High Availability For Key-Value Stores Using Checkpoint/Restore

Fadhil Abubaker, Hussain Sadiq Abuwala

Introduction

- Modern distributed systems are expected to be highly available.

Introduction

- Modern distributed systems are expected to be highly available.
- High-availability is implemented through replication.
  - Synchronous vs Asynchronous.
  - Active-Active vs Active-Standby.

Introduction

- Modern distributed systems are expected to be highly available.
- High-availability is implemented through replication.
  - Synchronous vs Asynchronous.
  - Active-Active vs Active-Standby.
- But implementing HA is difficult.
  - Propagate updates.
  - Coordinate transactions.
  - Atomic handover.
  - Performance.

Introduction

- Modern distributed systems are expected to be highly available.
- High-availability is implemented through replication.
  - Synchronous vs Asynchronous.
  - Active-Active vs Active-Standby.
- But implementing HA is difficult.
  - Propagate updates.
  - Coordinate transactions.
  - Atomic handover.
  - Performance.
- Rely on an external layer to provide replication?

External Replication Layer

Push replication outside of the database system, delegating it to an external layer.

[1] J. Kim, K. Salem, K. Daudjee, A. Aboulnaga, and X. Pan. Database High Availability Using SHADOW Systems.

External Replication Layer

Push replication outside of the database system, delegating it to an external layer.

- Shared-Disk

[1] J. Kim, K. Salem, K. Daudjee, A. Aboulnaga, and X. Pan. Database High Availability Using SHADOW Systems.

External Replication Layer

Push replication outside of the database system, delegating it to an external layer.

- Shared-Disk
- SHADOW[1]

[1] J. Kim, K. Salem, K. Daudjee, A. Aboulnaga, and X. Pan. Database High Availability Using SHADOW Systems.

External Replication Layer

Push replication outside of the database system, delegating it to an external layer.

- Shared-Disk
- SHADOW[1]
- Distributed Replicated Block Device (DRBD)

[1] J. Kim, K. Salem, K. Daudjee, A. Aboulnaga, and X. Pan. Database High Availability Using SHADOW Systems.

External Replication Layer

Push replication outside of the database system, delegating it to an external layer.

- Shared-Disk
- SHADOW[1]
- Distributed Replicated Block Device (DRBD)
- Virtual Machine Replication

[1] J. Kim, K. Salem, K. Daudjee, A. Aboulnaga, and X. Pan. Database High Availability Using SHADOW Systems.

External Replication Layer

Push replication outside of the database system, delegating it to an external layer.

- Shared-Disk
- SHADOW[1]
- Distributed Replicated Block Device (DRBD)
- Virtual Machine Replication
- ...Checkpoint/Restore?

[1] J. Kim, K. Salem, K. Daudjee, A. Aboulnaga, and X. Pan. Database High Availability Using SHADOW Systems.

Checkpoint/Restore In Userspace (CRIU)

- Linux utility to checkpoint/restore in-memory state of a process/container.

Checkpoint/Restore In Userspace (CRIU)

- Linux utility to checkpoint/restore in-memory state of a process/container.
- A running application can be checkpointed to persistent storage as a set of files.

Checkpoint/Restore In Userspace (CRIU)

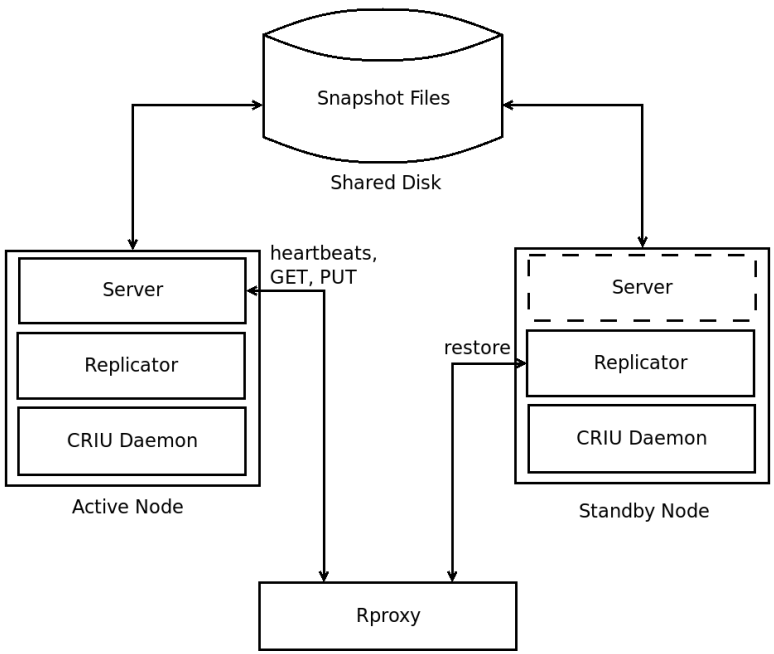
- Linux utility to checkpoint/restore in-memory state of a process/container.
- A running application can be checkpointed to persistent storage as a set of files.
- The application can be then restored back to the point it was frozen using the checkpoint files.

Checkpoint/Restore In Userspace (CRIU)

- Linux utility to checkpoint/restore in-memory state of a process/container.
- A running application can be checkpointed to persistent storage as a set of files.
- The application can be then restored back to the point it was frozen using the checkpoint files.
- Integrated with container runtimes such as OpenVZ, LXC/LXD, Docker, Podman, etc.
- Commonly used for container live-migration.

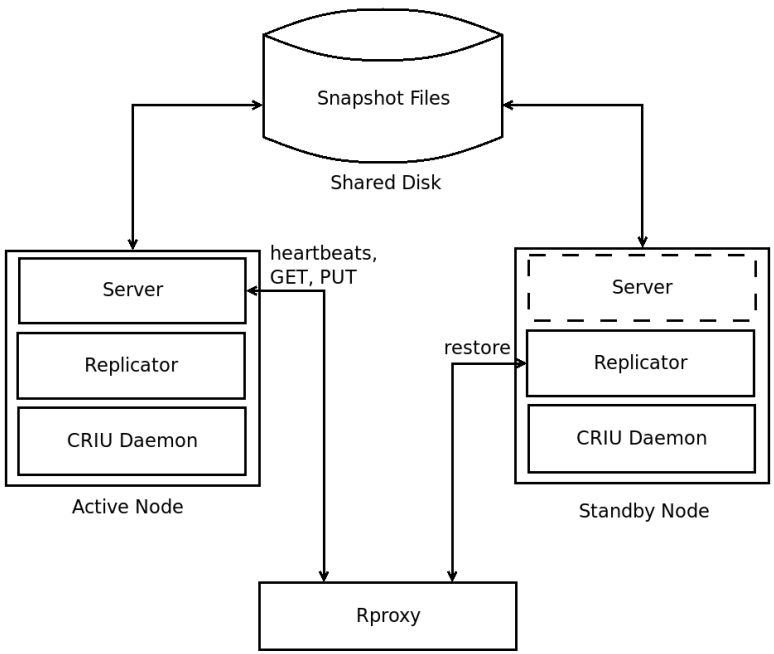
System Architecture

Normal Operation



System Architecture

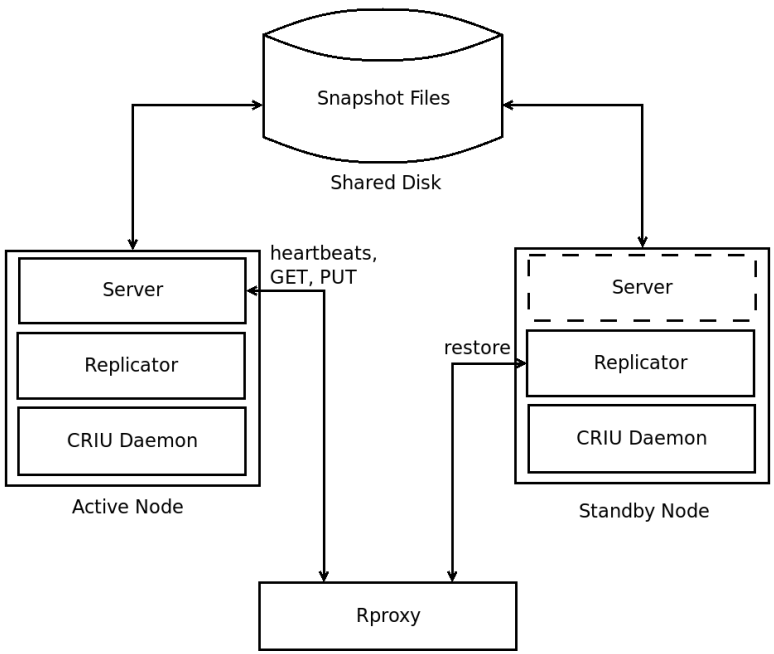
Normal Operation



- clients send requests to the **rproxy**.

System Architecture

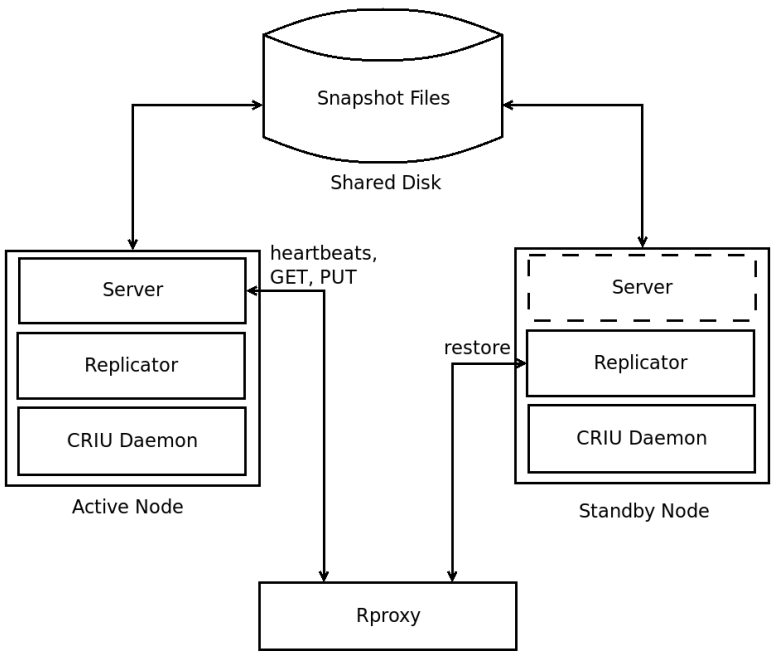
Normal Operation



- clients send requests to the **rproxy**.
- request is forwarded to the active **server**.

System Architecture

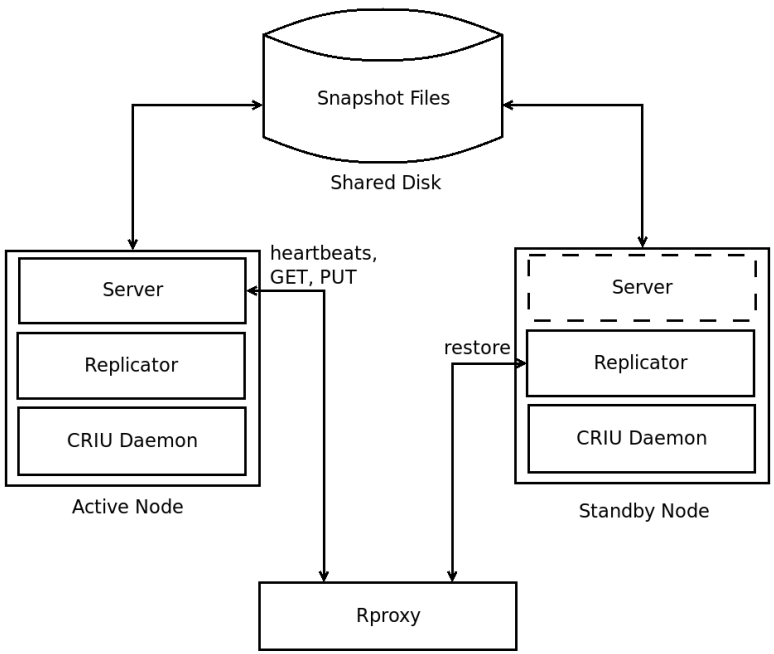
Normal Operation



- clients send requests to the **rproxy**.
- request is forwarded to the active **server**.
- **server** calls `checkpoint` endpoint on the **replicator** after *n* updates.

System Architecture

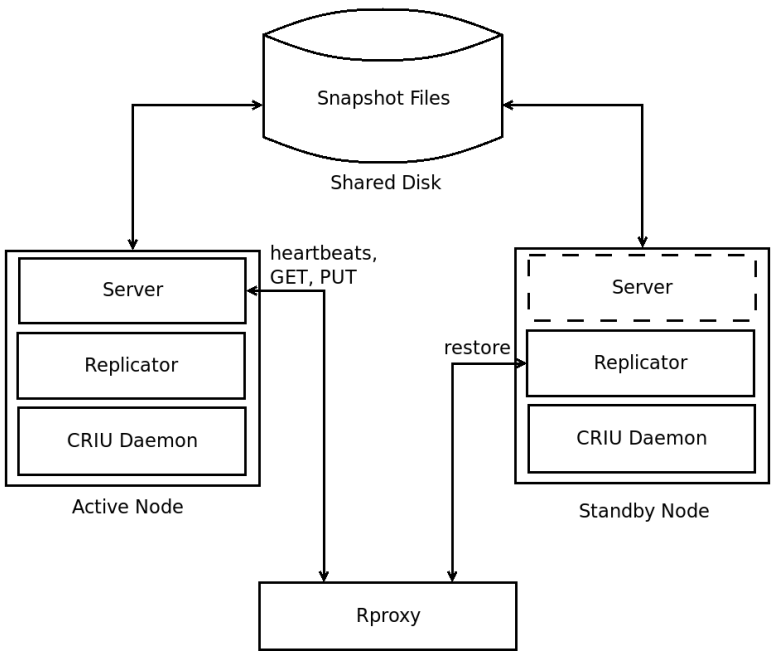
Normal Operation



- clients send requests to the **rproxy**.
- request is forwarded to the active **server**.
- **server** calls `checkpoint` endpoint on the **replicator** after *n* updates.
- **replicator** interfaces with the **C/R daemon** and saves a snapshot to the shared-disk.

System Architecture

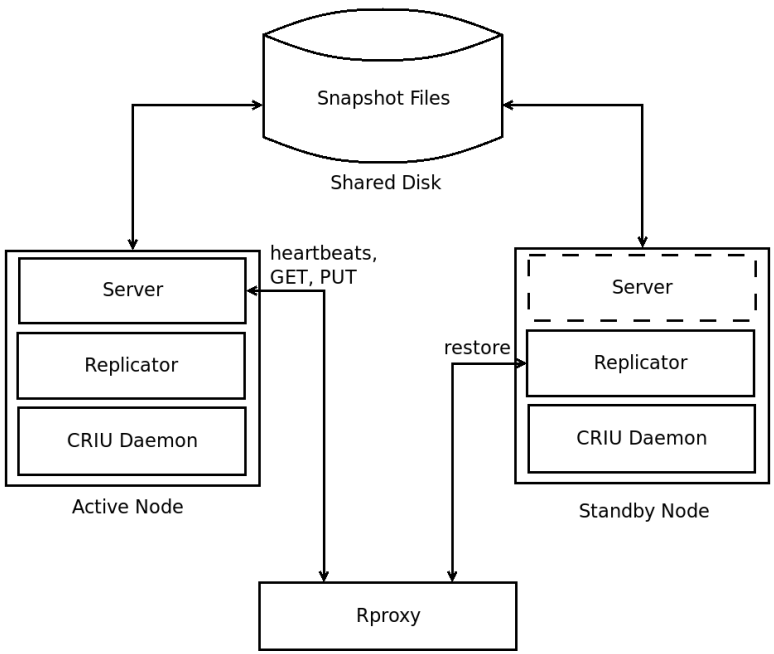
Failover Process



- **rproxy** sends regular heartbeats to active **server**.

System Architecture

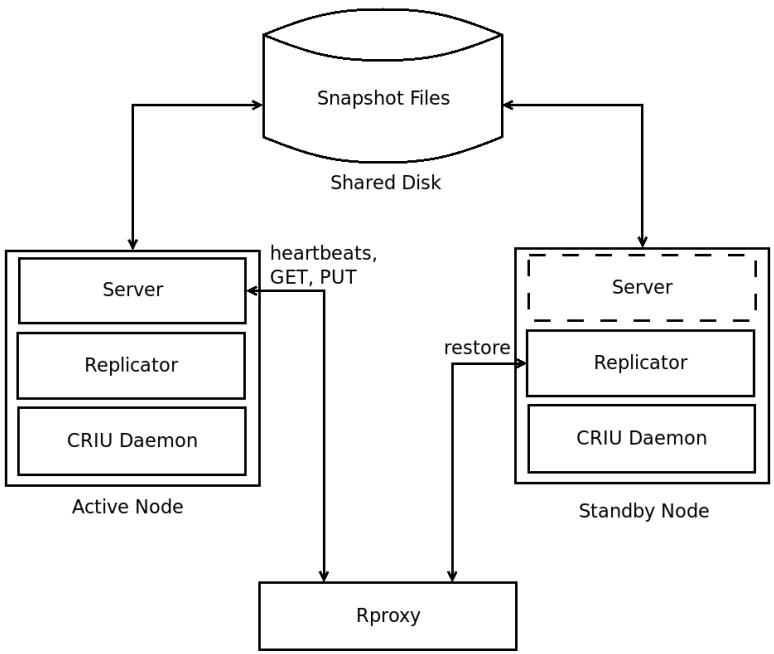
Failover Process



- **rproxy** sends regular heartbeats to active **server**.
- On timeout, calls the `restore` endpoint on standby **replicator**.

System Architecture

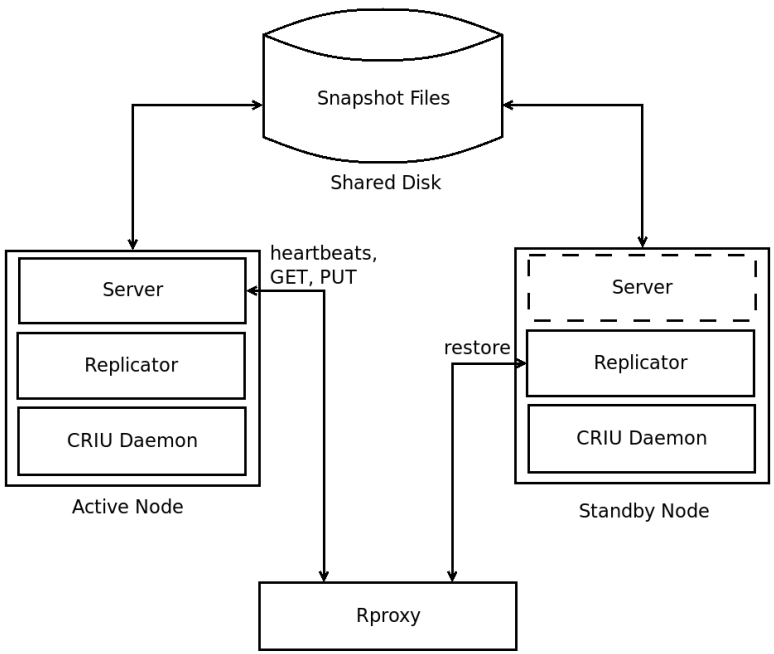
Failover Process



- **rproxy** sends regular heartbeats to active **server**.
- On timeout, calls the `restore` endpoint on standby **replicator**.
- **replicator** restores the **server** from the latest snapshot on the shared-disk.

System Architecture

Failover Process



- **rproxy** sends regular heartbeats to active **server**.
- On timeout, calls the `restore` endpoint on standby **replicator**.
- **replicator** restores the **server** from the latest snapshot on the shared-disk.
- Normal operations resume, with requests forwarded to the standby.

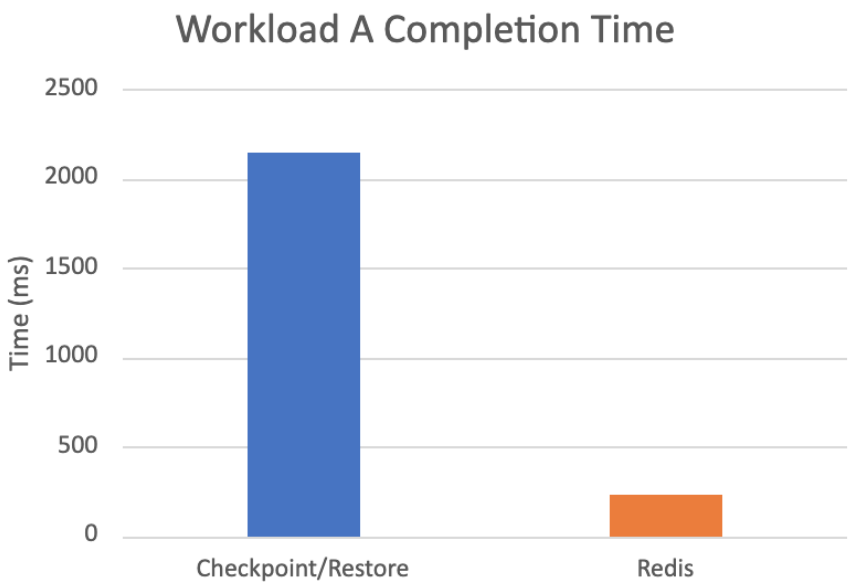
Evaluation

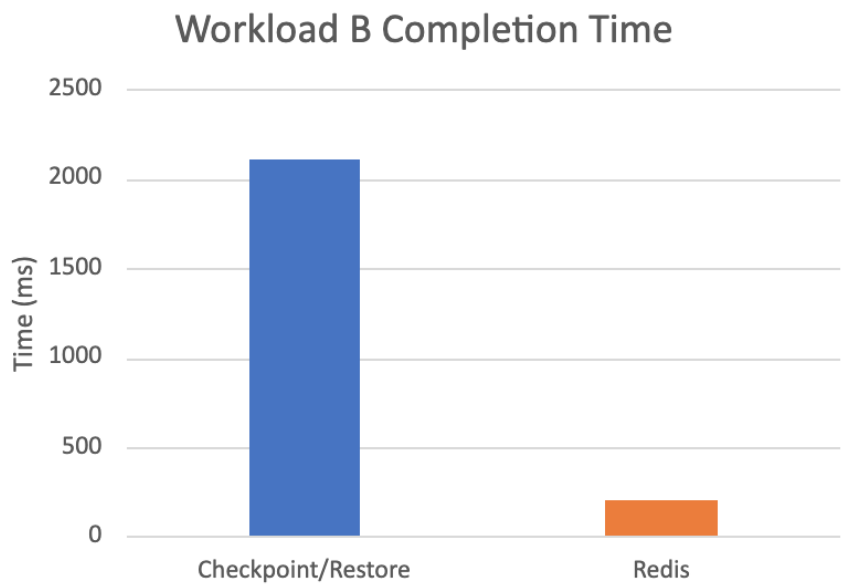
Methodology

- Yahoo! Cloud Serving Benchmark.
- Implemented a custom YCSB interface to support our key-value store.
- Each workload consists of 1000 operations.
- Workload A: 50/50 read/write mix.
- Workload B: 95/5 read/write mix.
- Snapshots were captured after every 250 updates.

Evaluation

Storage Performance vs Redis (No Replication)

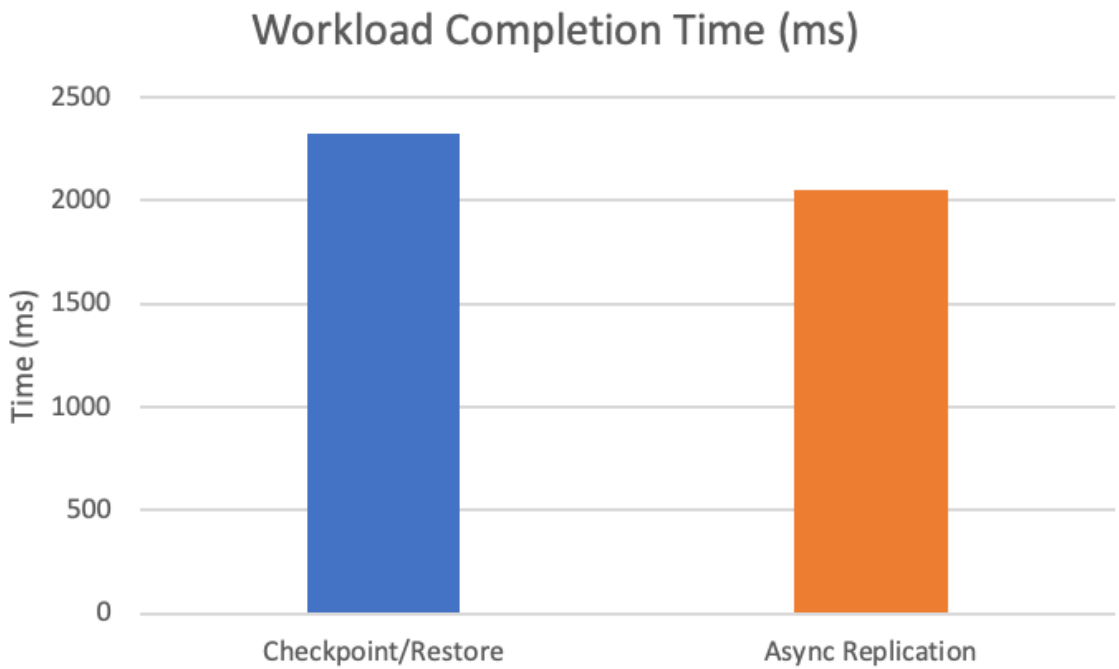




Storage performance likely suffers due to HTTP and JSON overhead.

Evaluation

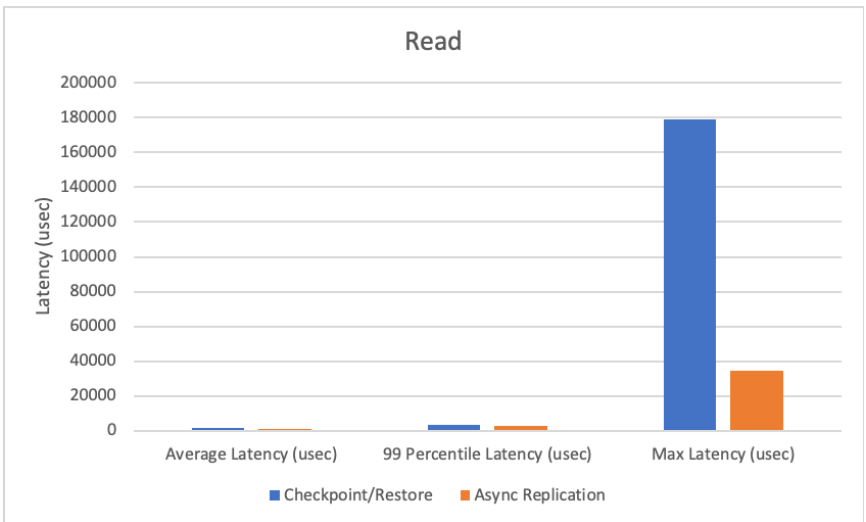
Replication Performance vs Asynchronous Replication

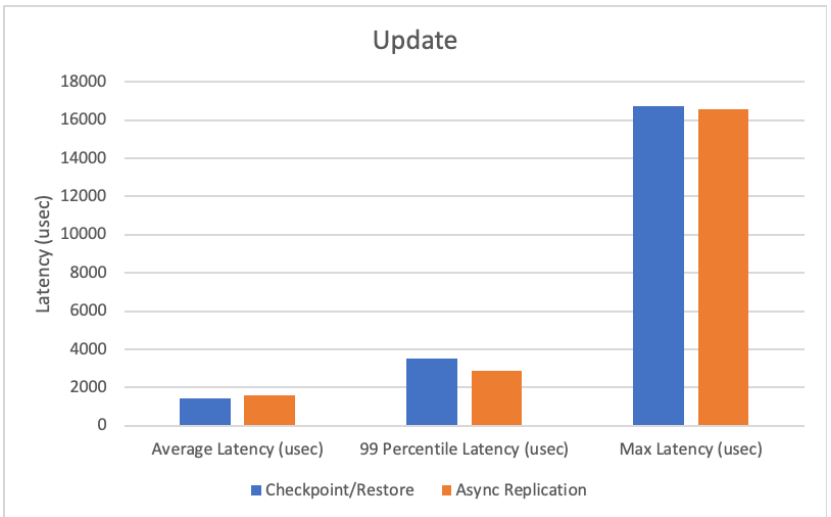


Requests are blocked while snapshots happen, leading to slightly higher completion times.

Evaluation

Replication Performance vs Asynchronous Replication

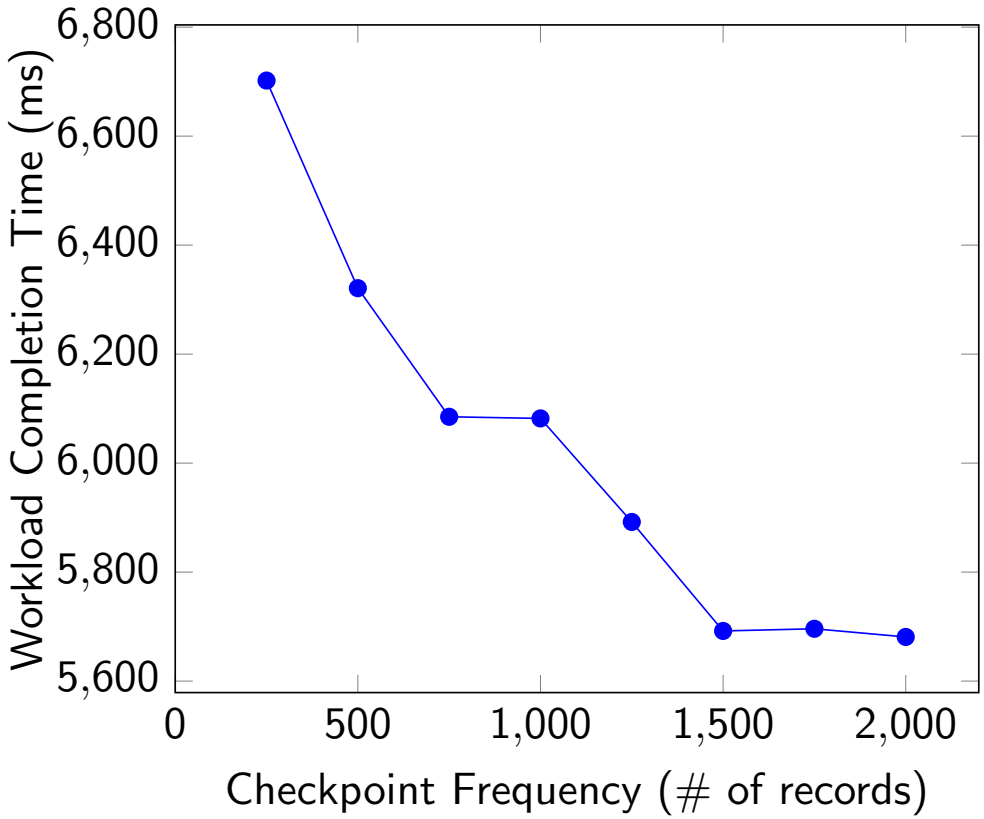




Discrepancy exists for max latency on reads due to snapshots blocking unlucky read requests.

Evaluation

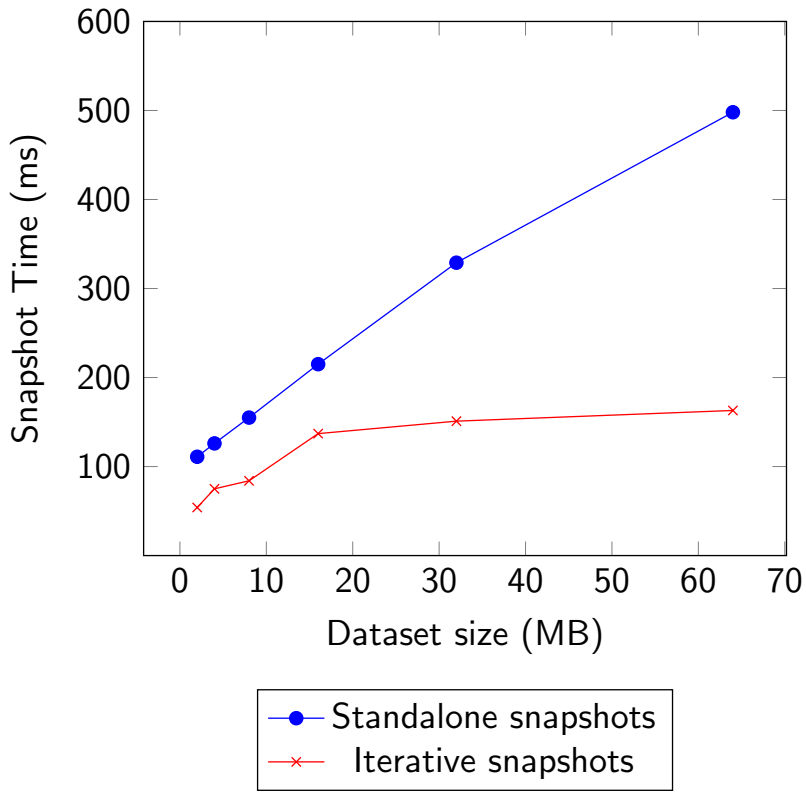
Impact of Checkpoint Frequency on Workload Completion



Custom workload with 5000 insert operations with varying checkpoint intervals.

Evaluation

Standalone vs Iterative Snapshots



Iterative snapshots track changes in memory pages across consecutive checkpoints, making it more efficient.

Demo

Questions?