Near-Optimal Latency Versus Cost Tradeoffs in Geo-Distributed Storage

Muhammed Uluyol, Anthony Huang, Ayush Goel, Mosharaf Chowdhury, Harsha V. Madhyastha

University of Michigan
Distribute Web Servers for Interactive Latency
Distribute **Data** for Availability
Distribute Data for Availability **and** Latency
Linearizability Imposes Unavoidable Trade-offs

- Read vs Write Latency
- Read Latency vs Cost
Linearizability Imposes **Unavoidable Trade-offs**

- **Read vs Write Latency**
- **Read Latency vs Cost**
Linearizability Imposes **Unavoidable** Trade-offs

- **Read vs Write Latency**
- **Read Latency vs Cost**

Overlap ensures consistency
Linearizability Imposes **Unavoidable** Trade-offs

- Read vs Write Latency
- Read Latency vs Cost
Linearizability Imposes **Unavoidable** Trade-offs

- Read vs Write Latency
- Read Latency vs Cost
Linearizability Imposes Unavoidable Trade-offs

- Read vs Write Latency
- Read Latency vs Cost
Linearizability Imposes **Unavoidable** Trade-offs

- Read vs Write Latency
- **Read Latency vs Cost**
Linearizability Imposes **Unavoidable Trade-offs**

- Read vs Write Latency
- **Read Latency vs Cost**
How Do Existing Approaches Perform?

![Diagram showing the relationship between storage overhead budget, write latency budget, and read latency. The lowest possible read latency is on the y-axis.](image)
How Do Existing Approaches Perform?
How Do Existing Approaches Perform?

- Lowest Possible Read Latency
- Storage Overhead Budget
- Write Latency Budget

![Diagram showing performance metrics]
How Do Existing Approaches Perform?

- **Lowest Possible Read Latency**
- **Storage Overhead Budget**
- **Write Latency Budget**

![3D Graph](image)

![Graphs](image)
How Do Existing Approaches Perform?

- EPaxos: state-of-the-art geo-replication protocol
How Do Existing Approaches Perform?

- EPaxos: state-of-the-art geo-replication protocol
- Compare with estimate of theoretical lower bound
How Do Existing Approaches Perform?

- EPaxos: state-of-the-art geo-replication protocol
- Compare with estimate of theoretical lower bound
  - No particular protocol
  - Respects consistency and fault-tolerance constraints
How Do Existing Approaches Perform?

- **EPaxos**: state-of-the-art geo-replication protocol

- Compare with estimate of theoretical lower bound
  - No particular protocol
  - Respects consistency and fault-tolerance constraints

![Graph showing performance comparison between lower bound and EPaxos.](image)
How Do Existing Approaches Perform?

- EPaxos: state-of-the-art geo-replication protocol
- Compare with estimate of theoretical lower bound
  - No particular protocol
  - Respects consistency and fault-tolerance constraints

Core problem: replication
Each site stores a full copy
Lowering Cost with Erasure Coding

- Each site stores $1/k$th of the data
- RS-Paxos: Paxos on erasure-coded data
Lowering Cost with Erasure Coding

- Each site stores $1/k$th of the data
- RS-Paxos: Paxos on erasure-coded data
Lowering Cost with Erasure Coding

- Each site stores $\frac{1}{k}$th of the data
- RS-Paxos: Paxos on erasure-coded data

RS-Paxos Limitations

- Two-round writes
- $k$-site intersection between quorums
Recap of the Problem

- Want to spread data across DCs, but constraints that impose trade-offs
- State-of-the-art falls short of the optimal
- Use erasure coding → hurts latency
**Pando**: Near-Optimal Trade-off

- **Two-round writes**
  Approximates latency of one-round writes

- **k-site intersection between quorums**
  1-site intersection (common-case)
Paxos Made Moderately Complex

ROBBERT VAN RENESSE and DENIZ ALTINBUKEN, Cornell University

This article explains the full reconfigurable multidegree Paxos (or multi-Paxos) protocol. Paxos is by no means a simple protocol, even though it is based on relatively simple invariants. We provide pseudocode and explain it guided by invariants. We initially avoid optimizations that complicate comprehension. Next we discuss liveness, list various optimizations that make the protocol practical, and present variants of the protocol.

Categories and Subject Descriptors: C.2.4 [Computer-Communication Networks]: Distributed Systems—Network operating systems; D.4.5 [Operating Systems]: Reliability—Fault-tolerance

General Terms: Design, Reliability

Additional Key Words and Phrases: Replicated state machines, consensus, voting

ACM Reference Format:
DOI: http://dx.doi.org/10.1145/2673577

1. INTRODUCTION

Paxos [Lamport 1998] is a protocol for state machine replication in an asynchronous environment that admits crash failures. It is useful to consider the terms in this sentence carefully:
Paxos Review

- 2-Phase writes: first become leader
Paxos Review

- 2-Phase writes: first become leader, then write
Paxos Review

- 2-Phase writes: first become leader, then write
Quickly Executing 2-Phase Writes

- **Step 1: faster Phase 1**
  - Flexible Paxos [OPODIS’16]: need Phase 1, 2 quorums to intersect
  - Phase 1 quorums need not overlap

10 ms

10 ms

30 ms

30 ms
Quickly Executing 2-Phase Writes

- Step 1: faster Phase 1
- Step 2: overlap latency cost of Phase 1 with Phase 2
  - RPC Chains [NSDI’09]: start Phase 2 at a delegate

10 ms

10 ms

15 ms

30 ms

I am leader

Ack.

Write data

Ack.
Pando: Near-Optimal Trade-off

- Two-round writes
  Approximates latency of one-round writes

- $k$-site intersection between quorums
  1-site intersection (common-case)
Write to All

\( k=2 \)
Write to All, **Wait for Quorum**

$k=2$
Write to All, **Wait for Quorum**

$k=2$  

**Phase 2**

**Rare Case**

**Common Case**

Read

$k=2$ Read
Achieving 1-Site Intersection

k=2

Phase 2

Rare Case

Maybe a write finished

Common Case

Read

Read
Achieving 1-Site Intersection

$k=2$

**Phase 2**

**Rare Case**

**Common Case**

Maybe a write finished

Read
Pando: Near-Optimal Trade-off

- Two-round writes
  Approximates latency of one-round writes
- k-site intersection between quorums
  1-site intersection (common-case)
See paper:
- Correctness
- Bounding latency under conflicts
Evaluation: Proximity to Lower Bound

- **Access set**: DCs hosting web servers reading/writing data
- **MIP solver** selects data sites to minimize latency
- 500 access sets

---

**Sample access set**

---

**Measure gap**
Pando is Close to the Lower Bound

CDF across access sets

Volume of Gap to the Lower Bound (lower is better)

Pando > EPaxos > RS-Paxos

13x higher

11x higher
Pando is Close to the Lower Bound

CDF across access sets

Volume of Gap to the Lower Bound (lower is better)
Pando is Close to the Lower Bound

Potential for true 1-round writes

8x higher

13x higher

11x higher
Pando is Close to the Lower Bound

- Cloud deployment confirms solver latency estimates
- Up to 46% cost ($) savings
Conclusion

- Pando: linearizability across geo-distributed DCs
- Achieves a near-optimal read–write–storage trade-off
  - Allow for erasure-code data to minimize cost
  - Rethink how to use Paxos in the wide-area setting
Backup Slides
Deployment Latency

95\%ile Read Latency (ms) per Front-end

GL

NA–AS

NA–EU

NA

Pan
EP
RSP
Latency Under Conflicts

![Graph showing latency under conflicts for different numbers of front-ends issuing requests. The graph compares Pando and Pando without leader fallback.](image)

- **Latency for successful writes**
  - X-axis: Number of front-ends issuing requests (1, 2, 4, 8, 16)
  - Y-axis: Time in milliseconds (300ms, 1s, 3s, 10s, 30s)

**Legend**
- Pando
- Pando w/o leader fallback
Contributions of Each Technique

CDF across access sets

GapVolume (lower is better)
Throughput

![Throughput graph]

- **Throughput** using 3 replicas
- **Read**
  - 1KB
  - 10KB
  - 100KB
- **Write**
  - 1KB
  - 10KB
  - 100KB

Colors:
- **k=2,r=3** (Dark purple)
- **k=2,r=4** (Yellow)

**Note:** The graph shows the throughput for different file sizes (1KB, 10KB, 100KB) for read and write operations with two different configurations (k=2,r=3 and k=2,r=4). The throughput is measured in units that are not specified in the image.
Read Latency After Failure

![Graph showing CDF across access sets and max read latency across front-ends (ms) for Pando and EPaxos with and without failure.](image-url)