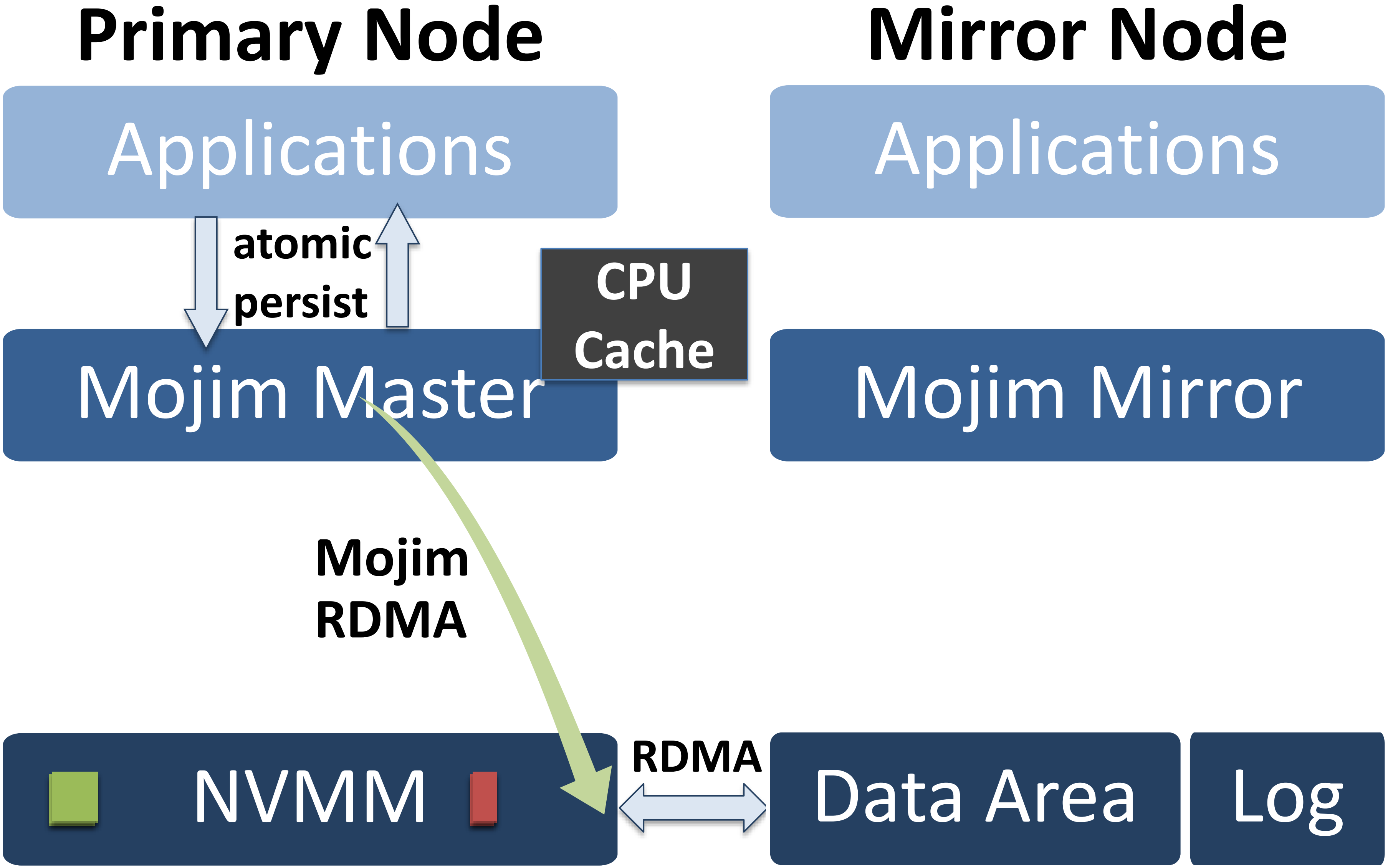
opt1

opt2



Mojim Architecture

opt1 opt2 opt3



Primary Node

Mirror Node

Applications

Applications

atomic persist

CPU Cache

Mojim Master

Mojim Mirror

Mojim RDMA

NVMM

RDMA

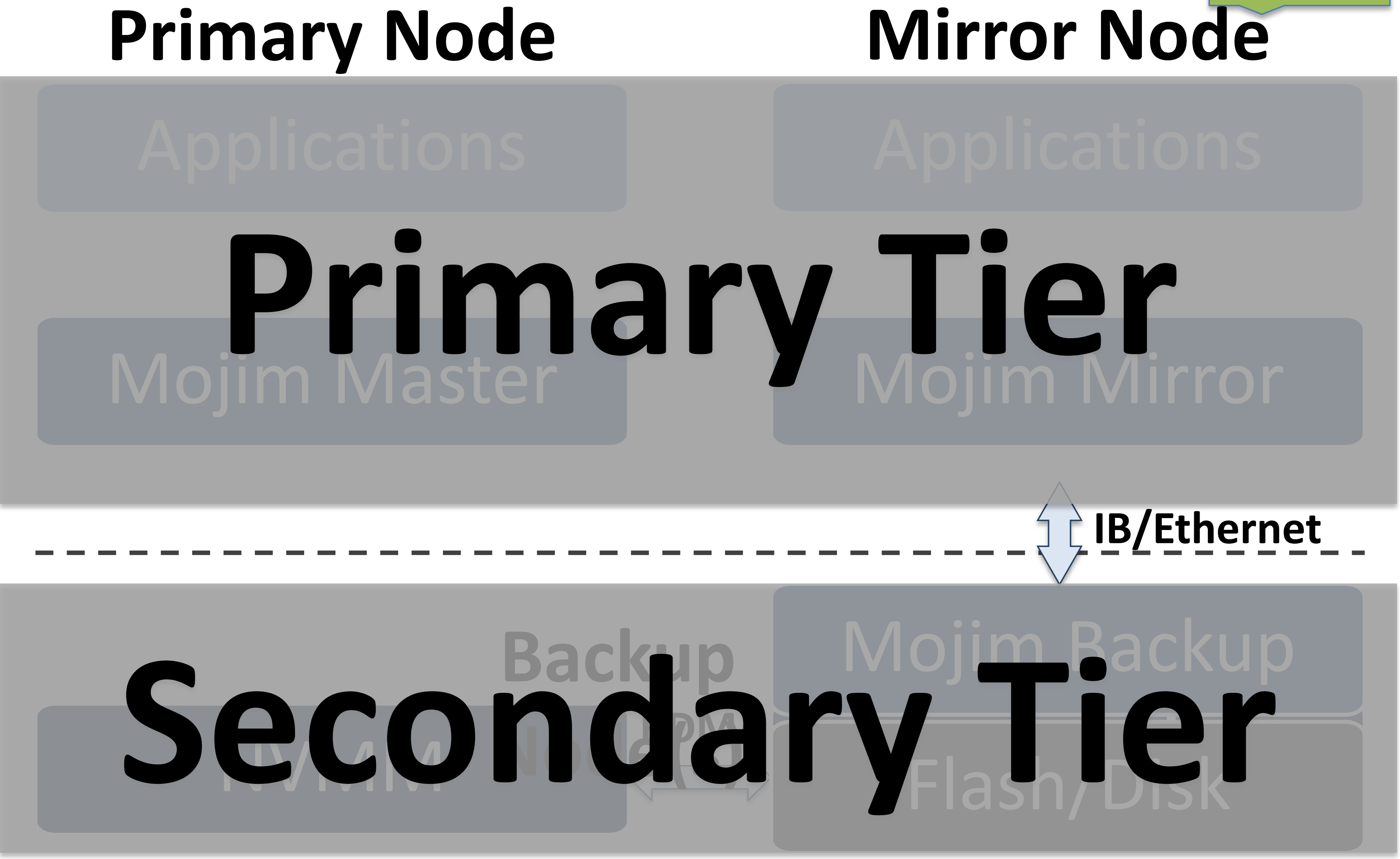
Data Area

Log

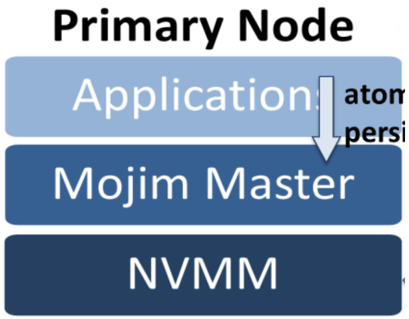
Mojim Architecture

opt1 opt2 opt3 opt4

opts ...



Flexible Modes



	Scheme	Performance	Reliability	Availability	Consistency	$ Cost
	Un-replicated	Good	0	Worst	N/A	Low
Mojim	Async	Good	1	Good	Weak	Fair
	Sync	Good	1	Good	Strong	Fair
	Sync-disk	Good	1	OK	Strong	Low
	Sync-two-tier	Good	N-1	Best	Strong+Weak	High
	Sync-twotier-ETH	Bad	N-1	Good	Strong+Weak	Fair
Existing	Write-all	Bad	N-1	Best	Strong	High
	Chain-rep	OK	N-1	Best	Strong	High
	Broadcast-rep	OK	N-1	Best	Strong	High

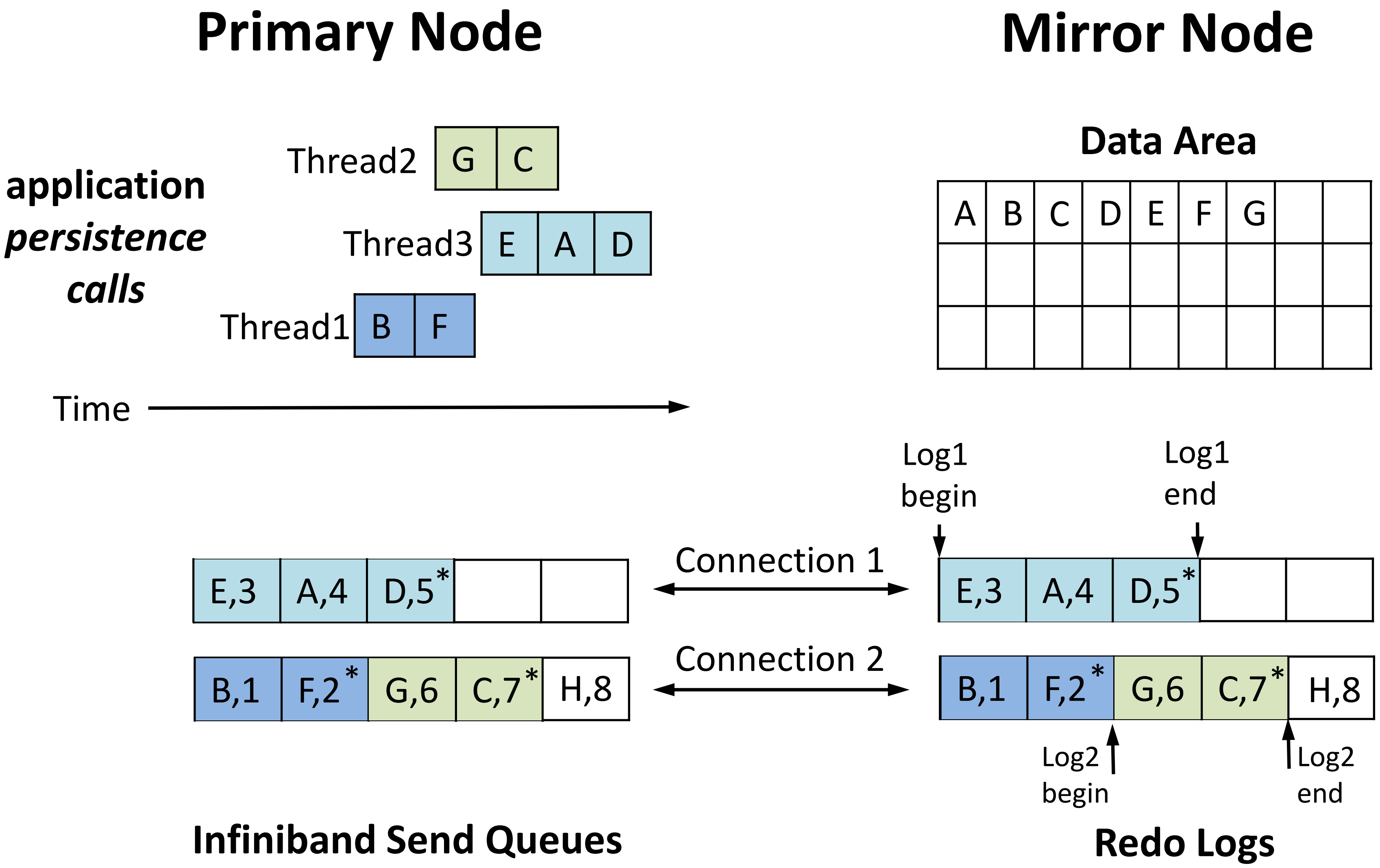
Outline

- Introduction
- Mojim Design and Architecture
- Mojim Implementation
- Evaluation Results
- Conclusion

Implementation

- Mojim as a generic layer in Linux kernel
- Networking
  - Optimized implementation of IBV-verbs in kernel
  - Zero copy, reliable
  - Multiple connections, multiple receiving threads polling
- Replication and recovery
  - Redo logs on mirror and backup nodes
  - Atomic operation support
  - Fast recovery (ensured by thresholds)

Primary Tier Replication



Primary Node

Mirror Node

application
persistence
calls

Thread2 G C

Thread3 E A D

Thread1 B F

Time

Data Area

A	B	C	D	E	F	G		

Log1 begin

Log1 end

E,3 | A,4 | D,5* | |

Connection 1

B,1 | F,2* | G,6 | C,7* | H,8

Connection 2

E,3 | A,4 | D,5* | |

B,1 | F,2* | G,6 | C,7* | H,8

Log2 begin

Log2 end

Infiniband Send Queues

Redo Logs

Mojim Applications

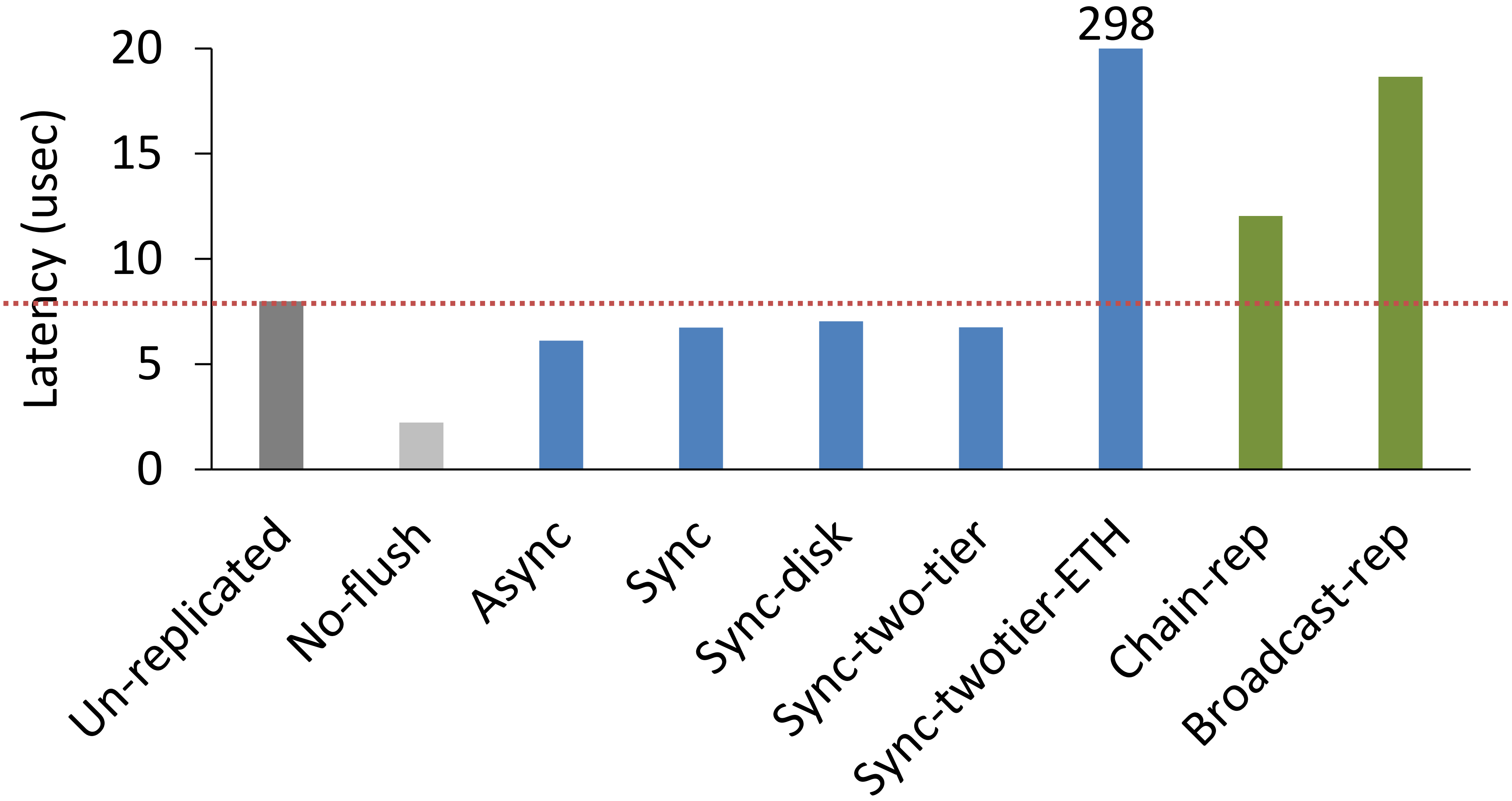
- Persistent Memory File System
- Google Hash Table
- MongoDB
- No or small change to applications

Outline

- Introduction
- Mojim Design and Architecture
- Mojim Implementation
- Evaluation Results
- Conclusion

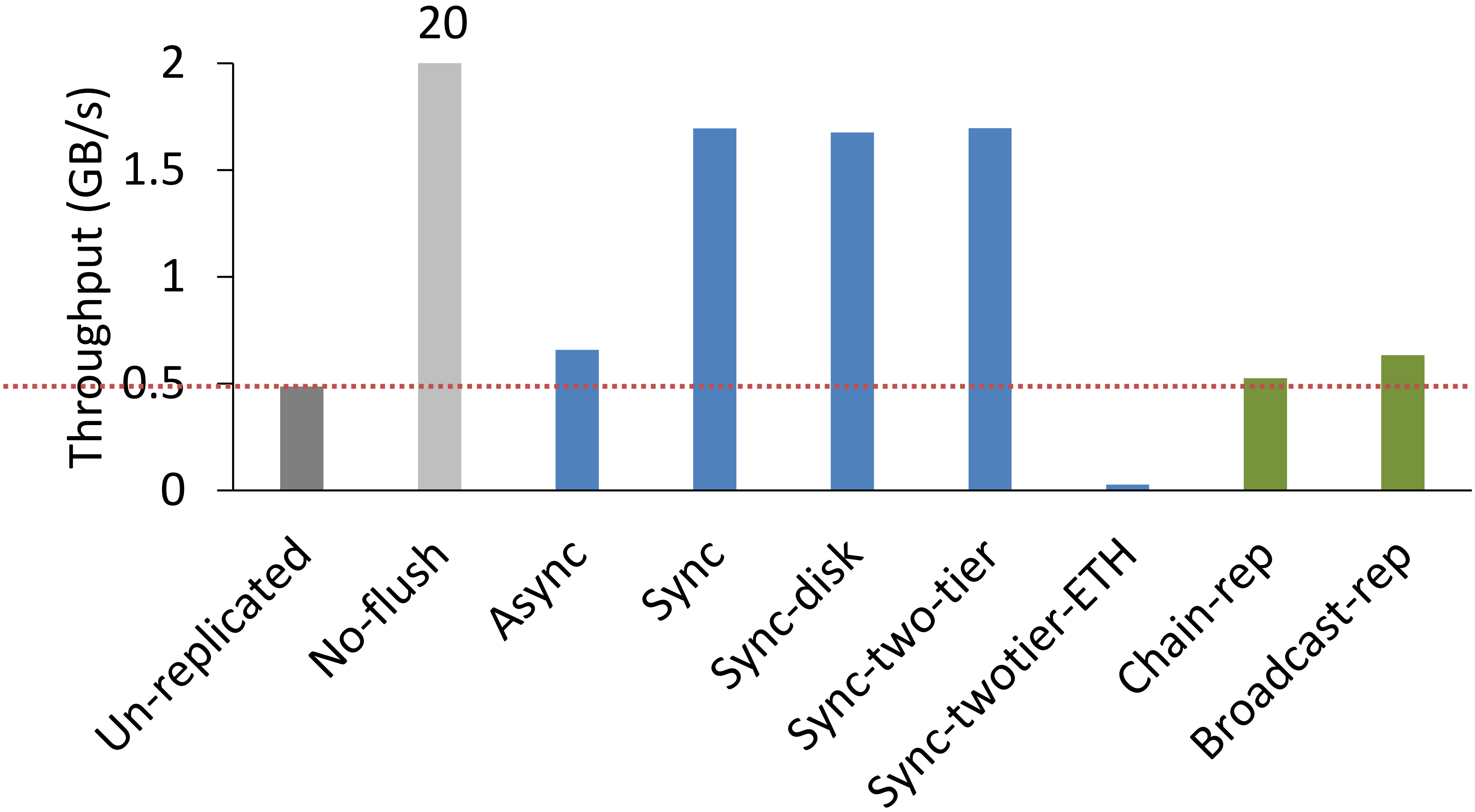
Data Persistence Latency

- Testbed: DRAM as NVM, 40Gbps Infiniband, 1Gbps Ethernet
- Workload: Persist random 4KB regions in a 4GB *mmap*'d file

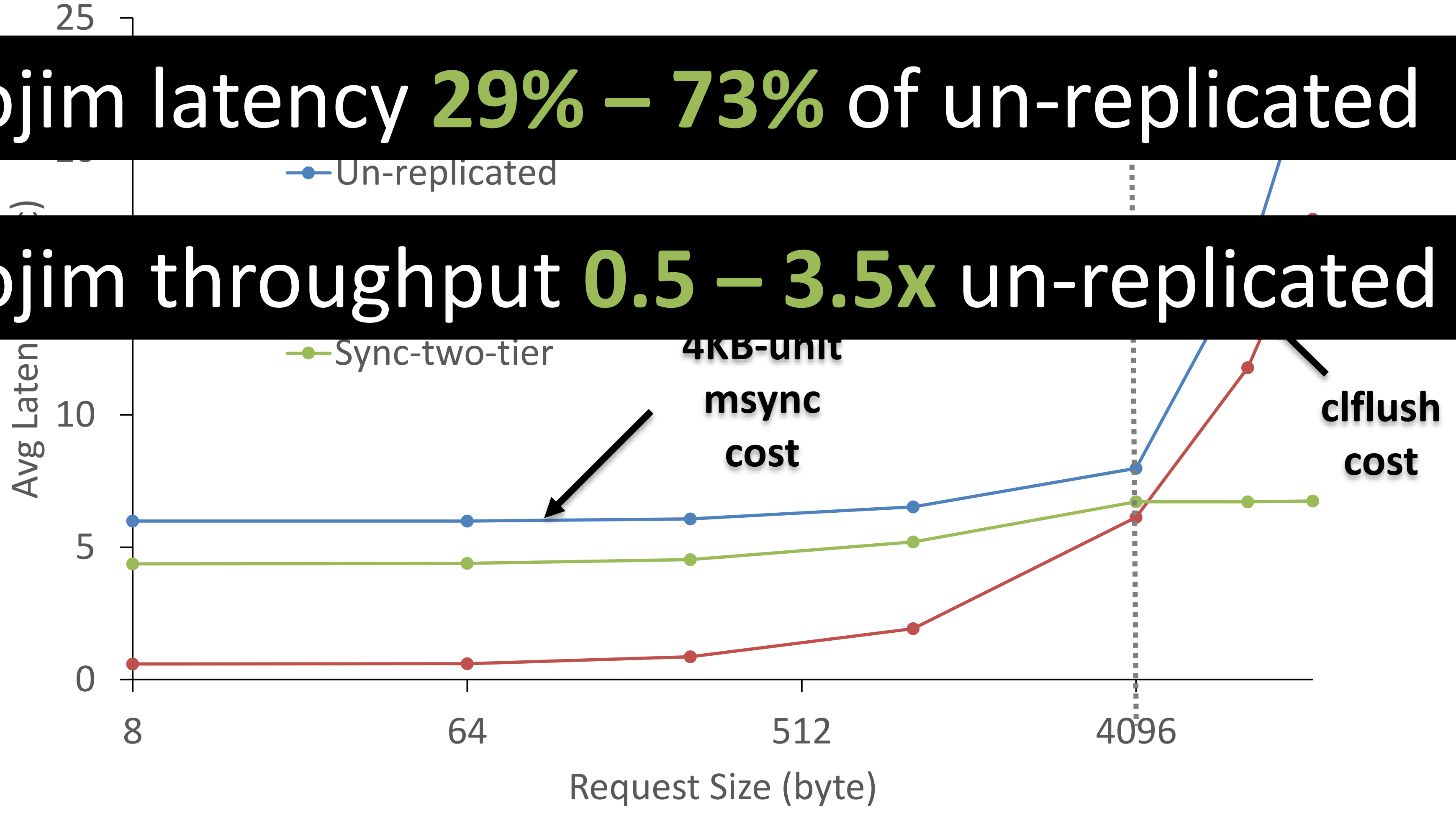


Data Persistence Throughput

- Workload: Persist random 12KB regions in a 4GB *mmap*’d file



Effect of Persisted Data Size



Mojim latency **29% – 73%** of un-replicated

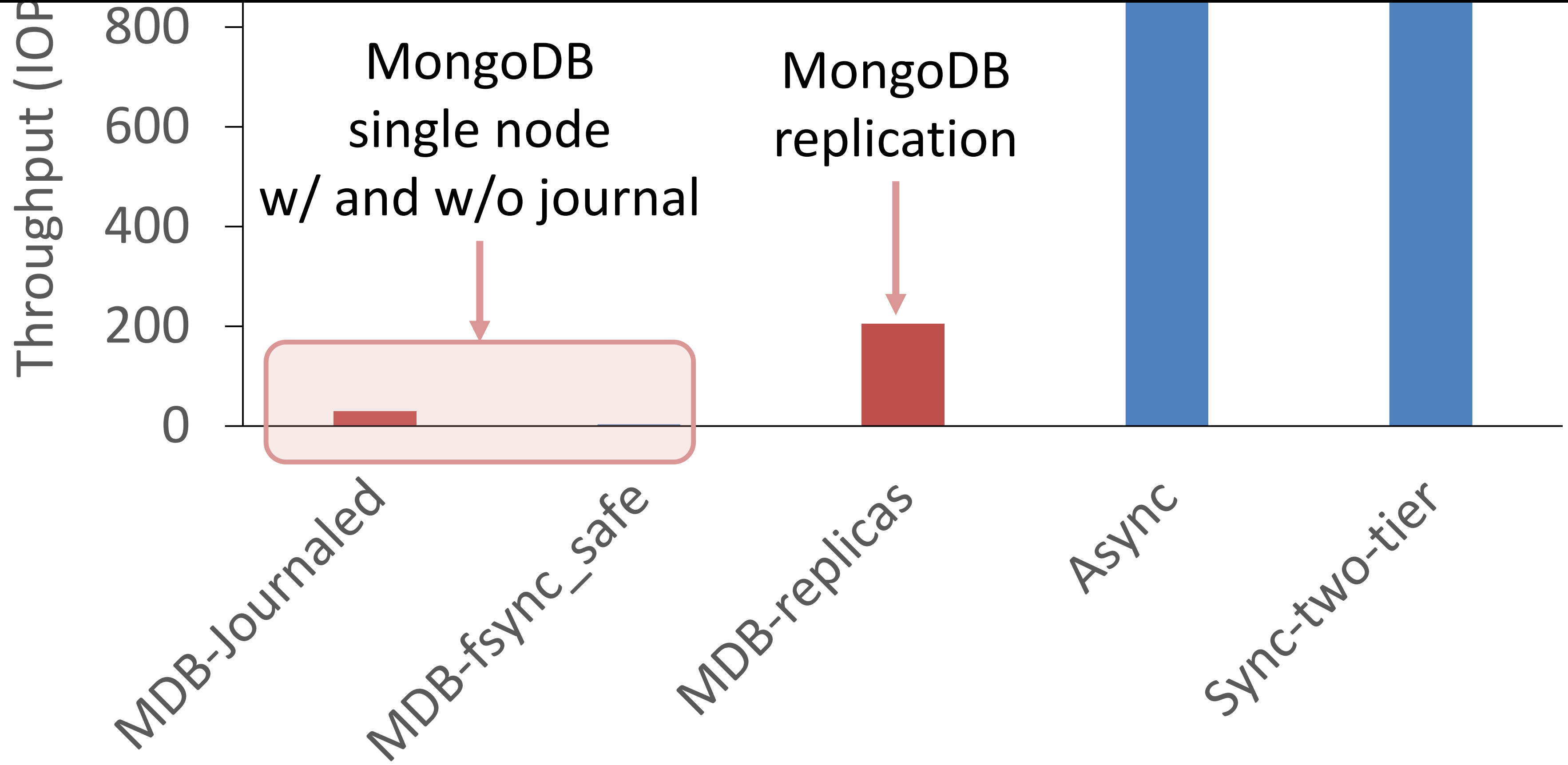
Mojim throughput **0.5 – 3.5x** un-replicated

4KB-unit msync cost

clflush cost

MongoDB Key-Value Pair Load

Mojim much faster than existing replication



MongoDB single node w/ and w/o journal

MongoDB replication

Mojim Summary

- Provide reliability and high availability to NVMM
- RDMA-based replication optimized for NVMM
- Two-tier architecture
- Flexible modes offering different guarantees
- Performance even better than un-replicated

Conclusion

- Gap between storage and memory getting smaller
- Time to rethink traditional software/networking
- More problems to be solved
  - Virtualization (e.g., replication/migration/snapshot)
  - Distributed systems (built on top of Mojim?)
  - Abstraction, programming language
  - Mobile devices

Thank you !
Questions ?

Yiying Zhang yiyingzhang@cs.ucsd.edu
Steven Swanson swanson@cs.ucsd.edu

NVSL
Non-volatile Systems Laboratory

UCSDCSE
Computer Science and Engineering

UCSD