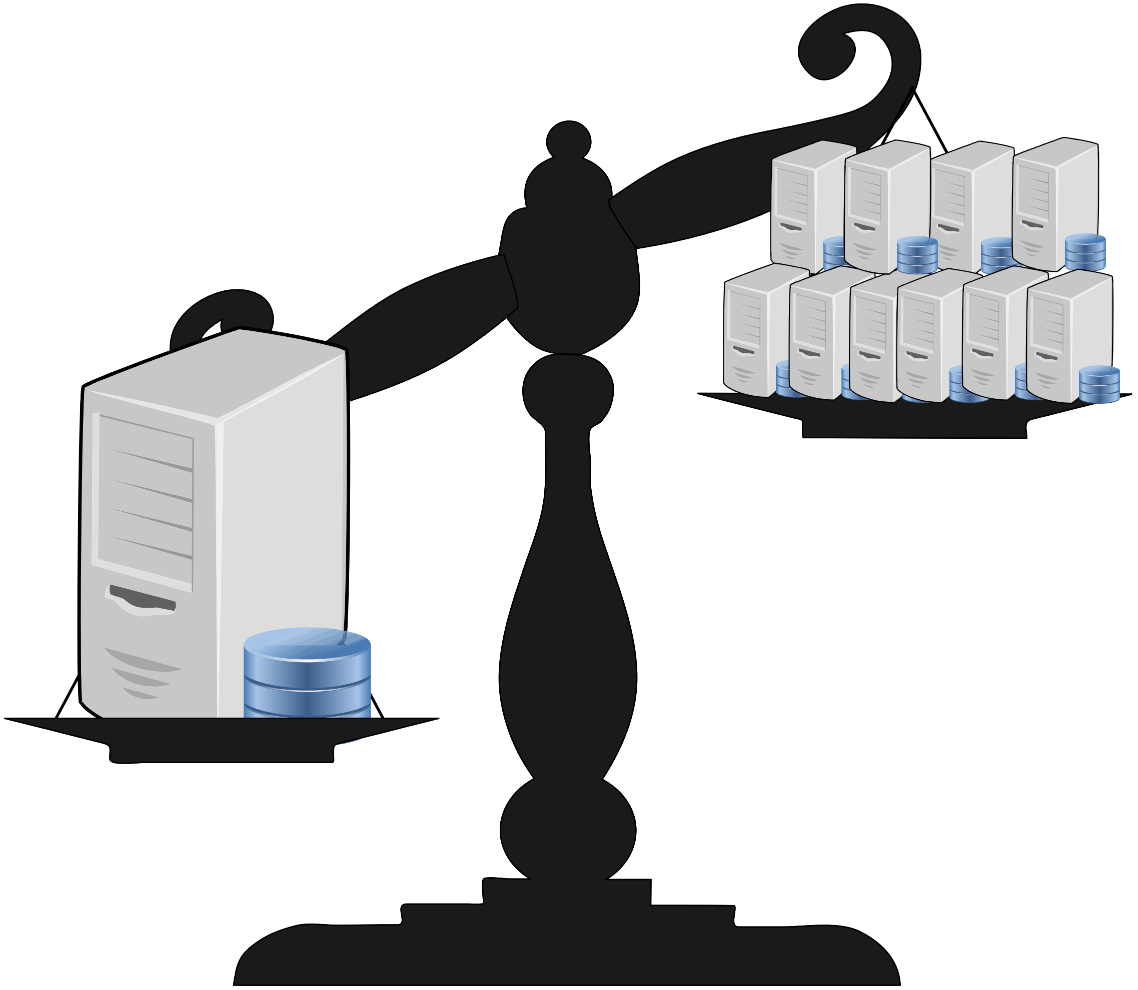
The End of a Myth: Distributed Transactions Can Scale

Erfan Zamanian, Carsten Binnig, Tim Harris, and Tim Kraska

Rafael Ketsetsides

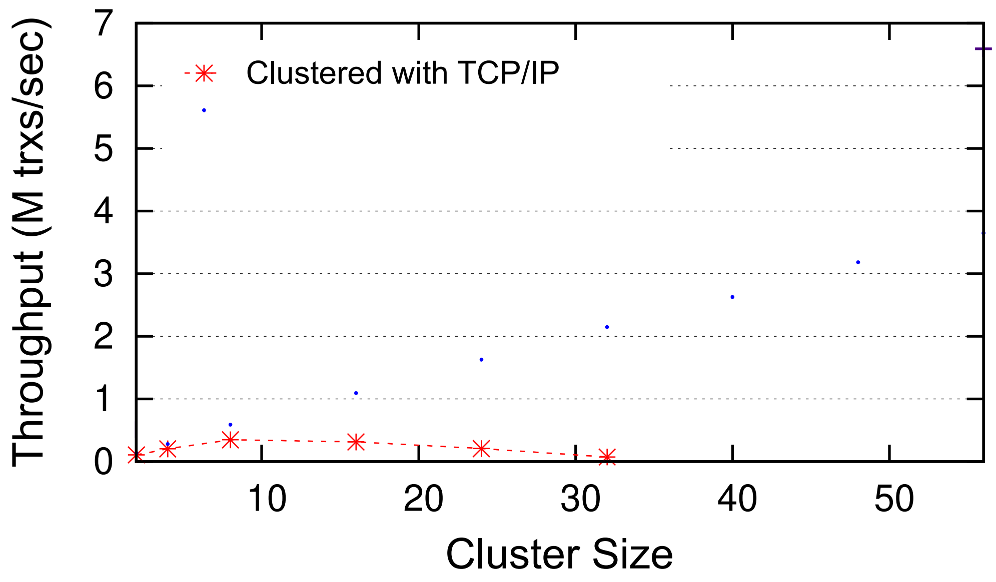
Why distributed transactions?

Cheaper for the same processing power

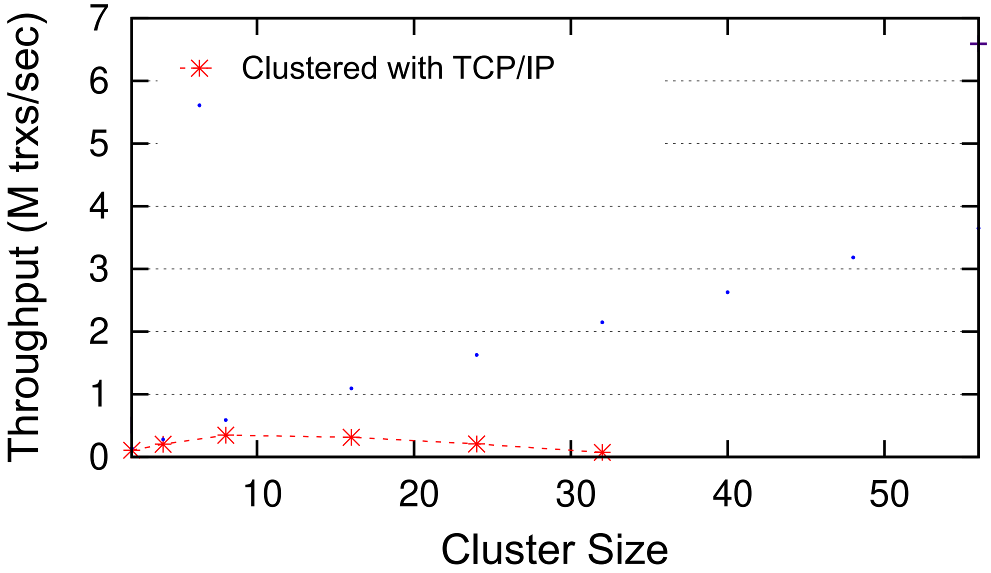


The problem with distributed transactions

Adding more machines doesn't increase performance



Adding more machines doesn't increase performance



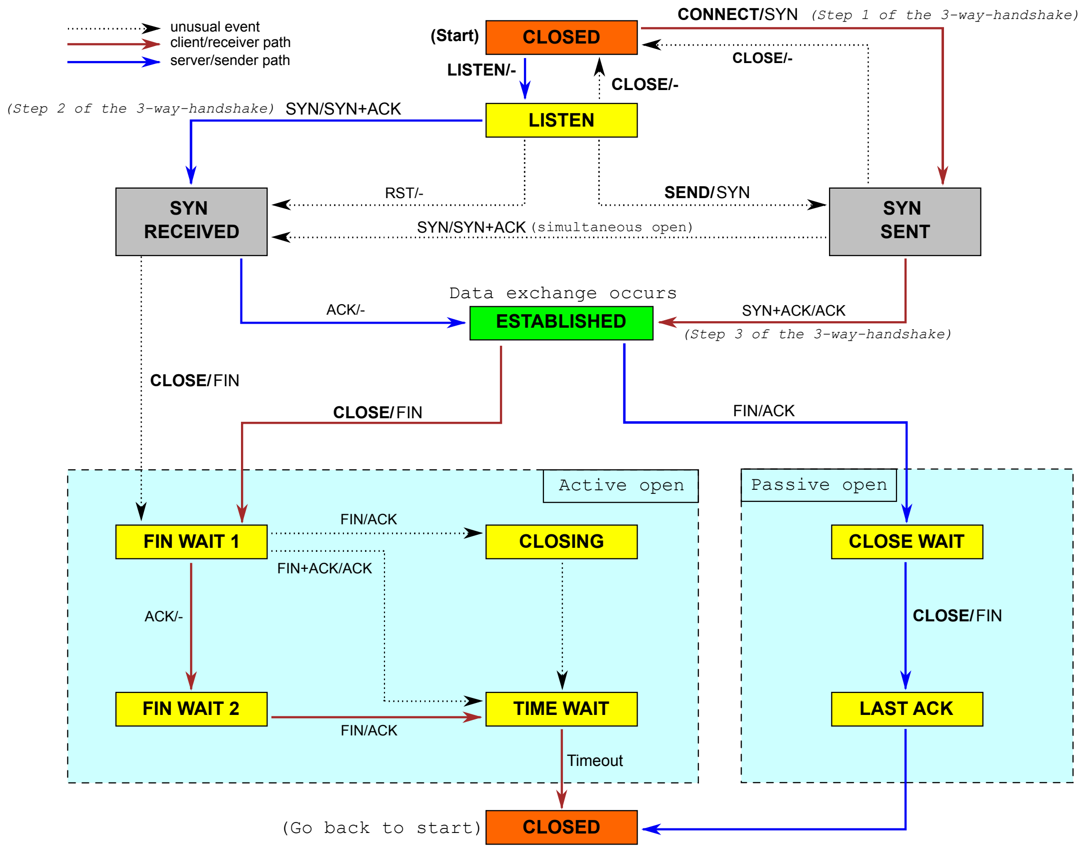
Accurate representation of a DB administrator

Why can’t they scale?

Background: TCP/IP

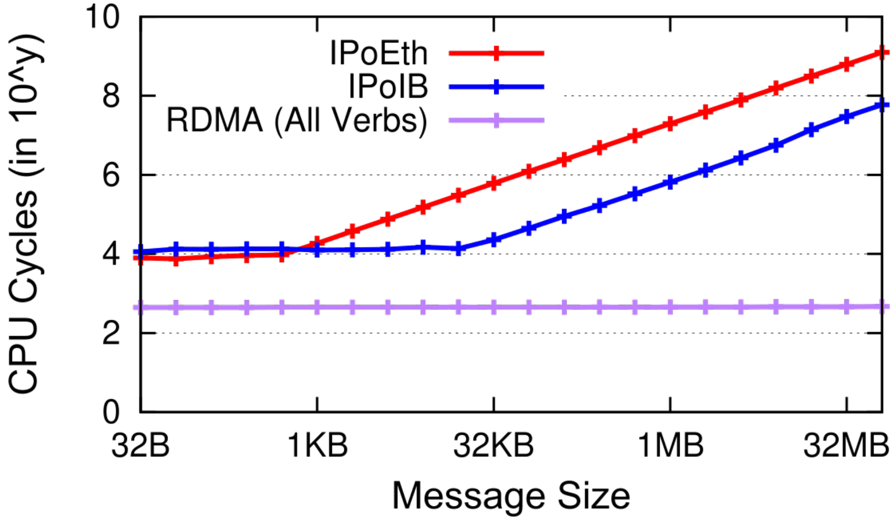
TCP/IP is computationally expensive

Complex design

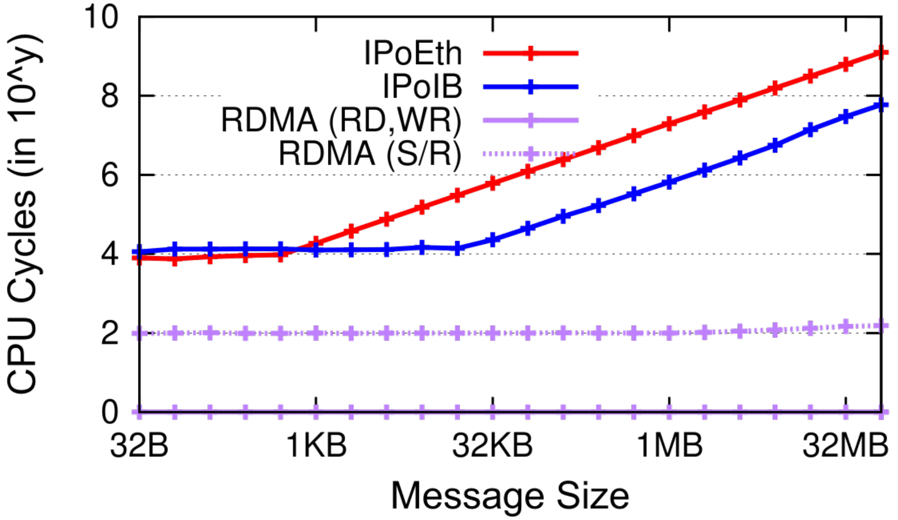


TCP/IP is computationally expensive

Fixed window size causes linear overhead



Client

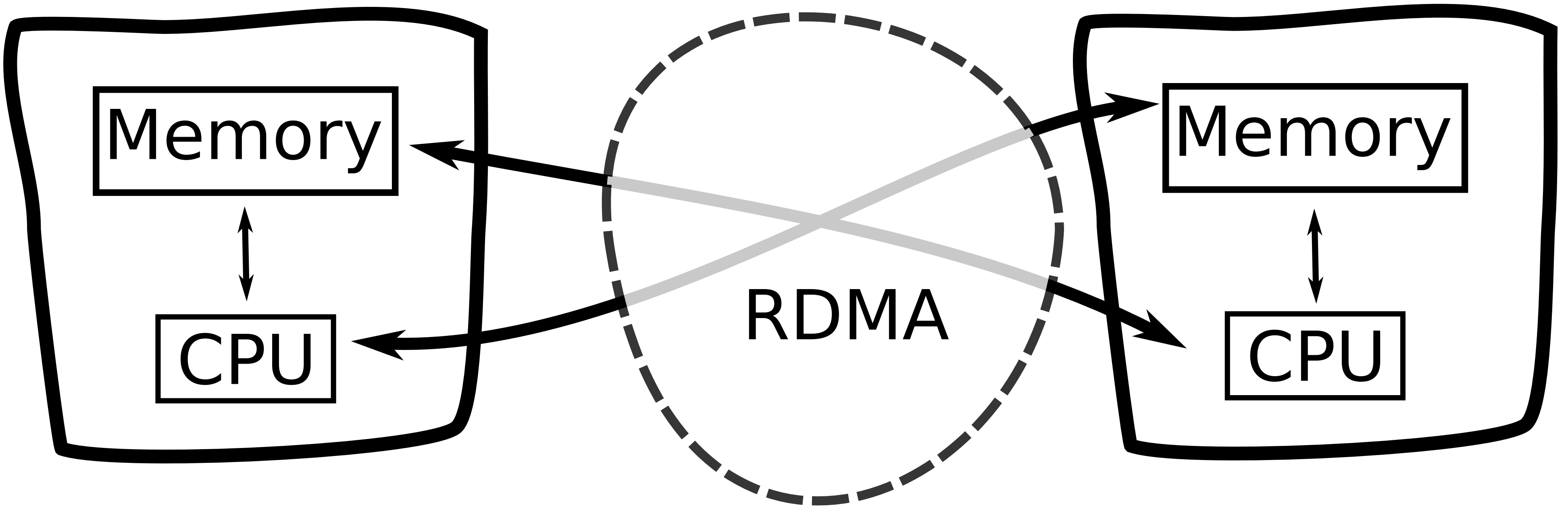


Server

How do we replace TCP/IP?

Background: RDMA

Remote Direct Memory Access (RDMA)



RDMA is great!

- Low latency
- High throughput
- Supported by InfiniBand

RDMA is hard to use

One-sided communication:

Receiver is not notified of connection

RDMA is hard to use

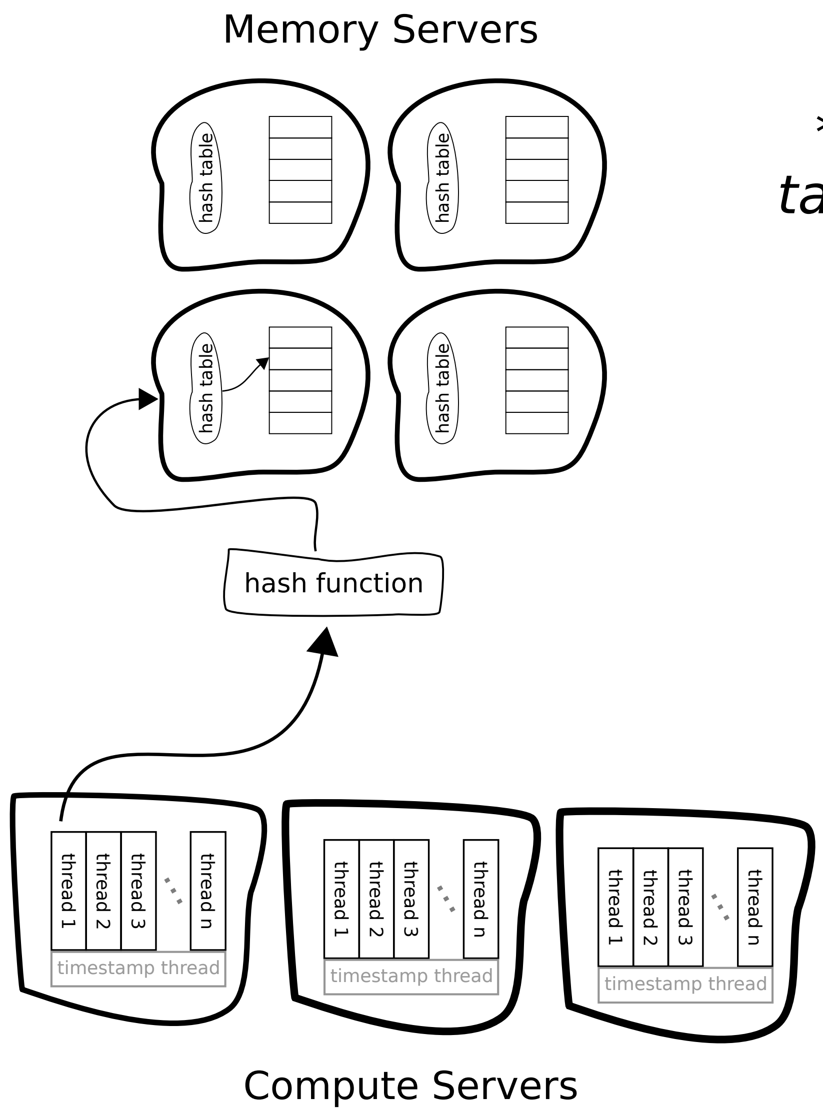
One-sided communication:

Receiver is not notified of connection

So far, most solutions that use RDMA, only do so for part of the design

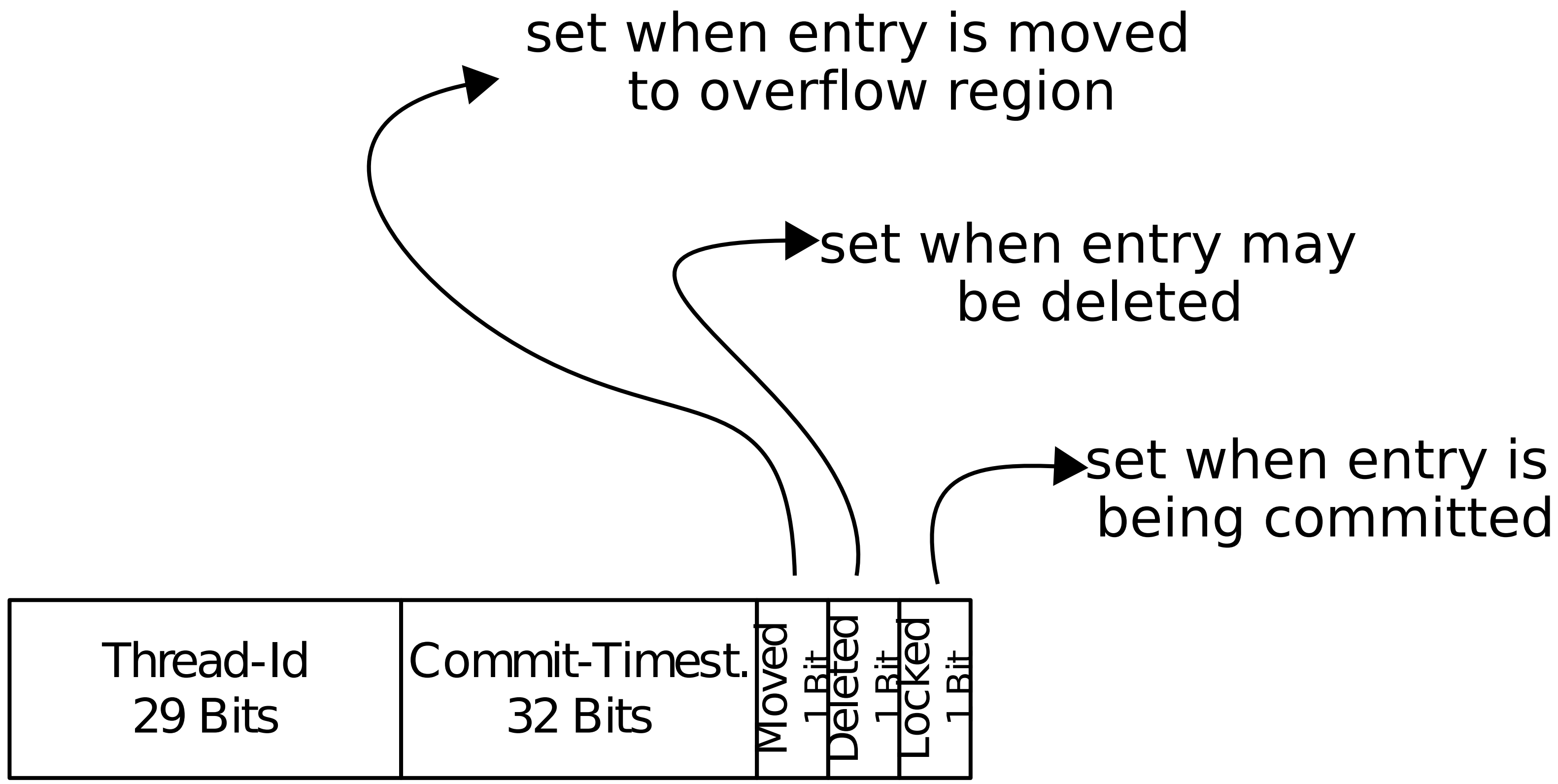
We need a system redesign

Main design

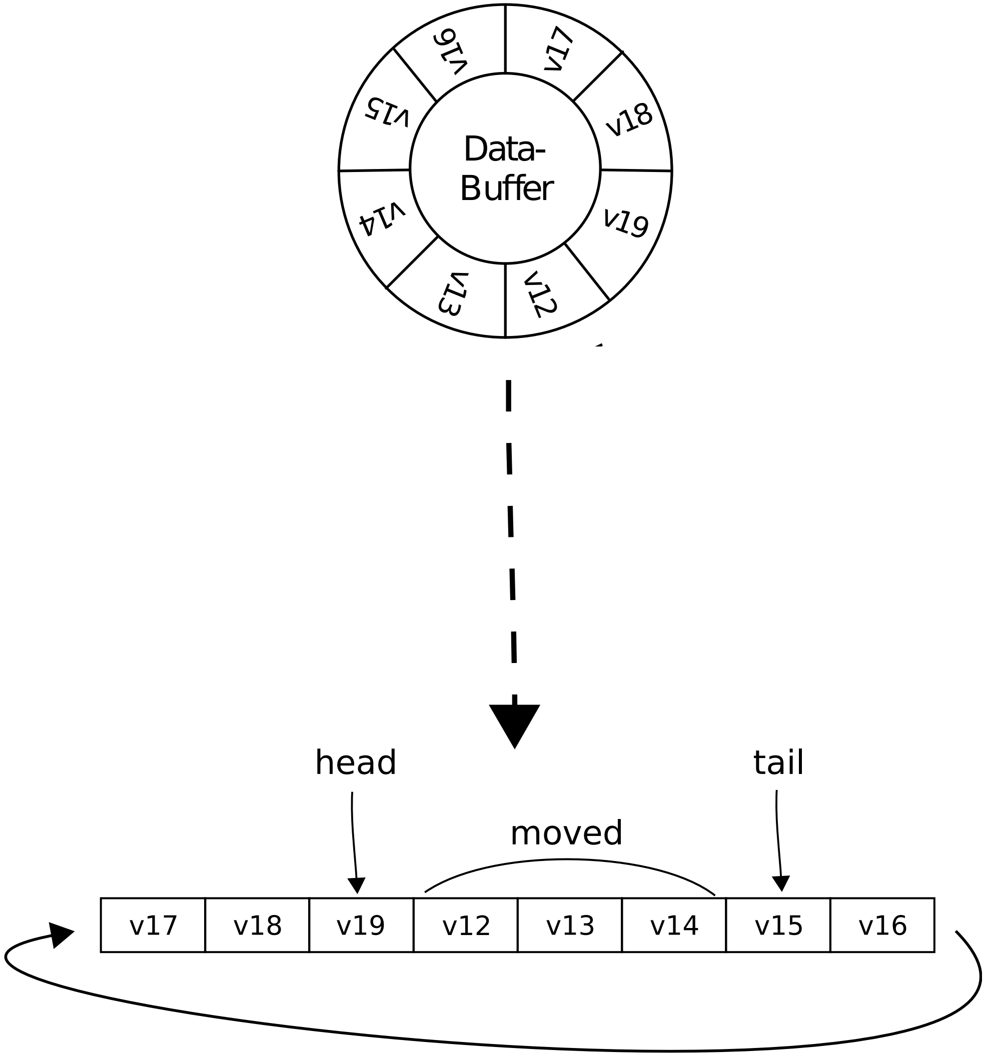


**Actually, the hash table might point to a different memory server*

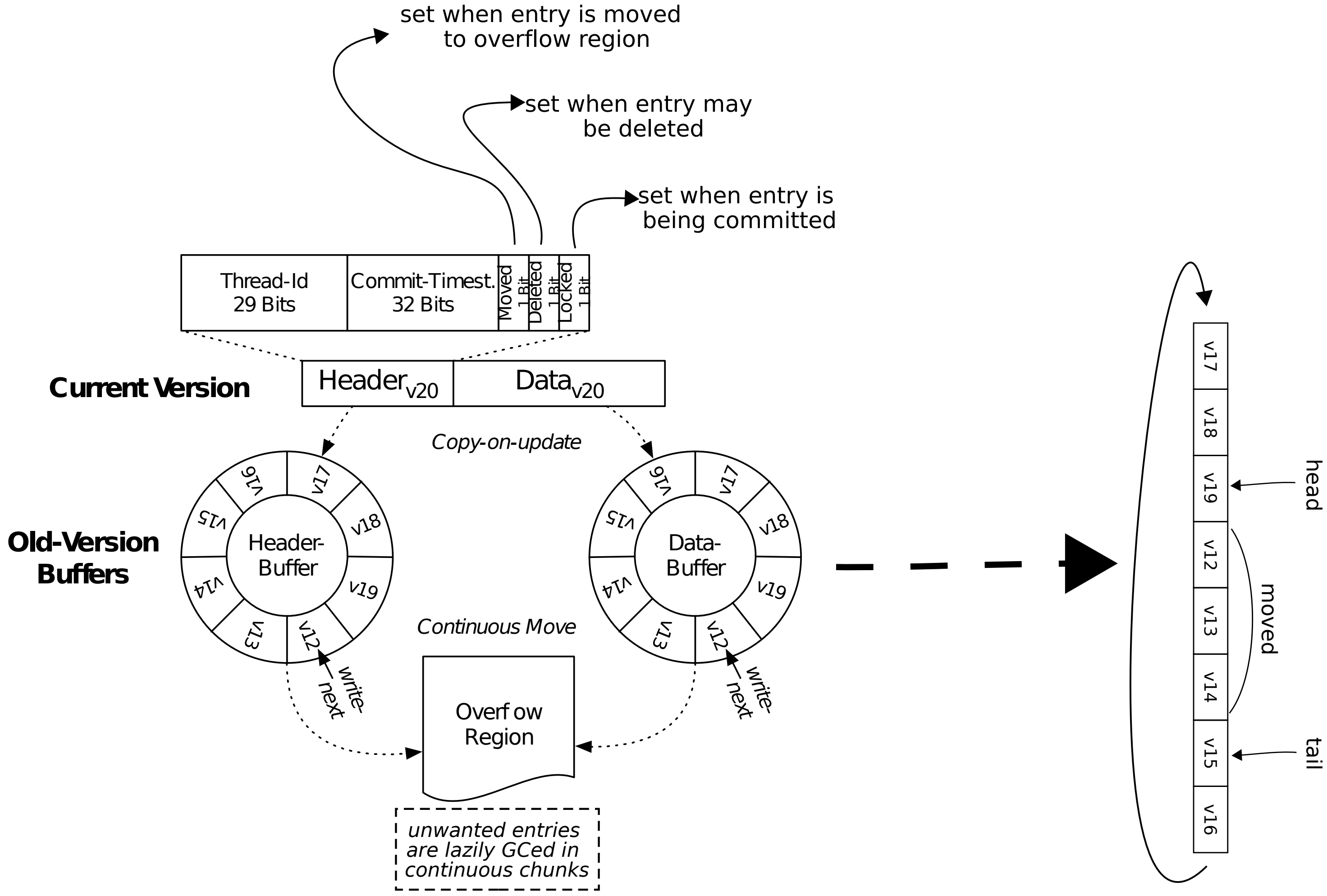
Data entries



Old-version Buffers



Data entries



Timestamp vector (Read timestamp)

Main design

- Read before fetching data
- Each cell is the commit timestamp of a thread
- Stored in a single Memory server

Optimizations

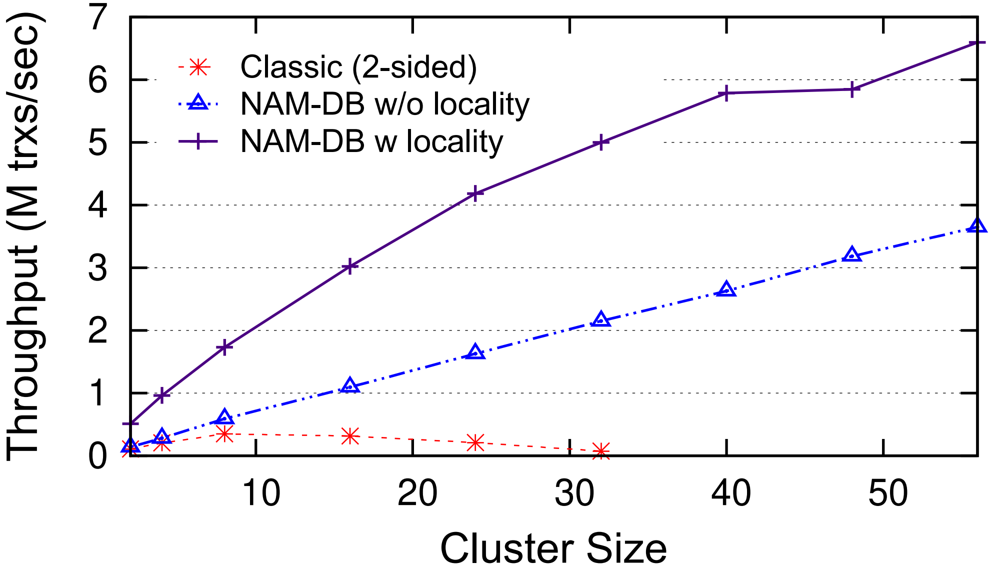
- Fetched by a dedicated thread in each Compute server (big ts reader)
- Threads in a Compute server might share commit timestamp (compression)

Further notes

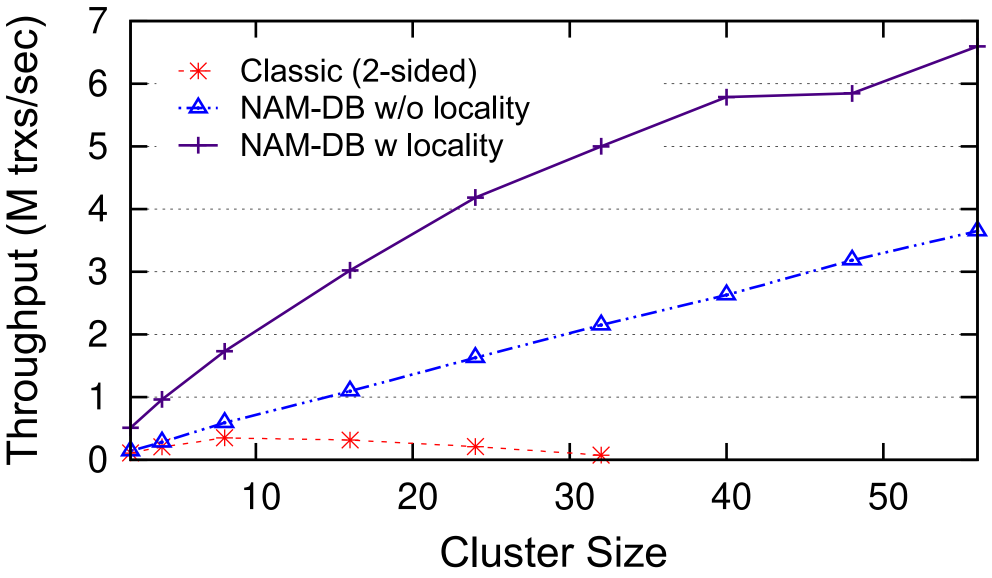
- Compute and Memory servers might coexist, taking advantage of locality
- Timestamp vectors may be partitioned
- Secondary indexes (B+-trees, hash tables)

Is it good enough?

Linear scalability



Linear scalability

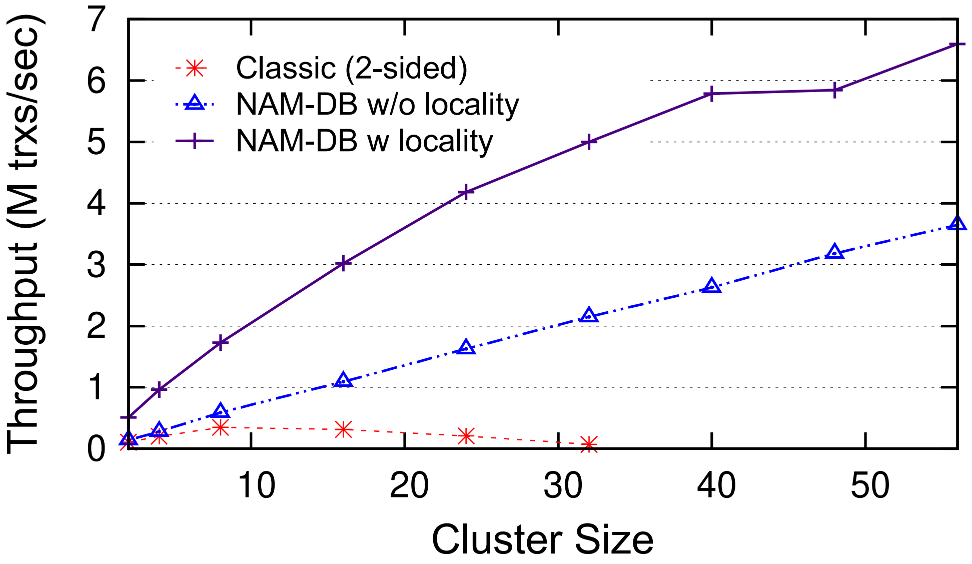


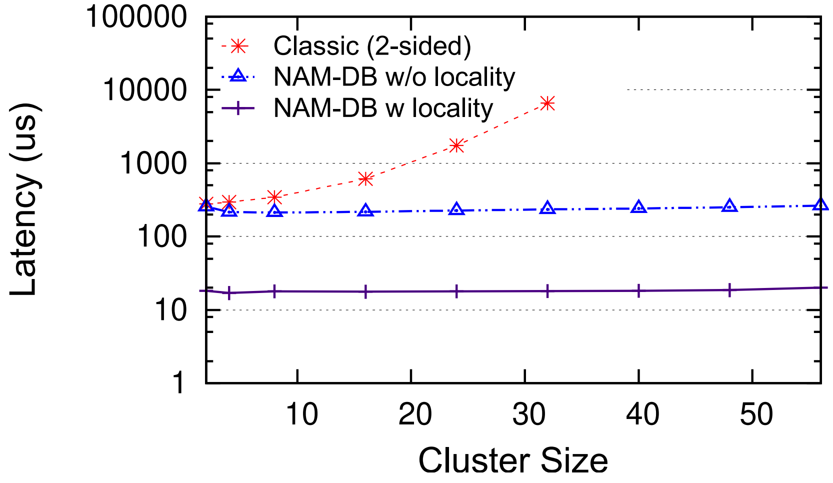
Experiments

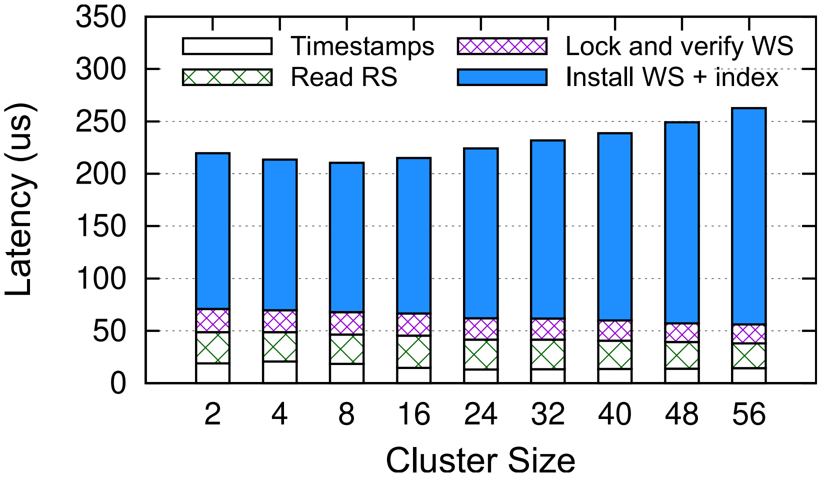
Setup

- **TPC-C benchmark**
- **2011-released InfiniBand (FDR)**
- **Two clusters, with 8 and 57 machines**

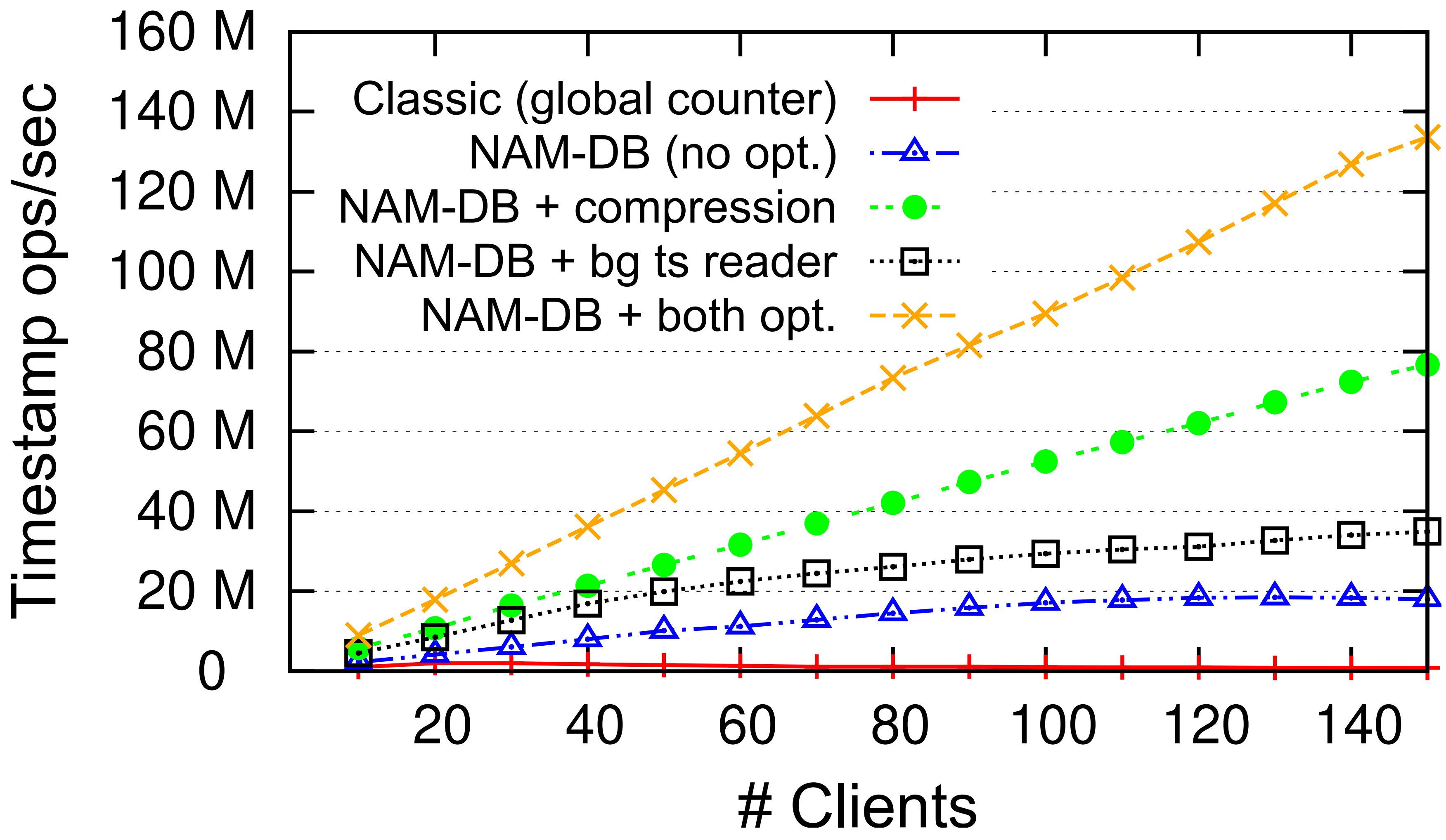
Experiment 1: System scalability



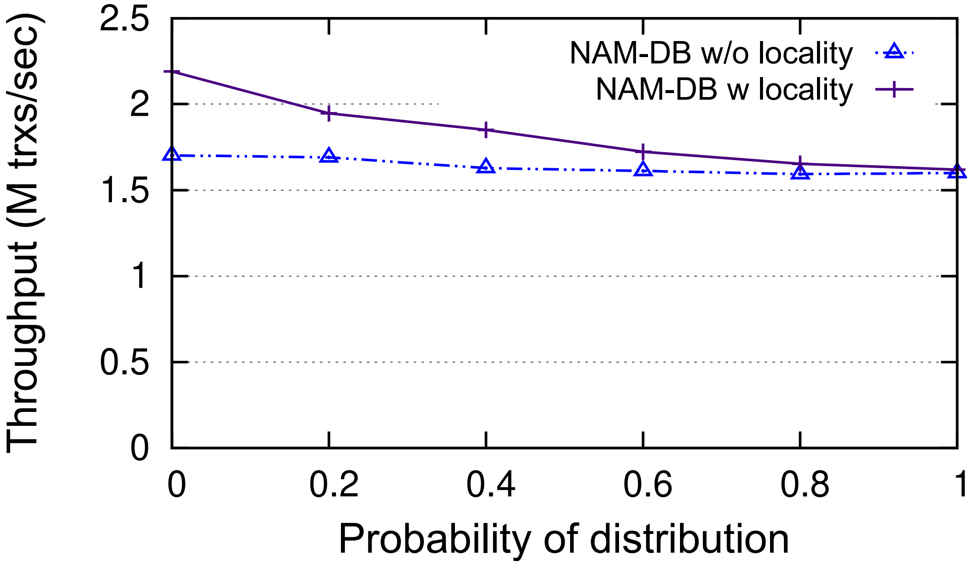


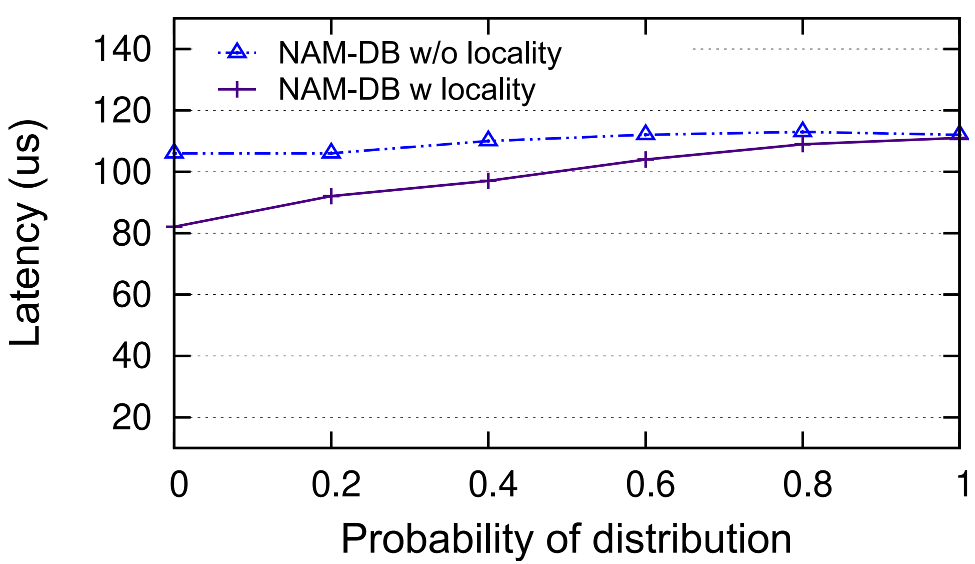


Experiment 2: Scalability of the Oracle

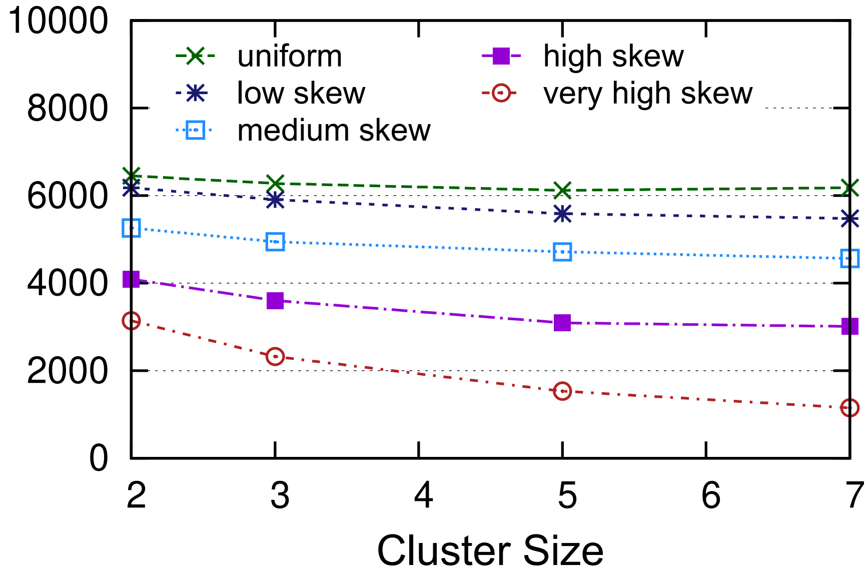


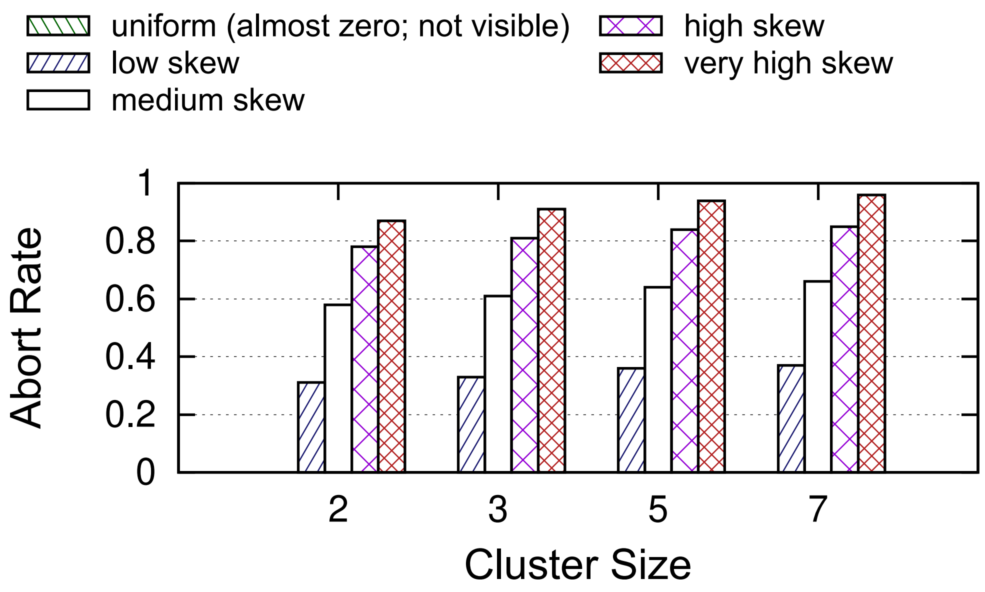
Experiment 3: Effect of Locality



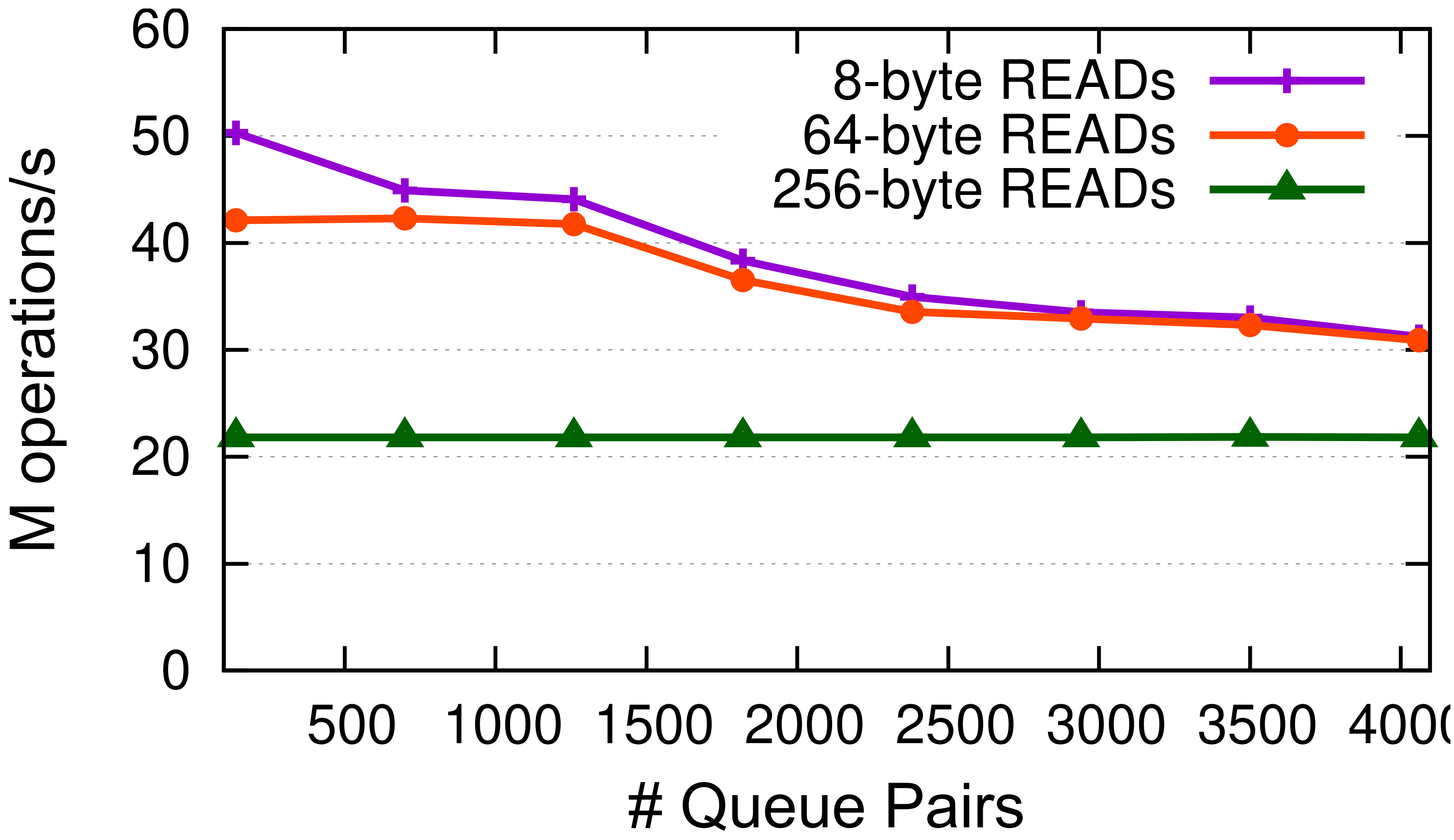


Experiment 4: Effect of Contention





Experiment 5: Scalability of RDMA Queue Pairs



Gaps in the logic

Future work

Future work

- Optimize for OLAP
- Reliably emulate large clusters and perform experiments
- Analyze performance, and optimize constants
- Explore collocation methods
- Explore secondary indexes

Thank you!

Media sources

E. Zamanian et al. The end of a myth: Distributed transactions can scale

C. Binnig et al. The end of slow networks: It’s time for a redesign