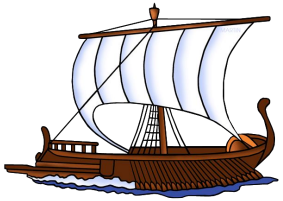
THE UNIVERSITY of EDINBURGH
informatics

Odyssey: The Impact of Modern Hardware on Strongly-Consistent Replication Protocols

Vasilis Gavrielatos, Antonios Katsarakis, Vijay Nagarajan



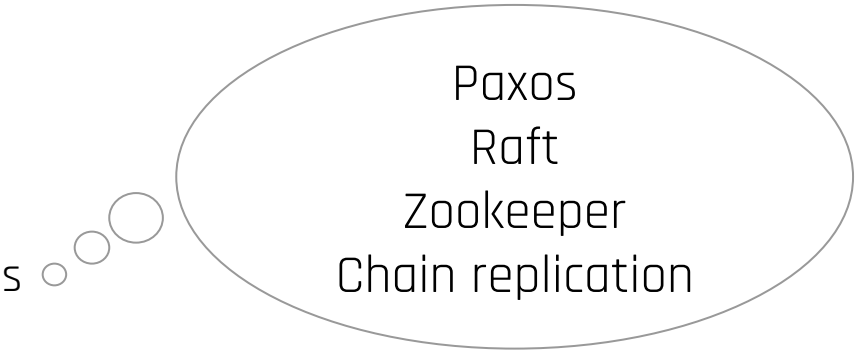
Thanks to: EPSRC Pioneering research and skills, arm, HUAWEI

Scope

-- Replication protocols
-- Strong consistency
-- Replicated KVS
-- Read/Write API
-- Datacenter

Scope

-- Replication protocols
-- Strong consistency
-- Replicated KVS
-- Read/Write API
-- Datacenter



Scope

-- Replication protocols
-- Strong consistency
-- Replicated KVS
-- Read/Write API
-- Datacenter

Old Hardware

-- Single-thread
-- Slow Disk
-- Slow network

Scope

-- Replication protocols
-- Strong consistency
-- Replicated KVS
-- Read/Write API
-- Datacenter

~~Old~~ Modern Hardware

-- ~~Single thread~~ Manycore
-- ~~Slow Disk~~ Big Memories
-- ~~Slow network~~ Fast Networks

Scope

-- Replication protocols
-- Strong consistency
-- Replicated KVS
-- Read/Write API
-- Datacenter

~~Old~~ Modern Hardware

-- ~~Single-thread~~ Manycore
-- ~~Slow Disk~~ Big Memories
-- ~~Slow network~~ Fast Networks

Scope

-- Replication protocols
-- Strong consistency
-- Replicated KVS
-- Read/Write API
-- Datacenter

~~Old~~ Modern Hardware

-- ~~Single-thread~~ Manycore
-- ~~Slow Disk~~ Big Memories
-- ~~Slow network~~ Fast Networks

Scope

-- Replication protocols
-- Strong consistency
-- Replicated KVS
-- Read/Write API
-- Datacenter

~~Old~~ Modern Hardware

-- ~~Single-thread~~ Manycore
-- ~~Slow Disk~~ Big Memories
-- ~~Slow network~~ Fast Networks

Scope

-- Replication protocols
-- Strong consistency
-- Replicated KVS
-- Read/Write API
-- Datacenter

~~Old~~ Modern Hardware

-- ~~Single-thread~~ Manycore
-- ~~Slow Disk~~ Big Memories
-- ~~Slow network~~ Fast Networks

Modern Hardware challenges conventional wisdom

How do protocols perform over modern hardware?

How do protocols perform over modern hardware?

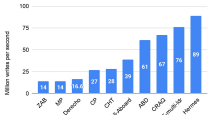
What are the best practices?

Taxonomy

Leader-based Total Order	Leader-based Per-key Order
Decentralized Total Order	Decentralized Per-key Order

Odyssey Framework

Design Space Characterization

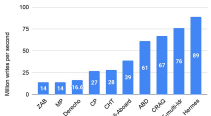


Taxonomy

Leader-based Total Order	Leader-based Per-key Order
Decentralized Total Order	Decentralized Per-key Order

Odyssey Framework

Design Space Characterization

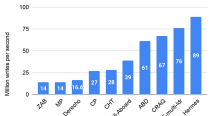


Taxonomy

Leader-based Total Order	Leader-based Per-key Order
Decentralized Total Order	Decentralized Per-key Order

Odyssey Framework

Design Space Characterization

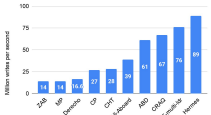


Taxonomy

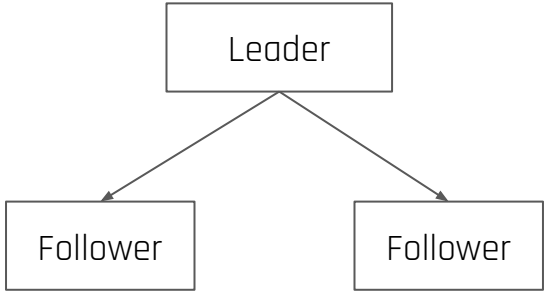
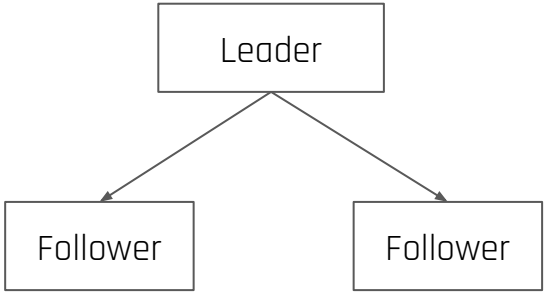
Leader-based Total Order	Leader-based Per-key Order
Decentralized Total Order	Decentralized Per-key Order

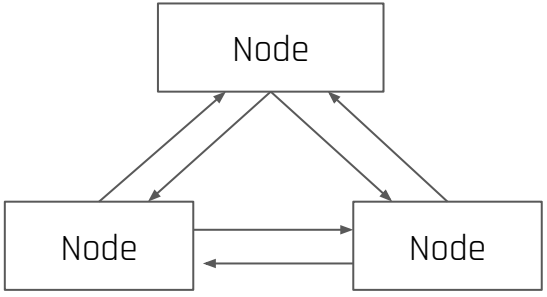
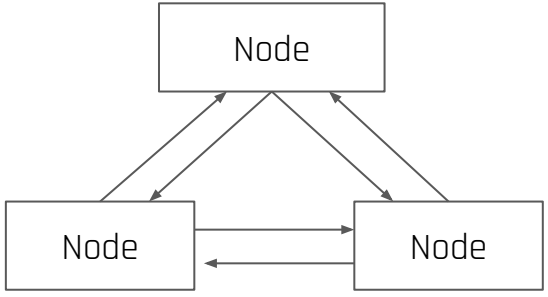
Odyssey Framework

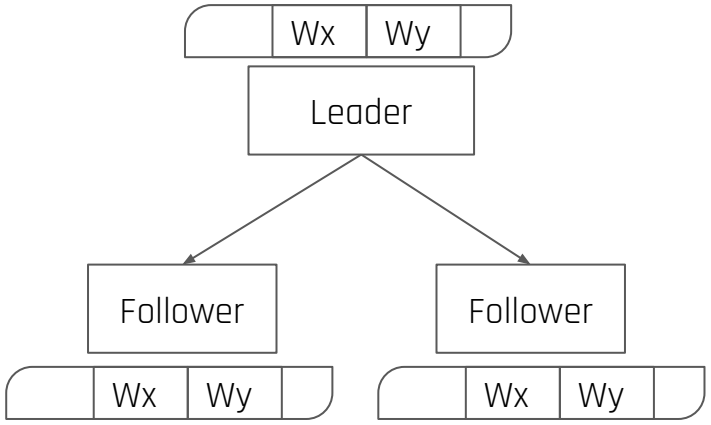
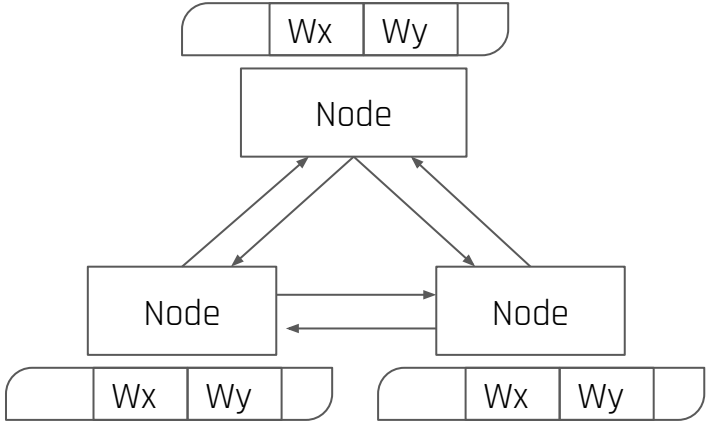
Design Space Characterization

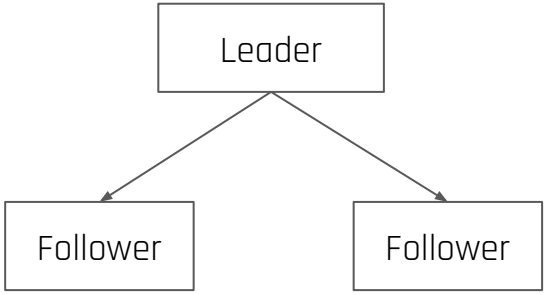
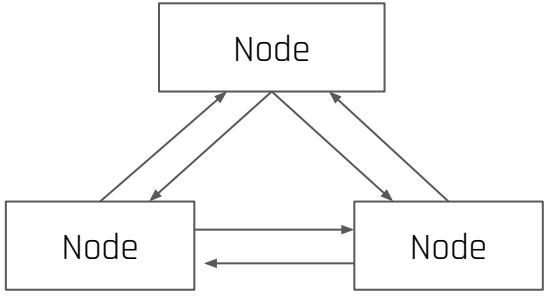


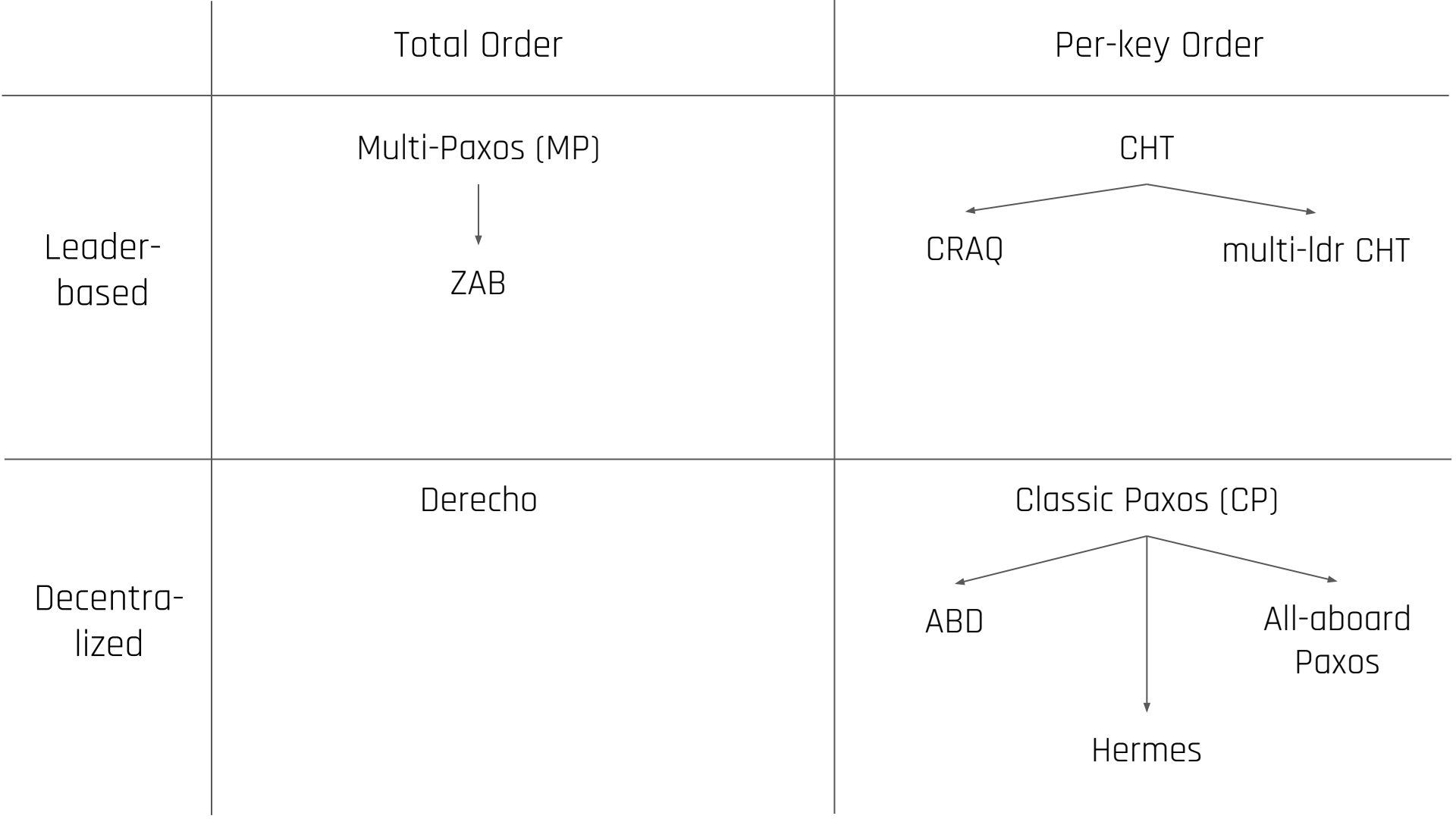
	Total Order	Per-key Order
Leader-based		
Decentra-lized		

	Total Order	Per-key Order
Leader-based		
Decentra-lized		

	Total Order	Per-key Order
Leader-based		
Decentra-lized		

	Total Order	Per-key Order
Leader-based		
Decentra-lized		

	Total Order	Per-key Order
Leader-based		
Decentra-lized		

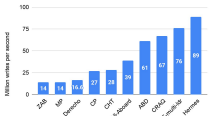

Total Order
Per-key Order
Leader-based
Multi-Paxos (MP)
ZAB
CHT
CRAQ
multi-ldr CHT
Decentra-lized
Derecho
Classic Paxos (CP)
ABD
All-aboard Paxos
Hermes

Taxonomy

Leader-based Total Order	Leader-based Per-key Order
Decentralized Total Order	Decentralized Per-key Order

Odyssey Framework

Design Space Characterization



Servers	5
Hardware threads	40
Network Bandwidth	56 Gbps (RDMA)
KVS	MICA
Key-value size	48 B

General Directives.

- Prioritize thread-scalability, then load-balance and then the work-per-request ratio.·Total order should be avoided in read/write systems.
- Leader-based protocols can achieve high-performance, butcare must be taken to ensure load balance.
- It is worth investing in the hardware multicast primitive only in the case of LPKO protocols.
- Local reads can deliver great performance, but it's not guaranteed.
- In order to minimize latency, choose protocols with high throughput.

Recommendations

- All-aboard is the most attractive design point for a scenario where: 1) availability is the most important concern and 2) conditional writes are required.
- If simple writes will do, then we recommend ABD.
- If a small window of unavailability on a failure is tolerable,then Hermes is the best candidate, while CHT-multi-ldr and CRAQ are good alternatives.

General Directives.

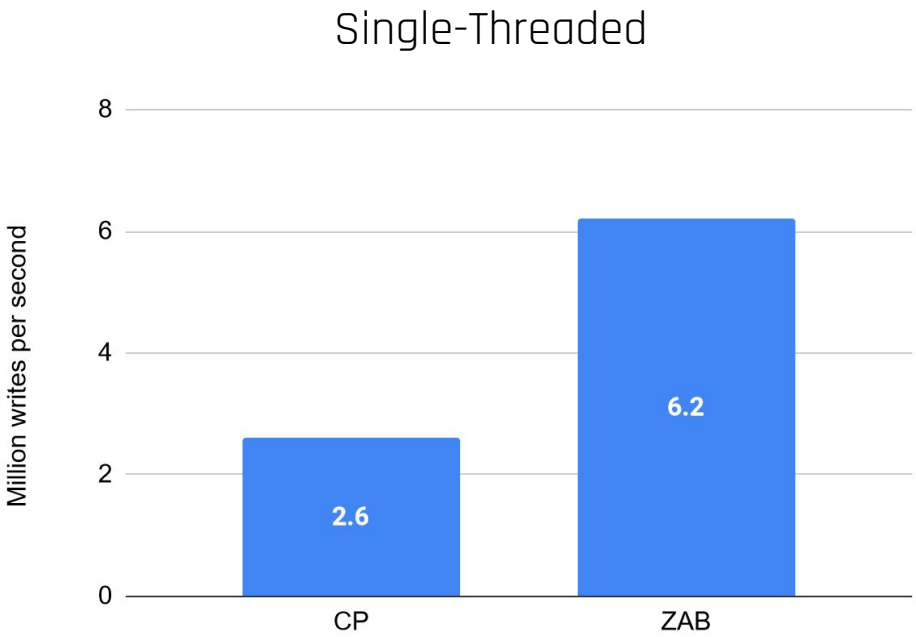
- Prioritize thread-scalability, then load-balance and then the work-per-request ratio.·Total order should be avoided in read/write systems.
- Leader-based protocols can achieve high-performance, butcare must be taken to ensure load balance.

Total order should be avoided in read/write systems.

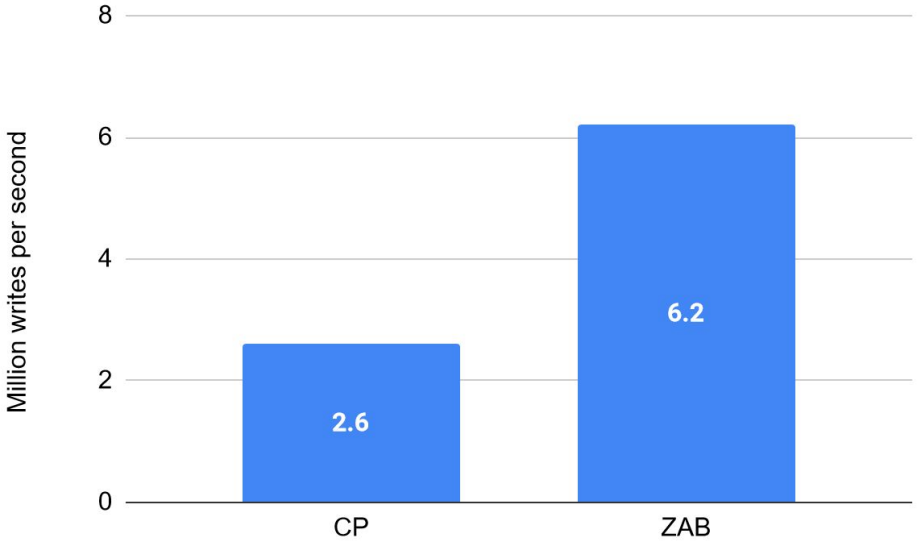
Recommendations

- All-aboard is the most attractive design point for a scenario where: 1) availability is the most important concern and 2) conditional writes are required.
- If simple writes will do, then we recommend ABD.
- If a small window of unavailability on a failure is tolerable,then Hermes is the best candidate, while CHT-multi-ldr and CRAQ are good alternatives.

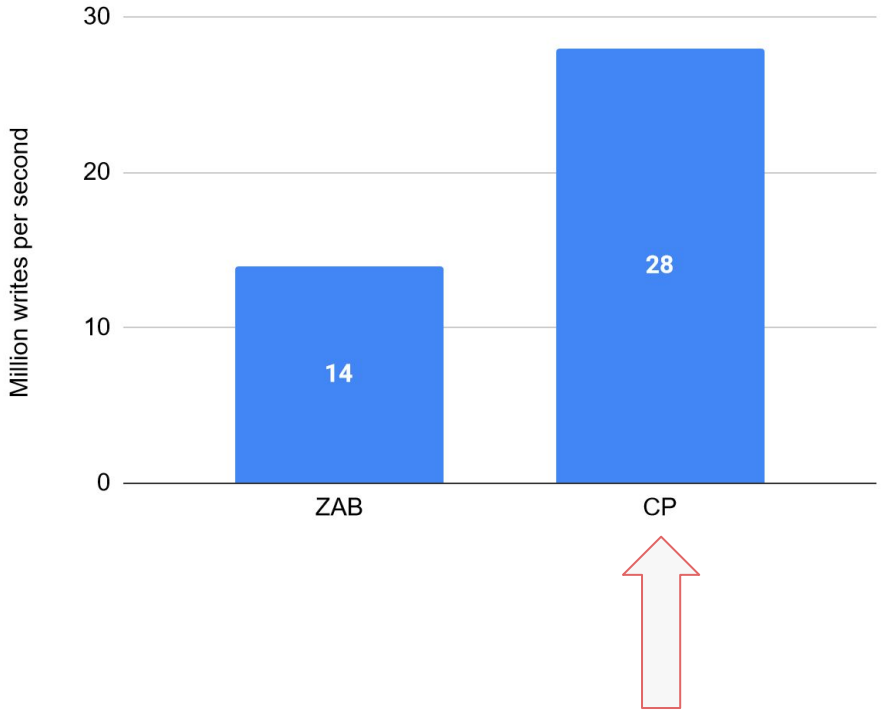
	Total Order	Per-key Order
Leader-based	Multi-Paxos (MP) ↓ ZAB [Hunt et al. ATC'10]	CHT CRAQ multi-ldr CHT
Decentra-lized	Derecho	Classic Paxos (CP) ABD All-aboard Paxos Hermes


Single-Threaded
Million writes per second
8
6
4
2
0
2.6
6.2
CP
ZAB

Single-Threaded

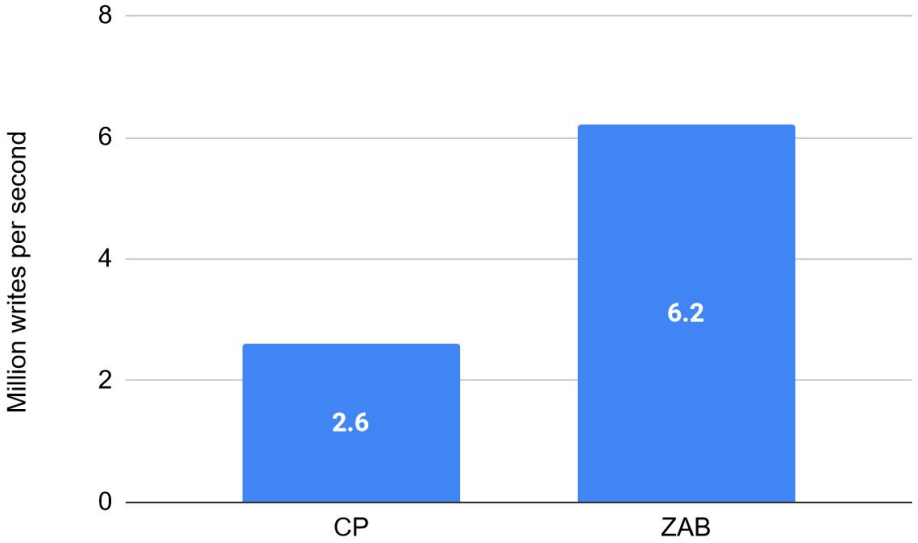


Multi-Threaded

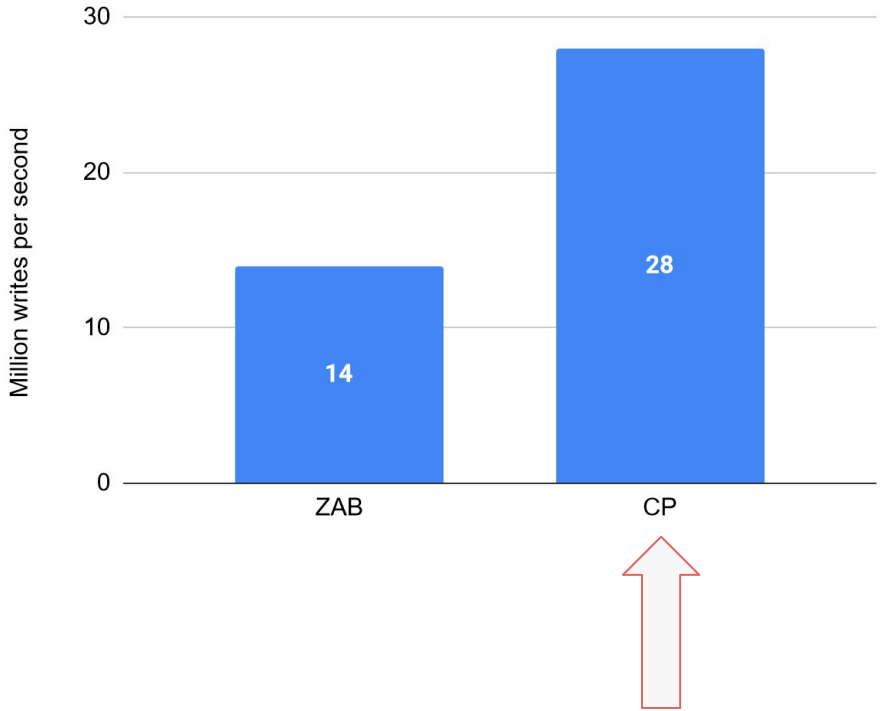


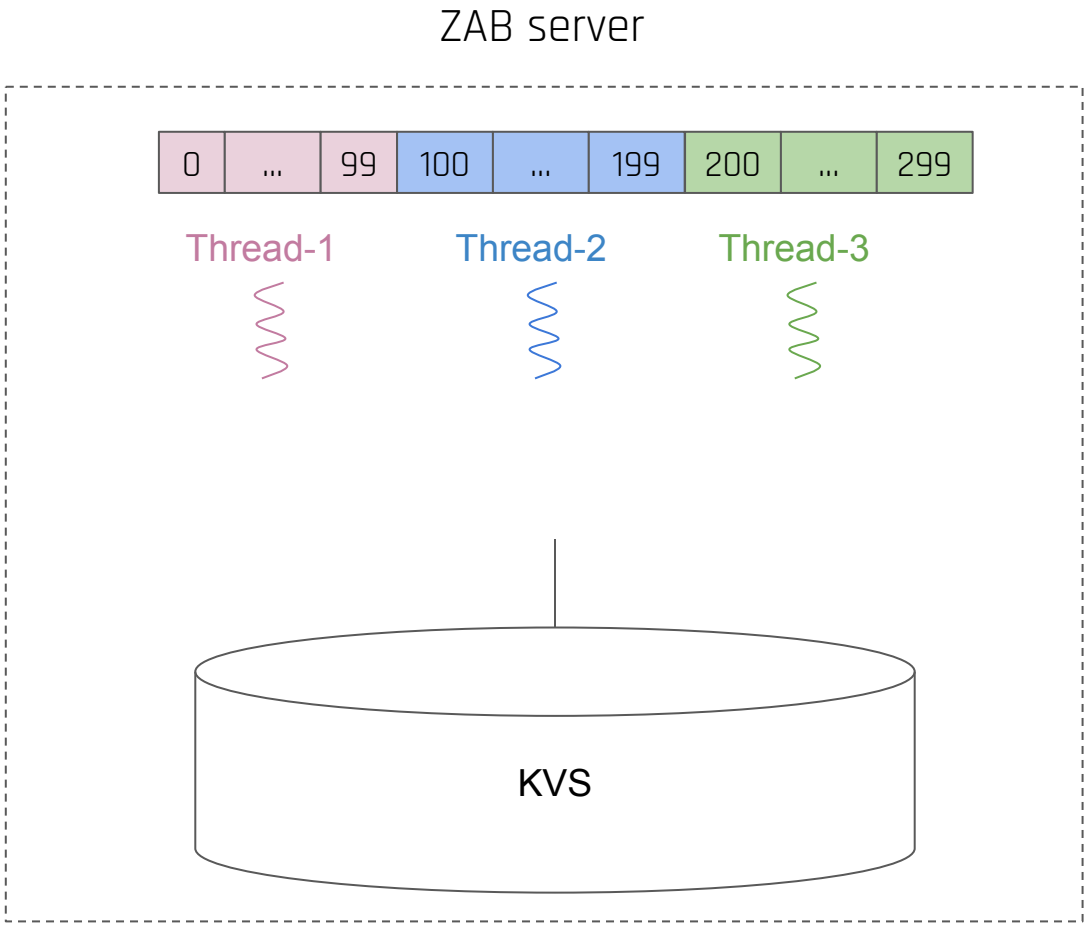

Extending Classic Paxos for High-performance Read-Modify-Write Registers
--All-aboard [Howard's thesis 2019]
--ABD reads/writes -- Carstamps [Burke NSDI '20]
https://arxiv.org/abs/2103.14701

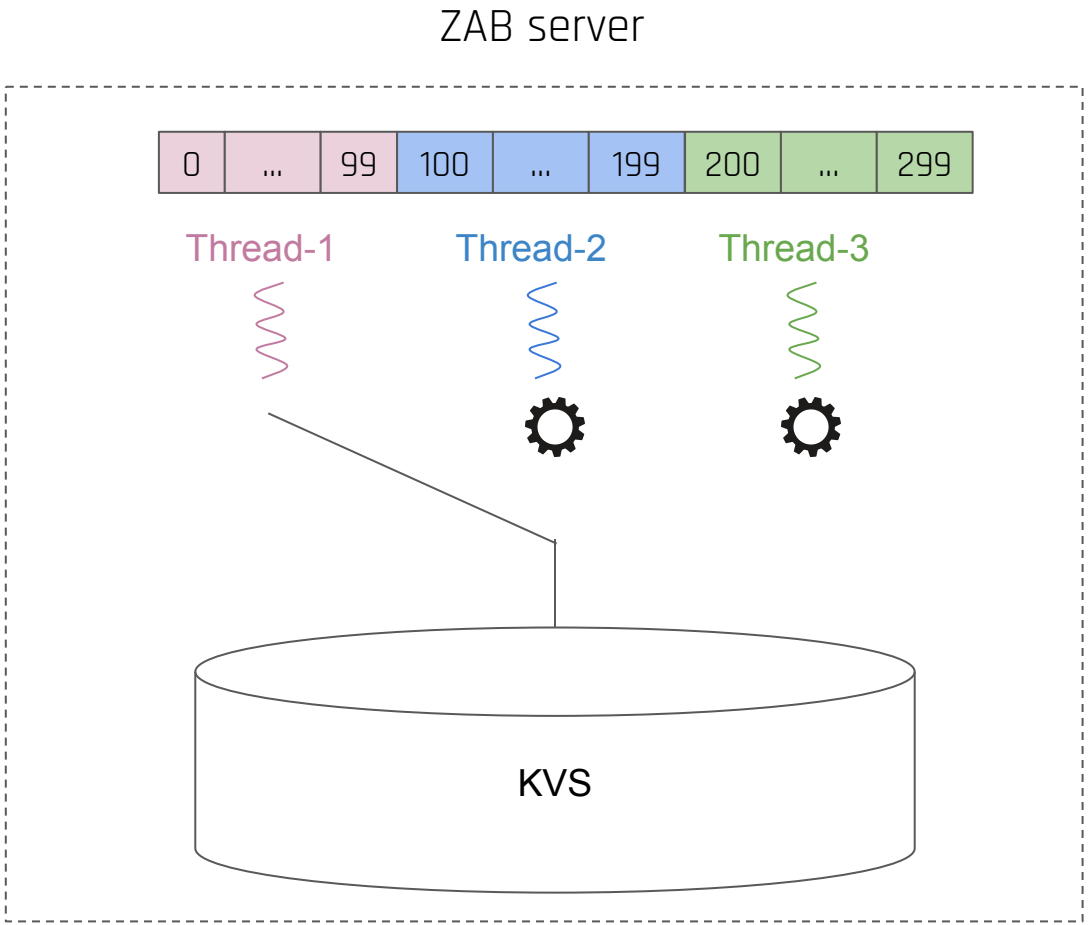
Single-Threaded

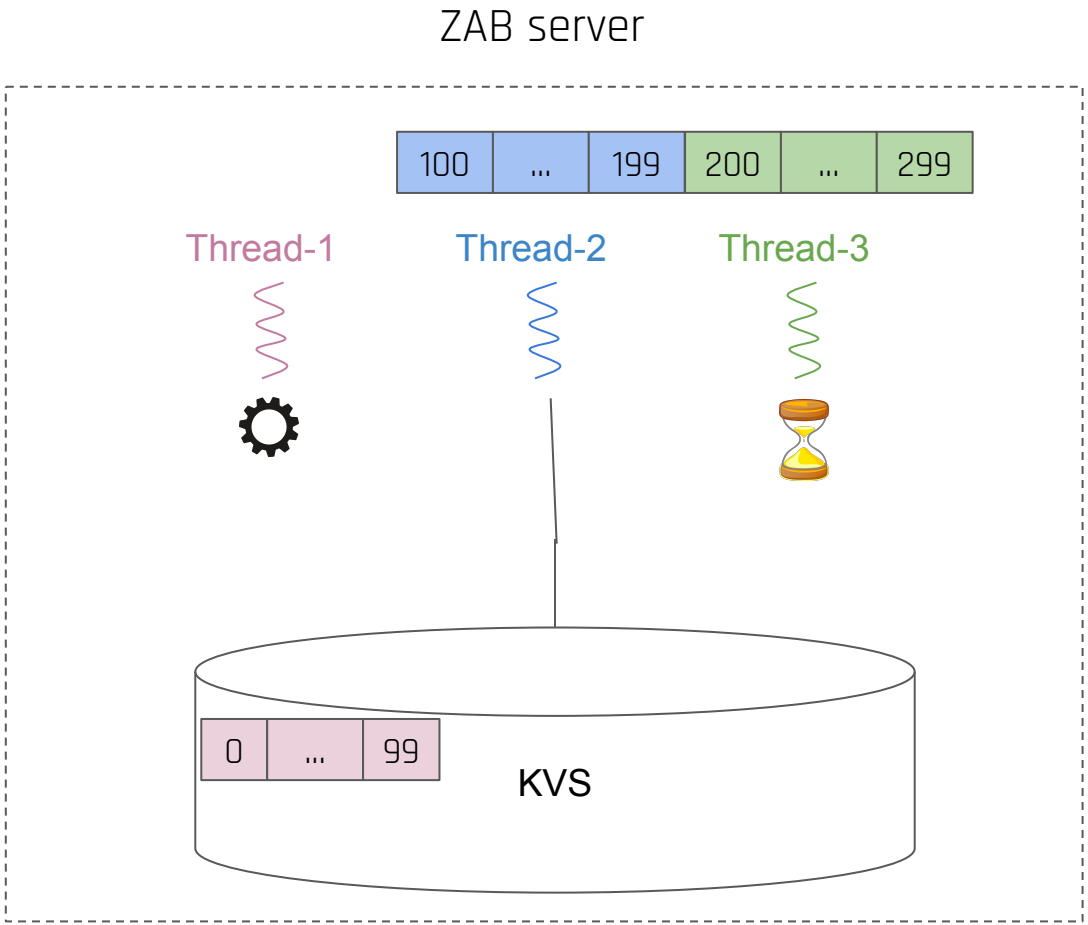


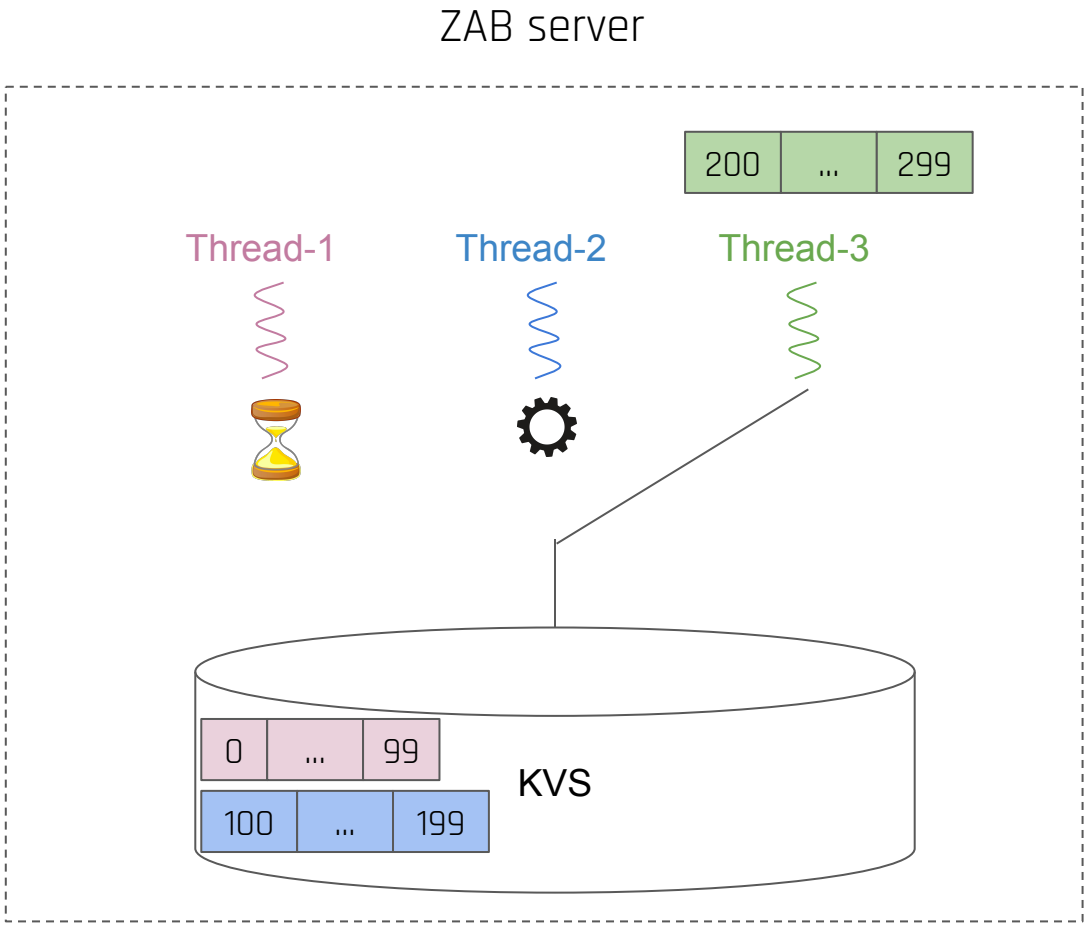
Multi-Threaded

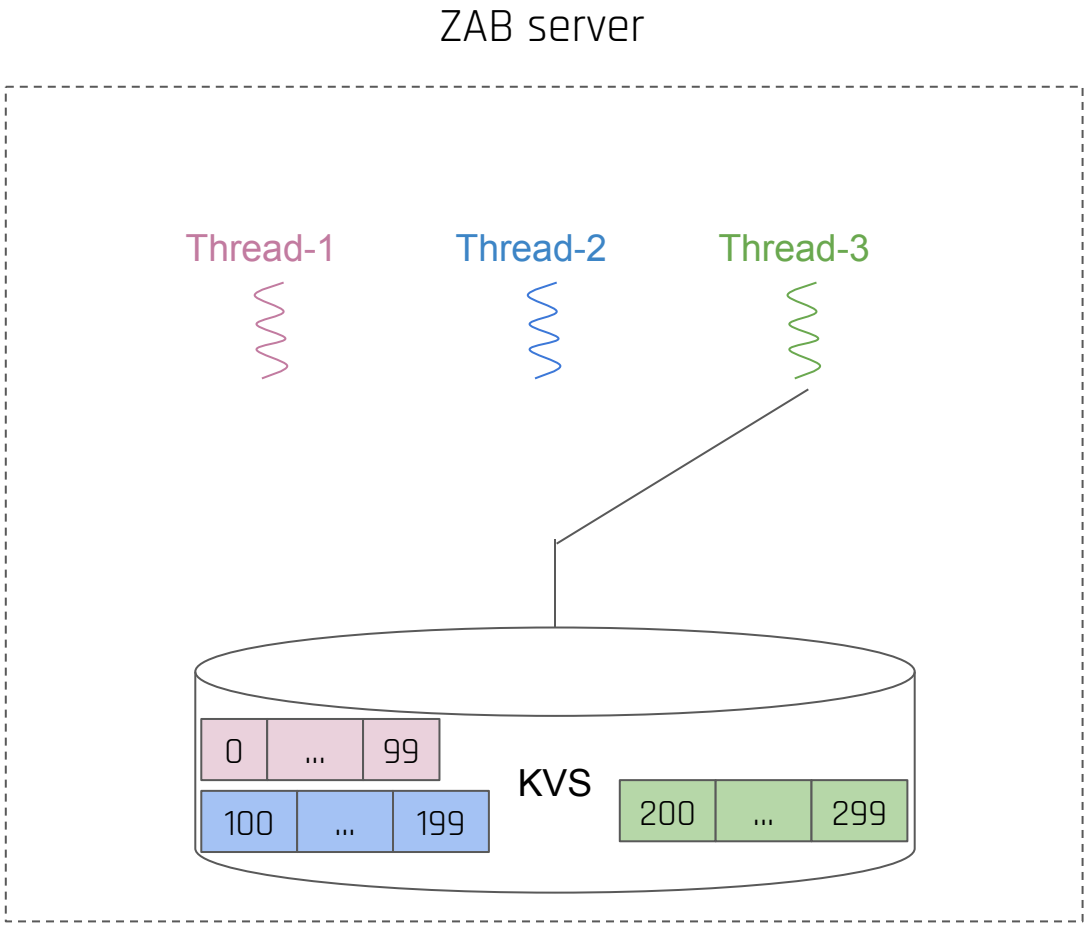


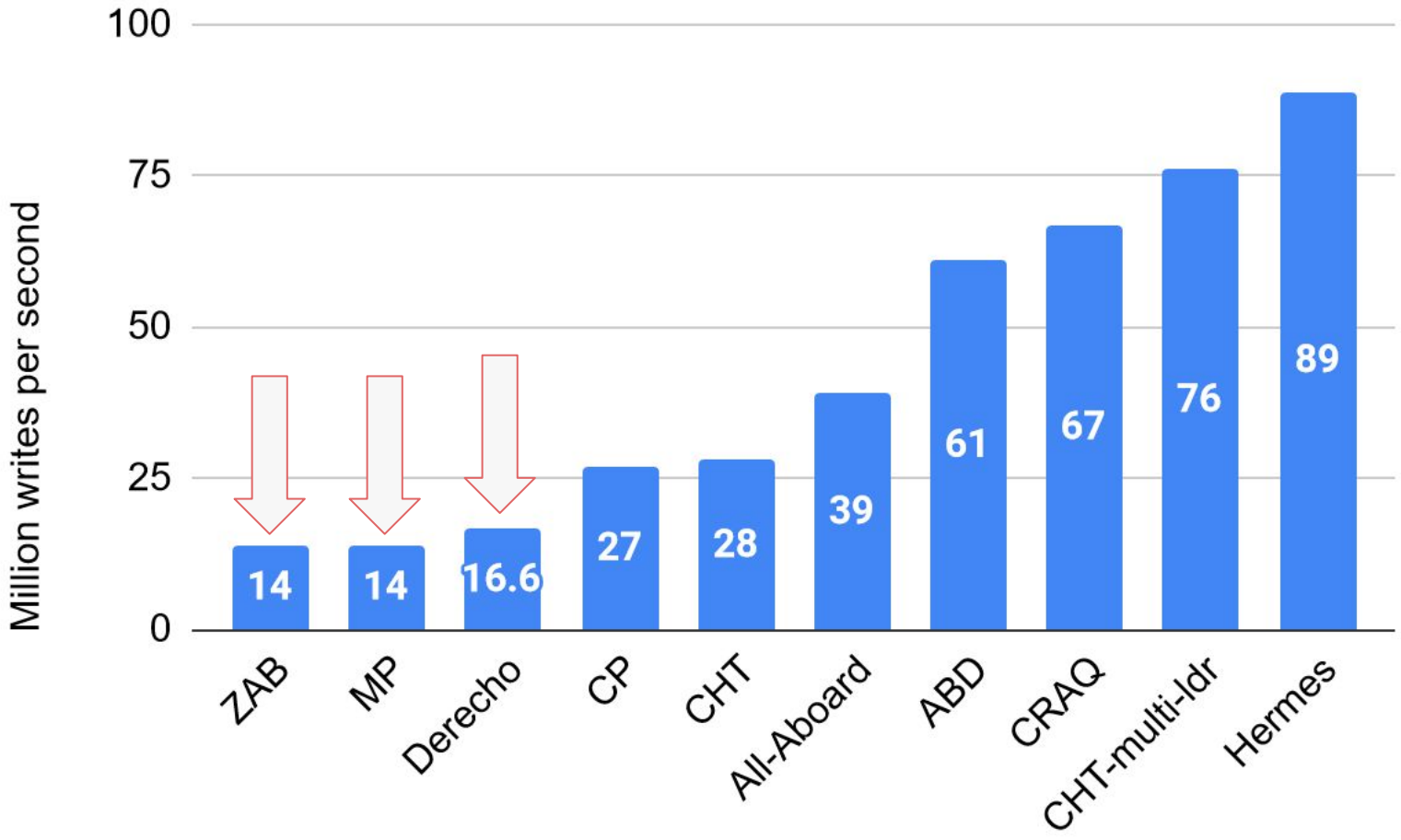

ZAB server
0
...
99
100
...
199
200
...
299
Thread-1
Thread-2
Thread-3
KVS

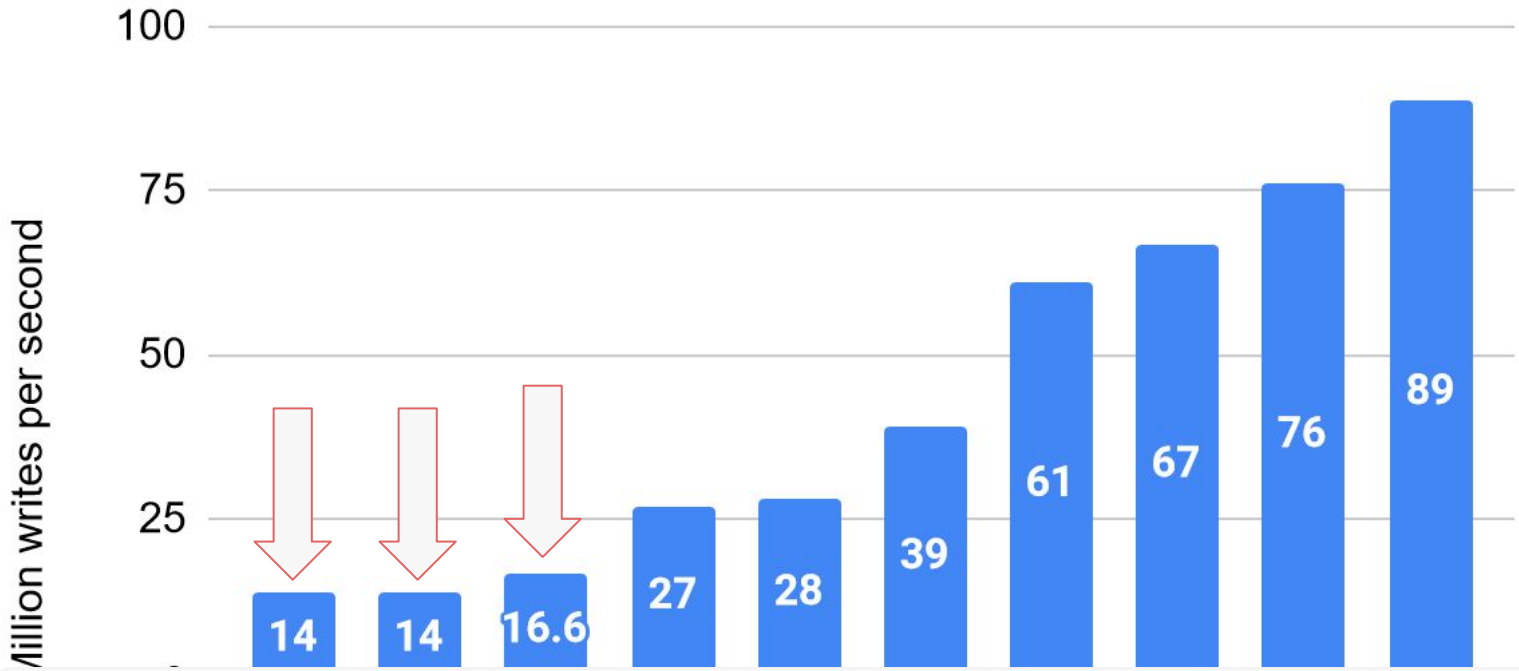

ZAB server
0
...
99
100
...
199
200
...
299
Thread-1
Thread-2
Thread-3
KVS


ZAB server
100
...
199
200
...
299
Thread-1
Thread-2
Thread-3
0
...
99
KVS


ZAB server
200
...
299
Thread-1
Thread-2
Thread-3
0
...
99
100
...
199
KVS


ZAB server
Thread-1
Thread-2
Thread-3
0
...
99
100
...
199
KVS
200
...
299


Million writes per second
100
75
50
25
0
14
14
16.6
27
28
39
61
67
76
89
ZAB
MP
Derecho
CP
CHT
All-Aboard
ABD
CRAQ
CHT-multi-ldr
Hermes



Total-order should be avoided

General Directives.

- Prioritize thread-scalability, then load-balance and then the work-per-request ratio.‣Total order should be avoided in read/write systems.
- Leader-based protocols can achieve high-performance, butcare must be taken to ensure load balance.

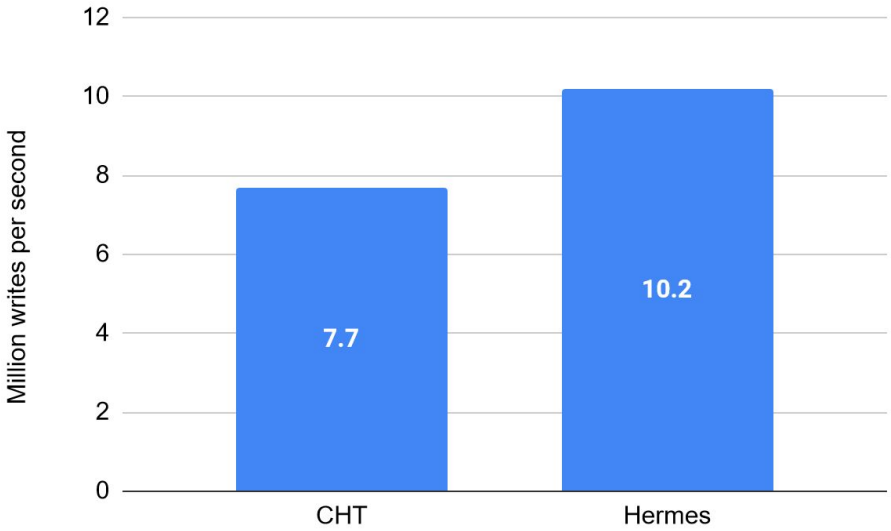
Leader-based protocols can achieve high-performance

Recommendations

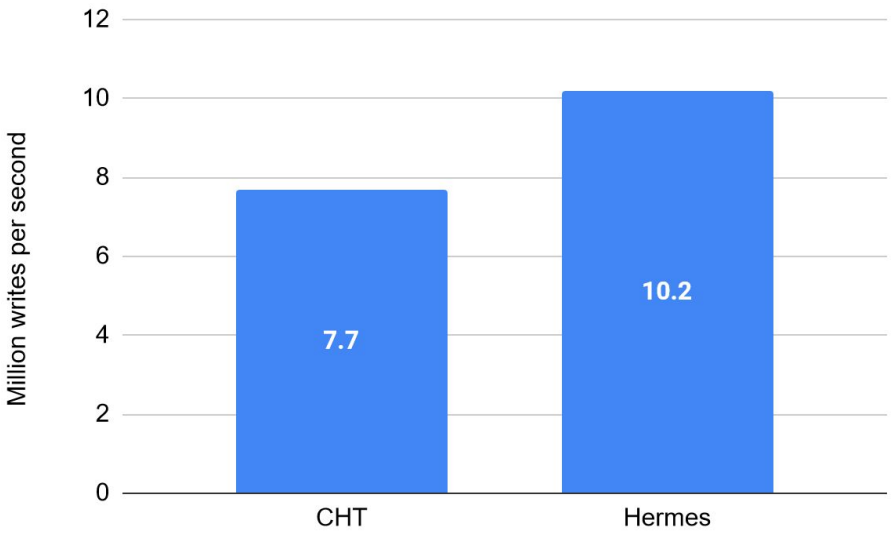
- All-aboard is the most attractive design point for a scenario where: 1) availability is the most important concern and 2) conditional writes are required.
- If simple writes will do, then we recommend ABD.
- If a small window of unavailability on a failure is tolerable,then Hermes is the best candidate, while CHT-multi-ldr and CRAQ are good alternatives.

	Total Order	Per-key Order
Leader-based	Multi-Paxos (MP) → ZAB	CHT [Chandra et al. PODC '16] → CRAQ, multi-ldr CHT
Decentra-lized	Derecho	Classic Paxos (CP) → ABD, All-aboard Paxos, Hermes

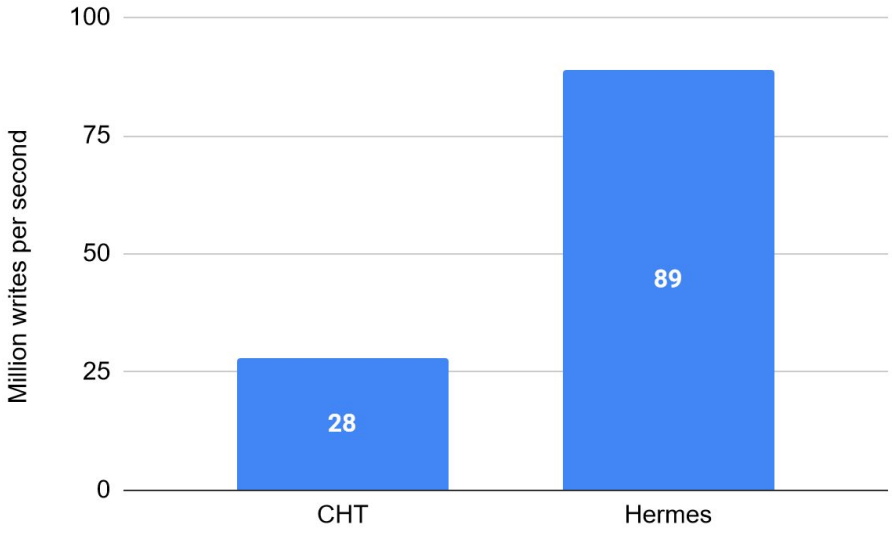
Single-threaded

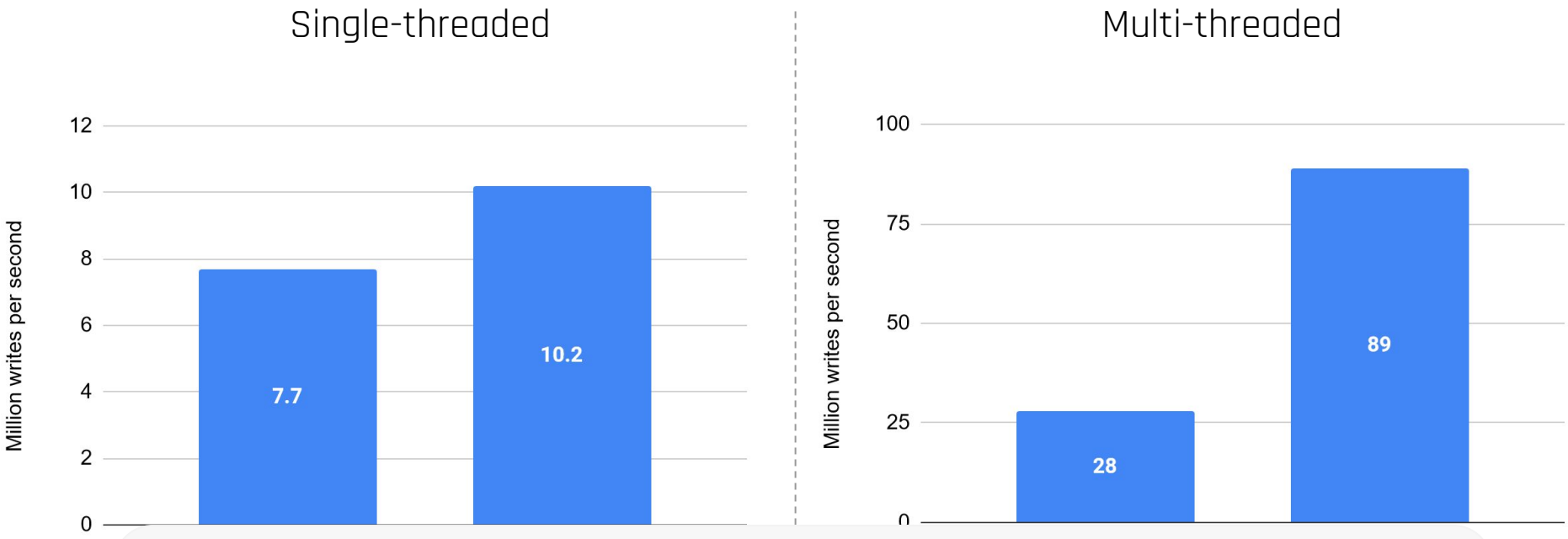


Single-threaded



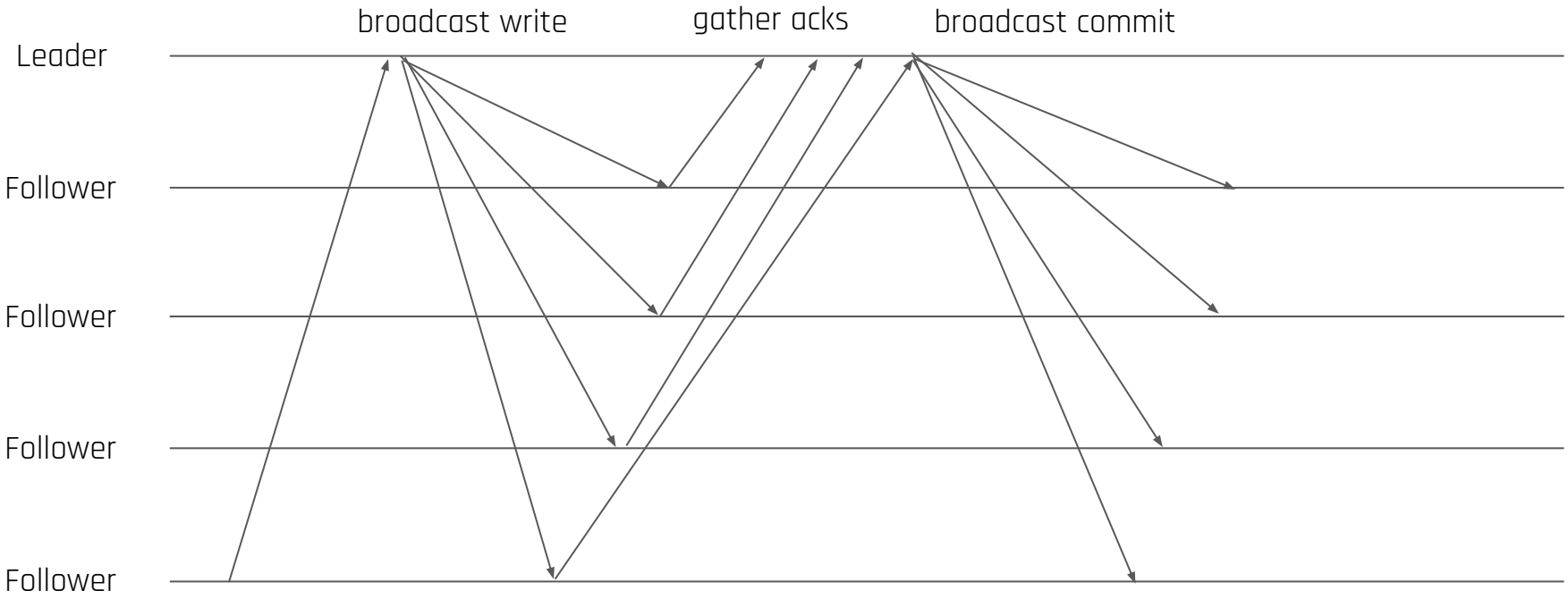
Multi-threaded



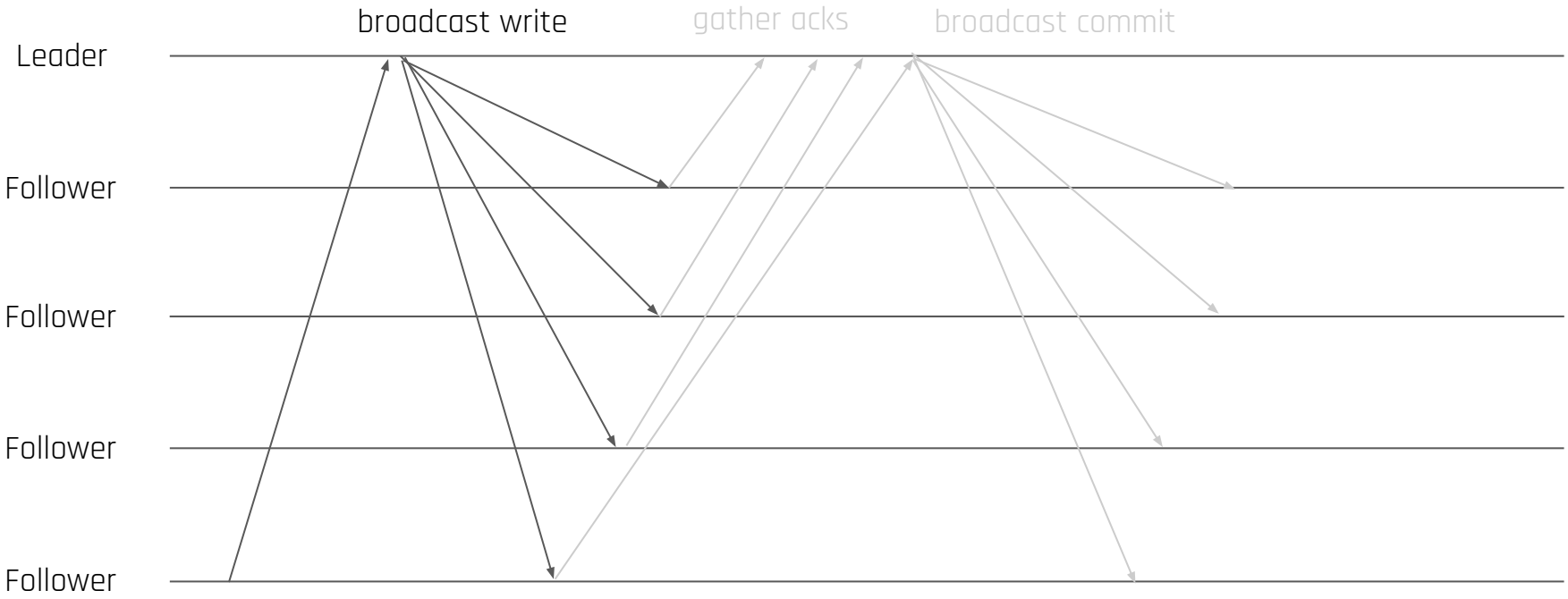


Bottleneck: Leader Send Bandwidth

CHT


broadcast write
gather acks
broadcast commit
Leader
Follower
Follower
Follower
Follower

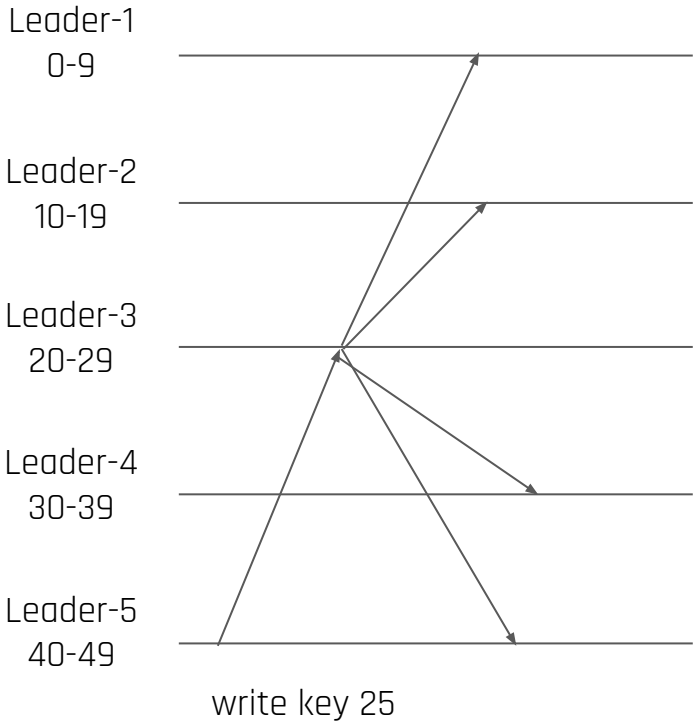
CHT


broadcast write
gather acks
broadcast commit
Leader
Follower
Follower
Follower
Follower

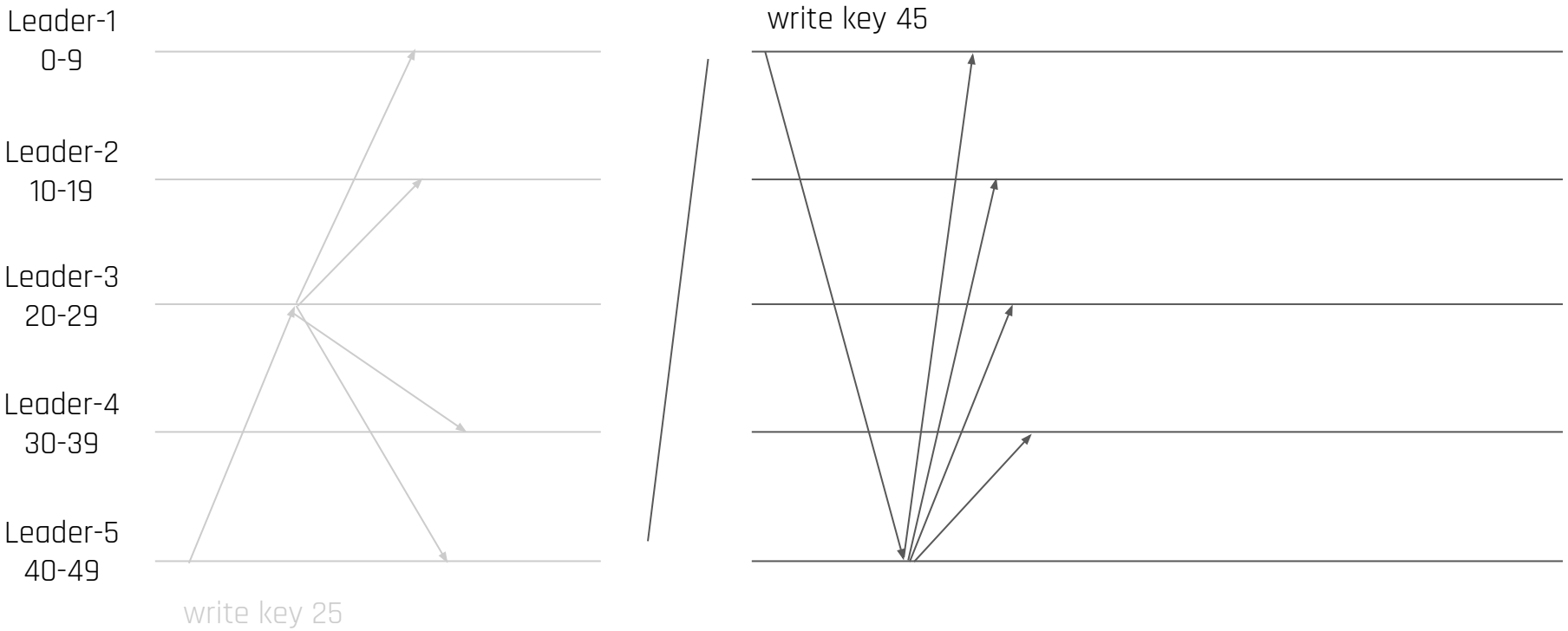
Two Protocol-level Solutions

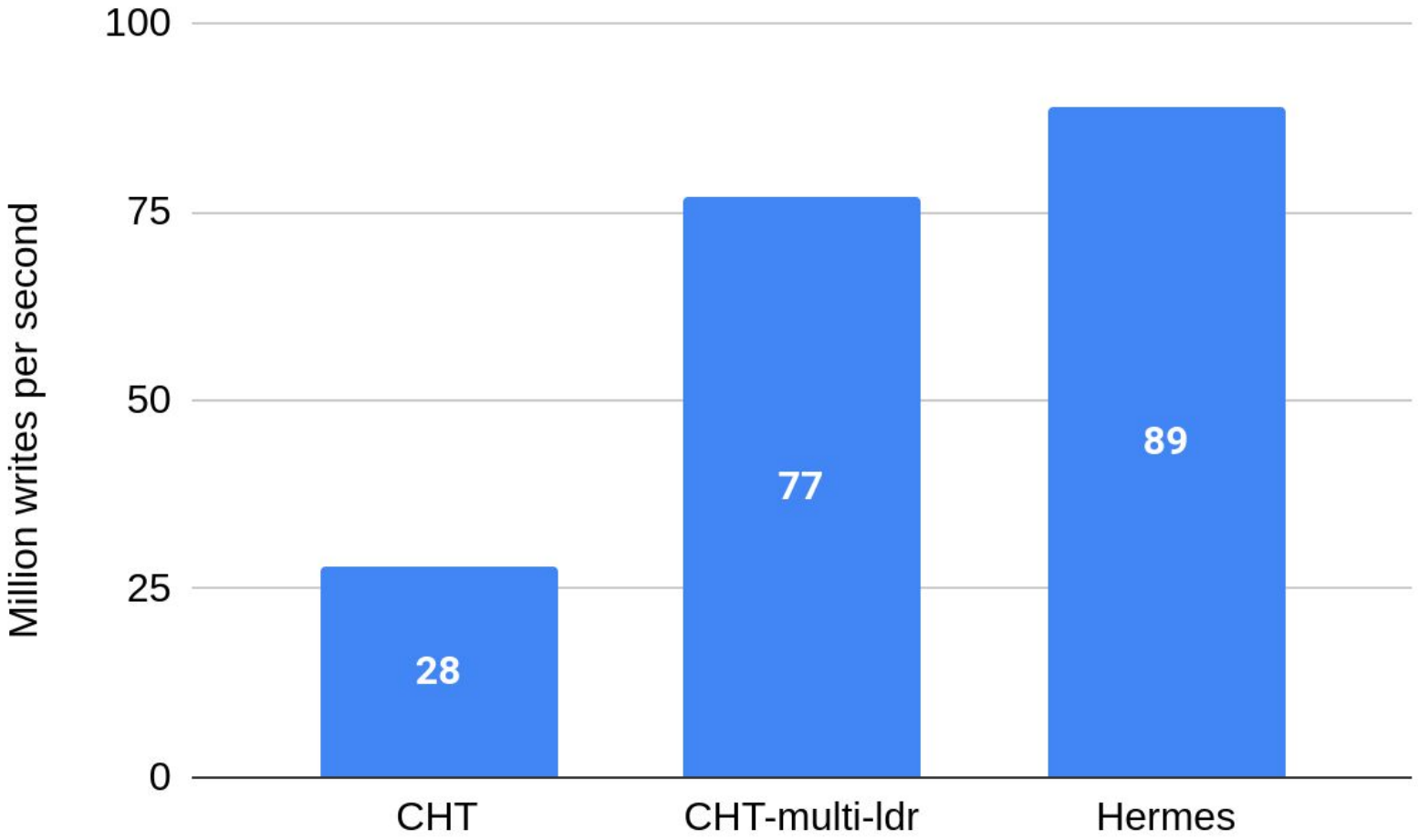
1. Multiple Leaders
2. Chain instead of broadcast

1. Multiple Leaders

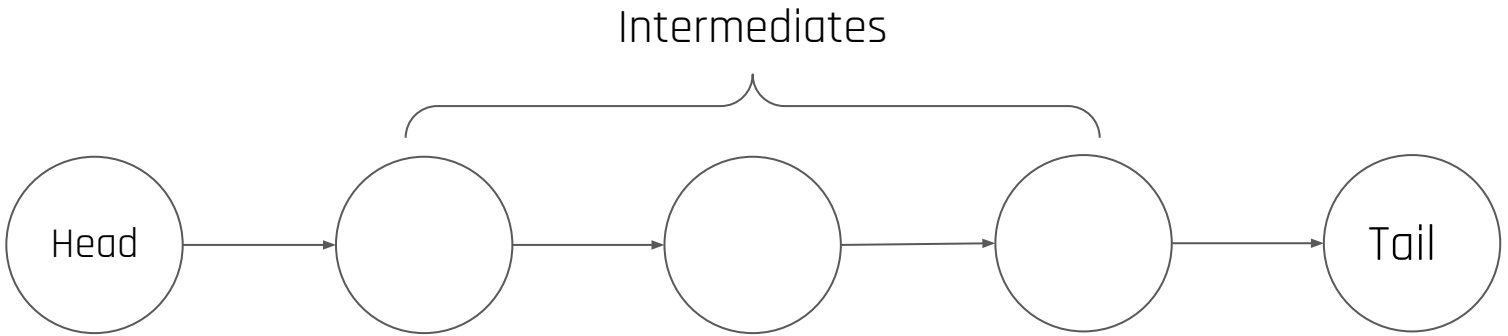


1. Multiple Leaders

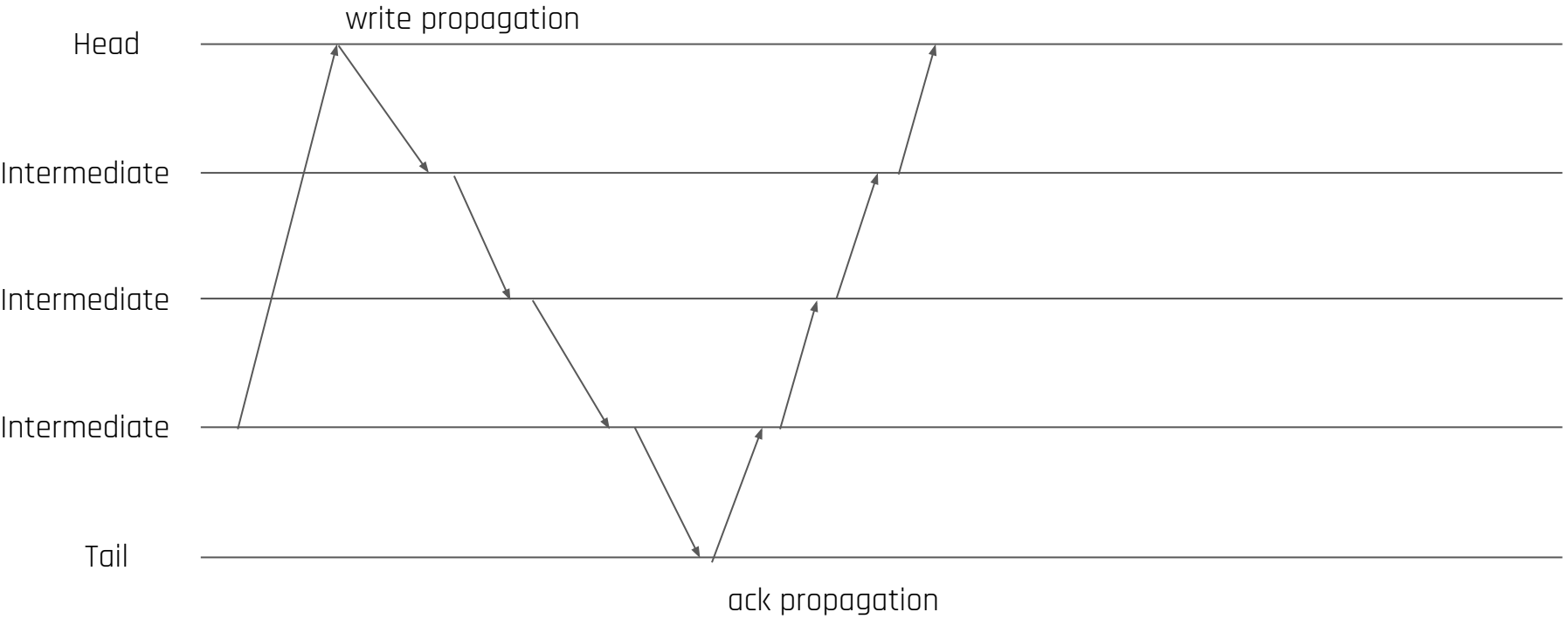


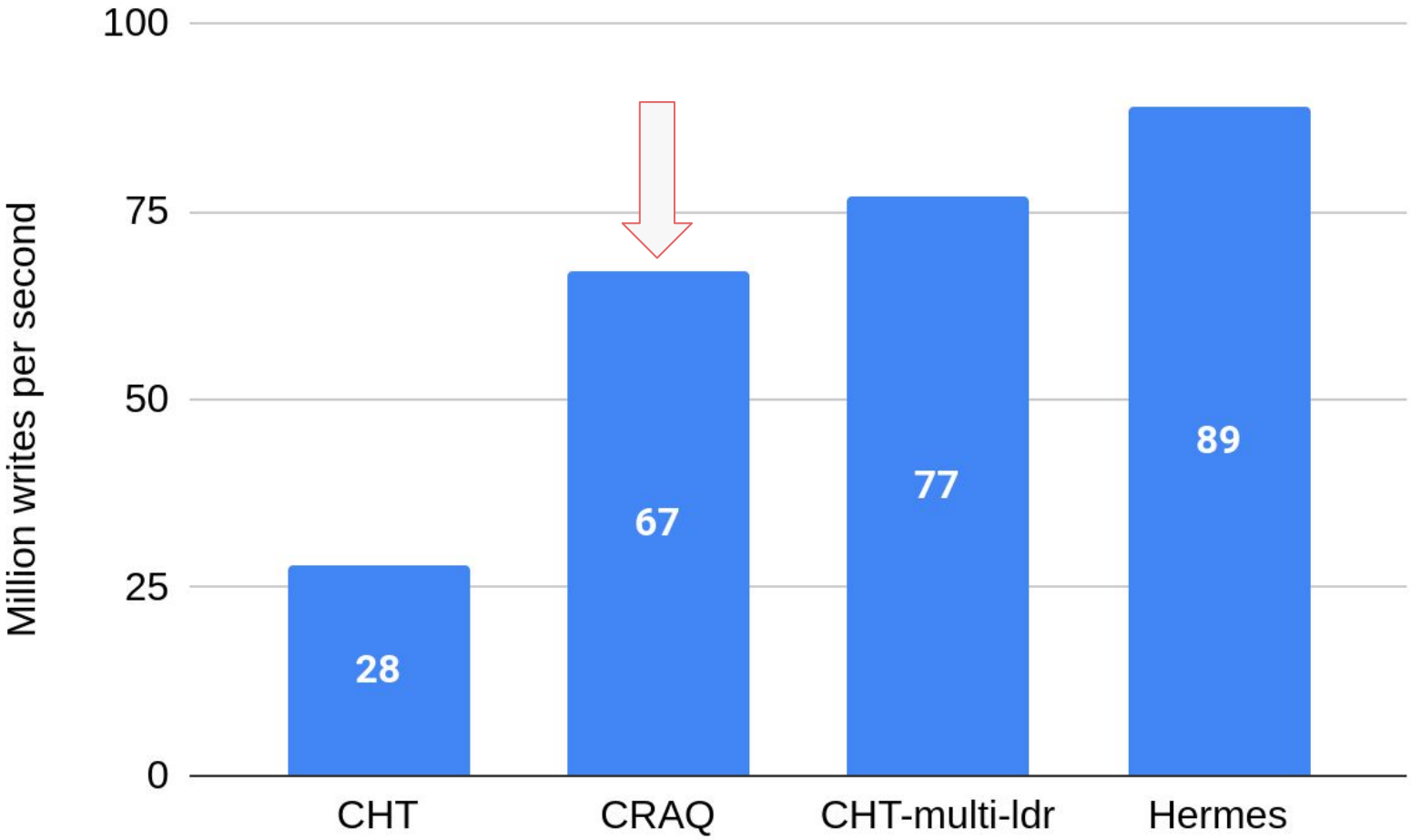

100
75
50
25
0
Million writes per second
28
77
89
CHT
CHT-multi-ldr
Hermes

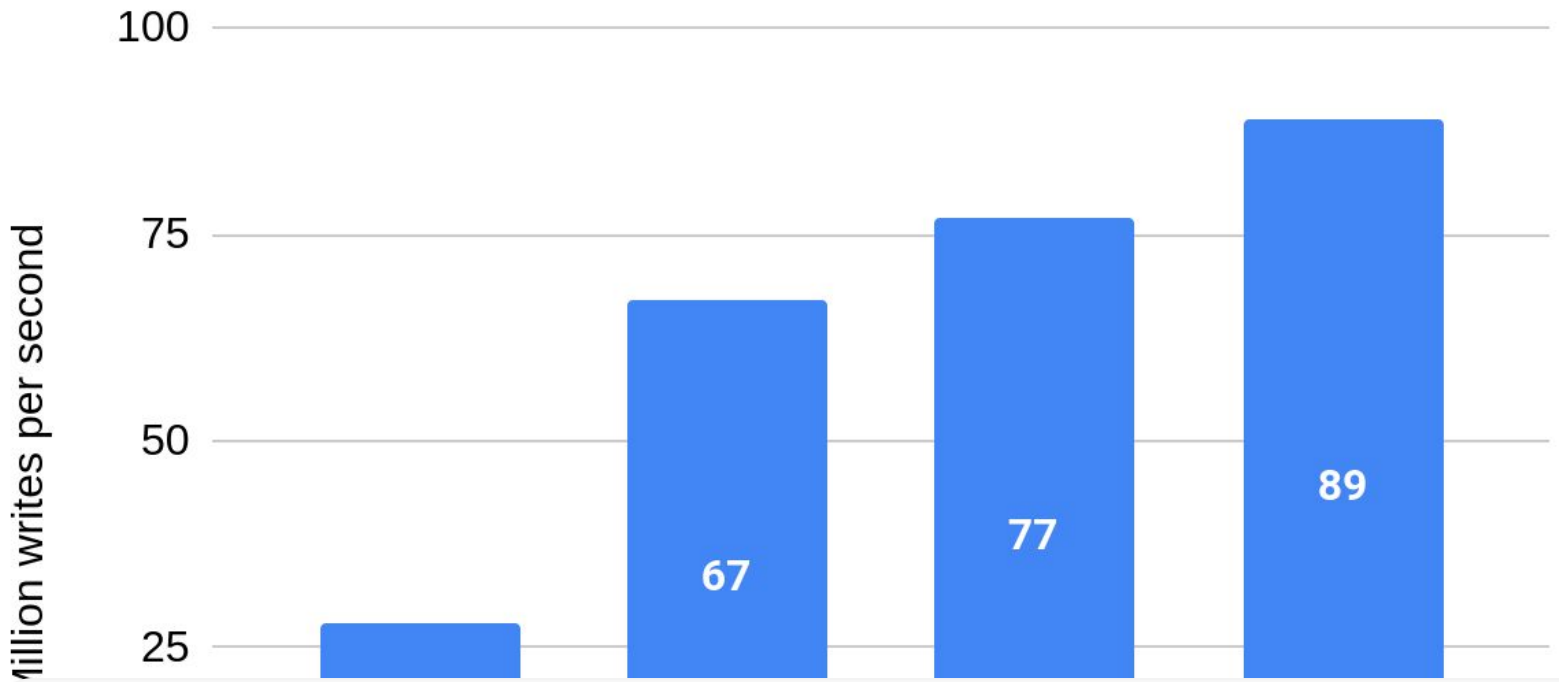
2. Chains instead of Broadcast



2. Chains instead of Broadcast




Million writes per second
100
75
50
25
0
28
67
77
89
CHT
CRAQ
CHT-multi-ldr
Hermes



Leader-based can achieve High-Performance

Odyssey Summary

Odyssey Summary

1. Taxonomy

Leader-based Total Order	Leader-based Per-key Order
Decentralized Total Order	Decentralized Per-key Order

   - Protocol selection

Odyssey Summary

1. Taxonomy

Leader-based Total Order	Leader-based Per-key Order
Decentralized Total Order	Decentralized Per-key Order

   - Protocol selection

2. Odyssey framework

   - Fast development of protocols
   - Apples-to-apples comparison

Odyssey Summary

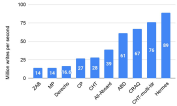
1. Taxonomy

   - Protocol selection

2. Odyssey framework

   - Fast development of protocols
   - Apples-to-apples comparison

3. Design space Characterization



   - Recommendations
   - Best practices