Paper ID: 274

Write-Optimized and Consistent RDMA-based Non-Volatile Main Memory Systems

Xinxin Liu, Yu Hua, Xuan Li, Qifan Liu

Huazhong University of Science and Technology

ICCD 2021

Background

- Non-Volatile Main Memory (NVMM)
  - Non-volatility, byte-addressability, high density and DRAM-scale latency.

- Remote Direct Memory Access (RDMA)
  - Allow to directly access remote memory via bypassing kernel and zero memory copy.
    - Two-sided RDMA operations (send and recv):
    - One-sided RDMA operations (read, write and atomic):
      - Provide higher bandwidth/lower latency than two-sided one.
      - Do not involve remote CPU.

Background

- Non-Volatile Main Memory (NVMM)
  - ✓ Non-volatility, byte-addressability, high density and DRAM-class latency.
- Remote Direct Memory Access (RDMA)
  - ✓ Allow to directly access remote memory via bypassing kernel and zero memory copy.
    - ✓ Two-sided RDMA operations (send and recv):
    - ✓ One-sided RDMA operations (read, write and atomic):
      - ✓ Provide higher bandwidth/lower latency than two-sided one.
      - ✓ Do not involve remote CPU.

NVMM can be directly accessed through the RDMA network.

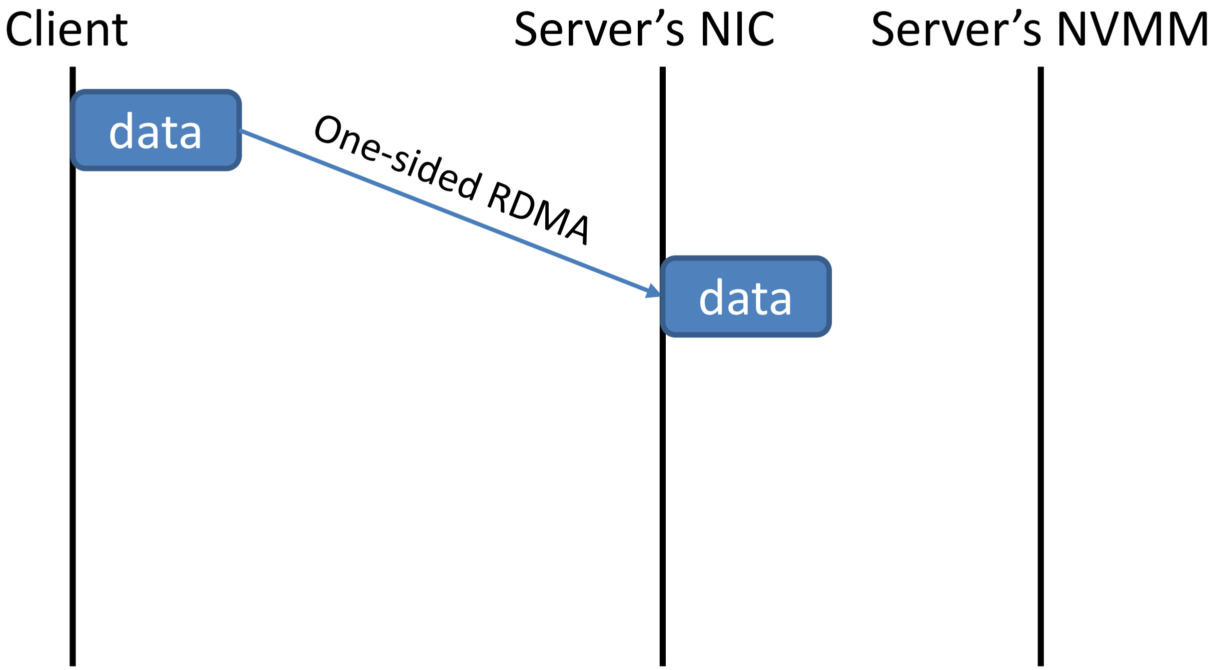
⇩

RDMA-based NVMM systems become an important research topic.

Challenges

➢ RDMA NICs fail to guarantee persistence with NVMM.

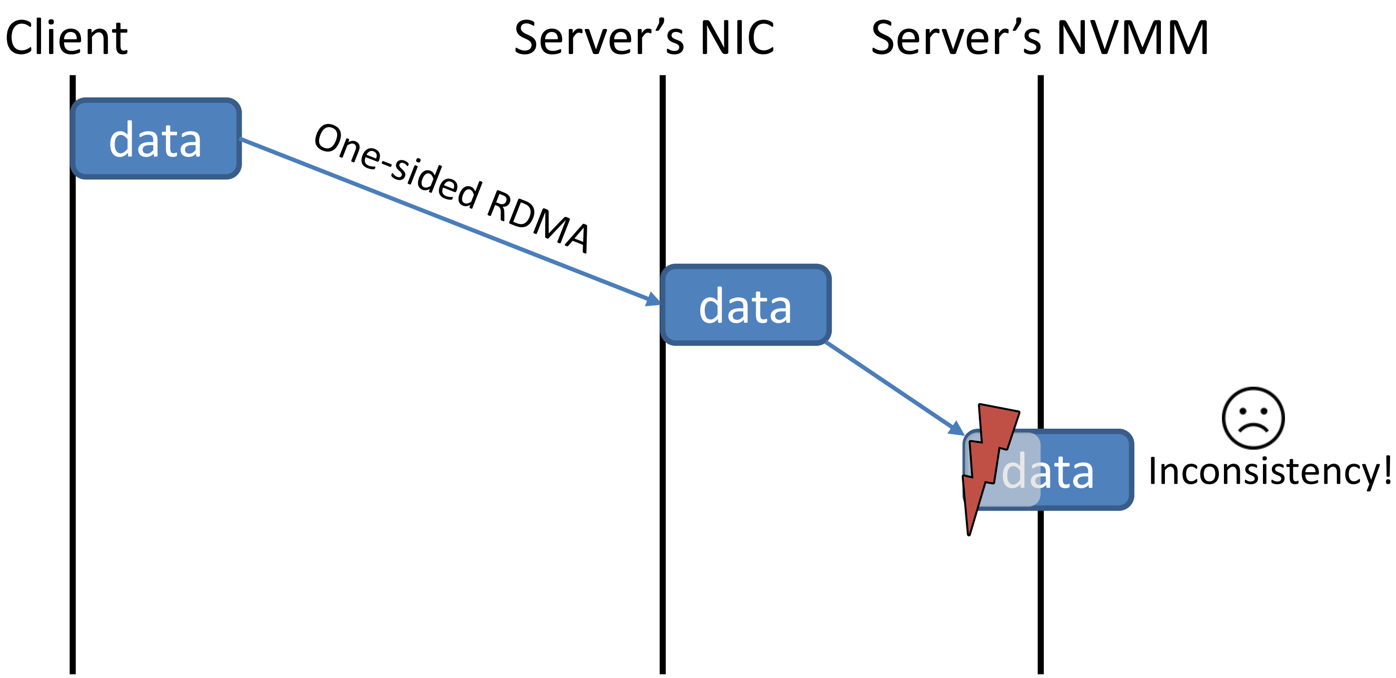
➢ Using one-sided RDMA to access remote NVMM needs to address the challenges of guaranteeing Remote Data Atomicity (RDA):



Challenges

- RDMA NICs fail to guarantee persistence with NVMM.

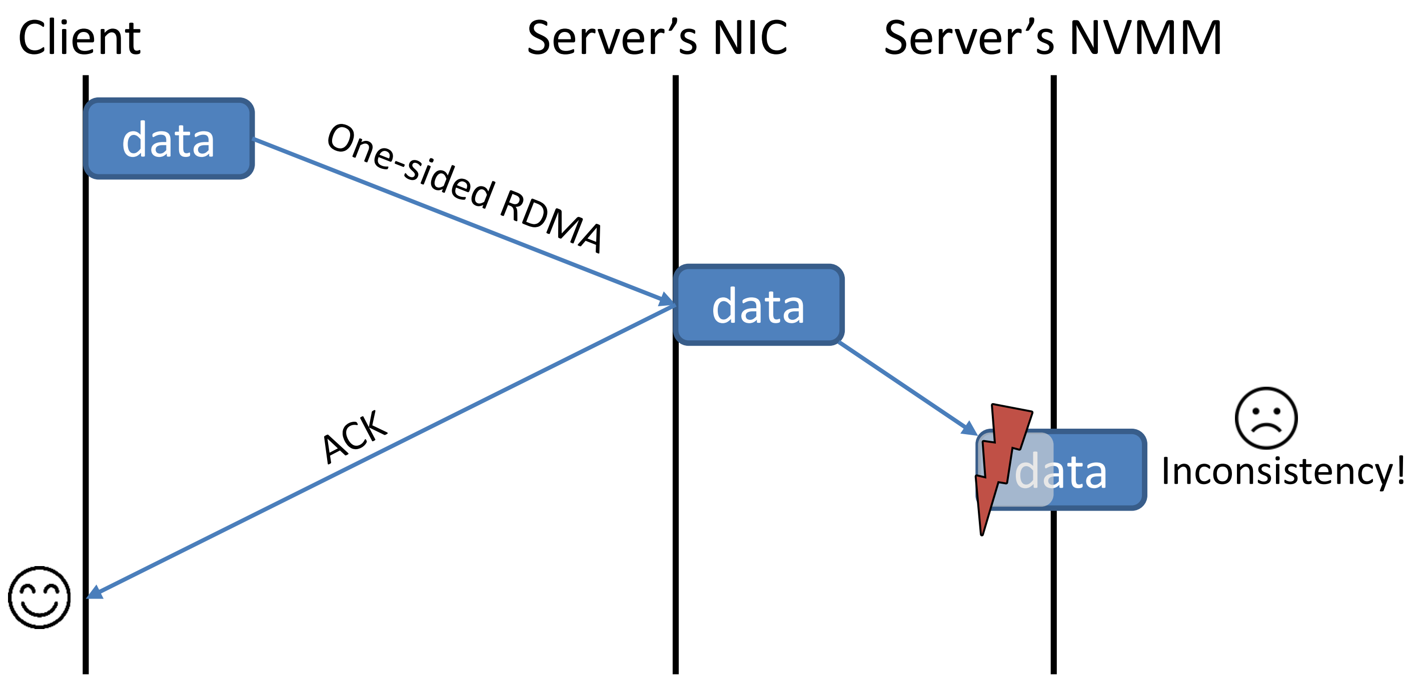
- Using one-sided RDMA to access remote NVMM needs to address the challenges of guaranteeing Remote Data Atomicity (RDA):



Challenges

- RDMA NICs fail to guarantee persistence with NVMM.

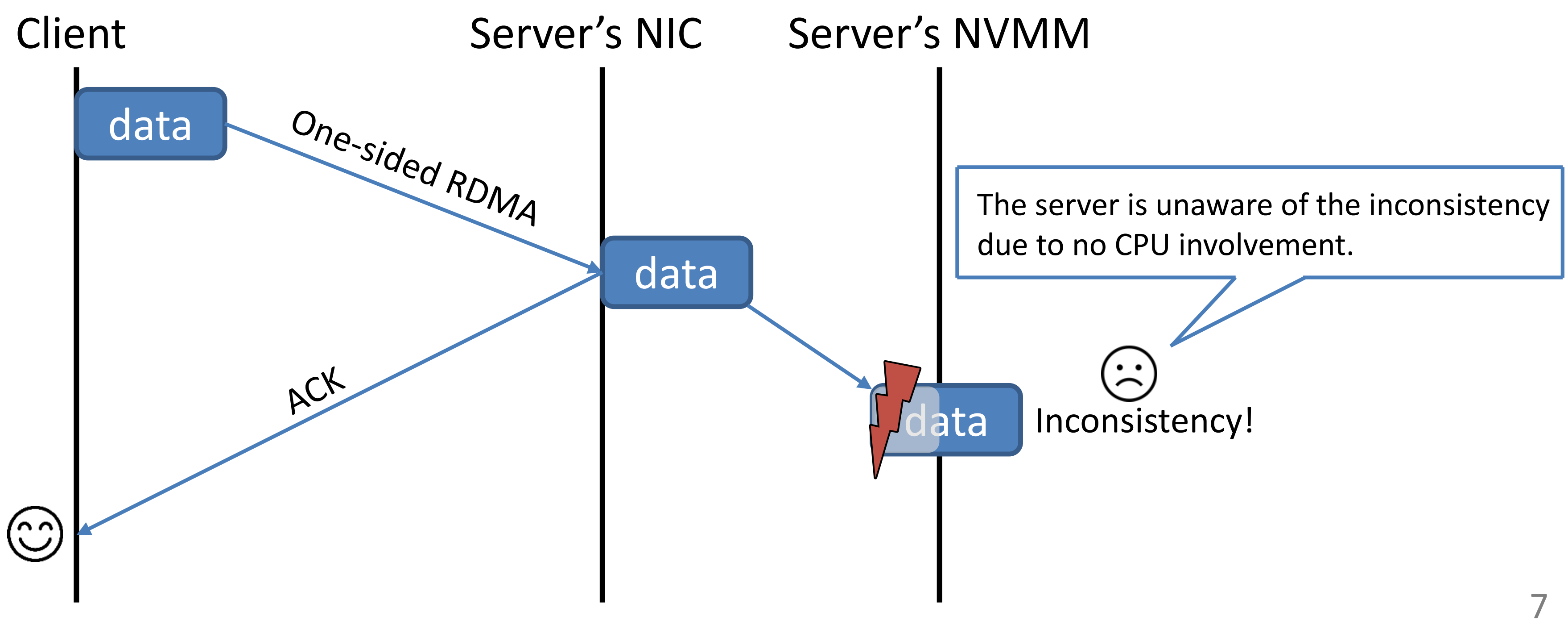
- Using one-sided RDMA to access remote NVMM needs to address the challenges of guaranteeing Remote Data Atomicity (RDA):



Challenges

- RDMA NICs fail to guarantee persistence with NVMM.

- Using one-sided RDMA to access remote NVMM needs to address the challenges of guaranteeing Remote Data Atomicity (RDA):

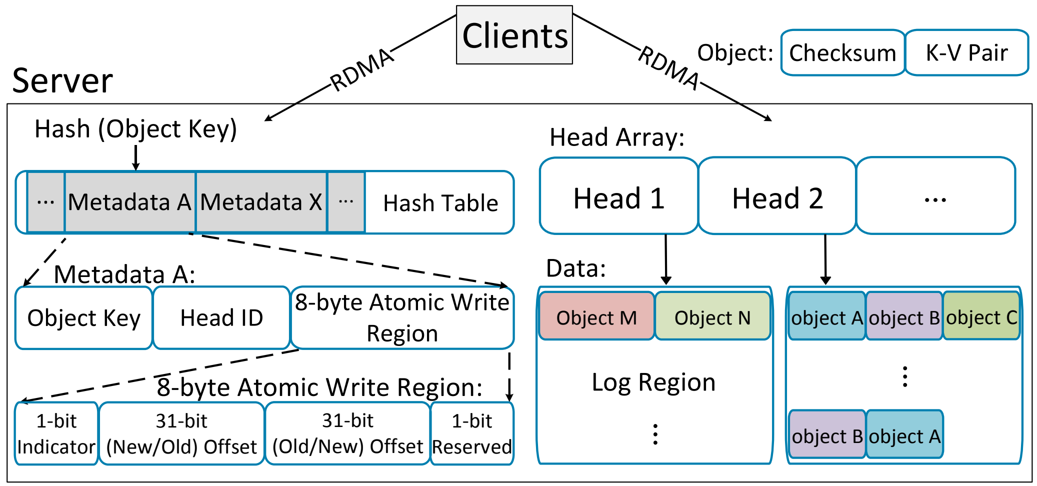


Existing Solutions

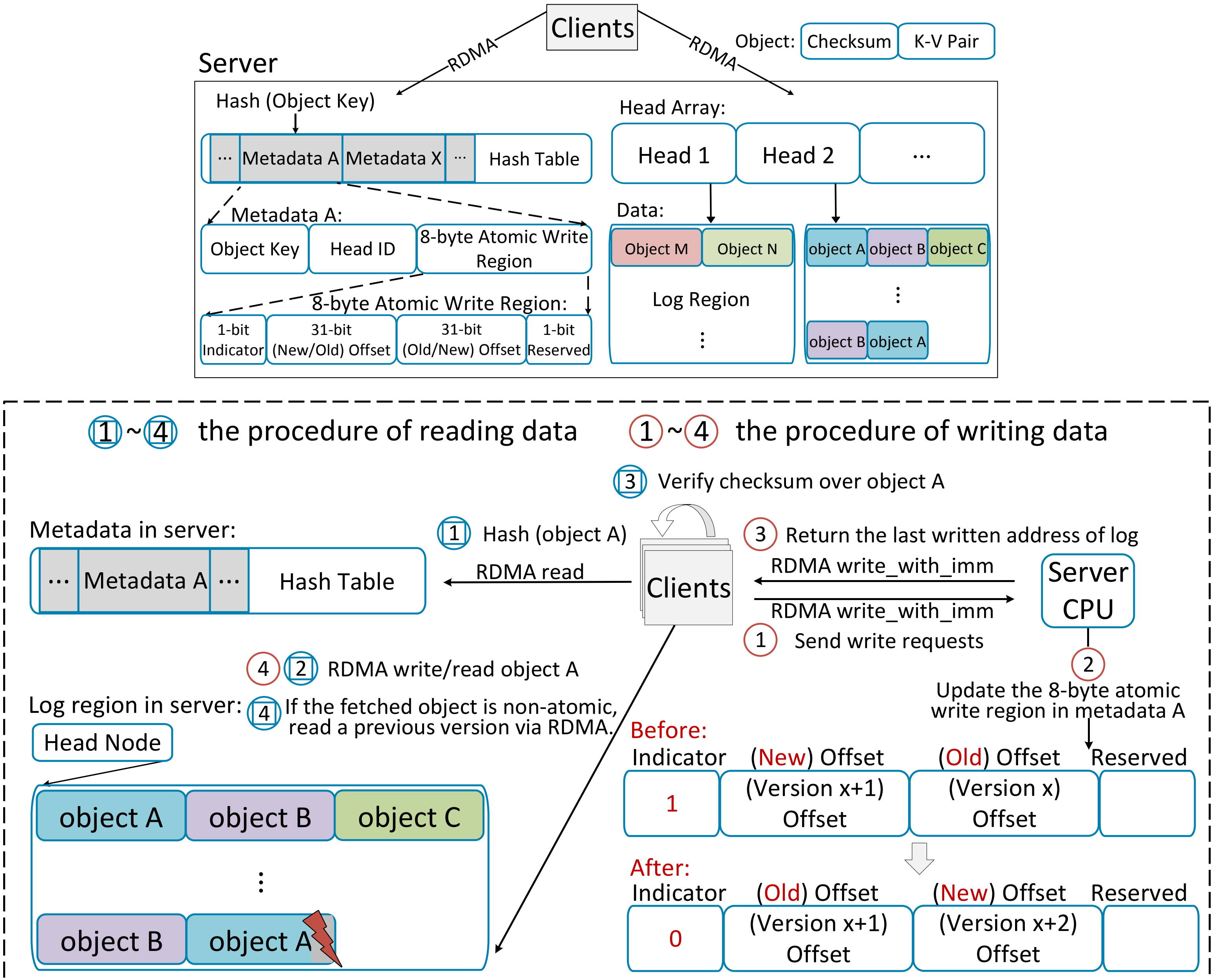
Inefficiency due to:

- **High Network Overheads**
  - ✓ Leverage an extra RDMA read after RDMA write(s)
- **High CPU Consumption**
  - ✓ Logging and COW require the remote CPU to control the sequence among operations.
- **Double NVMM Writes**
  - ✓ Consuming the limited NVMM endurance due to first checking the written data in buffers, and then applying them into the destination addresses.

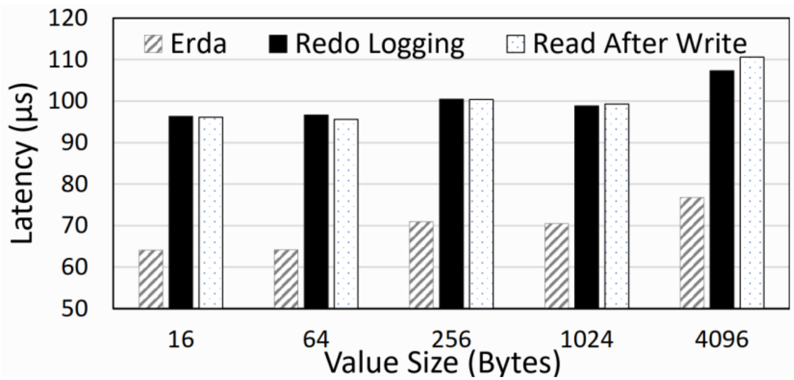
System Design of Erda



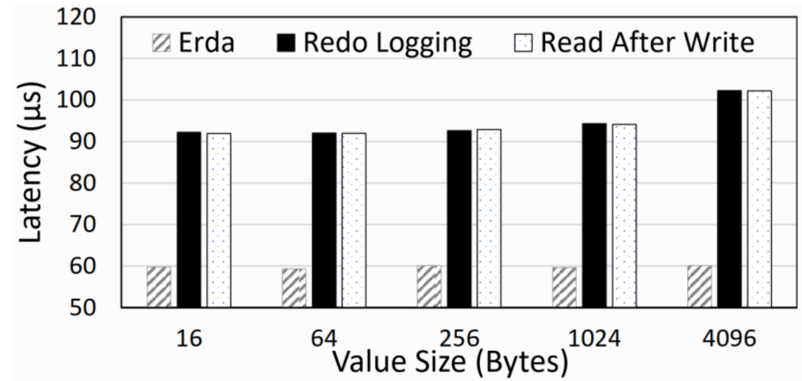
System Design of Erda



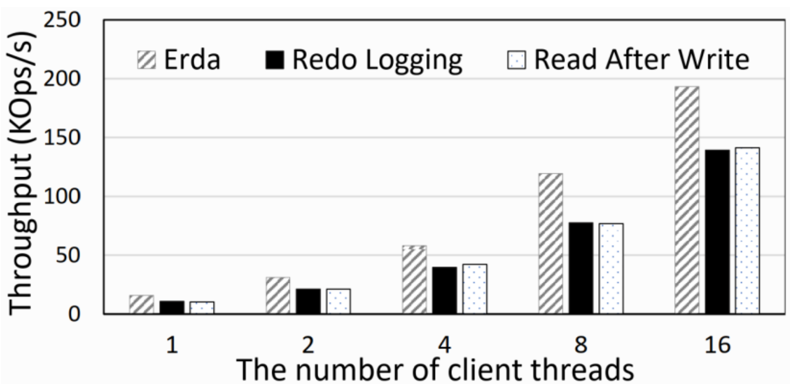
Evaluation



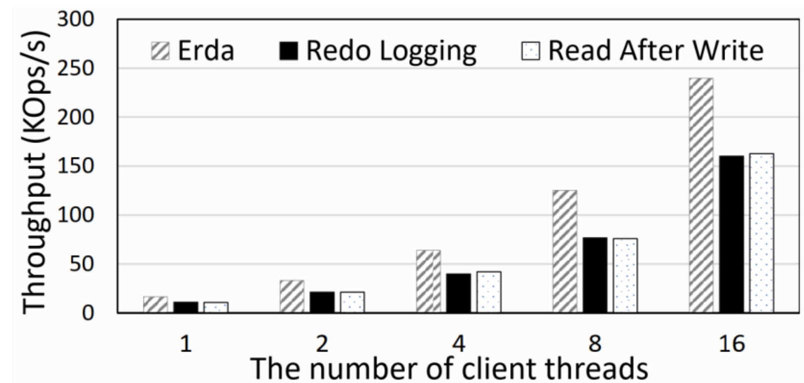
The latency of the update-heavy workload.



The latency of the read-mostly workload.



The throughput of the update-heavy workload.



The throughput of the read-mostly workload.

- The number of written bytes. ***N*** is the size of one KV pair. ***Size(key)*** is the key

	Create	Update	Delete
Erda	Size(key)+9+N	8+N	Size(key)+9
Redo Logging	Size(key)+12+2N	4+2N	Size(key)+8
Read After Write	Size(key)+12+2N	4+2N	Size(key)+8

Conclusion

- **Challenges of guaranteeing Remote Data Atomicity (RDA):**
  - ✓ High Network Overheads
  - ✓ High CPU Consumption
  - ✓ Double NVMM Writes

- **Erda:**
  - ✓ A write-optimized log-structured NVMM design for Efficient Remote Data Atomicity.
  - ✓ Leverage Out-of-Place Updates & CRC Checksum & 8-Byte Atomic Write.

- Compared with state-of-the-art schemes, **Erda reduces NVMM writes** by 50%, significantly **improves throughput** and **decreases latency.**

Thanks! Q&A