NVRAM-aware Logging in Transaction Systems

Jian Huang

Karsten Schwan Moinuddin K. Qureshi

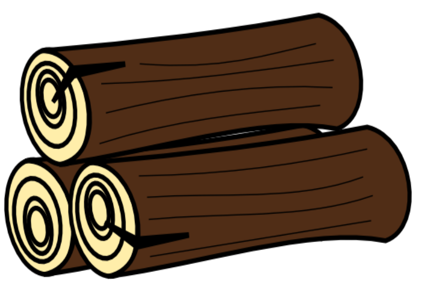
Logging Support for Transactions

Logging Support for Transactions

A C I D

ATOMICITY CONSISTENCY ISOLATION DURABILITY

ARIES: Disk-Based Approach (TODS'92)



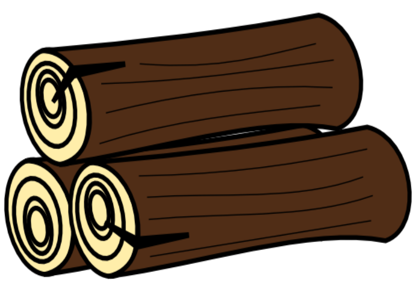
Write-ahead Logging

Logging Support for Transactions

A C I D

ATOMICITY CONSISTENCY ISOLATION DURABILITY

ARIES: Disk-Based Approach (TODS'92)



\+

Write-ahead Logging

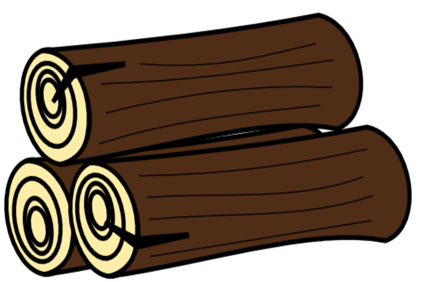
Rollback

Logging Support for Transactions

A C I D

ATOMICITY CONSISTENCY ISOLATION DURABILITY

ARIES: Disk-Based Approach (TODS'92)

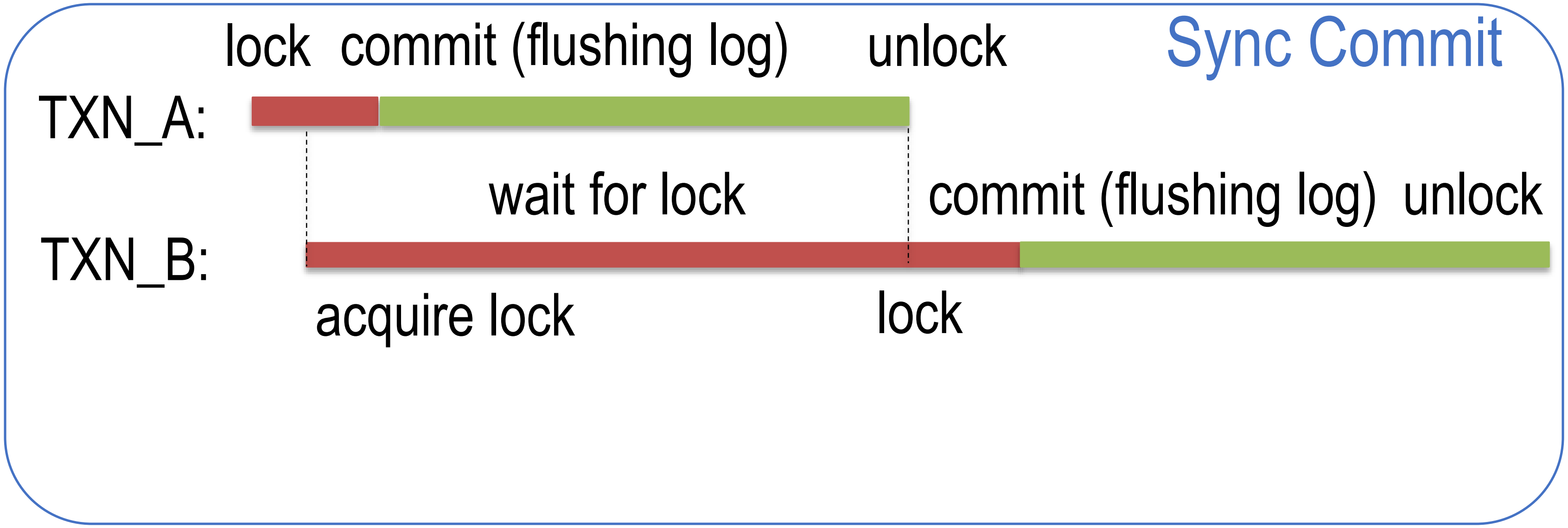


\+

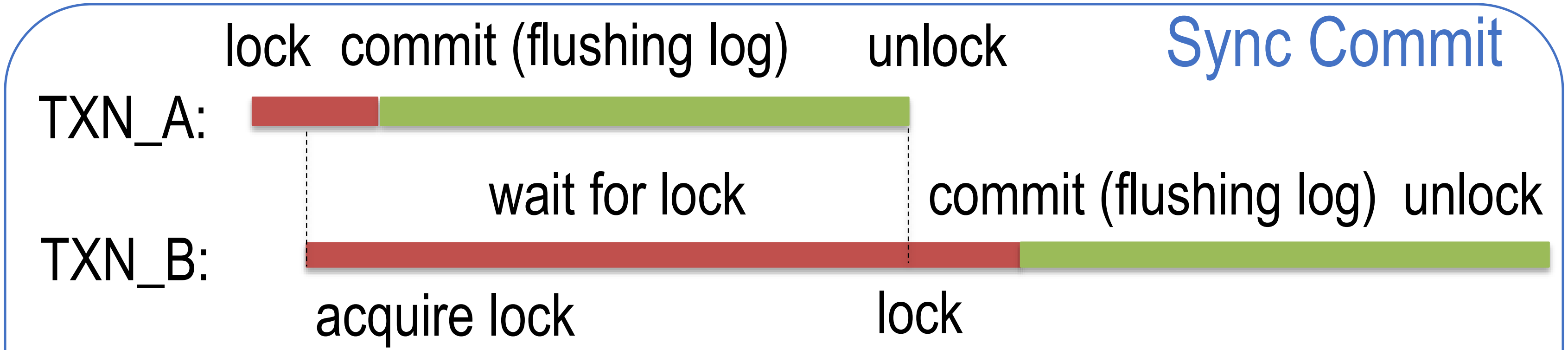
\+

Write-ahead Logging | Rollback | Centralized Log Buffer

On the Logging Persistence

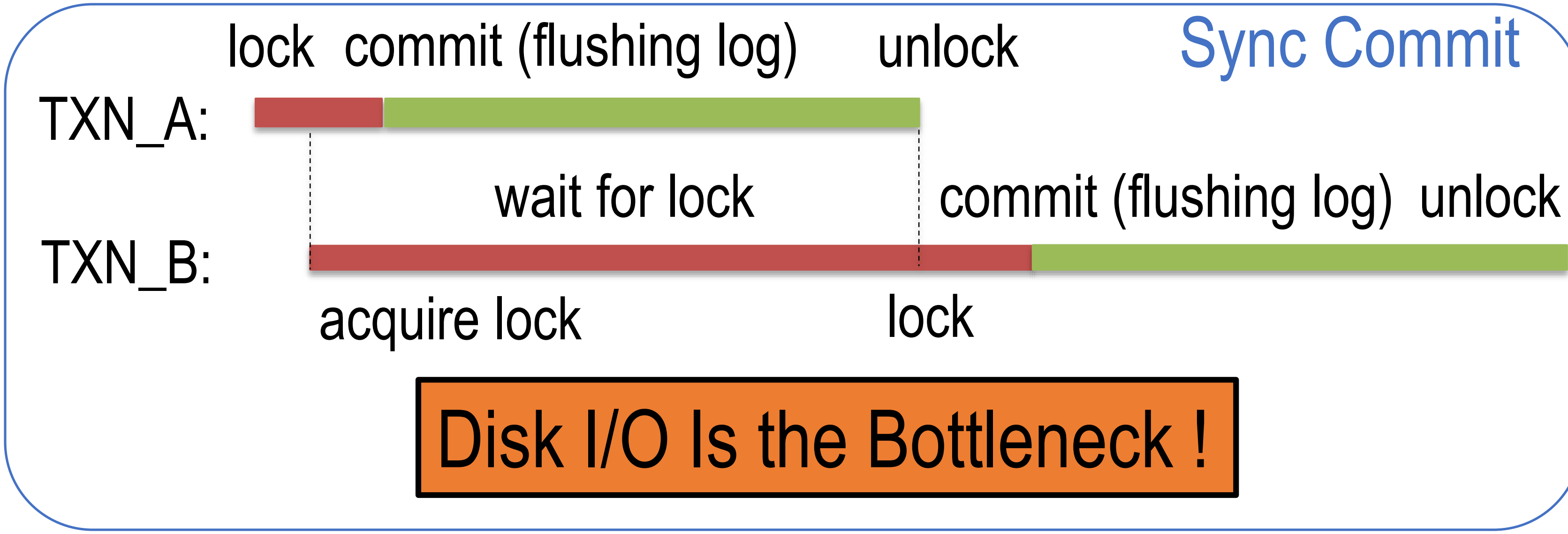


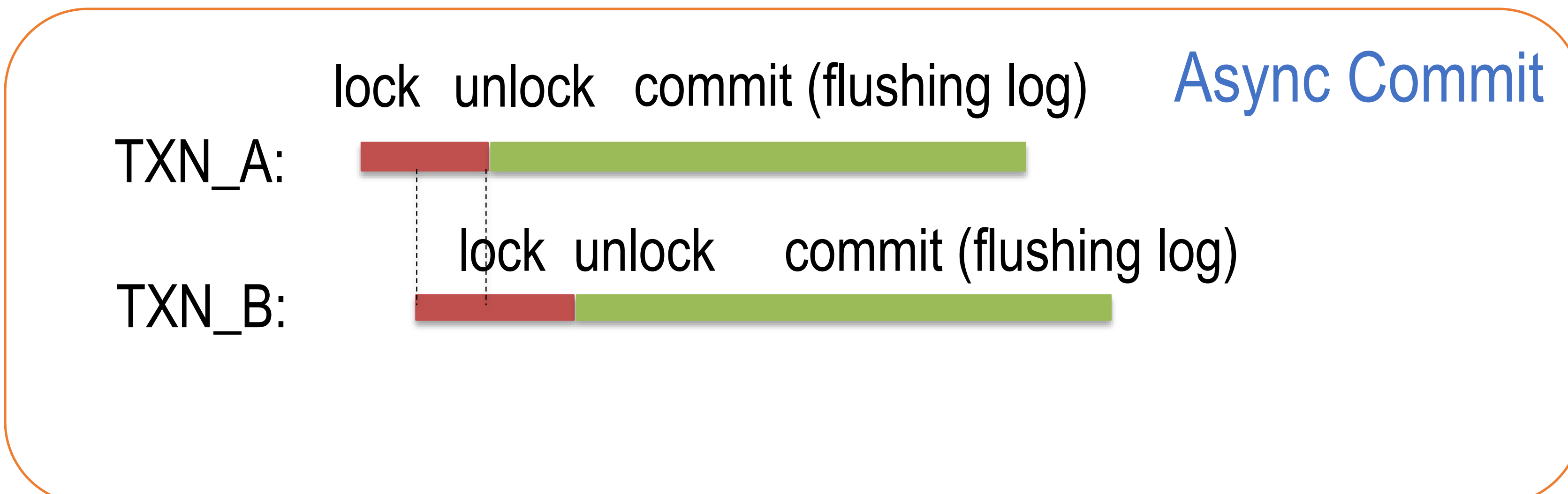
On the Logging Persistence



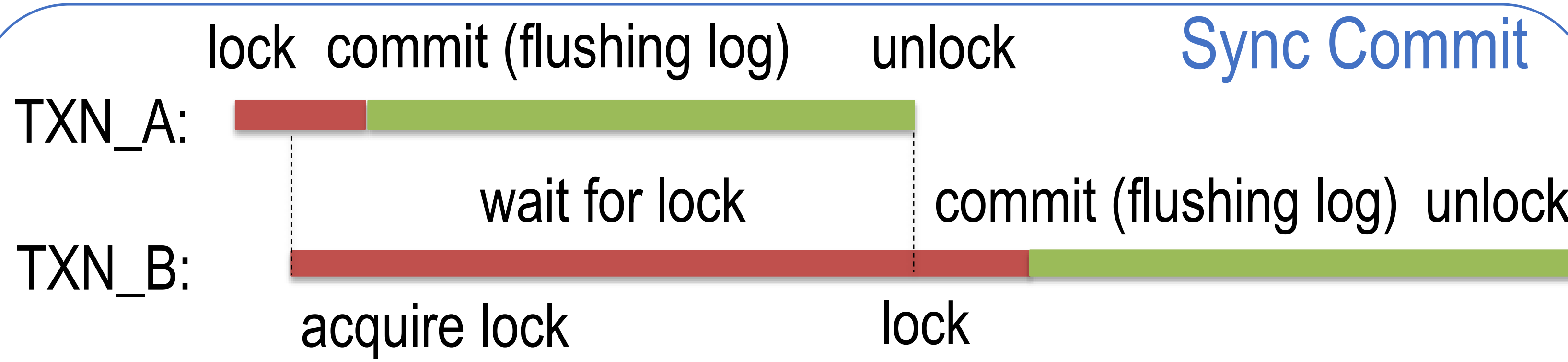
Disk I/O Is the Bottleneck !

On the Logging Persistence

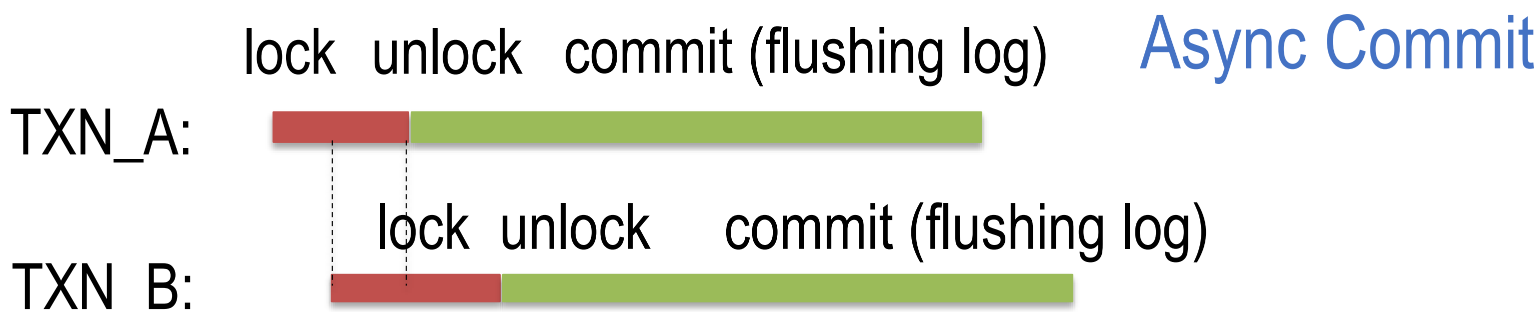




On the Logging Persistence



Disk I/O Is the Bottleneck !



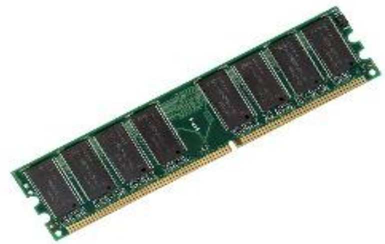
At the risk of partial data loss and incorrect result !

Rethink the Logging with NVRAM

Close to DRAM

Rethink the Logging with NVRAM

+

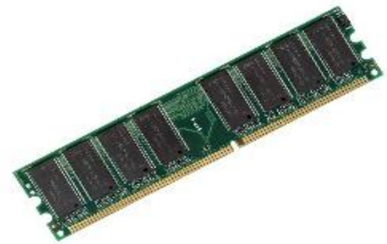


Close to DRAM

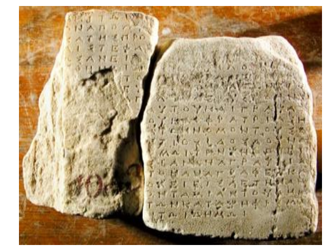
Byte Addressability

Rethink the Logging with NVRAM

+



+

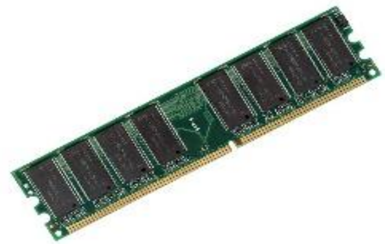


Close to DRAM

Byte Addressability

Non-Volatility

Rethink the Logging with NVRAM

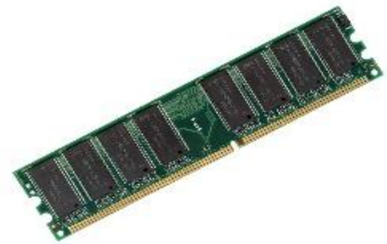


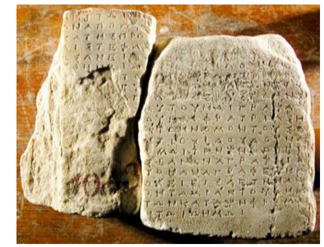
Close to DRAM + Byte Addressability + Non-Volatility

Disk-based logging

- Avoid frequent disk access
- Obtain sequential access pattern

Rethink the Logging with NVRAM





Close to DRAM + Byte Addressability + Non-Volatility

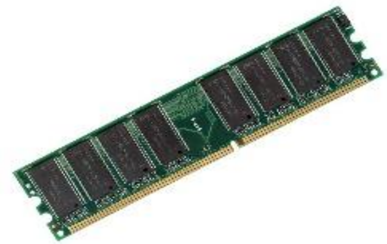
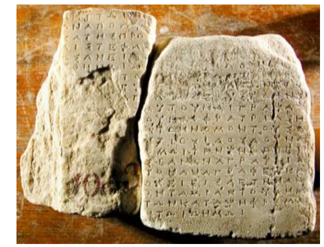
Disk-based logging

- Avoid frequent disk access
- Obtain sequential access pattern

Flash-based logging

- Reduce log flushing time
- Atomic Write (OSDI'08)

Rethink the Logging with NVRAM

 +  + 

Close to DRAM | Byte Addressability | Non-Volatility

Disk-based logging

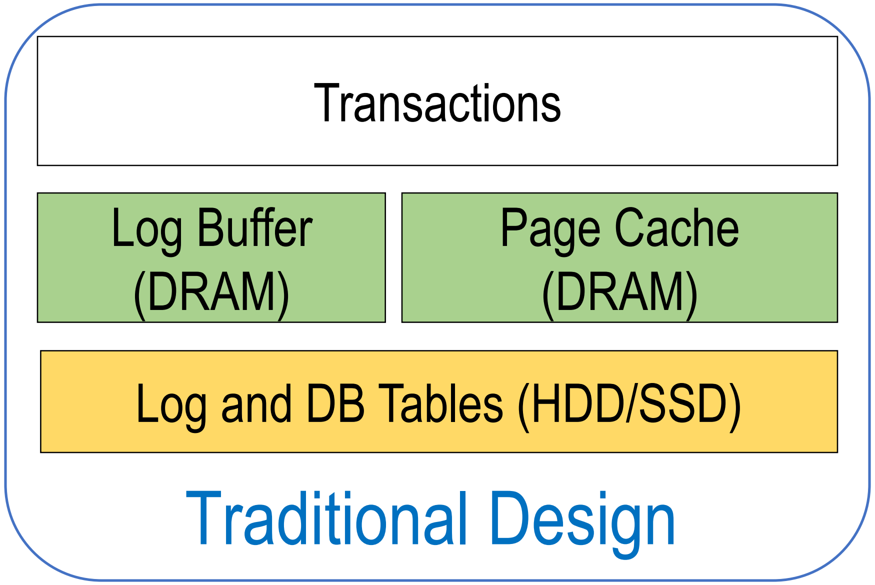
- Avoid frequent disk access
- Obtain sequential access pattern

Flash-based logging

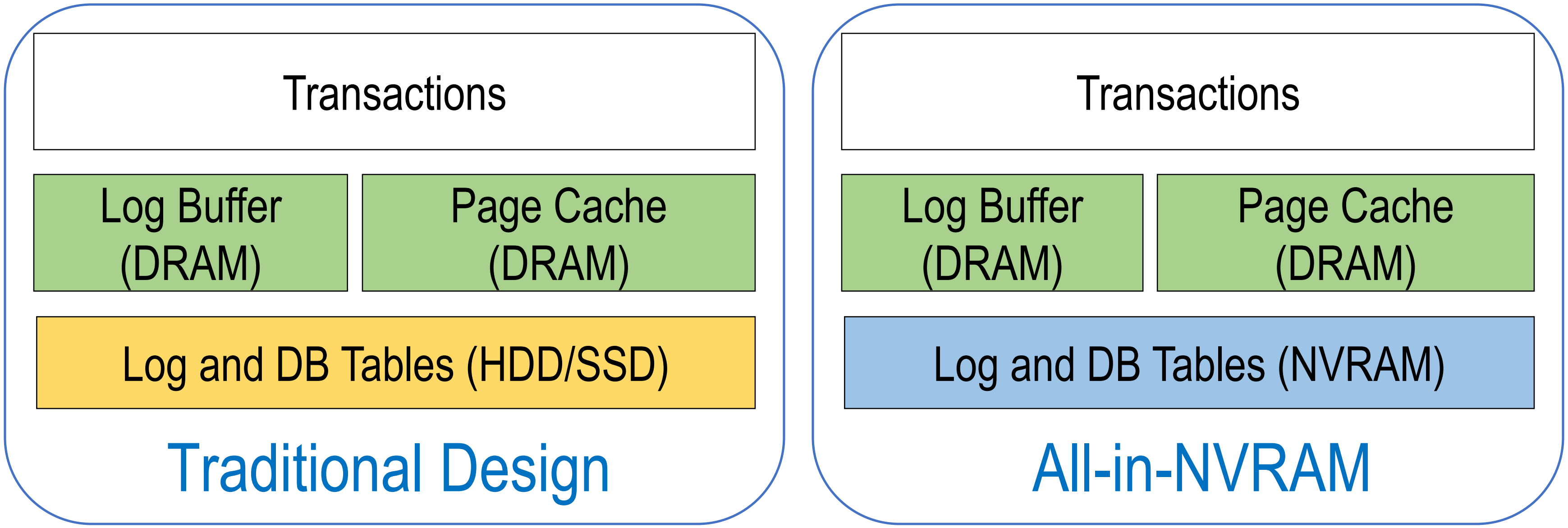
- Reduce log flushing time
- Atomic Write (OSDI'08)

NVRAM-based

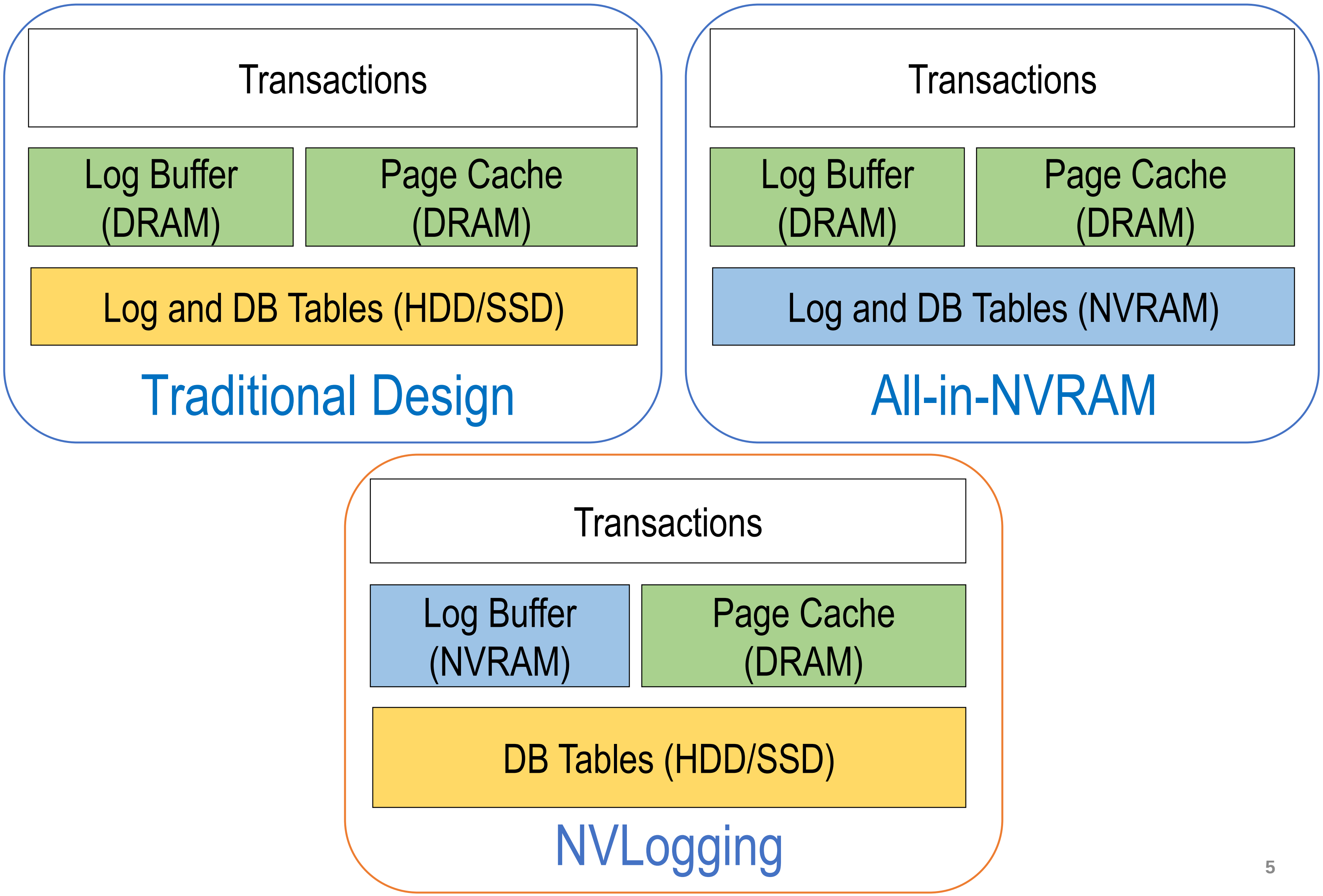
How to Use NVRAM in DB System?



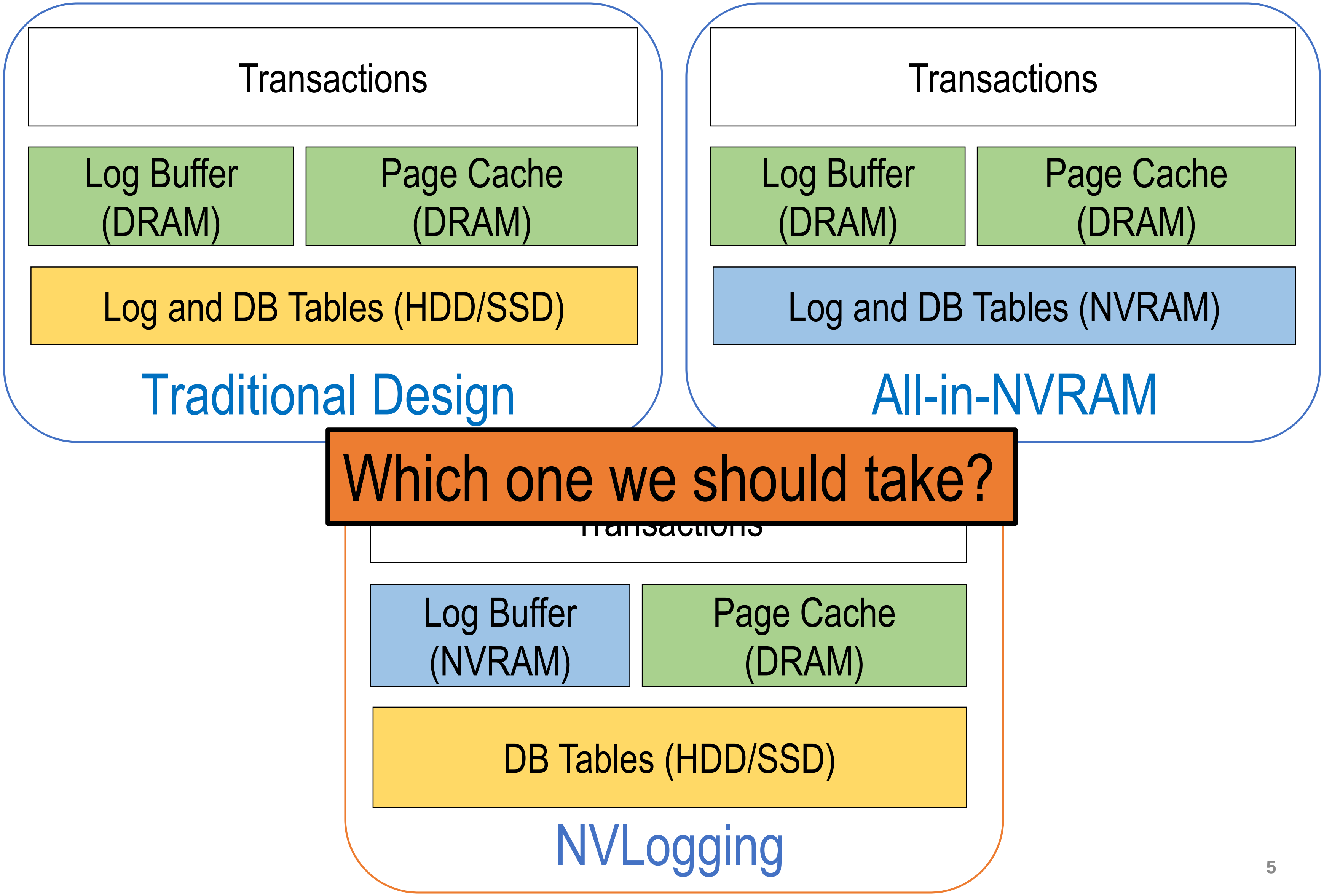
How to Use NVRAM in DB System?



How to Use NVRAM in DB System?

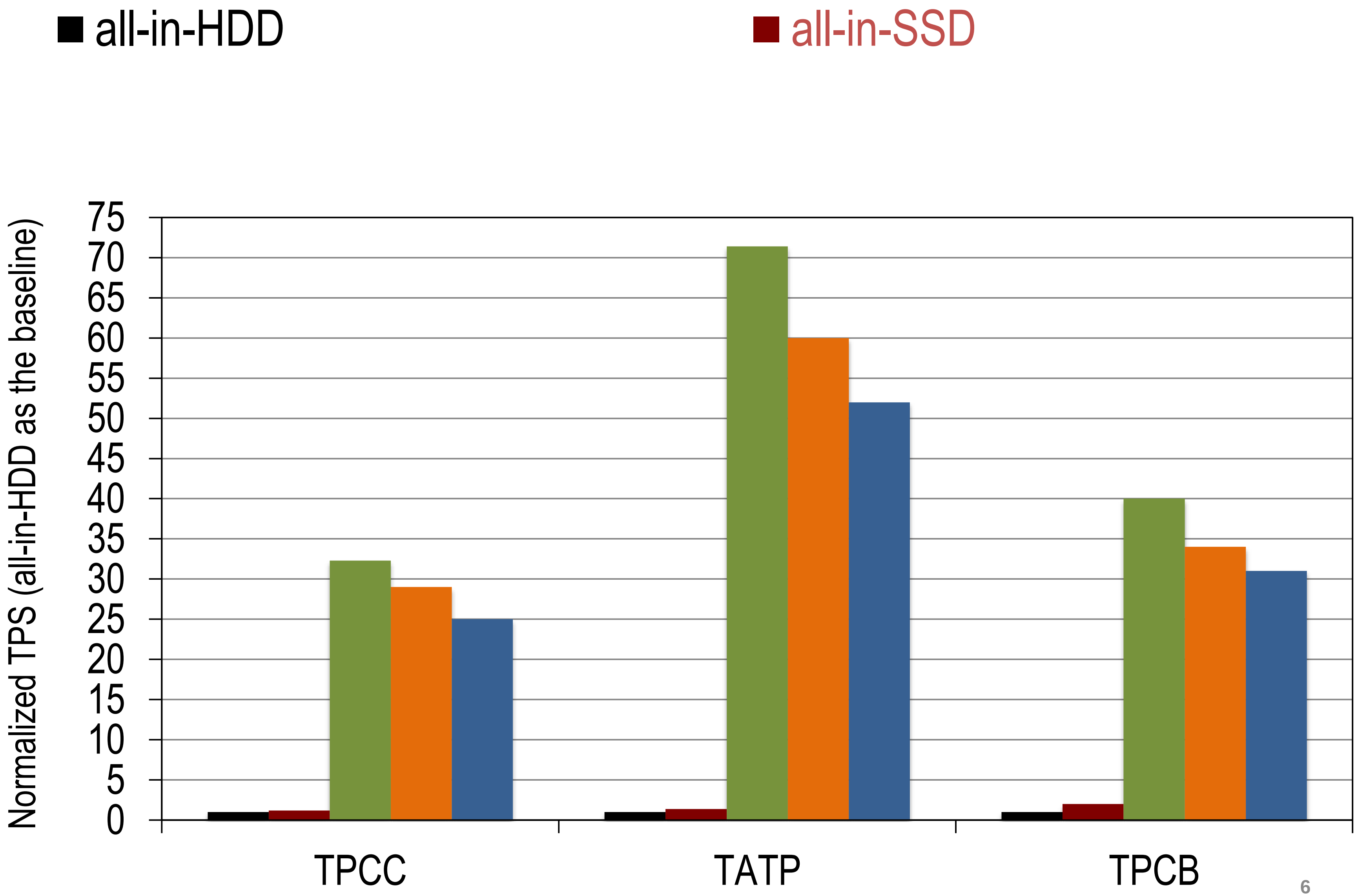


How to Use NVRAM in DB System?

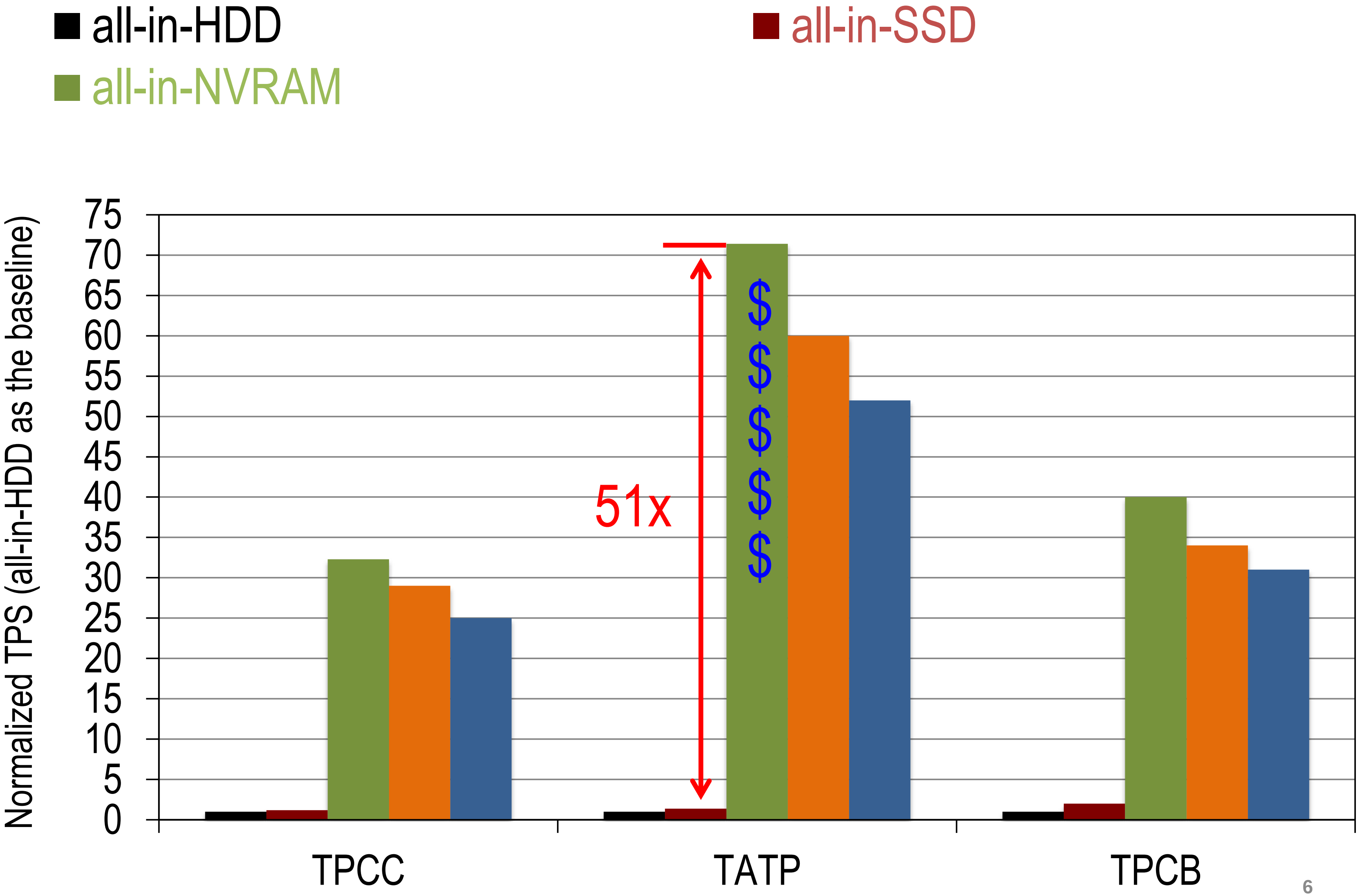


Using NVRAM in Cost-Effective Way

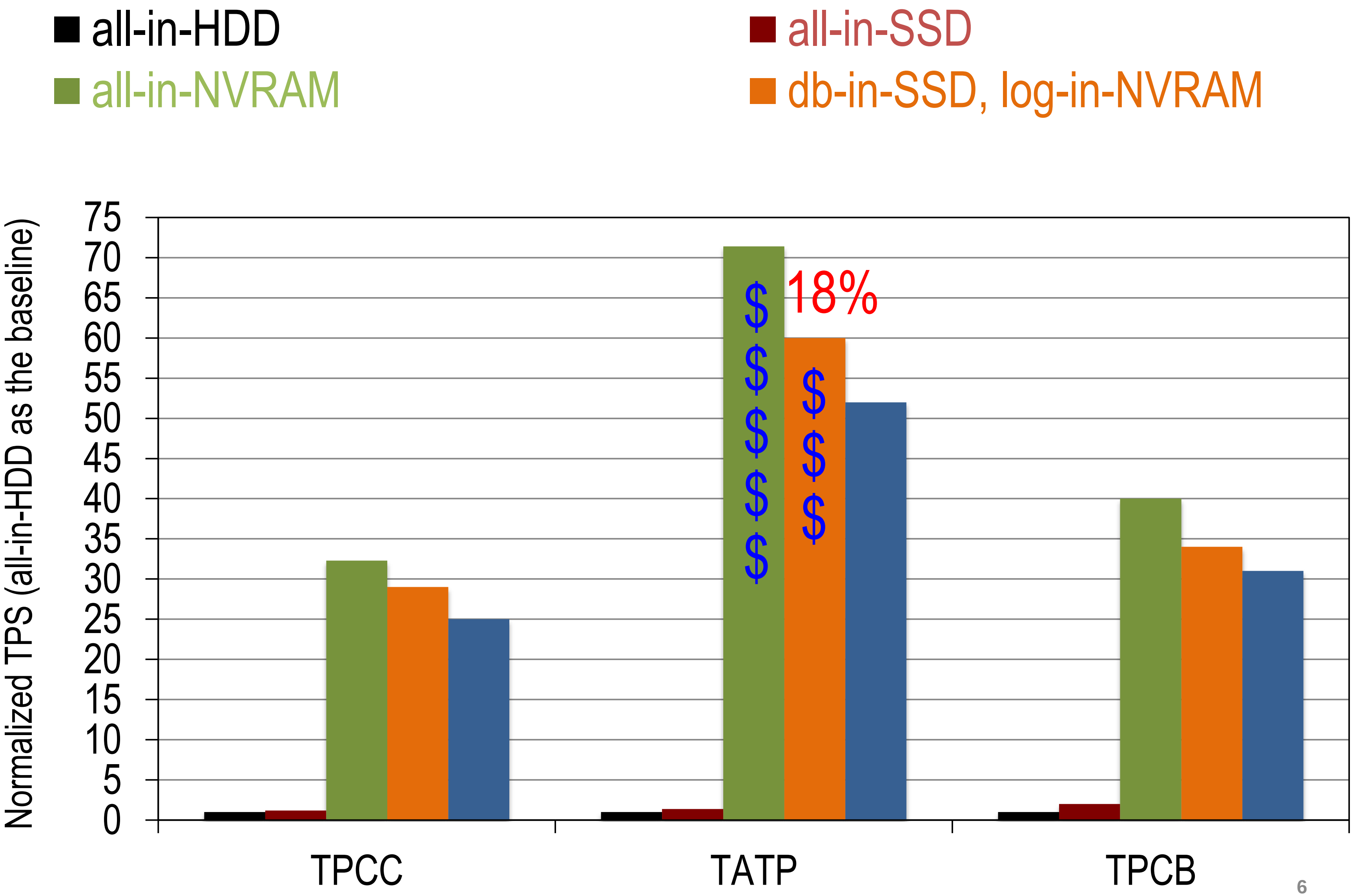
Using NVRAM in Cost-Effective Way



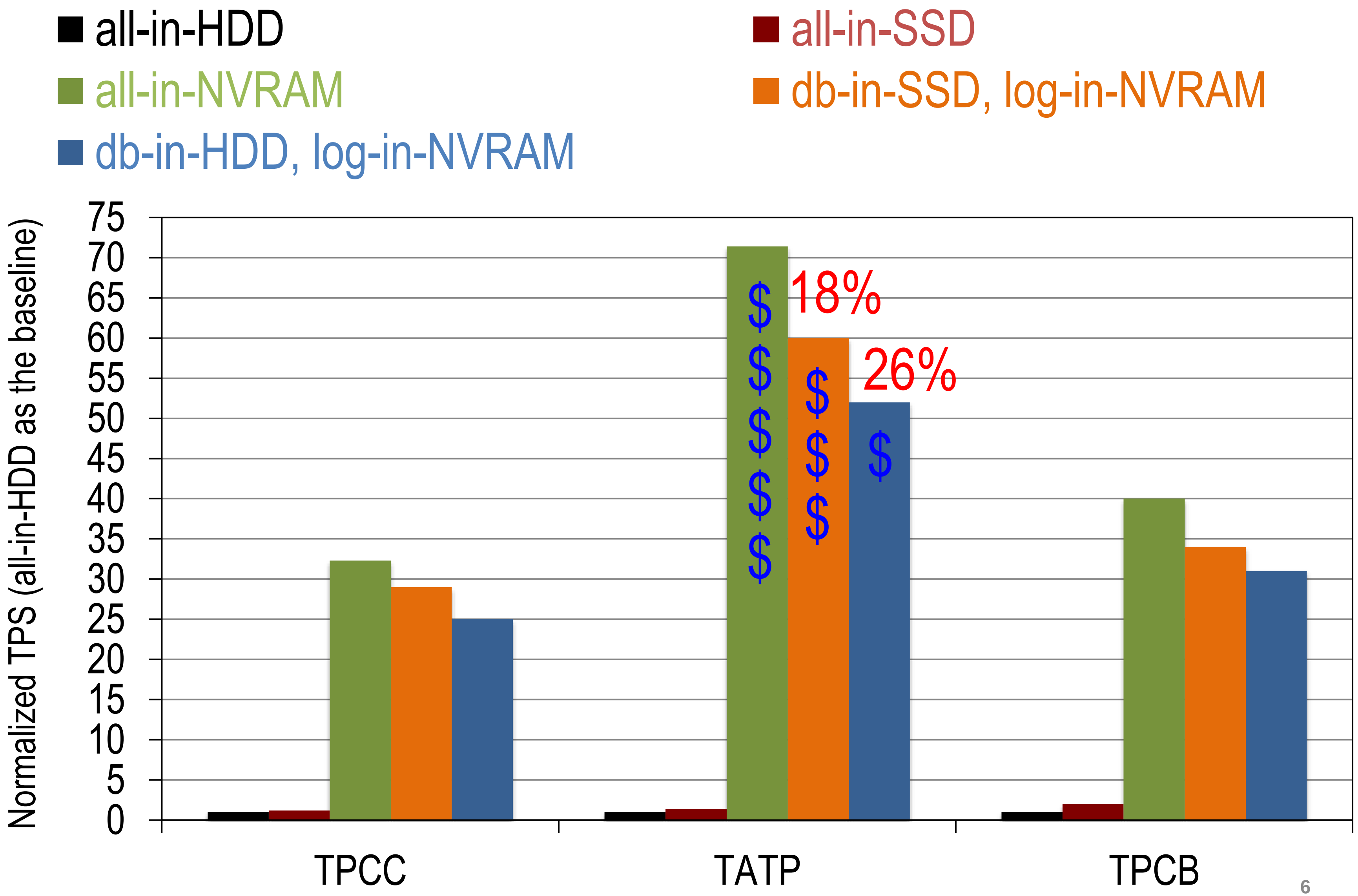
Using NVRAM in Cost-Effective Way



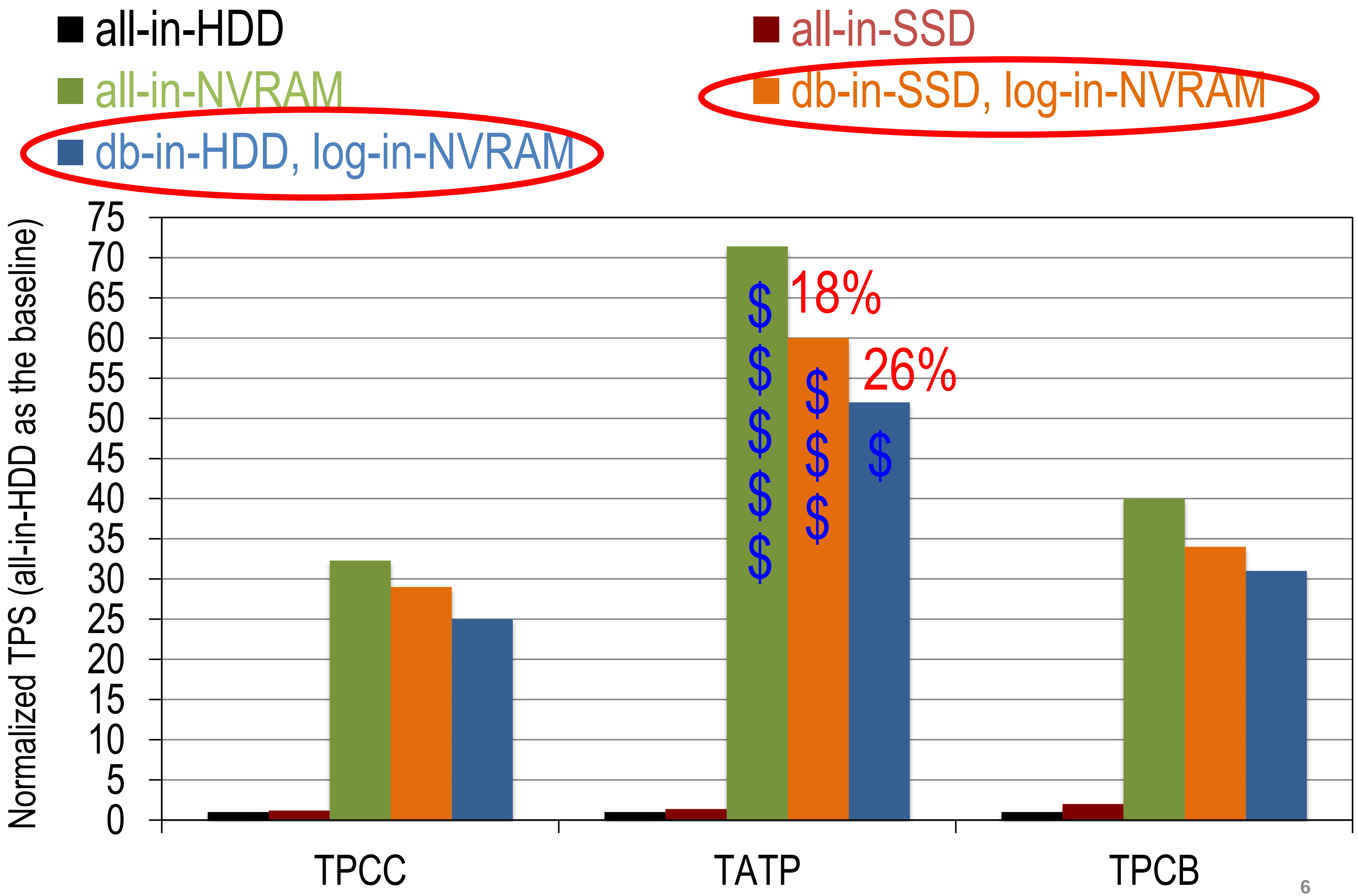
Using NVRAM in Cost-Effective Way



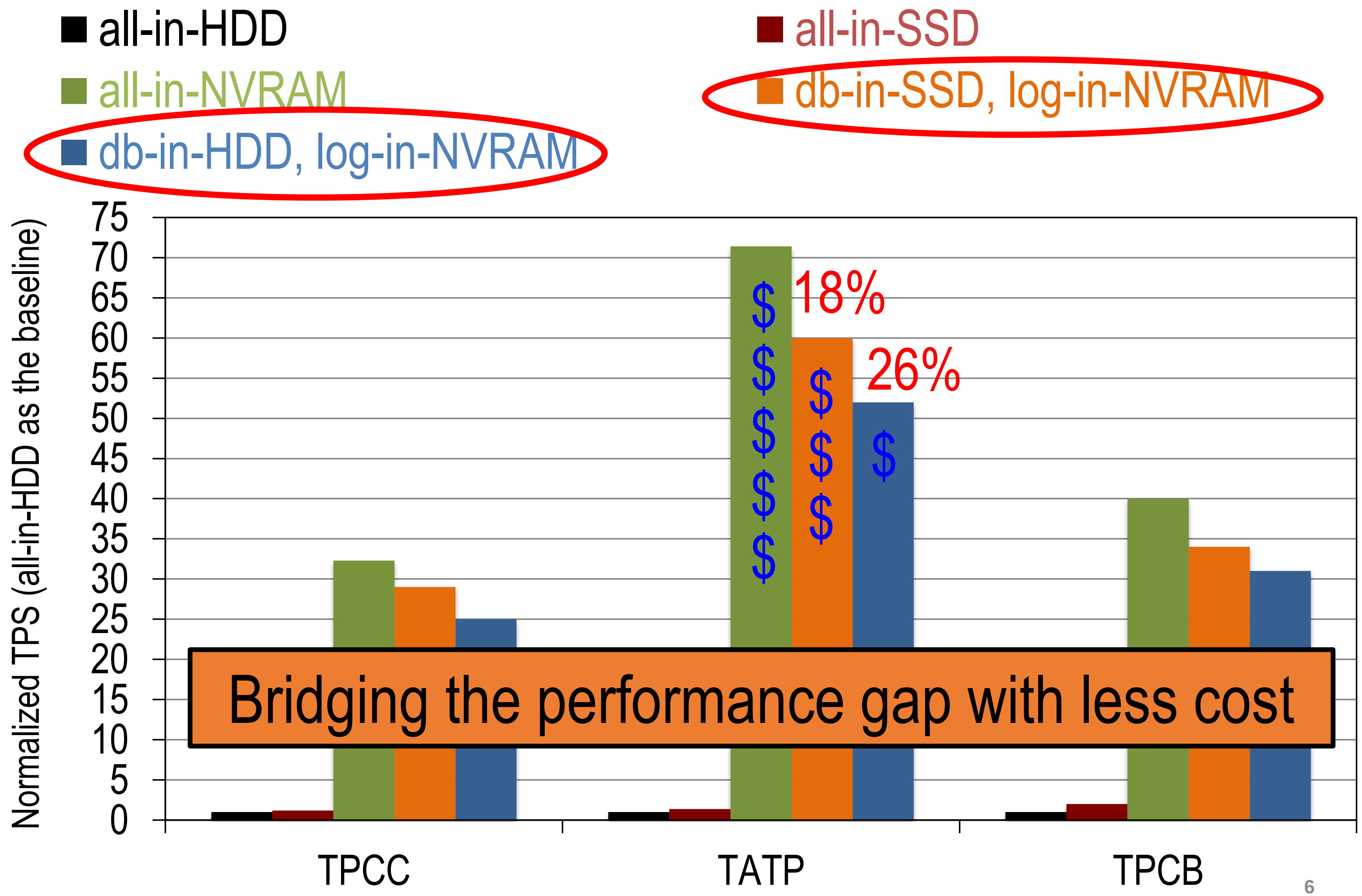
Using NVRAM in Cost-Effective Way



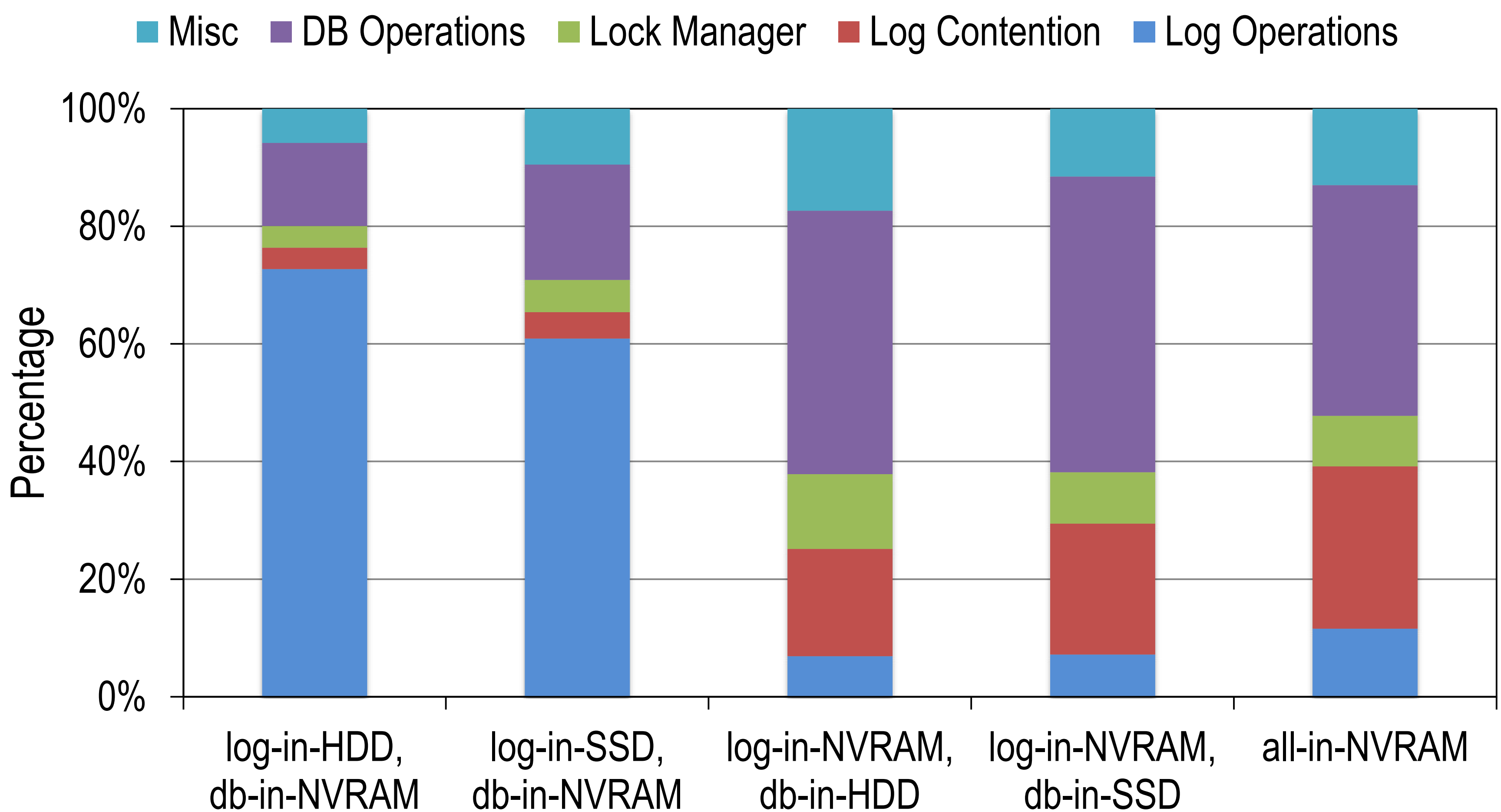
Using NVRAM in Cost-Effective Way



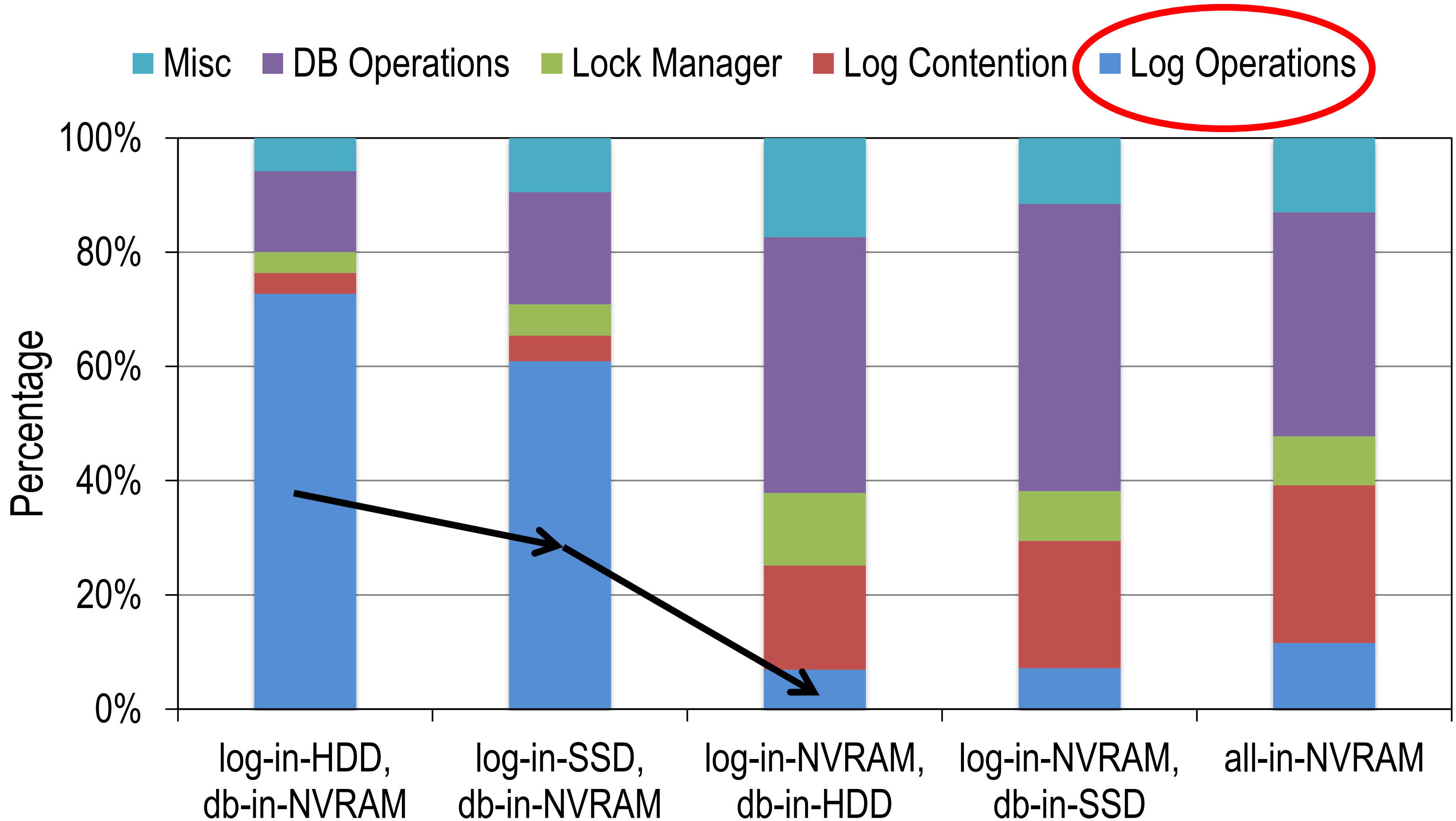
Using NVRAM in Cost-Effective Way



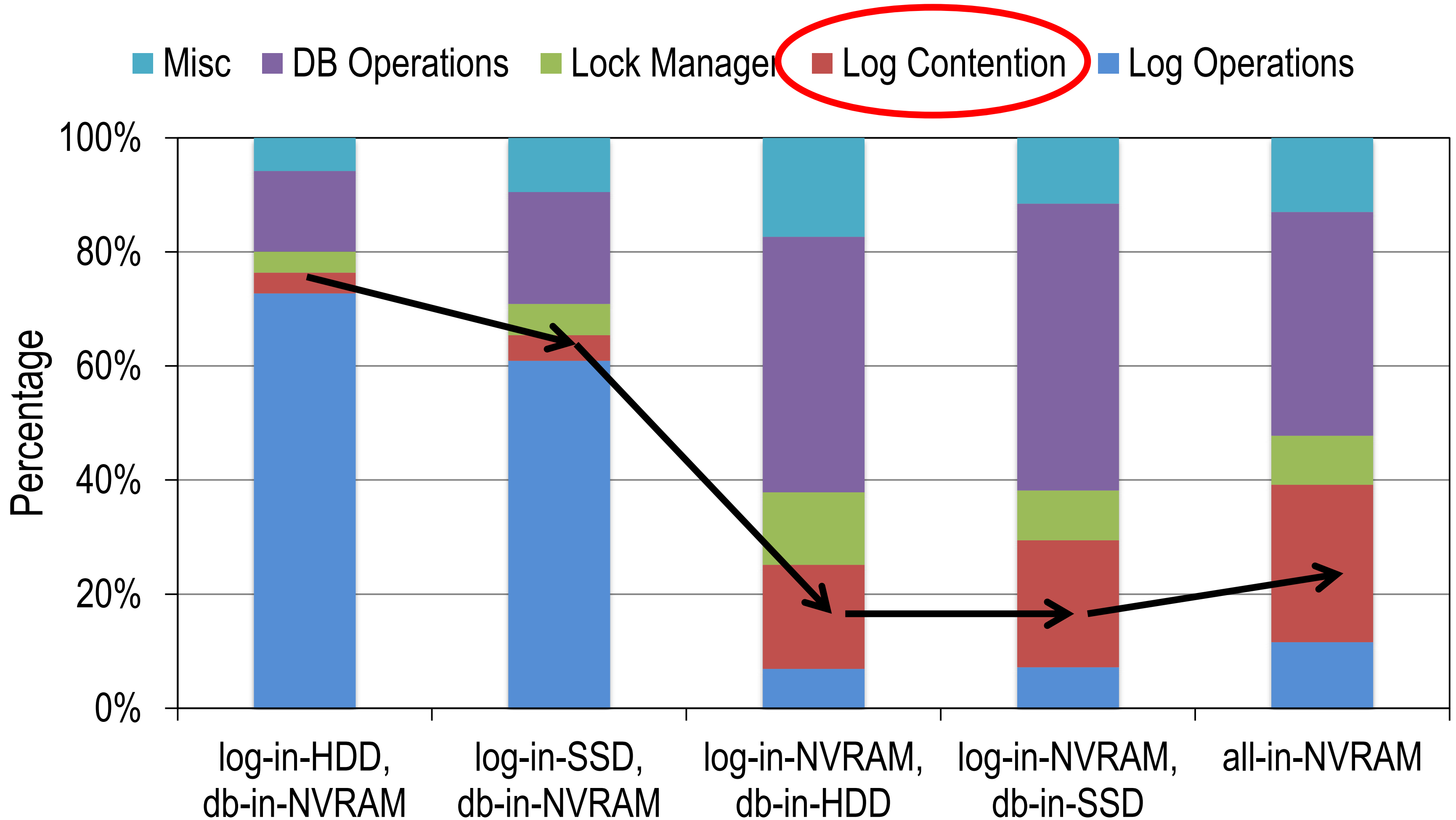
Bottlenecks Shifted from I/O to Software



Bottlenecks Shifted from I/O to Software

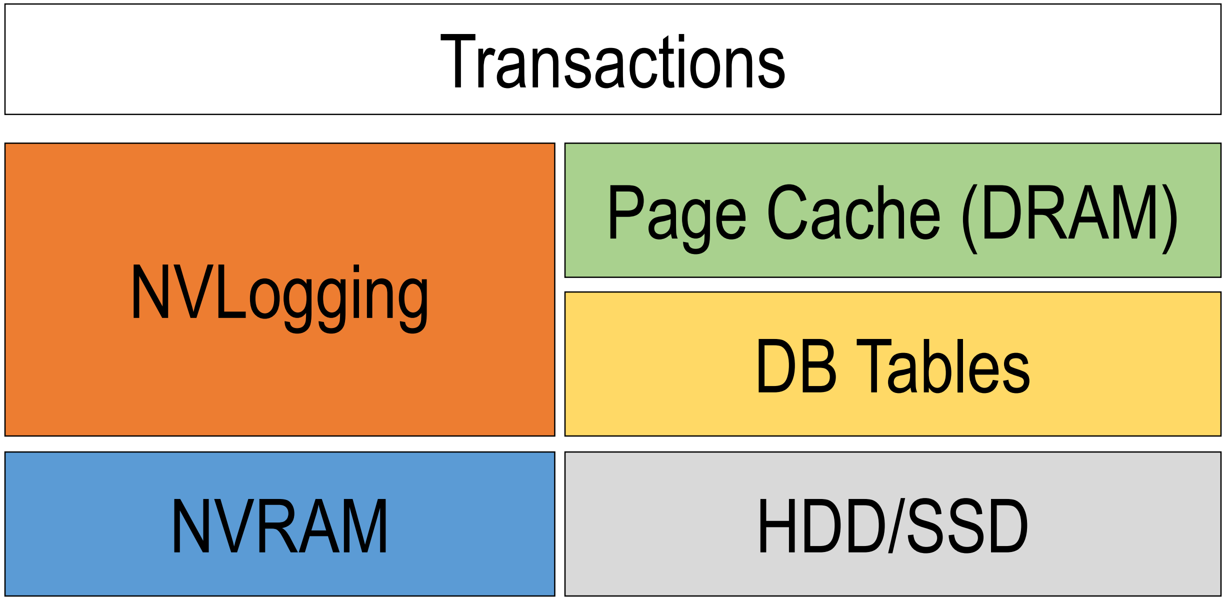


Bottlenecks Shifted from I/O to Software



Software-induced overhead becomes the new bottleneck !

NVLogging: Redesign the Logging with NVRAM

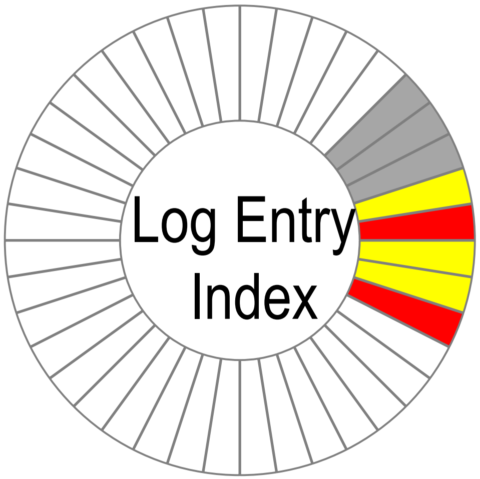


1. Use NVRAM in Cost-Effective Way
2. Avoid the Software Overhead
3. Minimal Code Modifications

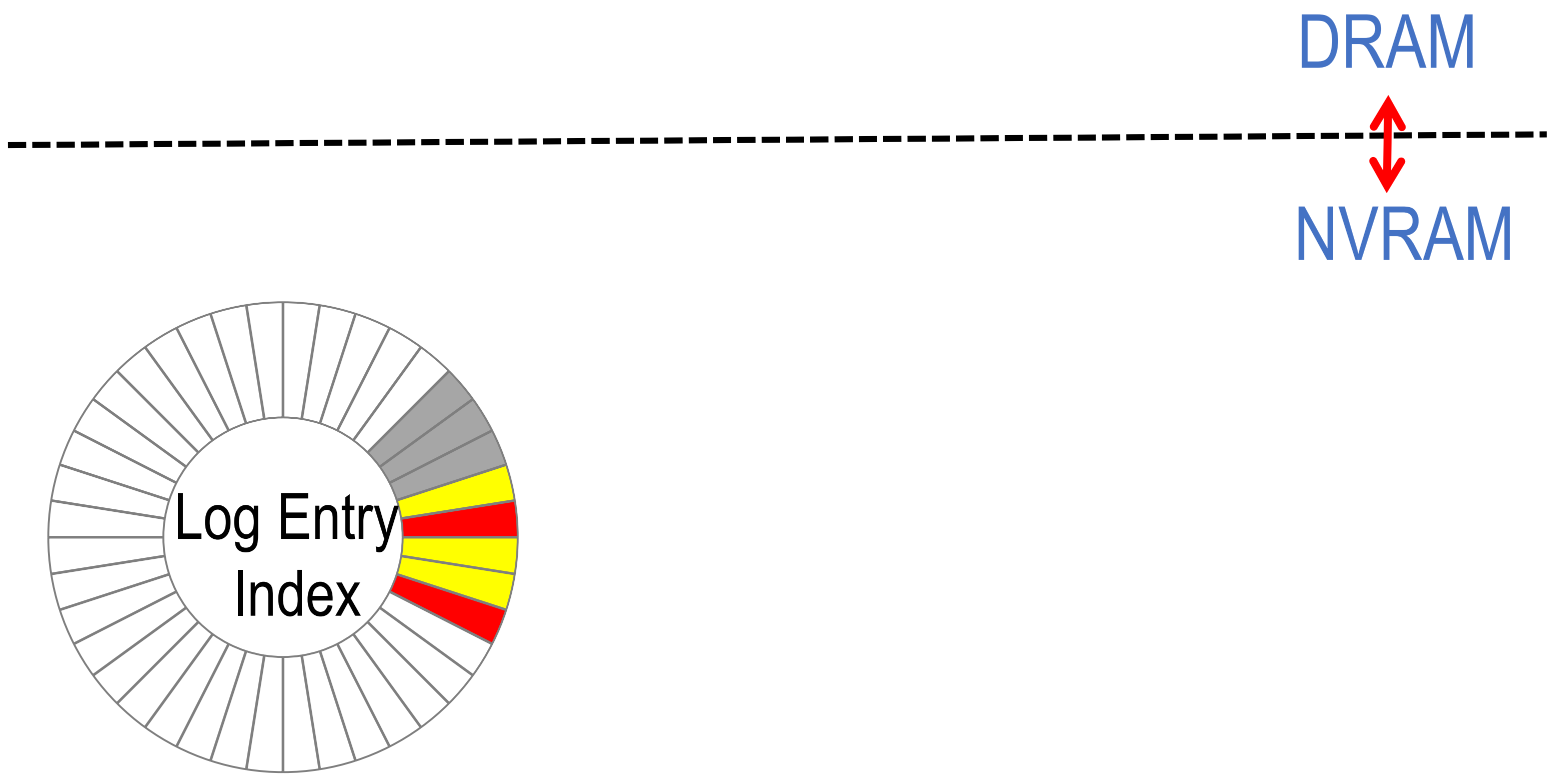
Log Management in NVLogging

DRAM

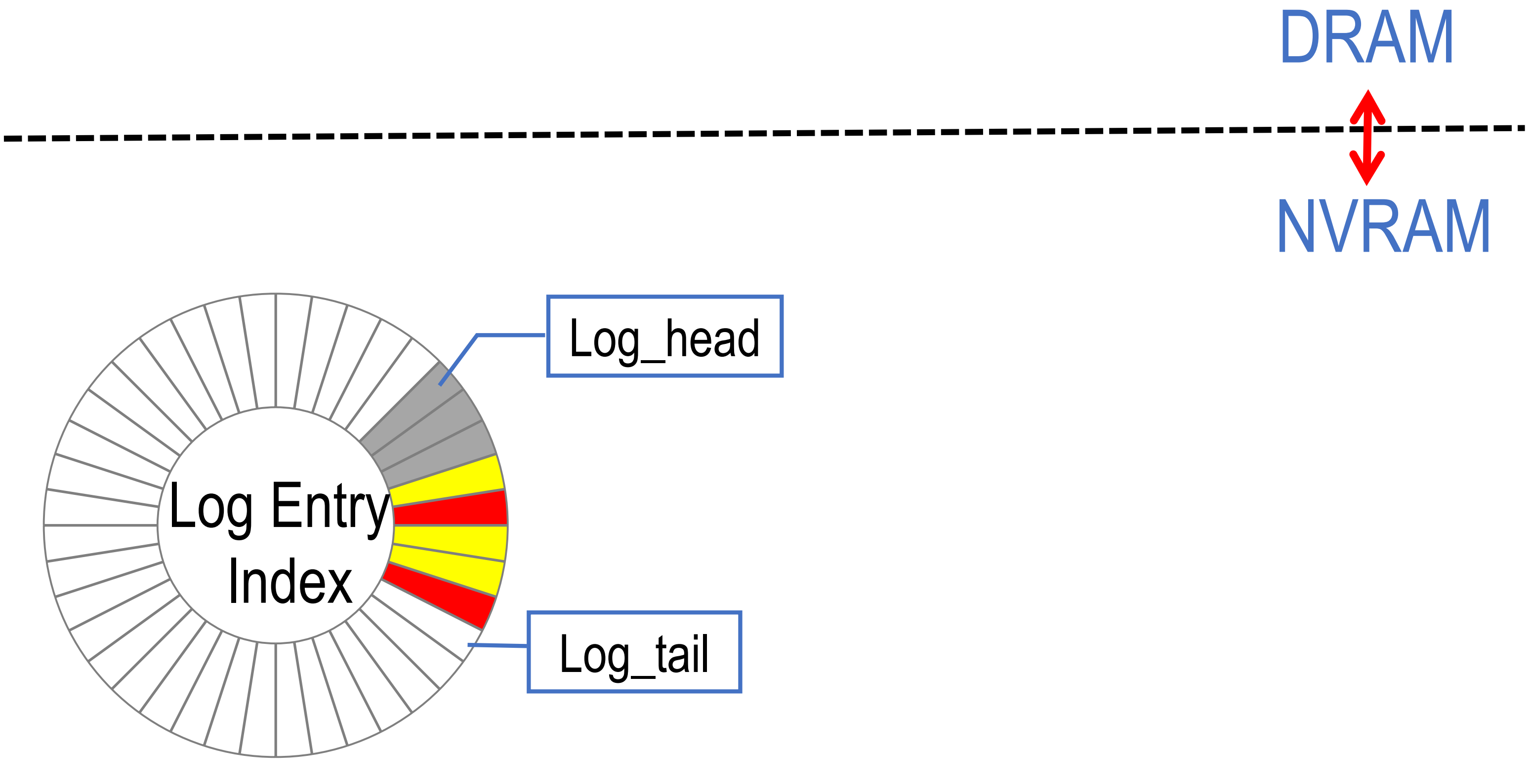
NVRAM



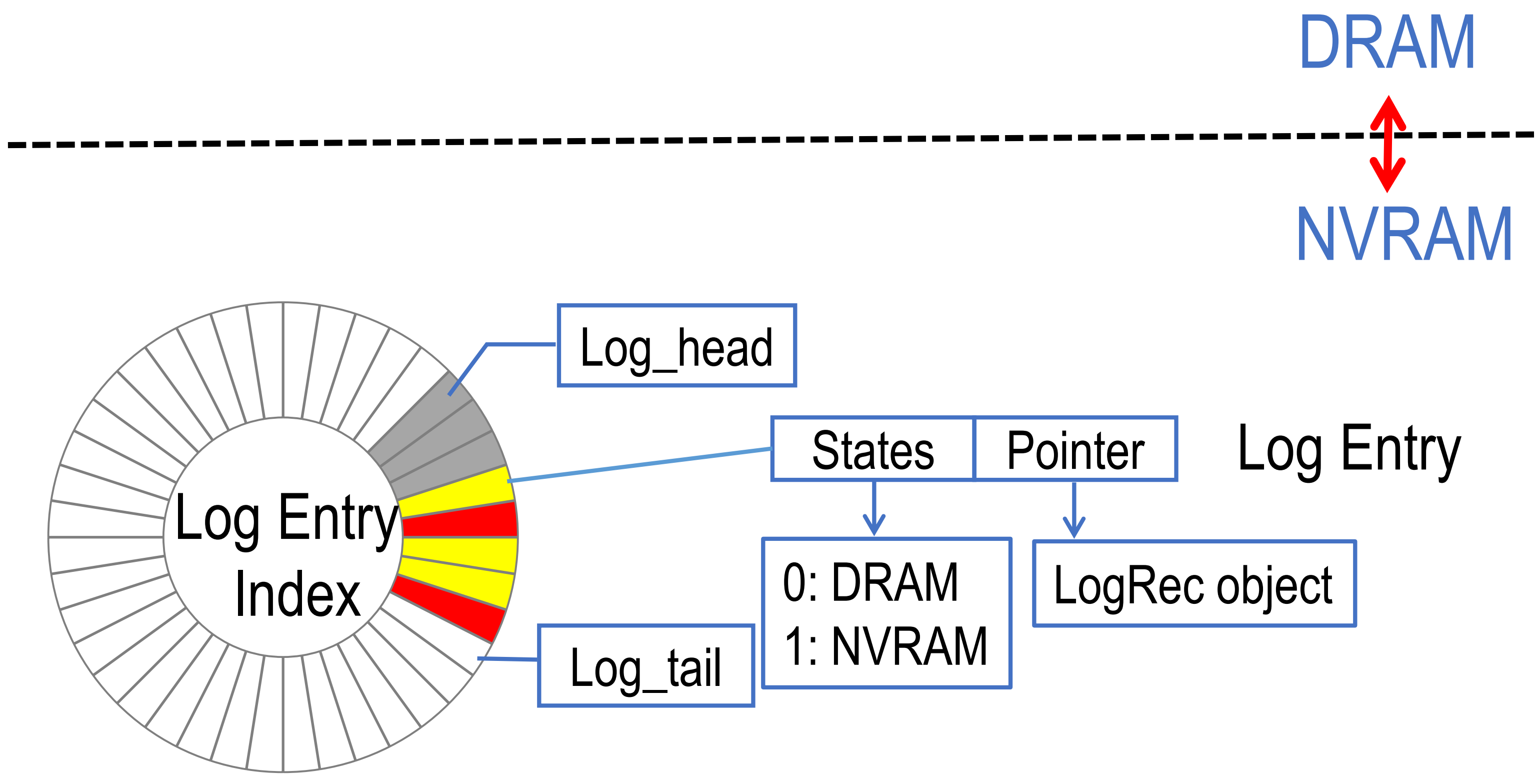
Log Management in NVLogging



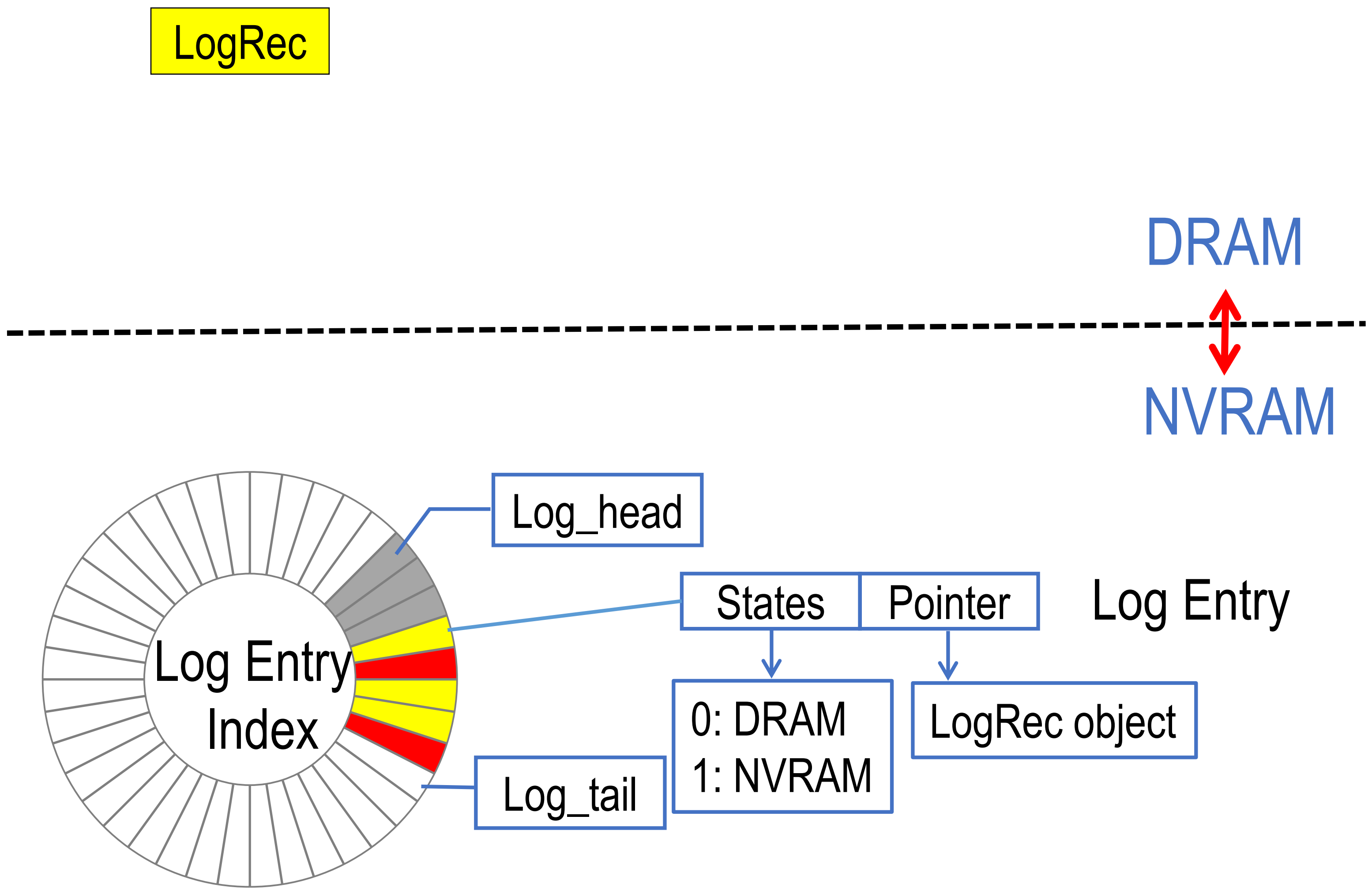
Log Management in NVLogging



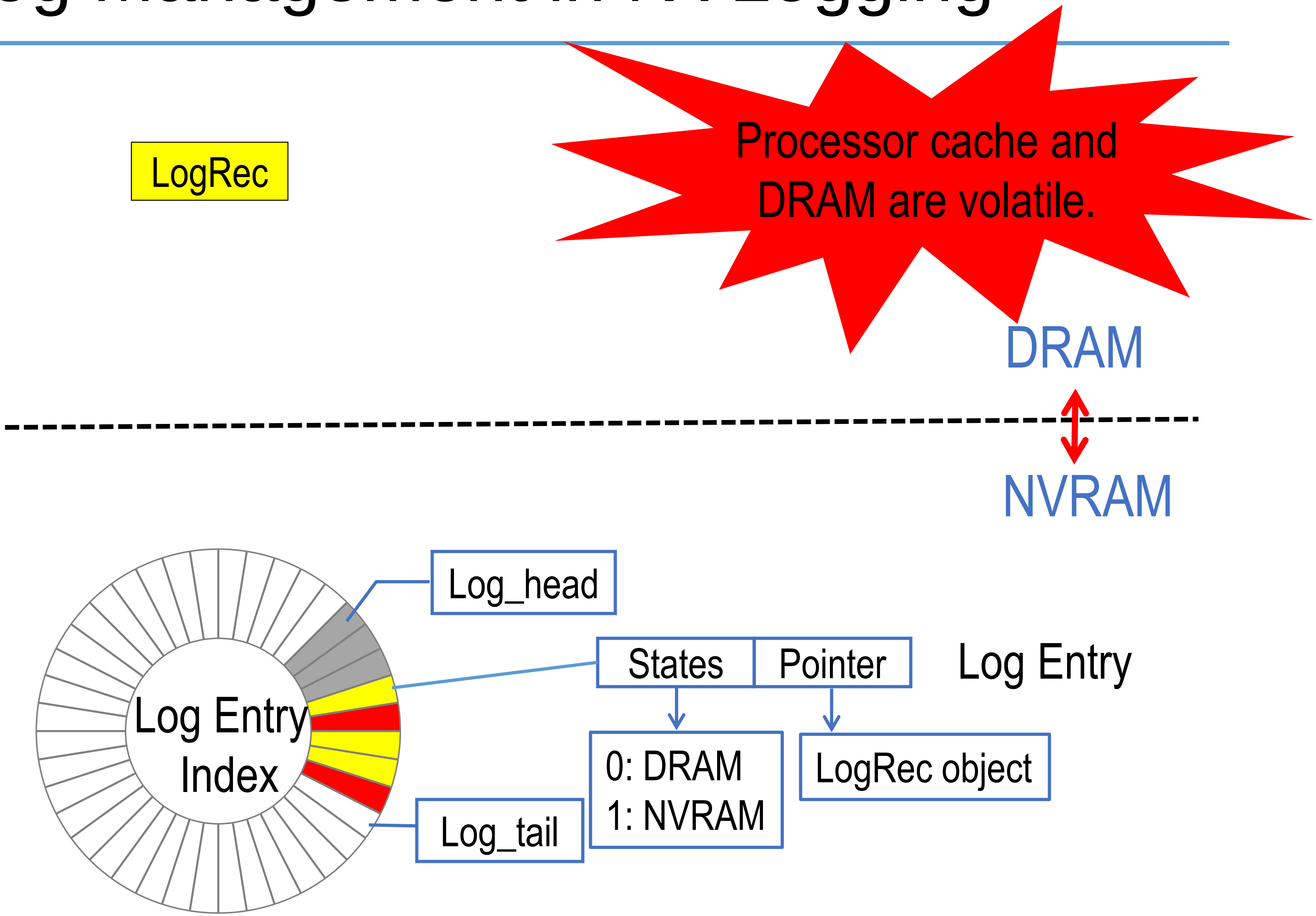
Log Management in NVLogging



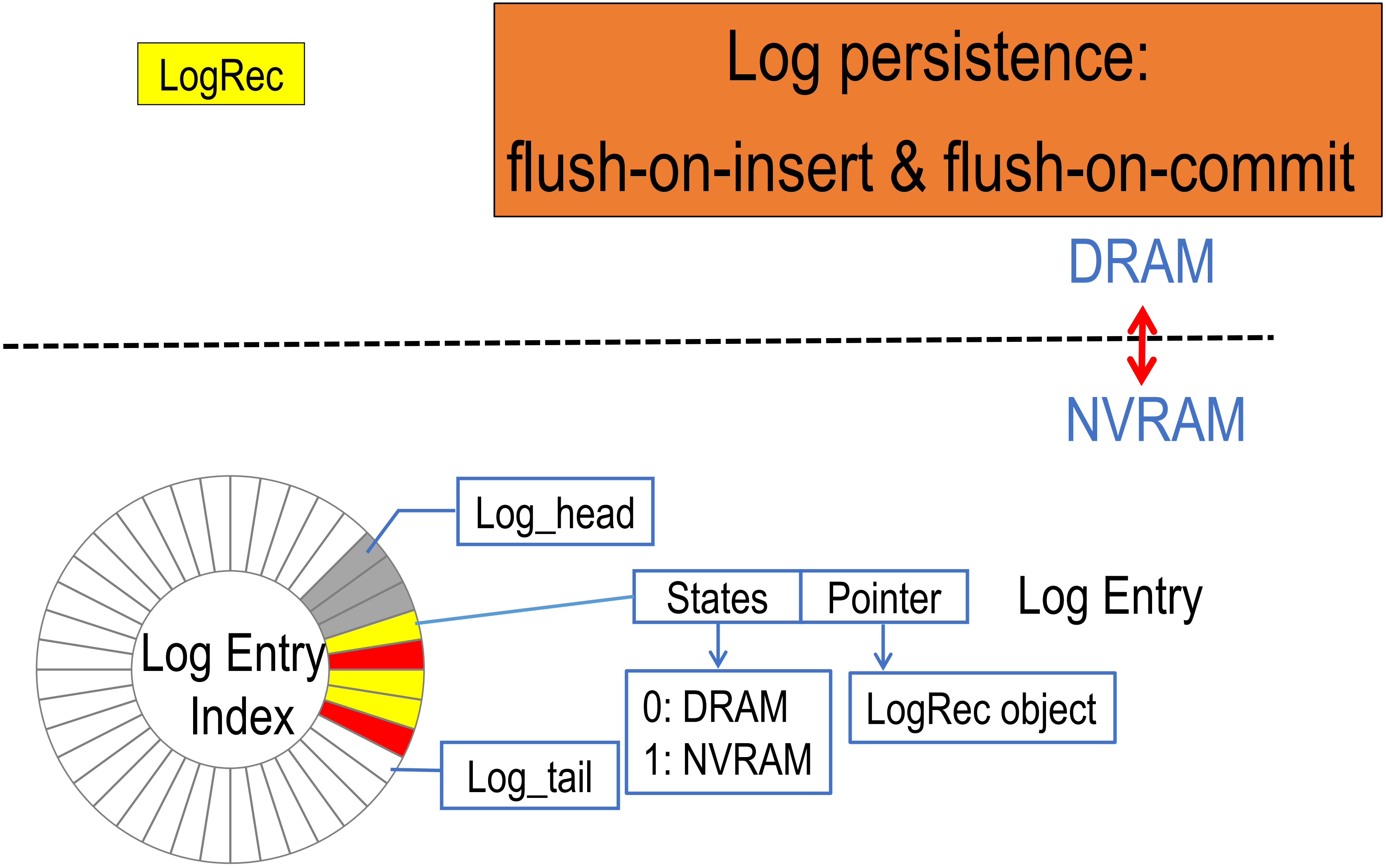
Log Management in NVLogging



Log Management in NVLogging



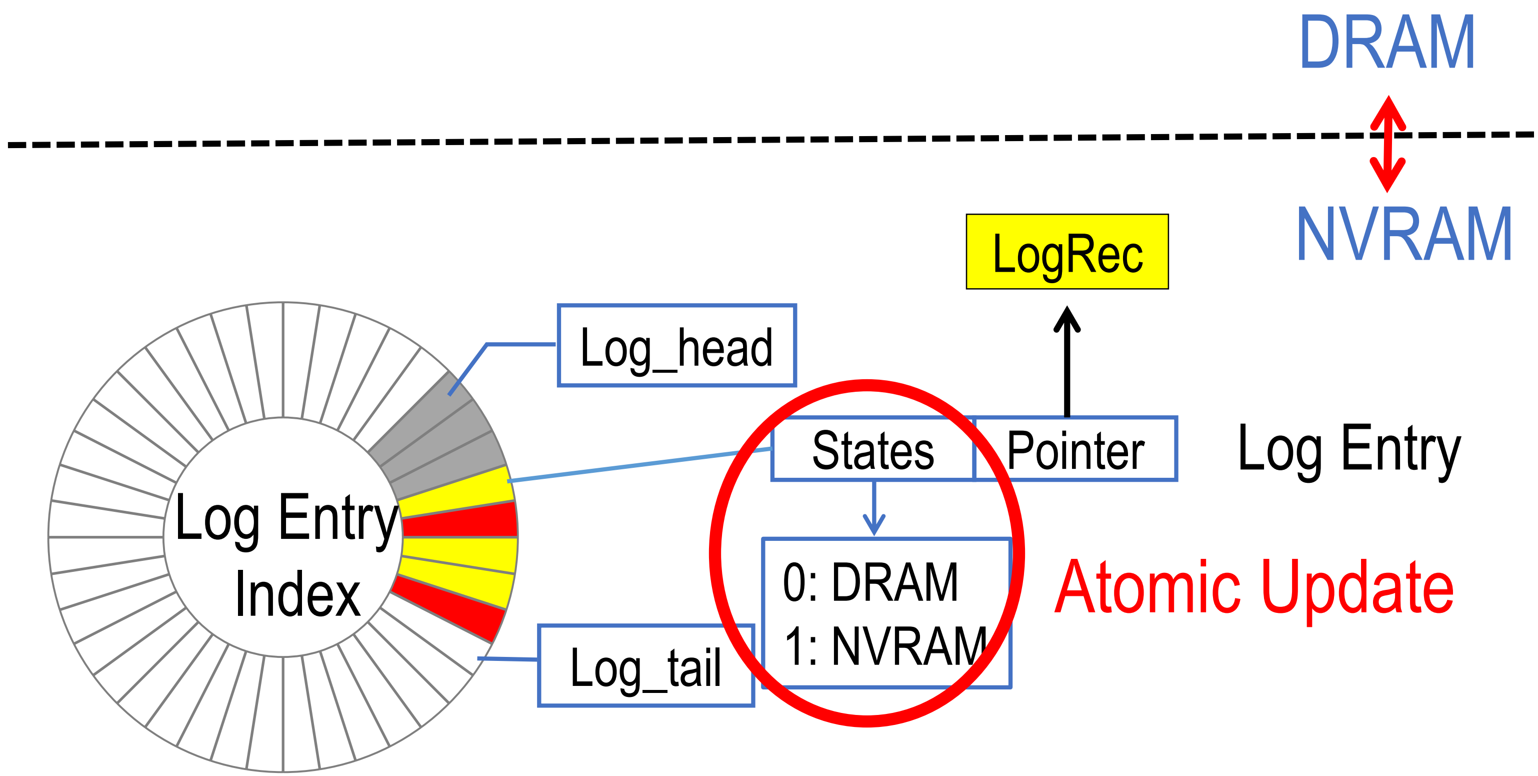
Log Management in NVLogging



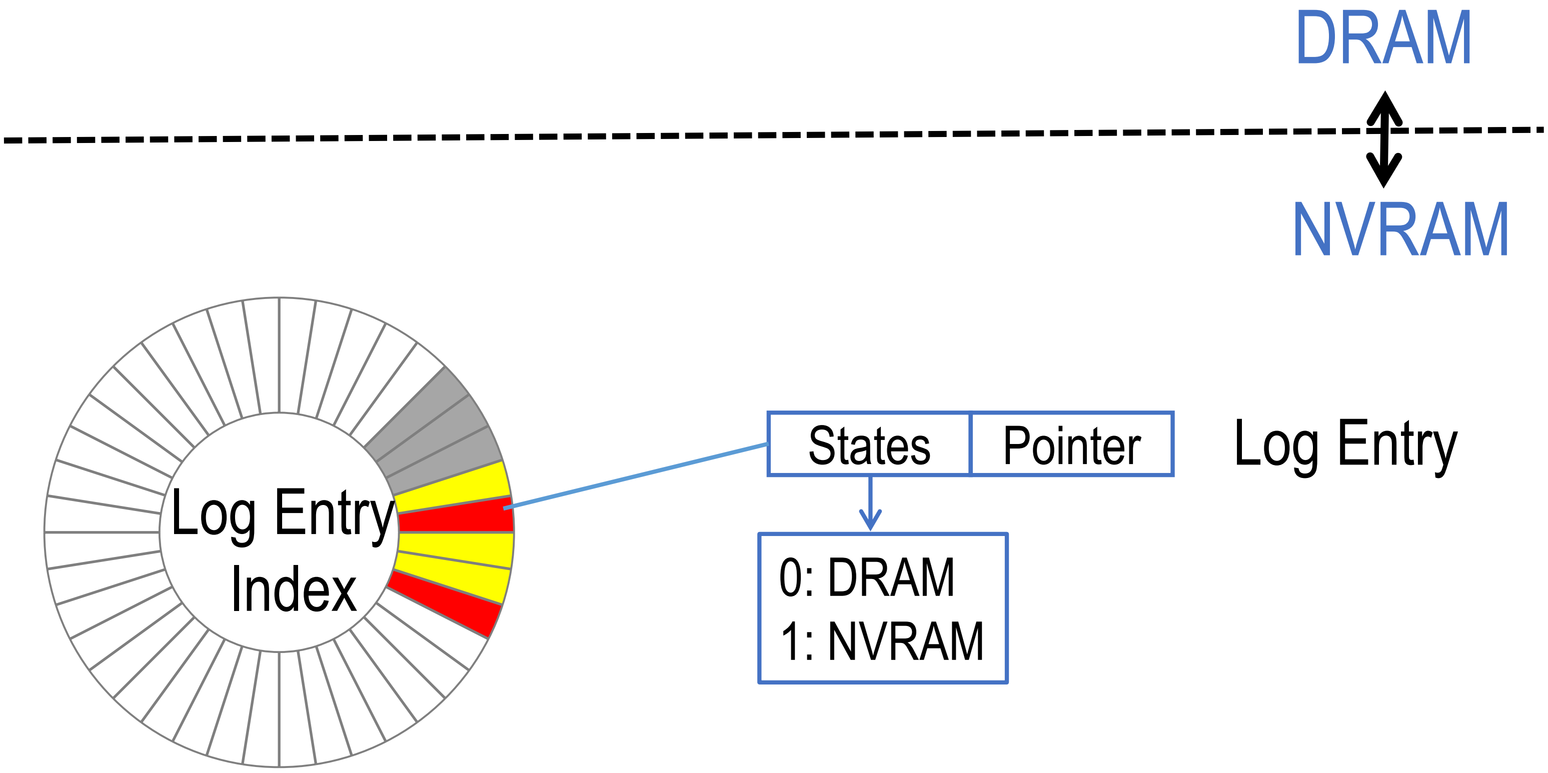
Log Management in NVLogging

Log persistence:

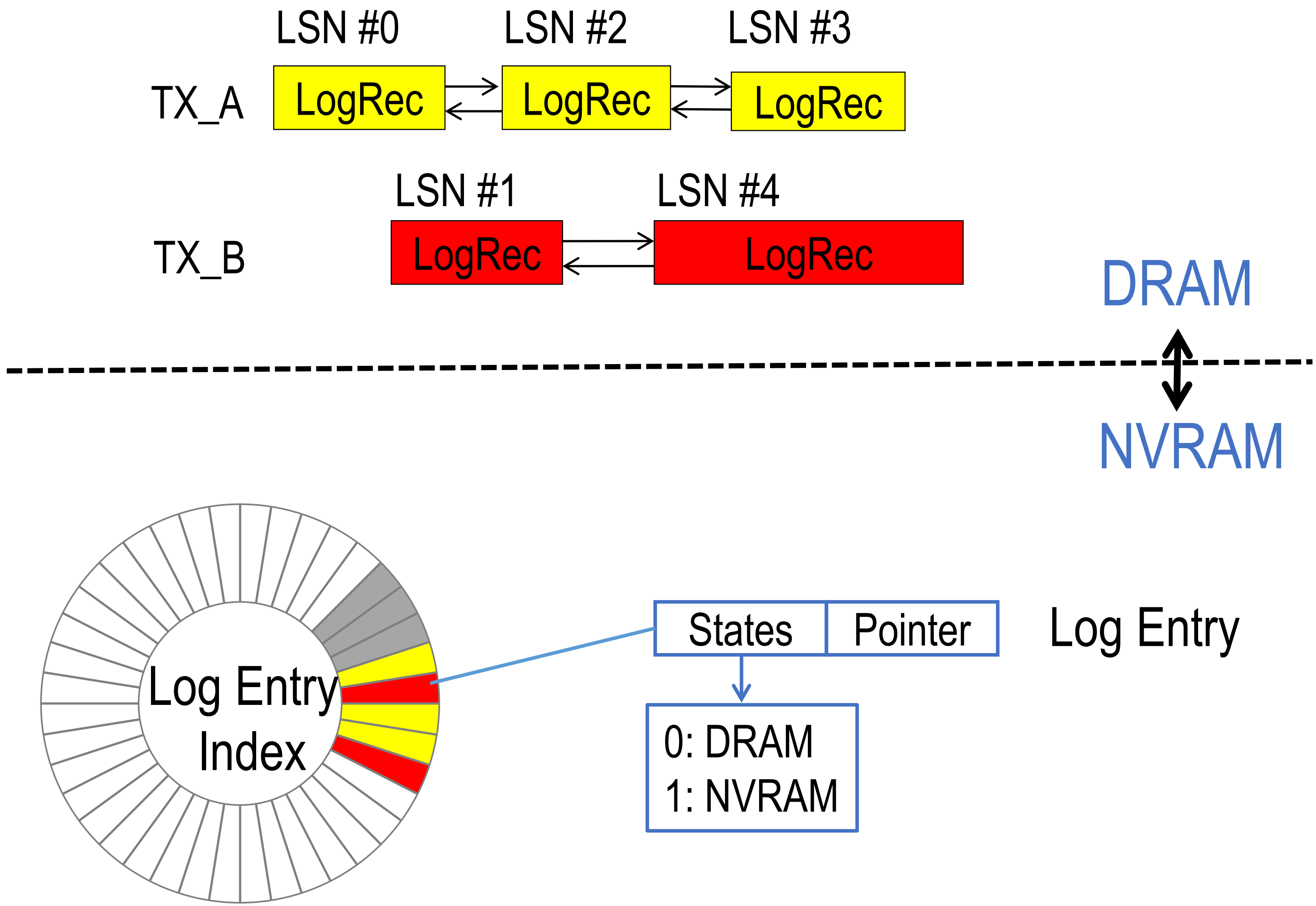
flush-on-insert & flush-on-commit



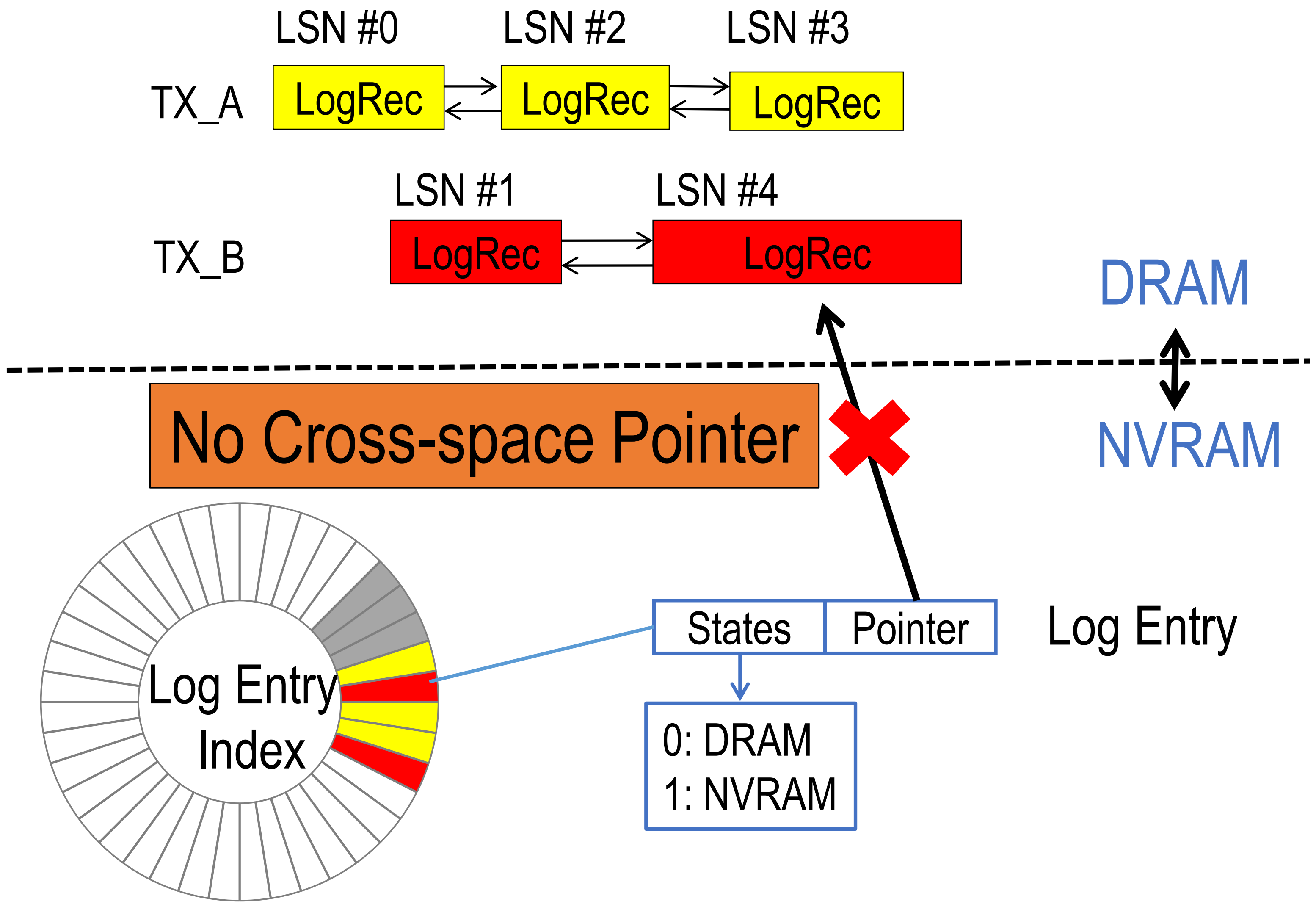
Avoid Log Contention with Per-Transaction Logging



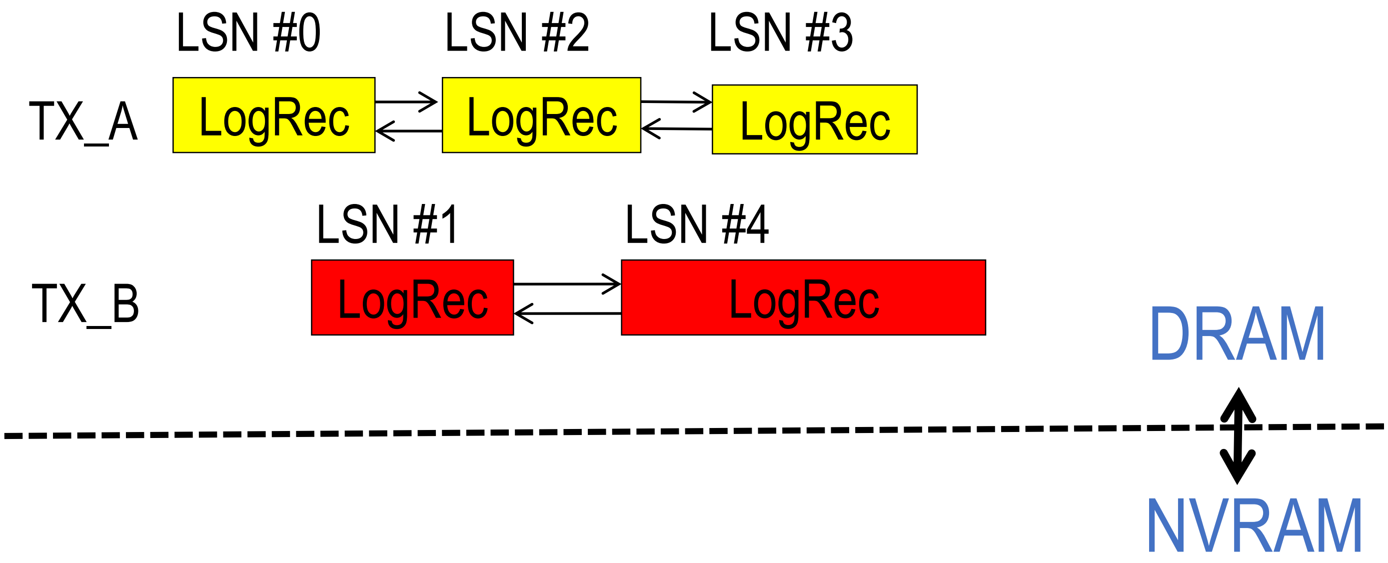
Avoid Log Contention with Per-Transaction Logging

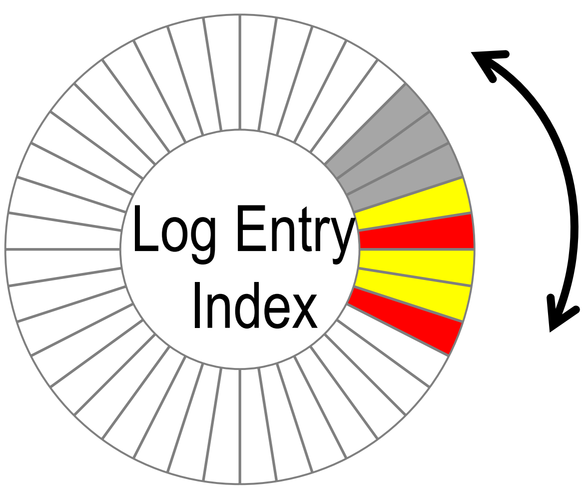


Avoid Log Contention with Per-Transaction Logging



Avoid Log Contention with Per-Transaction Logging

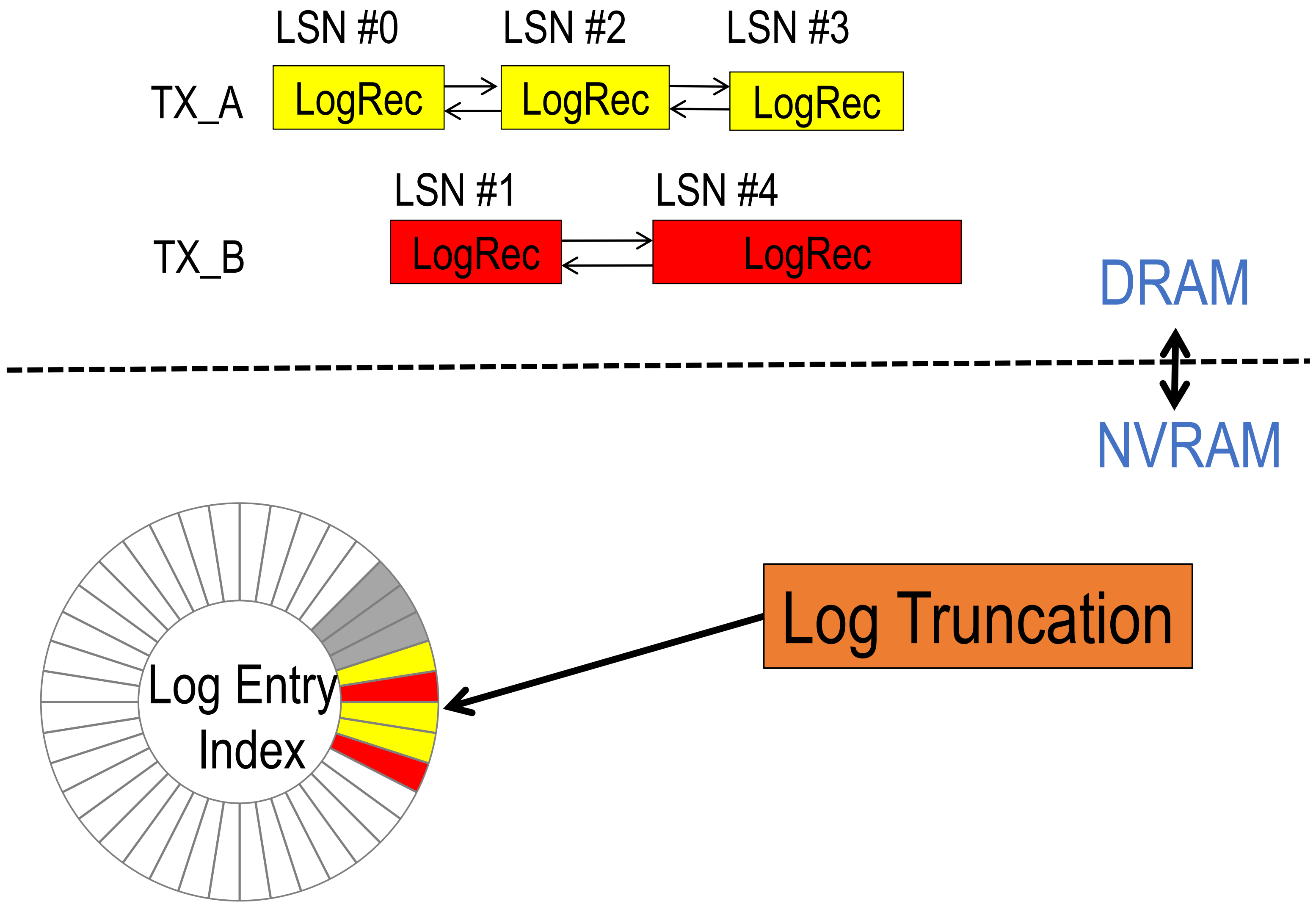




Recovery

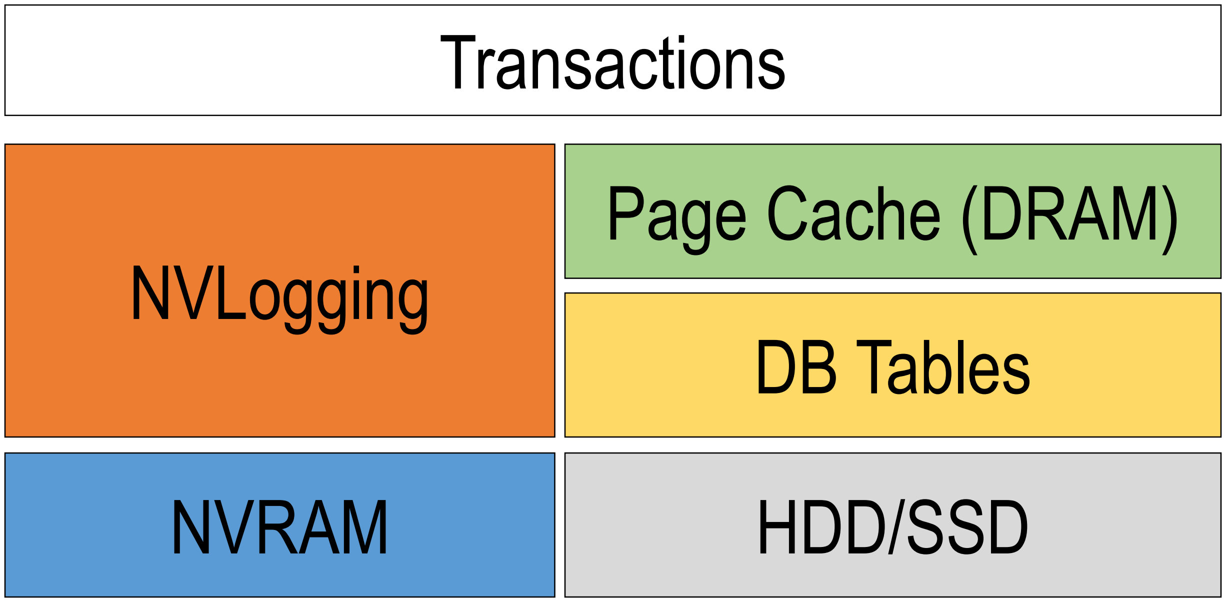
Using back pointer to access log objects

Avoid Log Contention with Per-Transaction Logging



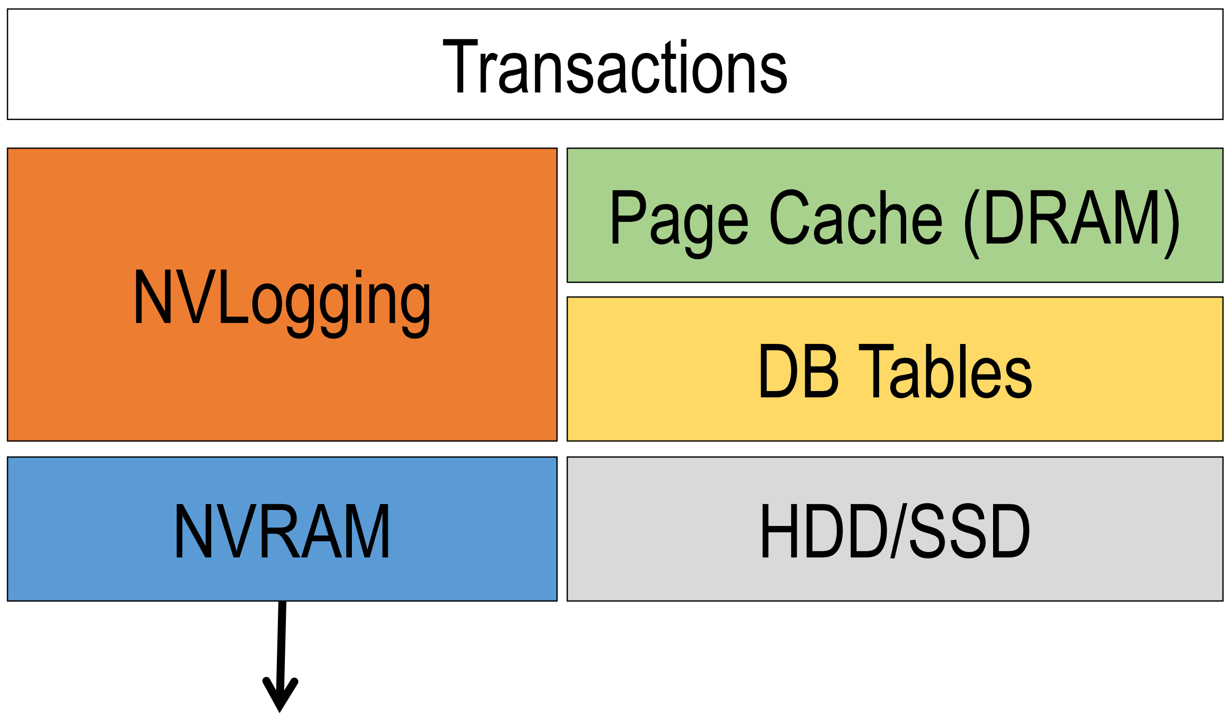
Implementation and Experimental Setup

Shore-MT + Shore-Kits Benchmark



Implementation and Experimental Setup

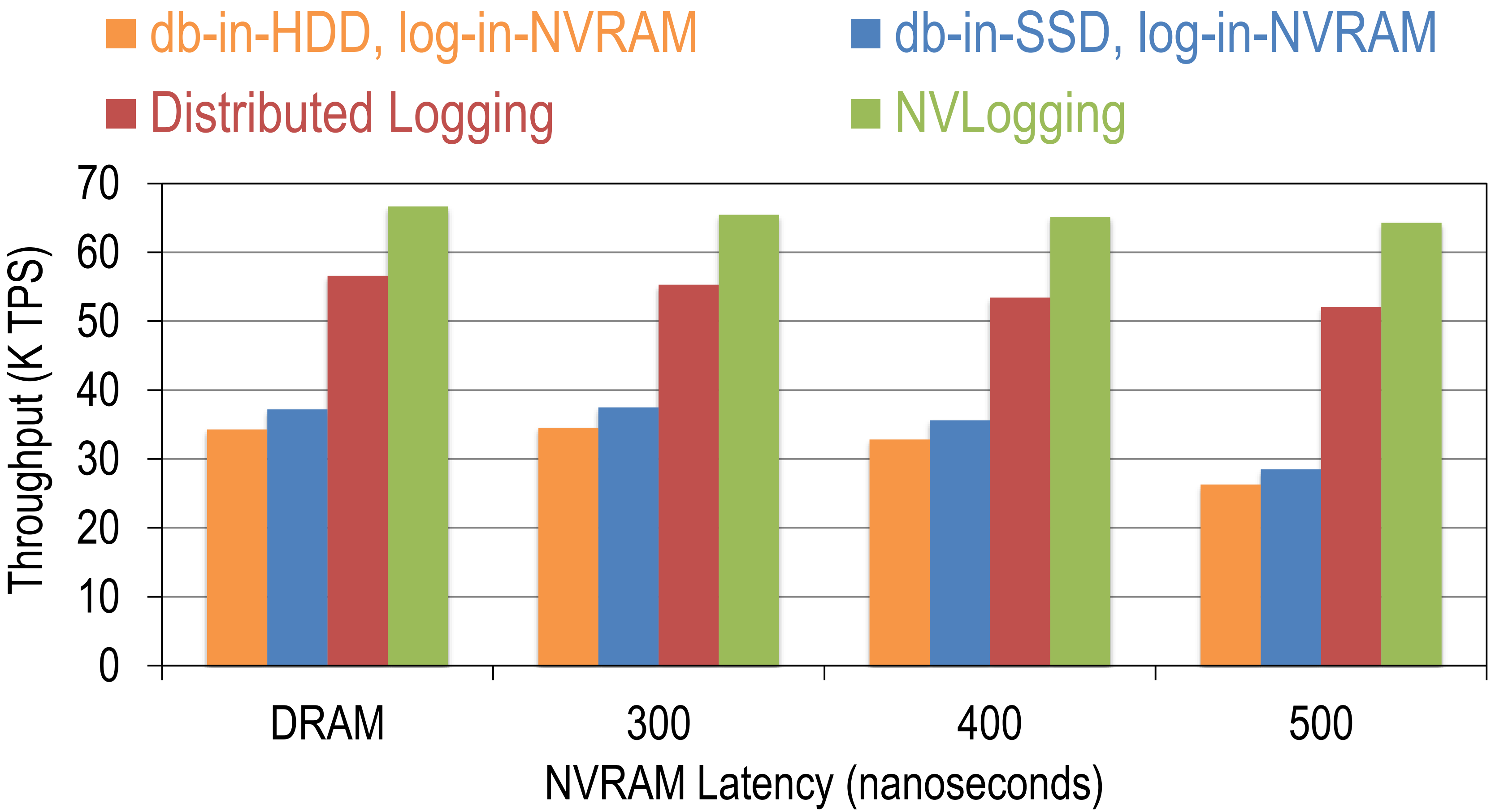
Shore-MT + Shore-Kits Benchmark



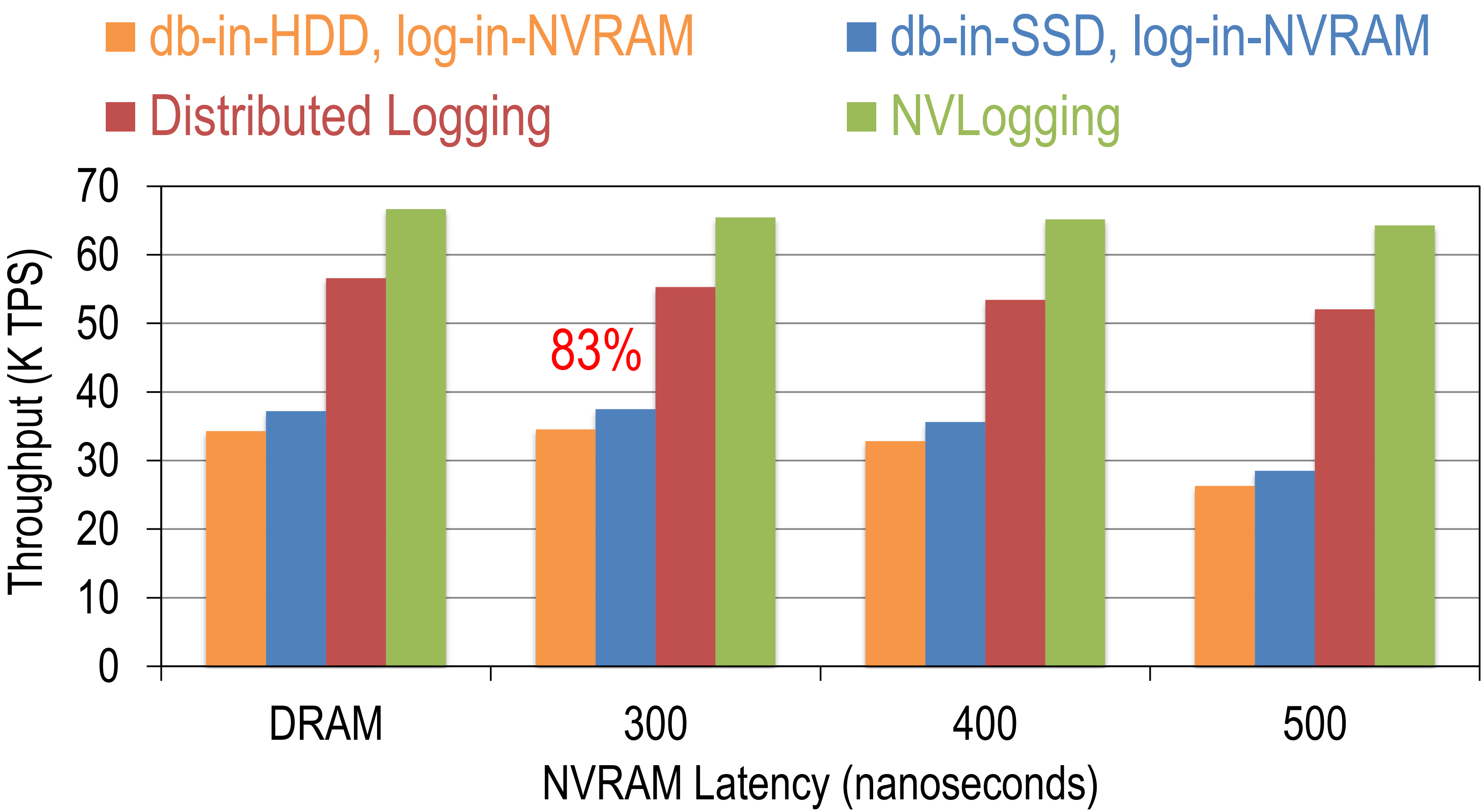
Intel Persistent Memory Emulator Platform

Xeon-EP @ 2.6 GHz, 64 GB DRAM + 256 GB NVRAM

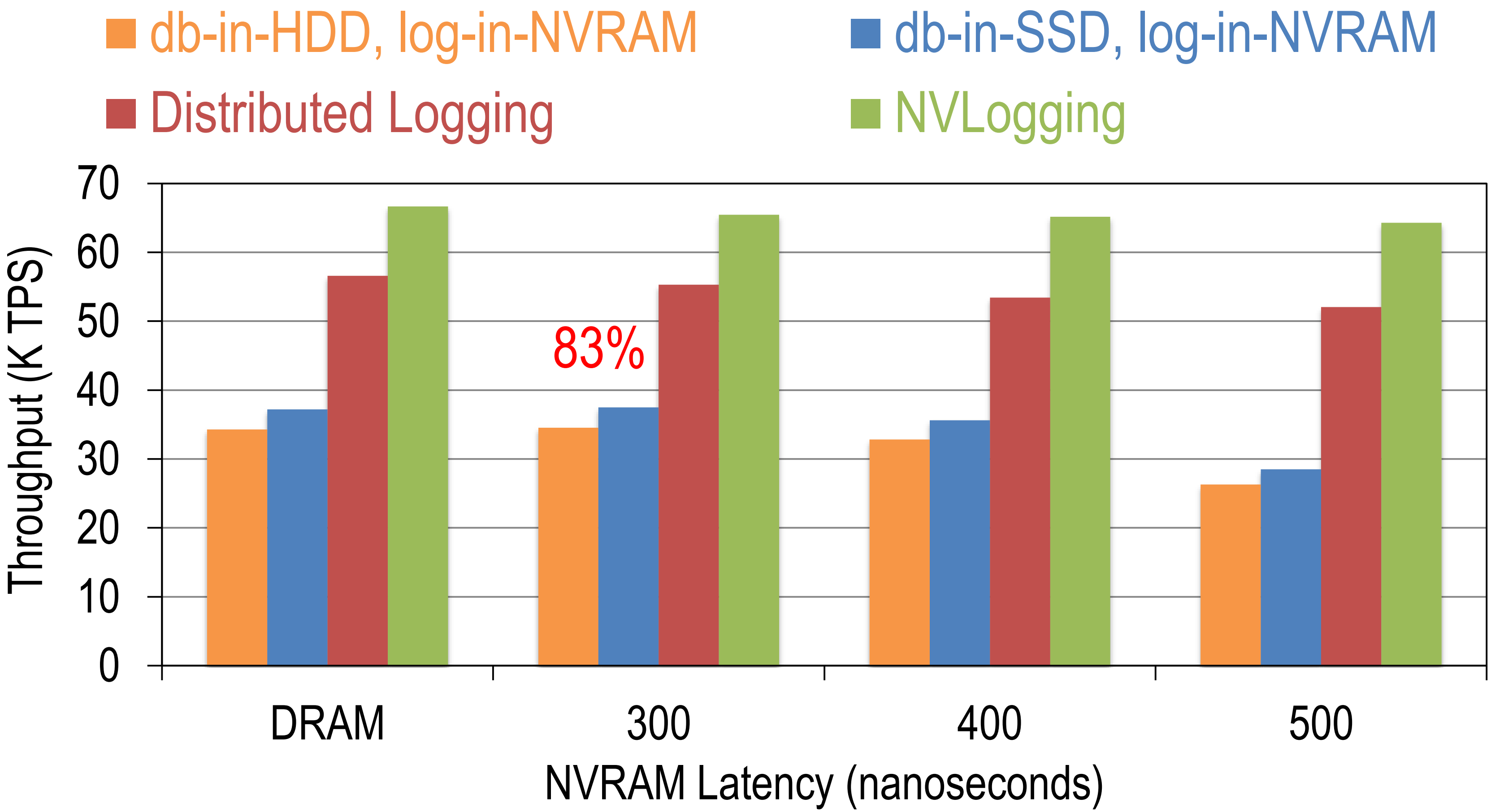
Benefits on Throughput with TPCB



Benefits on Throughput with TPCB

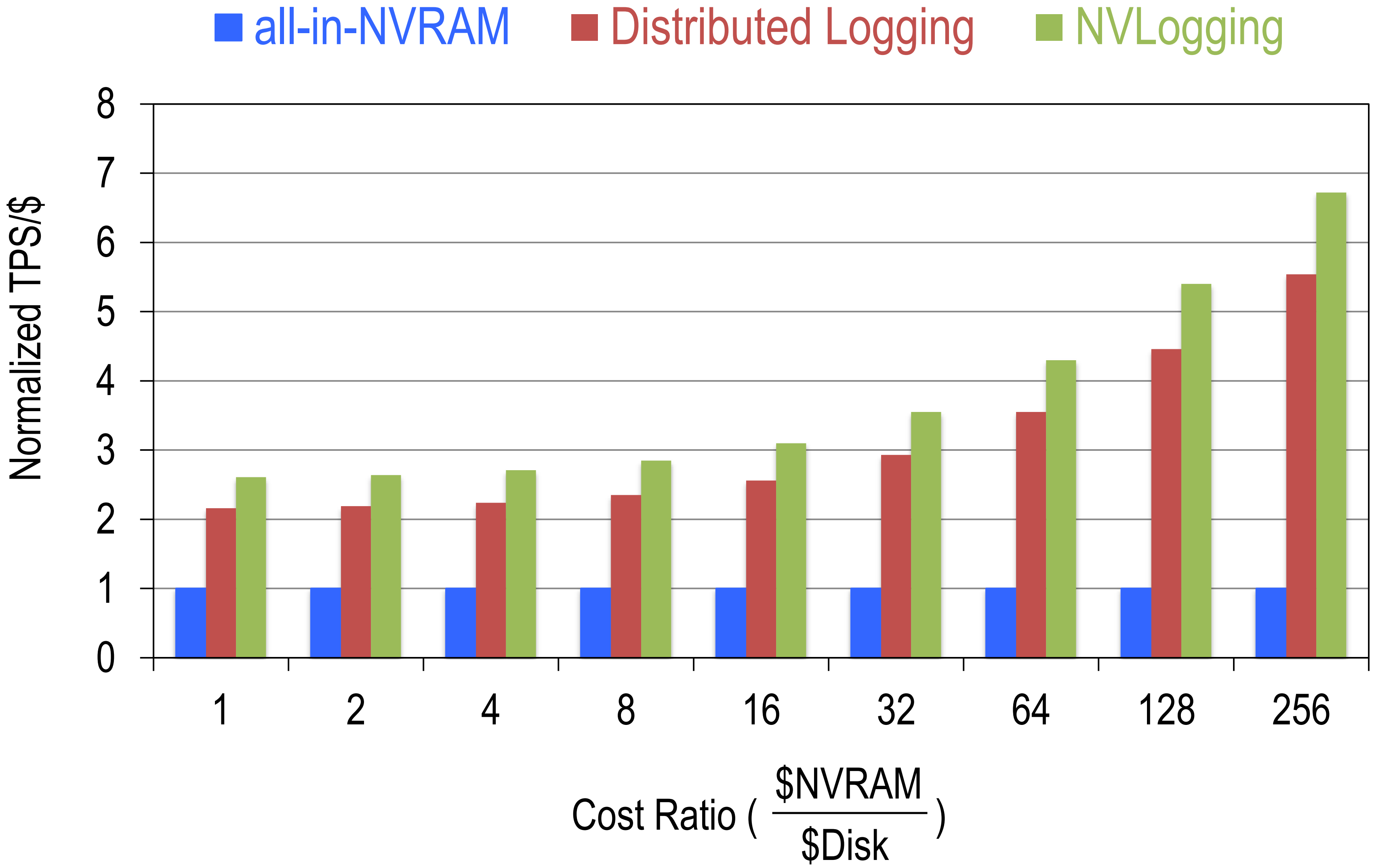


Benefits on Throughput with TPCB

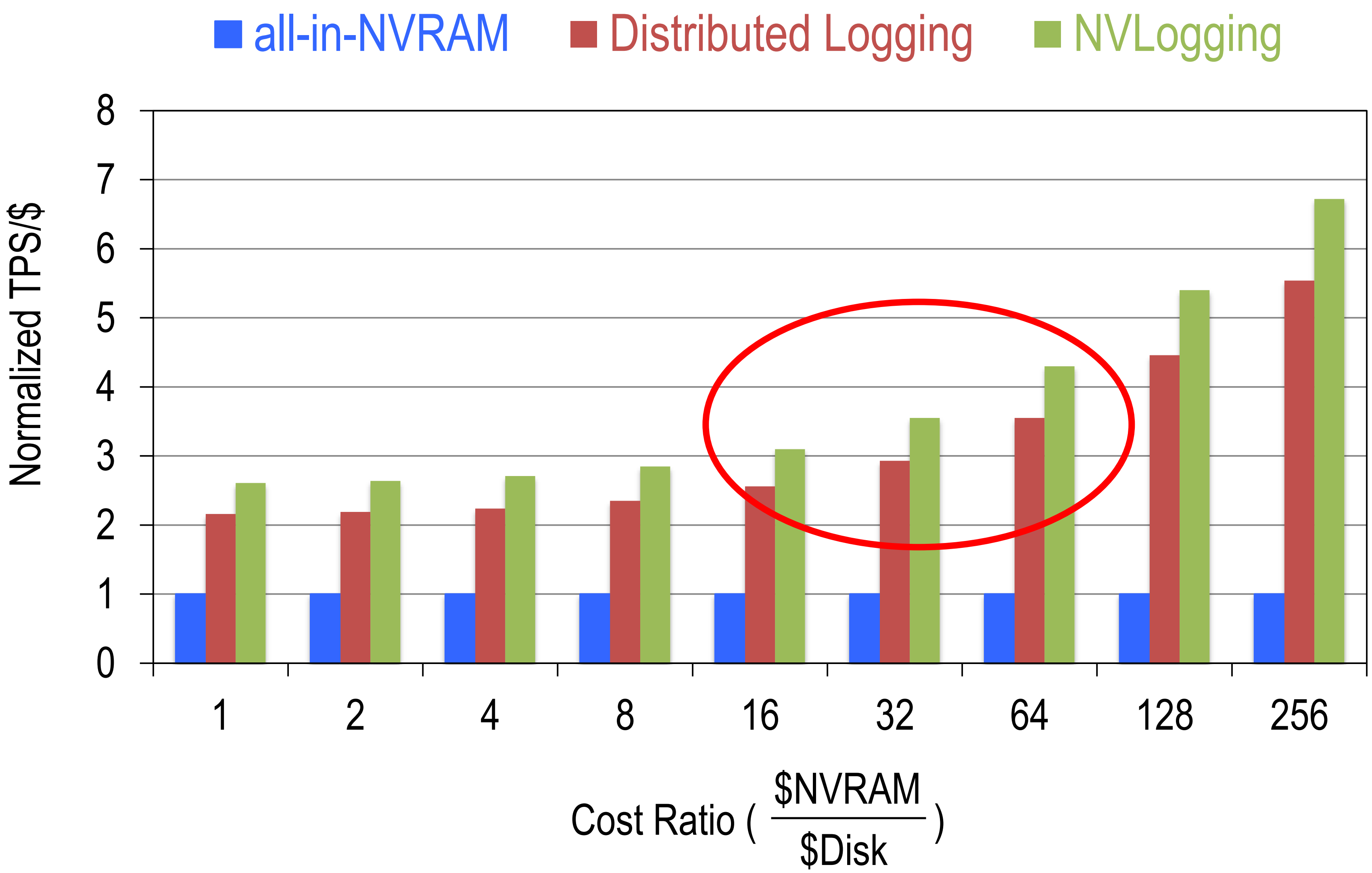


NVLogging improves TPS by 2.3x,
compared to the baseline (db-in-SSD, log-in-NVRAM)

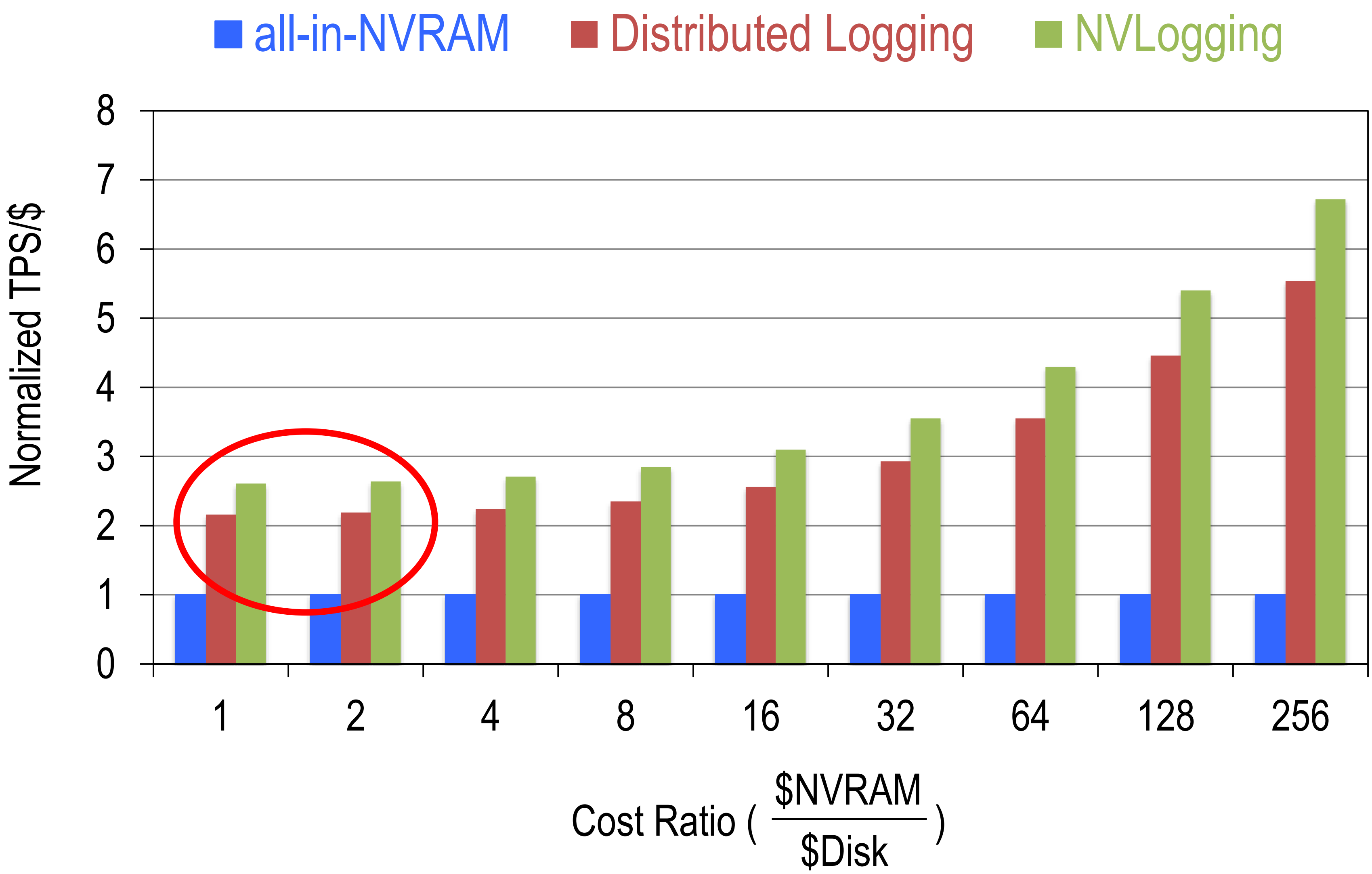
Benefits on Cost Savings



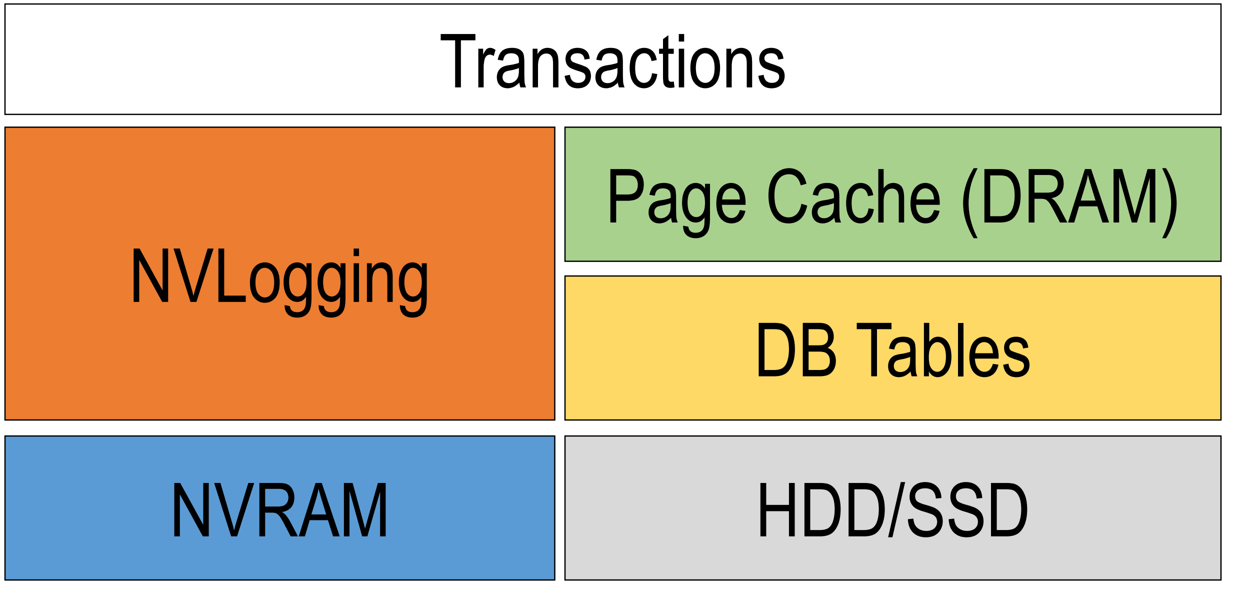
Benefits on Cost Savings



Benefits on Cost Savings



Conclusion



1 **Cost-effective Solution for DB system**

Higher TPS/$

2 **Per-transaction Logging with NVRAM**

Exploit NVRAM's both non-volatility and byte-addressability

Thanks!

Jian Huang

jian.huang@gatech.edu

Karsten Schwan   Moinuddin K. Qureshi

Q&A