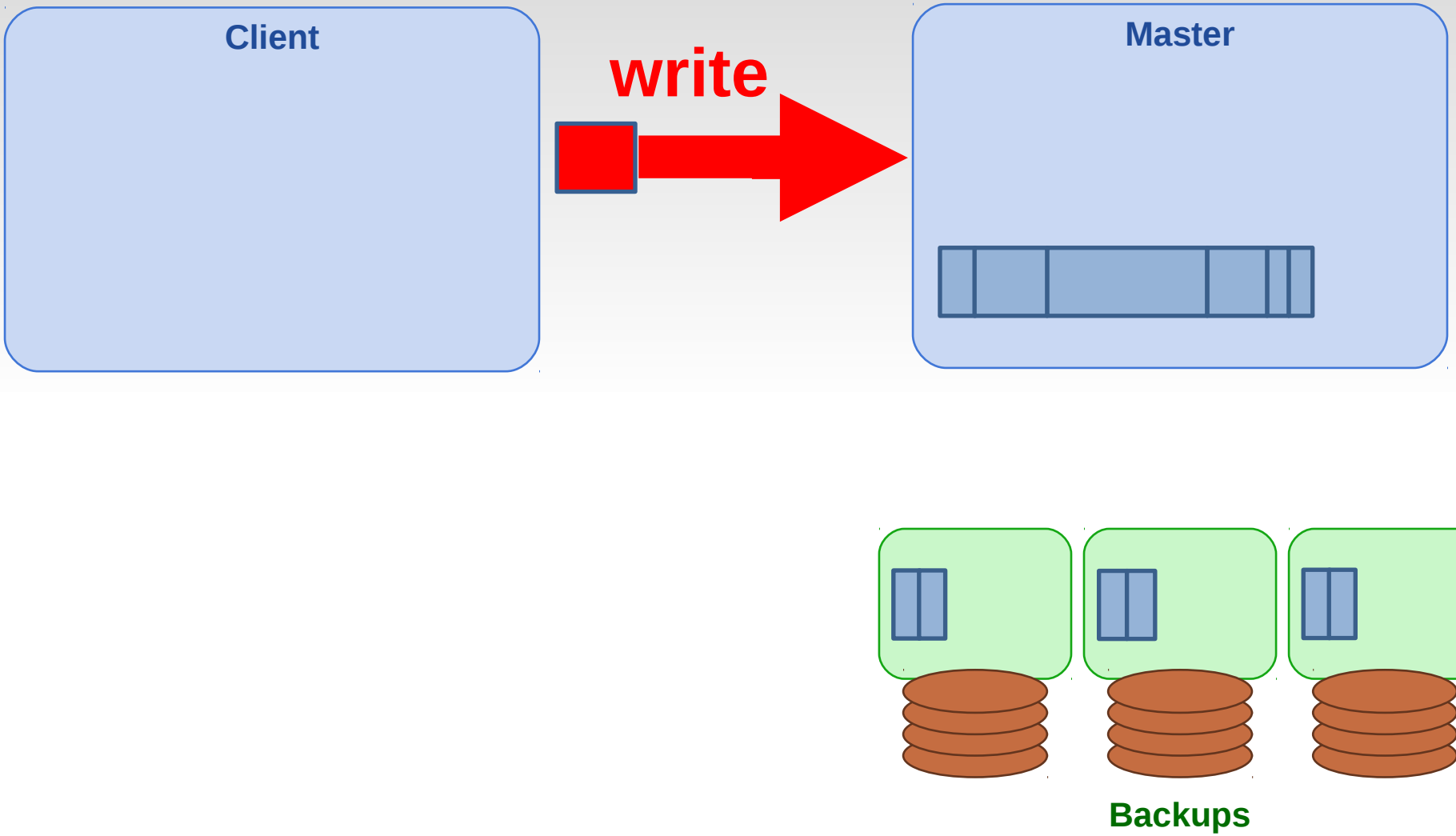
RDMA in RAMCloud

Alex Mordkovich

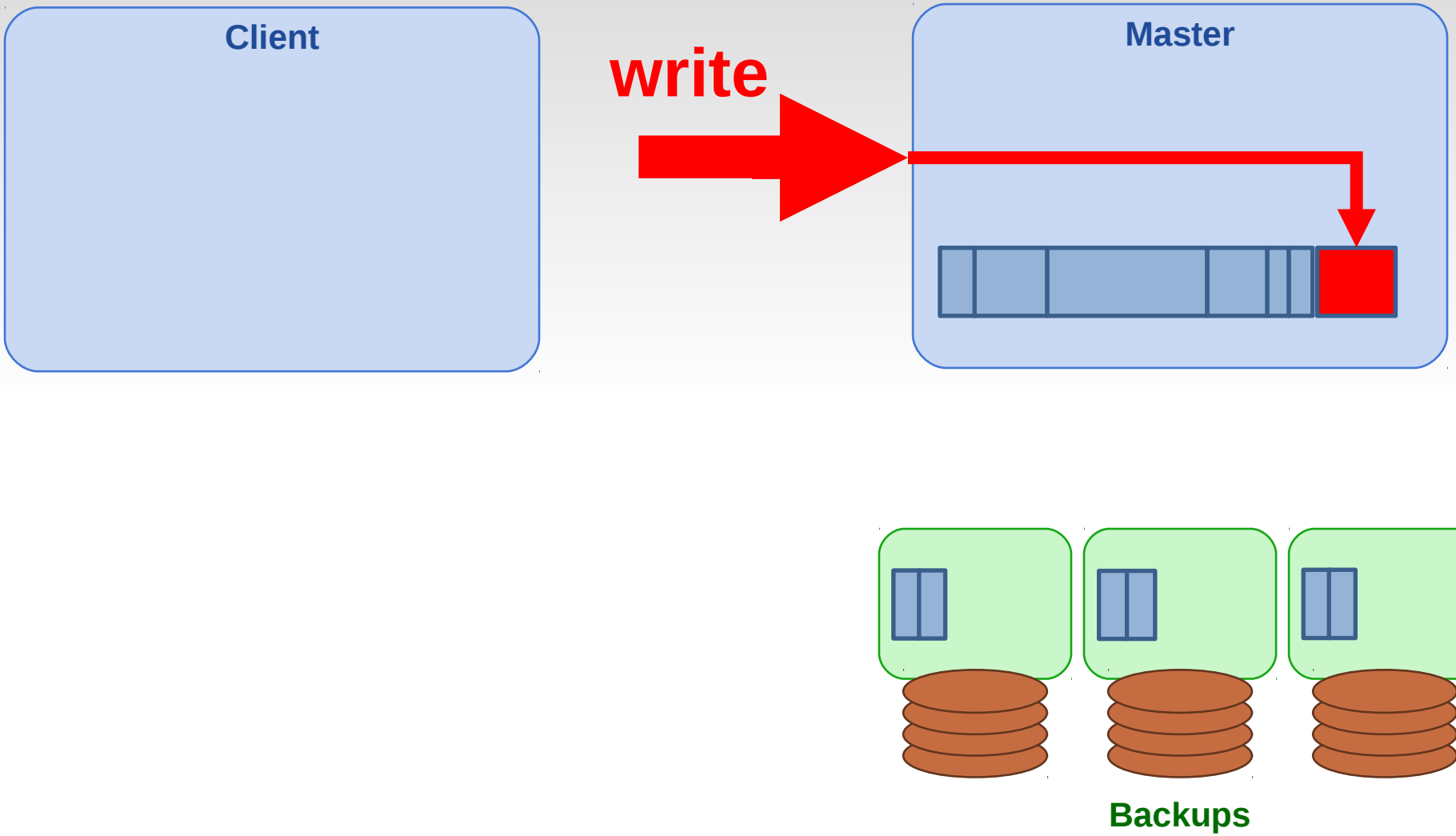
Agenda

- Motivation: Durable WRITEs
- Baseline system
- RDMA system
- Comparison
- Exploring Infiniband latency

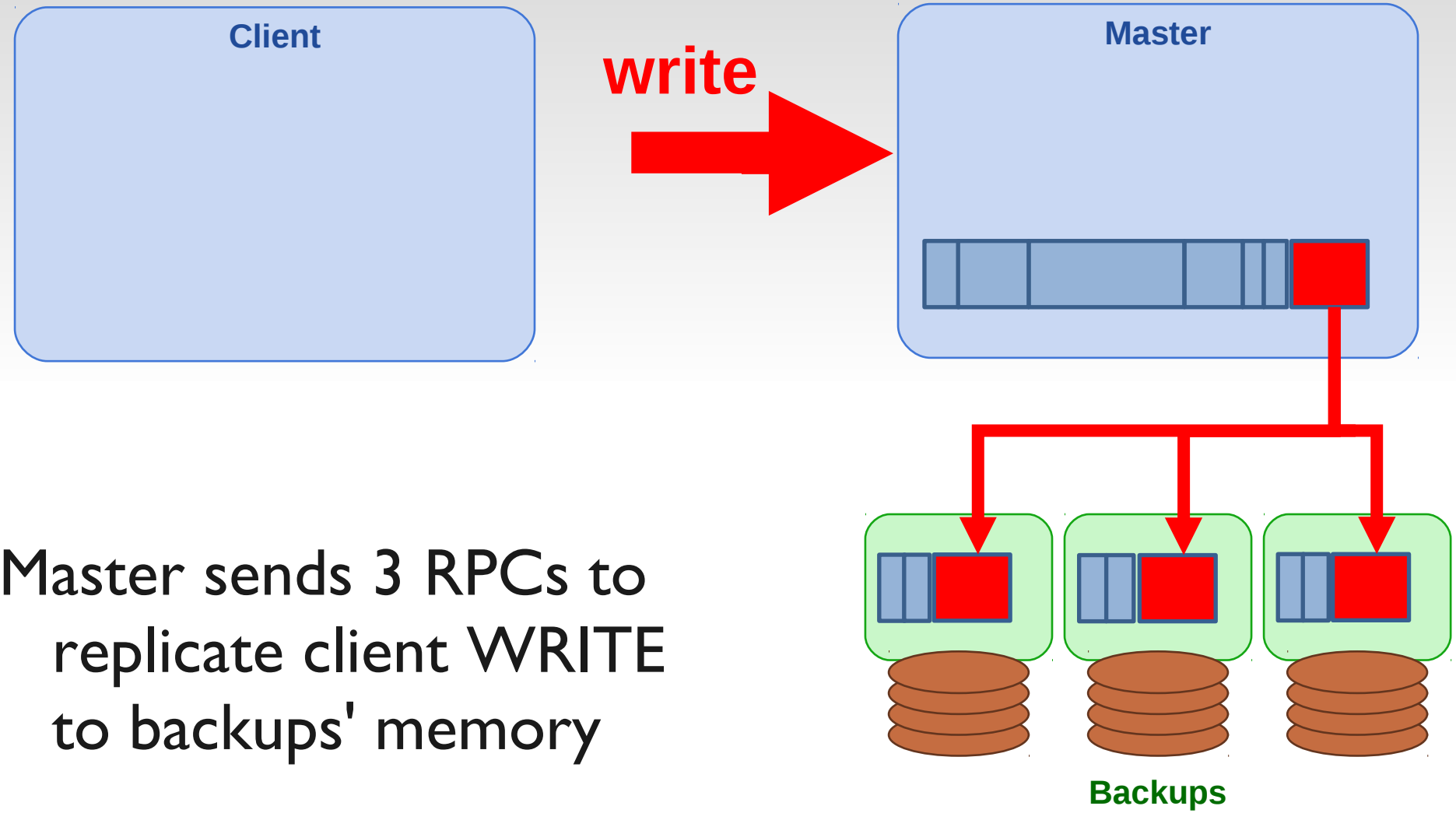
Motivation: Durable WRITEs



Motivation: Durable WRITEs



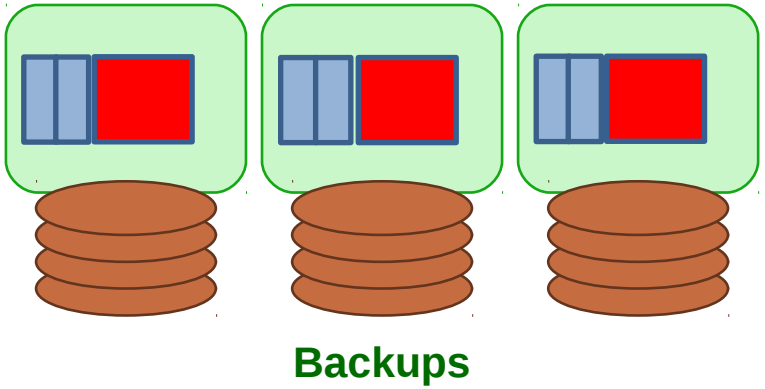
Motivation: Durable WRITEs



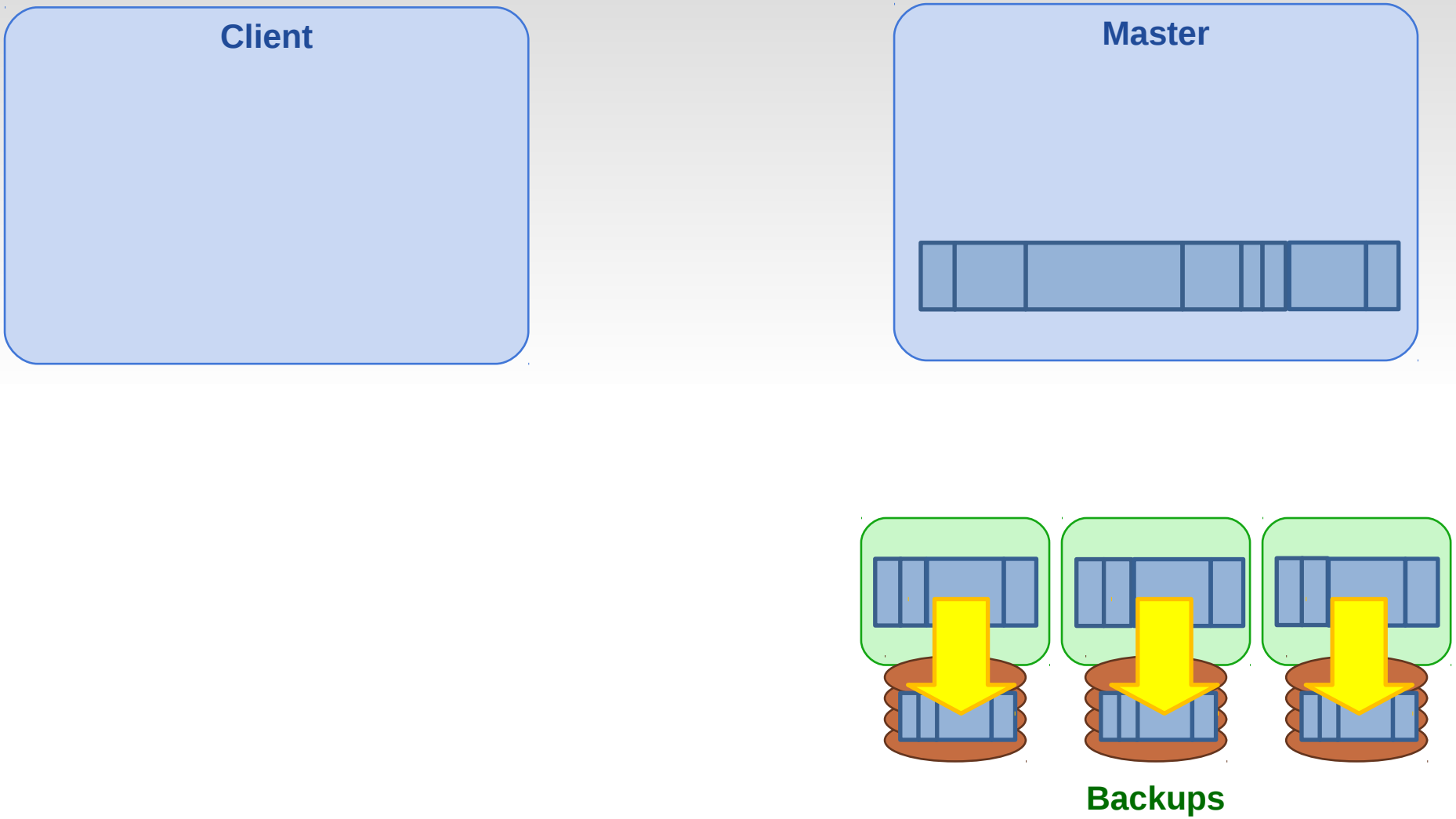
Master sends 3 RPCs to replicate client WRITE to backups' memory

Motivation: Durable WRITEs

Master acknowledges WRITE to client once all replication RPCs complete



Motivation: Durable WRITEs



Baseline Replication

- End-to-end latency:

100-byte read	~ 5 us
100-byte write	~15us

- ~10 us to replicate to backups.

Baseline Replication: Timeline

```
0 us     --- Begin replication (100-byte object)

1.8 us   --- Replication RPC #1 sent out

3.2 us   --- Replication RPC #2 sent out

4.4 us   --- Replication RPC #3 sent out

            [3.8 us "dead time"]

8.2 us   --- Replication RPC #1 completes (duration: 6.4 us)

9.8 us   --- Replication RPC #2 completes (duration: 6.6 us)

10.7 us --- Replication RPC #3 completes (duration: 6.3 us)
10.8 us --- End replication
```

time

Baseline Replication: Overheads

- For each RPC:

Acquire coarse lock protecting transport layer	300 ns
Talk to HCA to *send*	200 ns
Wait for RPC to complete	~6.4 us (albeit pipelined)

Baseline Replication: Overheads

- For each RPC:

Acquire coarse lock protecting transport layer	300 ns
Talk to HCA to *send*	200 ns
Wait for RPC to complete	~6.4 us (albeit pipelined)

Even though RPC framework system already quite optimized, RPCs still big time sink.

Baseline Replication: Overheads

- For each RPC:

Acquire coarse lock protecting transport layer	300 ns
Talk to HCA to *send*	200 ns
Wait for RPC to complete	~6.4 us (albeit pipelined)

Even though RPC framework system already quite optimized, RPCs still big time sink.

Can we we write 100 bytes faster?

What is Remote Direct Memory Access?

- HCA on Host A writes to main memory on Host B
- Bypass software stack on Host B
  - CPU on Host B not involved
- From perspective of software on Host B, contents just "appear" at memory location

- `rdmaWrite(localAddr, numBytes, remoteKey, remoteAddr)`

RDMA Replication

- Master just needs to copy data from its memory directly into log in backup's memory at some address
- Very conducive to RDMA
- Cleanly replace RPC code with RDMA

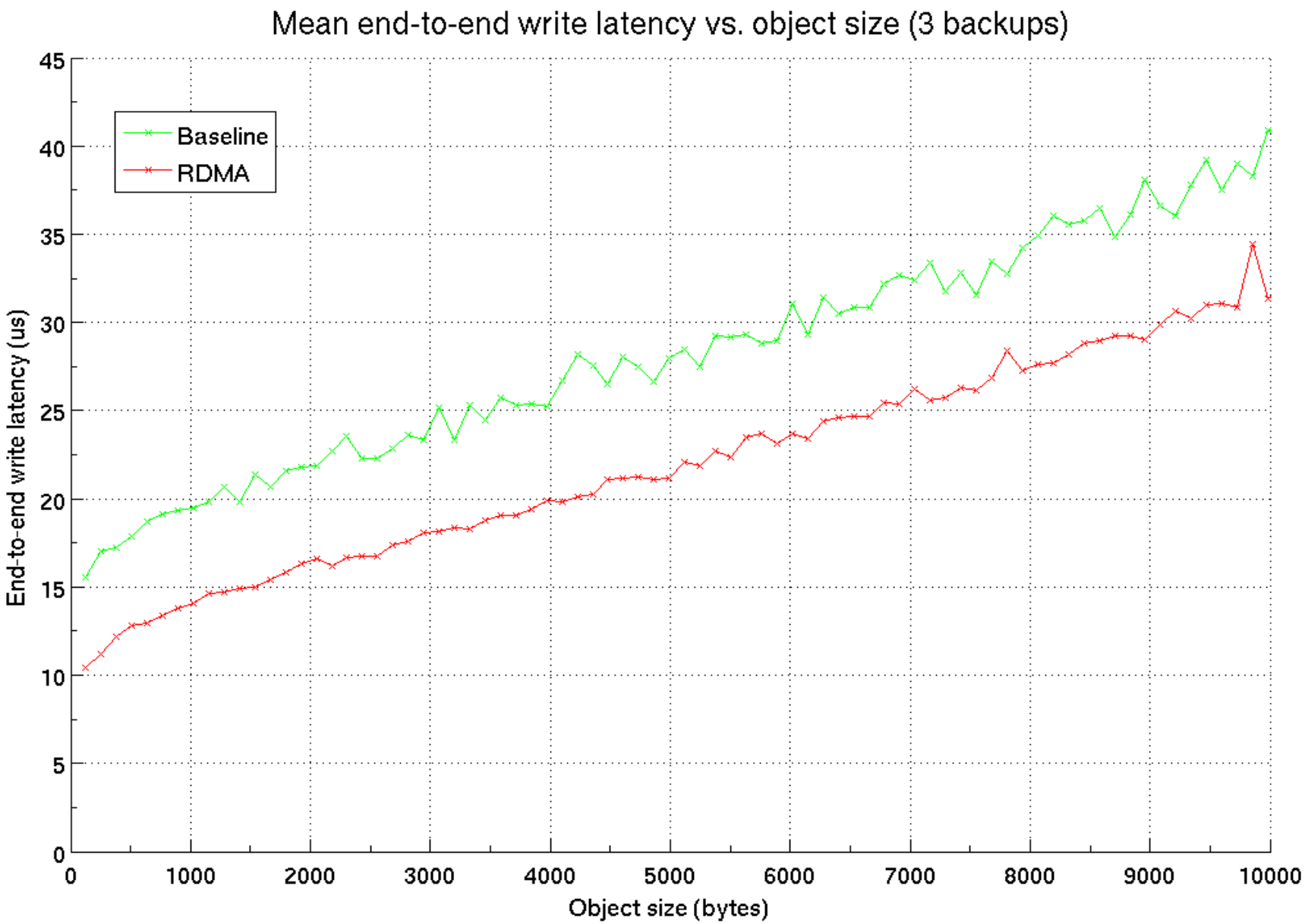
RDMA Replication: Timeline

4.1 us to complete replication

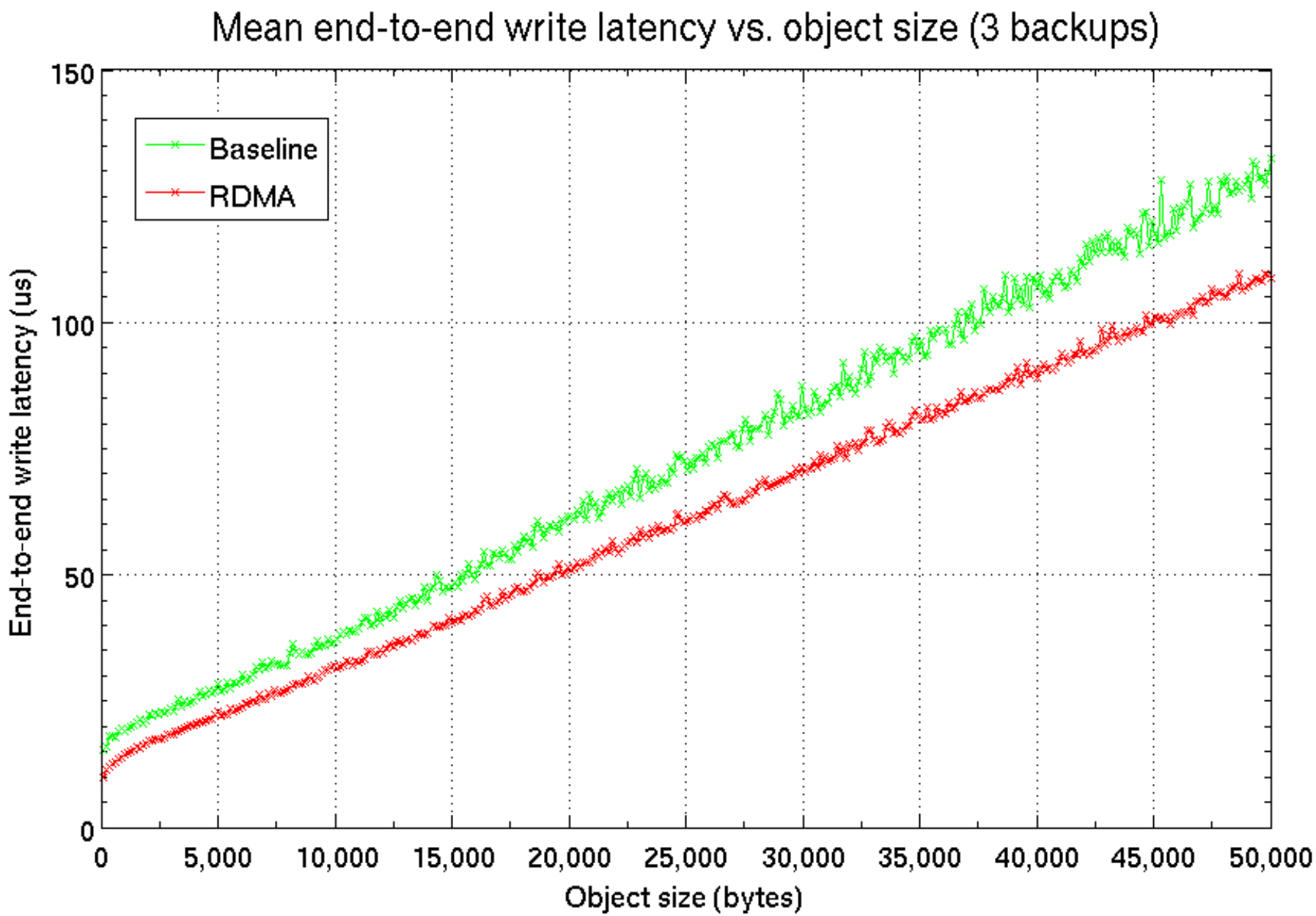
```
0 us     --- Begin replication (100 byte object)


1.0 us   --- Replication RDMA #1 sent out

1.6 us   --- Replication RDMA #2 sent out

2.2 us   --- Replication RDMA #3 sent out
            [0.6 us "dead time"]
2.8 us   --- Replication RDMA #1 completes (duration: 1.8 us)

3.4 us   --- Replication RDMA #2 completes (duration: 1.8 us)

3.9 us   --- Replication RDMA #3 completes (duration: 1.7 us)
4.1 us   --- End replication
```

time

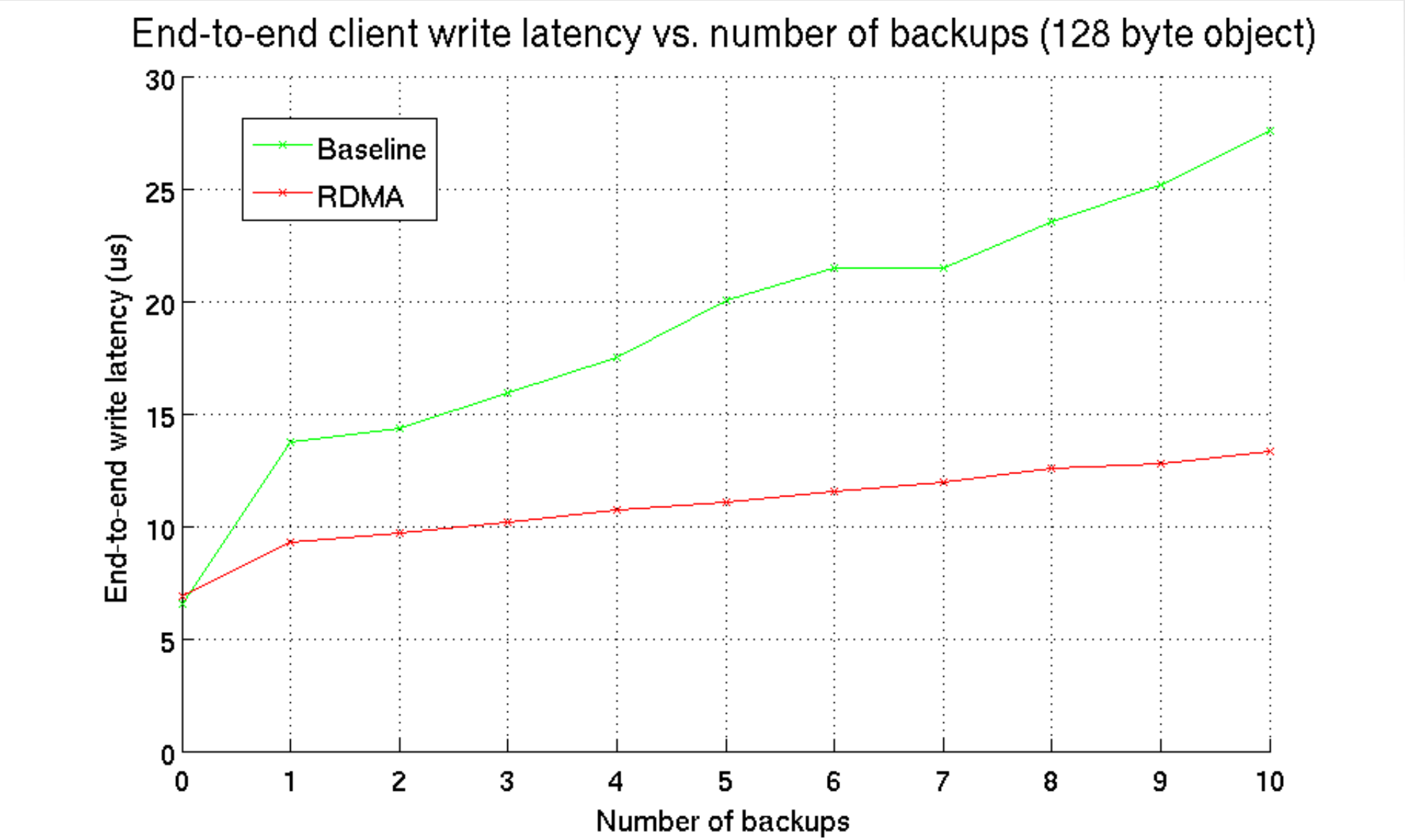
RDMA vs. Baseline



RDMA vs. Baseline



RDMA vs. Baseline

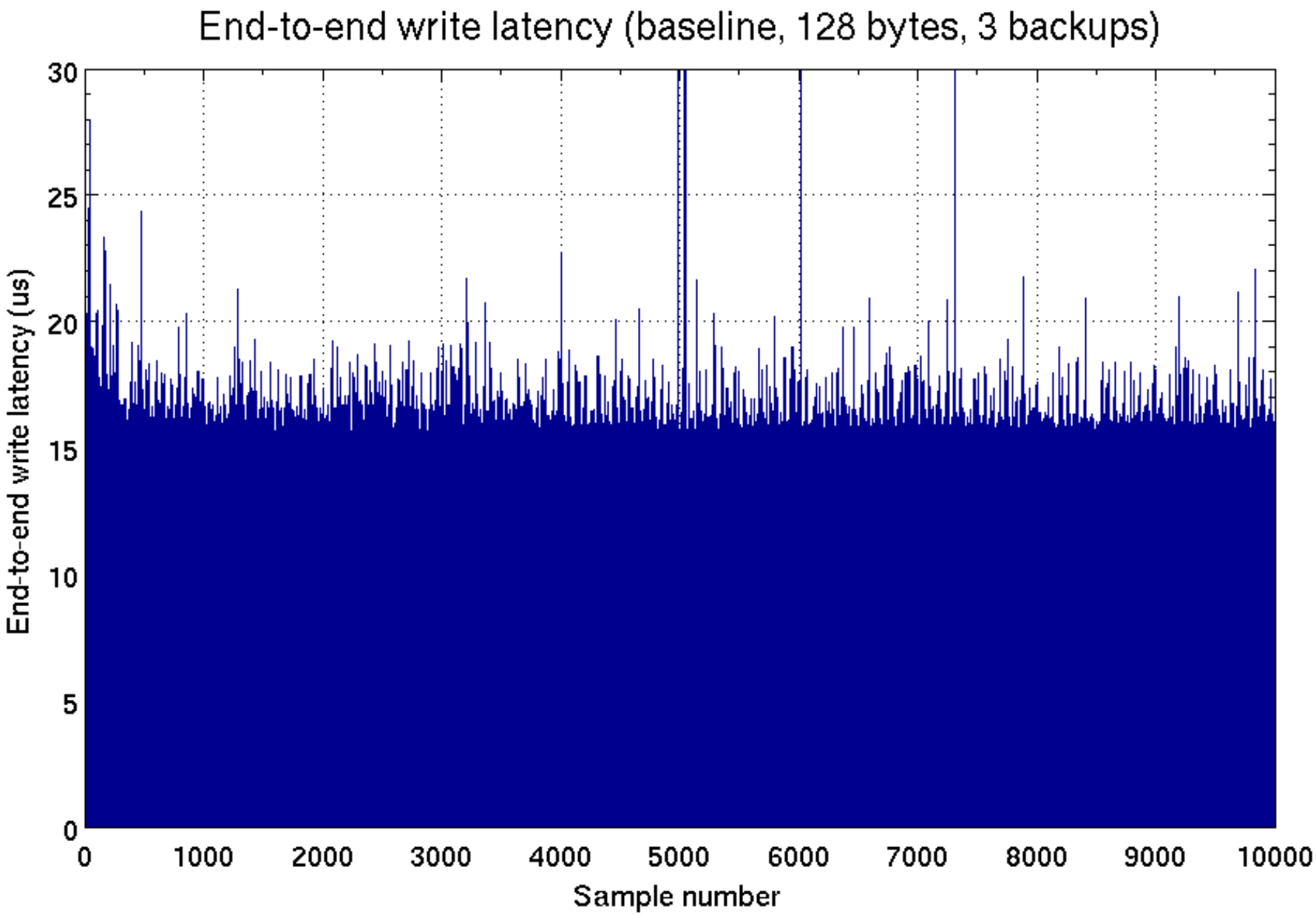


Variance

Observed variance in latencies, especially on baseline system.

Focused in on latencies for small (128-byte) objects...

Variance

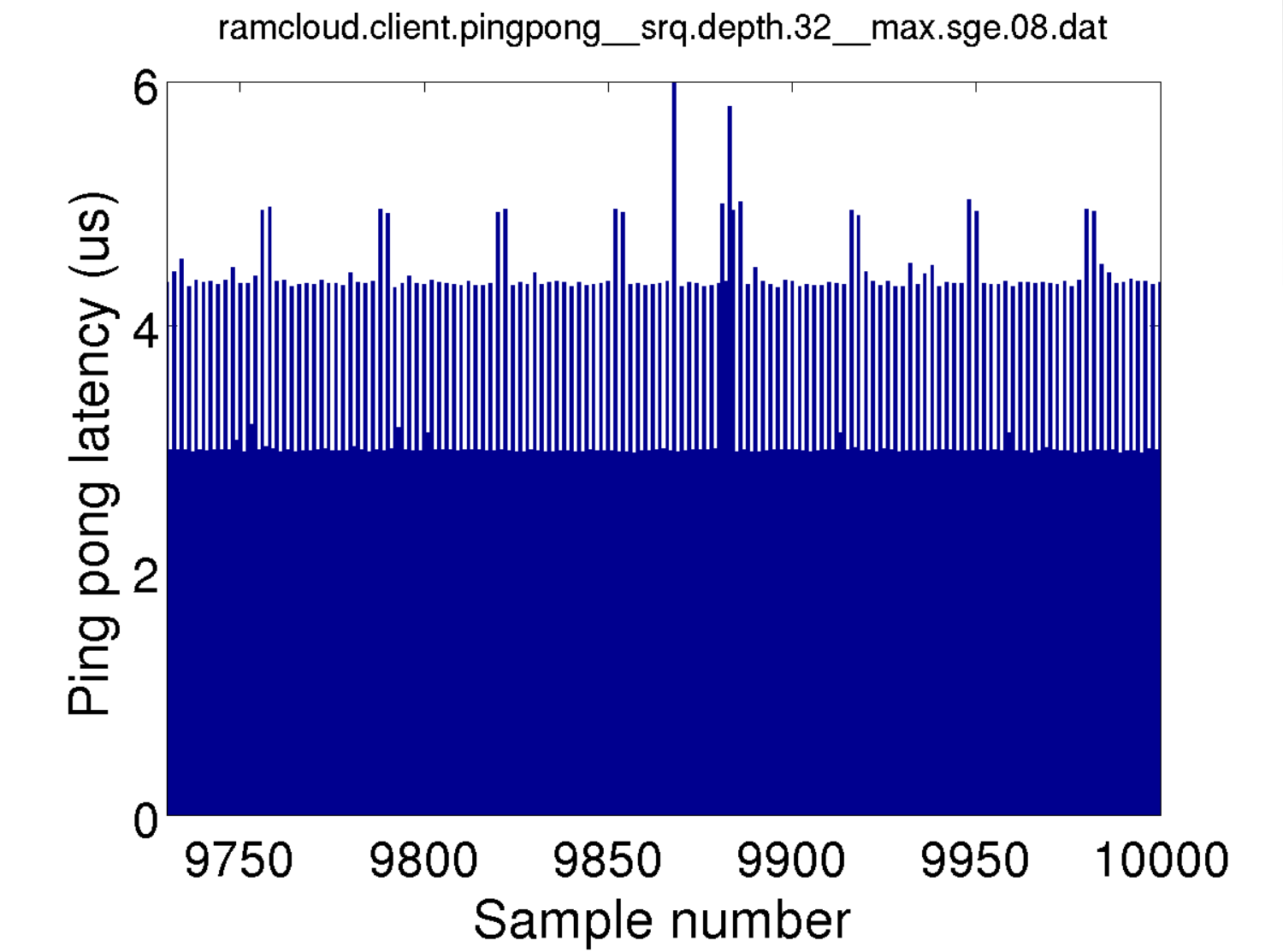

End-to-end write latency (baseline, 128 bytes, 3 backups)
End-to-end write latency (us)
Sample number
0
5
10
15
20
25
30
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000

Variance

- To rule out RAMCloud-related effects, stripped away RPC layer
- Modified RAMCloud client/server to simply ping-pong messages on top of Infiniband transport
- No backups
- Network and hosts otherwise quiesced

RAMCloud ping-pong

Default configuration



Effect of Infiniband parameters

Wrote simple ping-pong program to explore effect of Infiniband configurations.

Initially, program did not exhibit RAMCloud patterns.

Settings tested:

- Maximum number of SGE in a receive request
- Number of receive buffer registrations
- …

Maximum SGE in receive request

To receive a message, need to post a receive *request* to a receive *queue*.

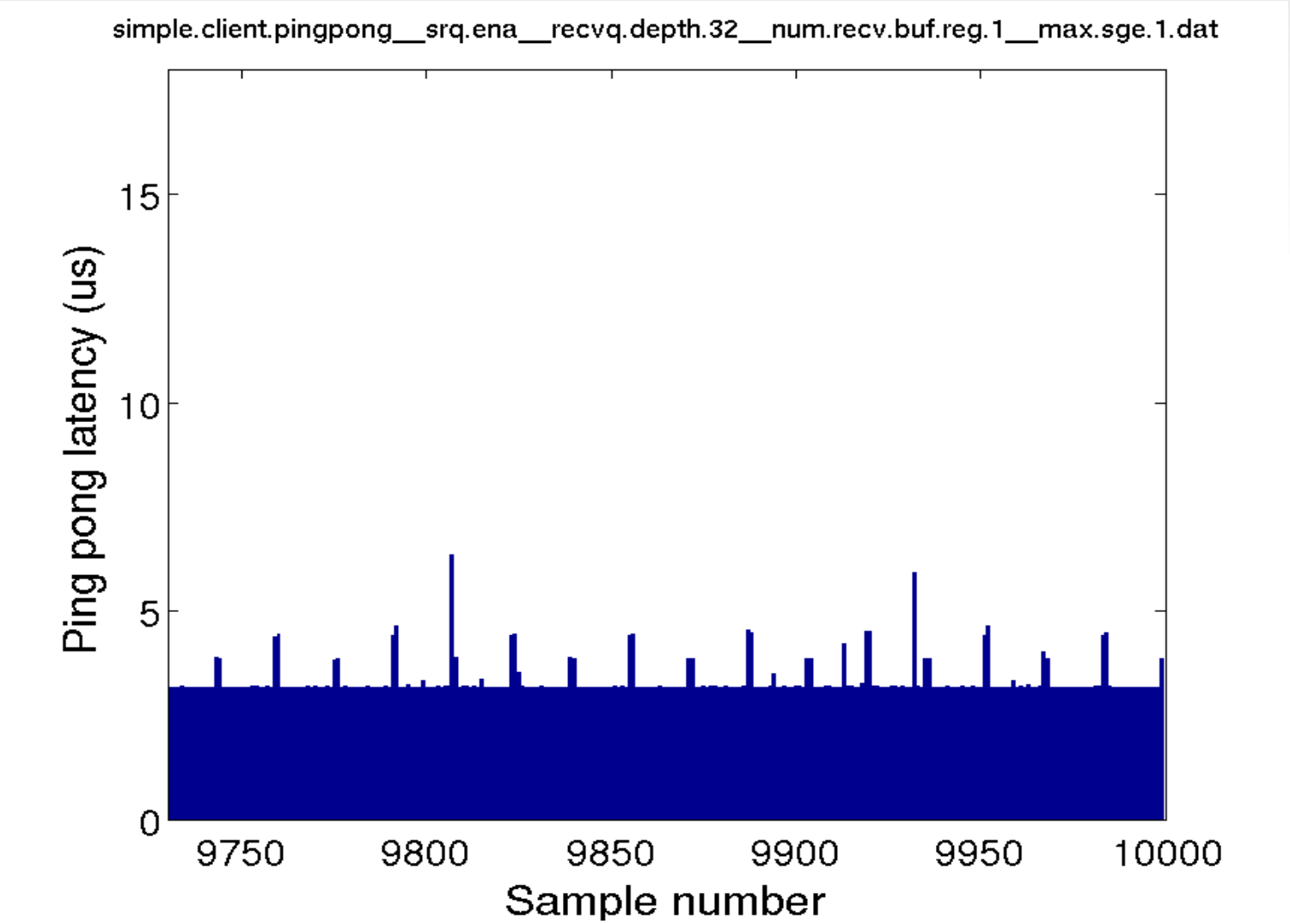
A receive request can specify multiple scatter/gather entries.

When a receive queue is created, the *maximum* number of SGE entries per (future) receive request is specified.

Both RAMCloud and simple program only actually use one SGE per receive request.

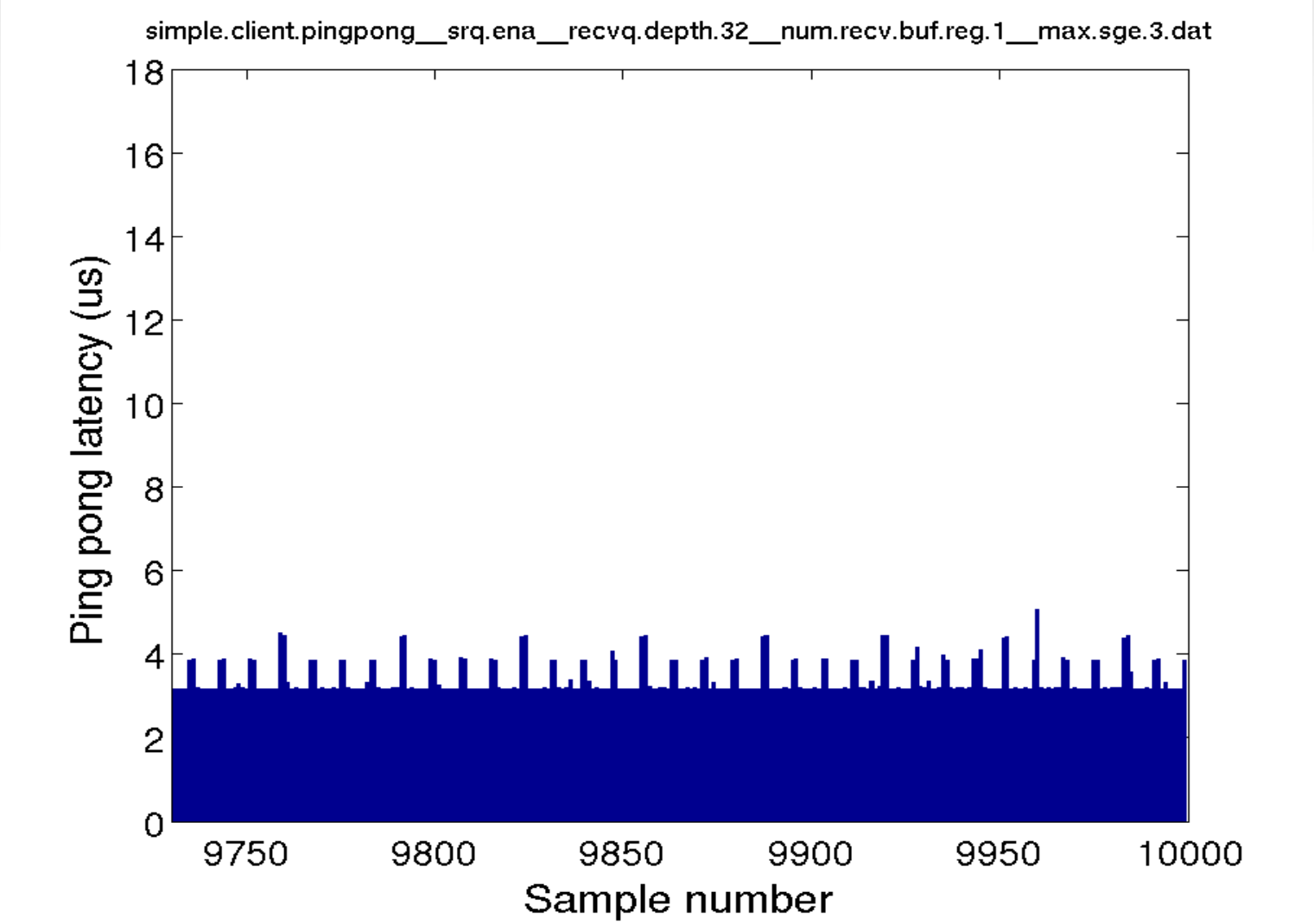
Maximum SGE in receive request

N = 32 max_sge = 1



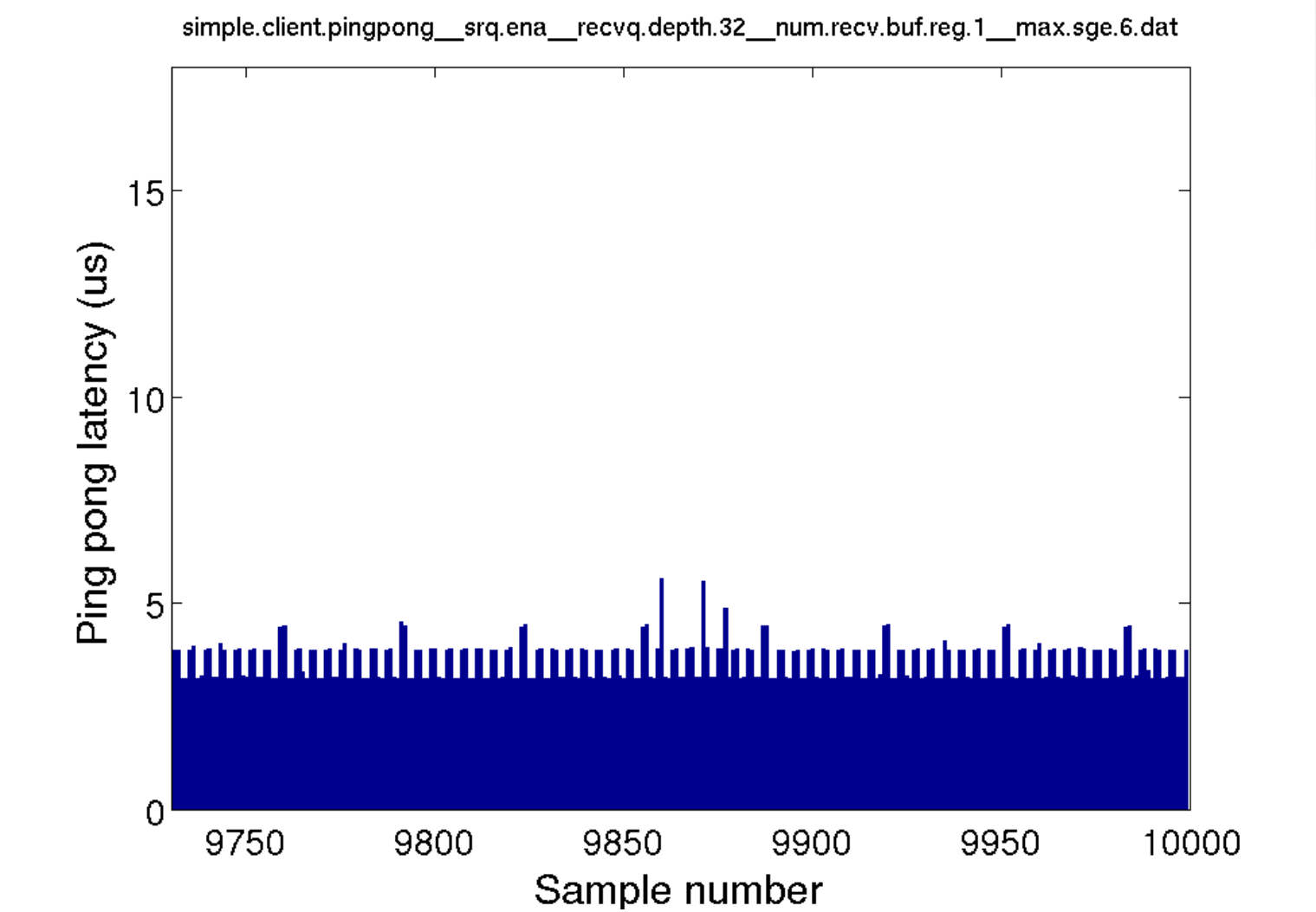
Maximum SGE in receive request

N = 32 max_sge = 2 thru 3



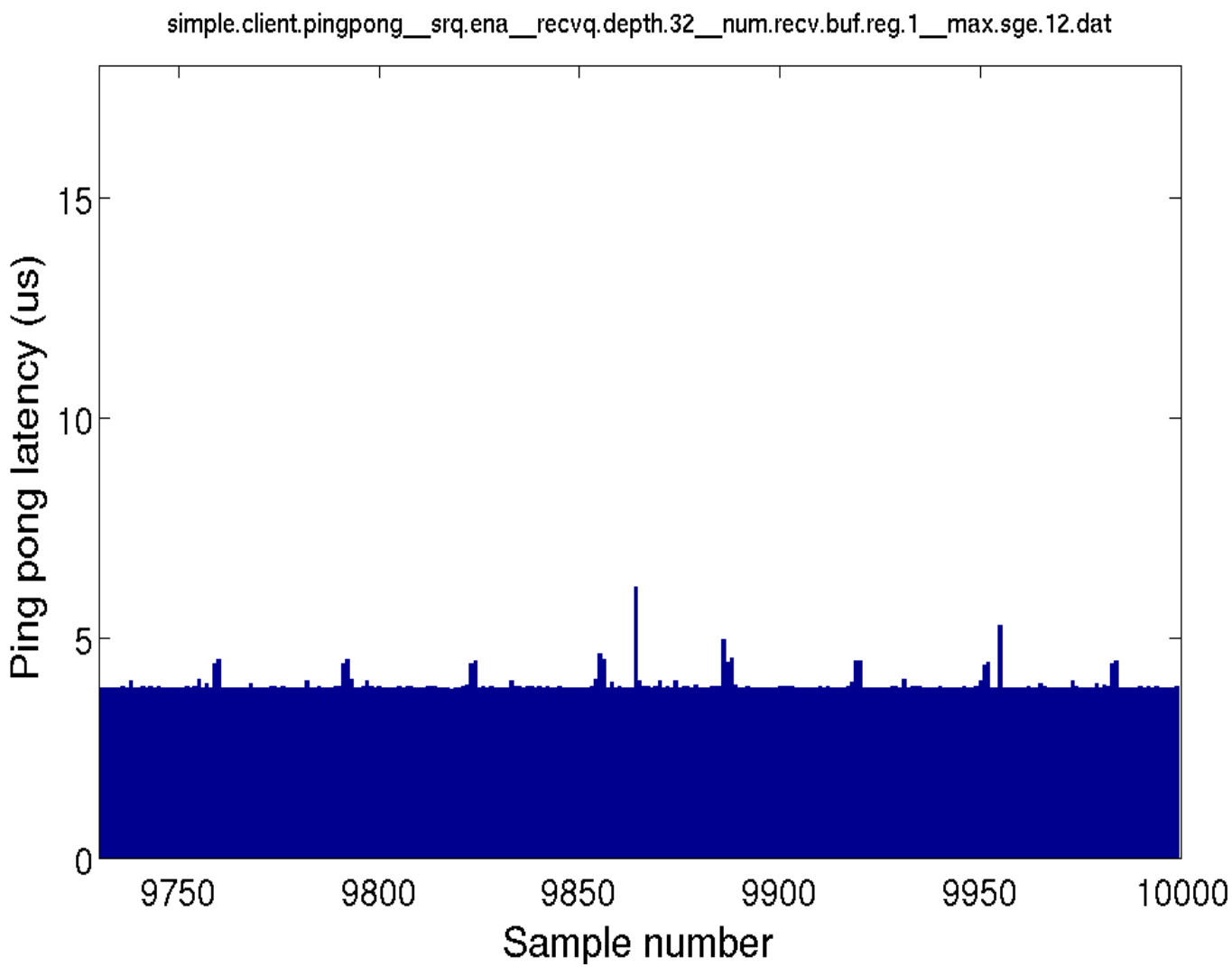
Maximum SGE in receive request

N = 32 max_sge = 4 thru 7



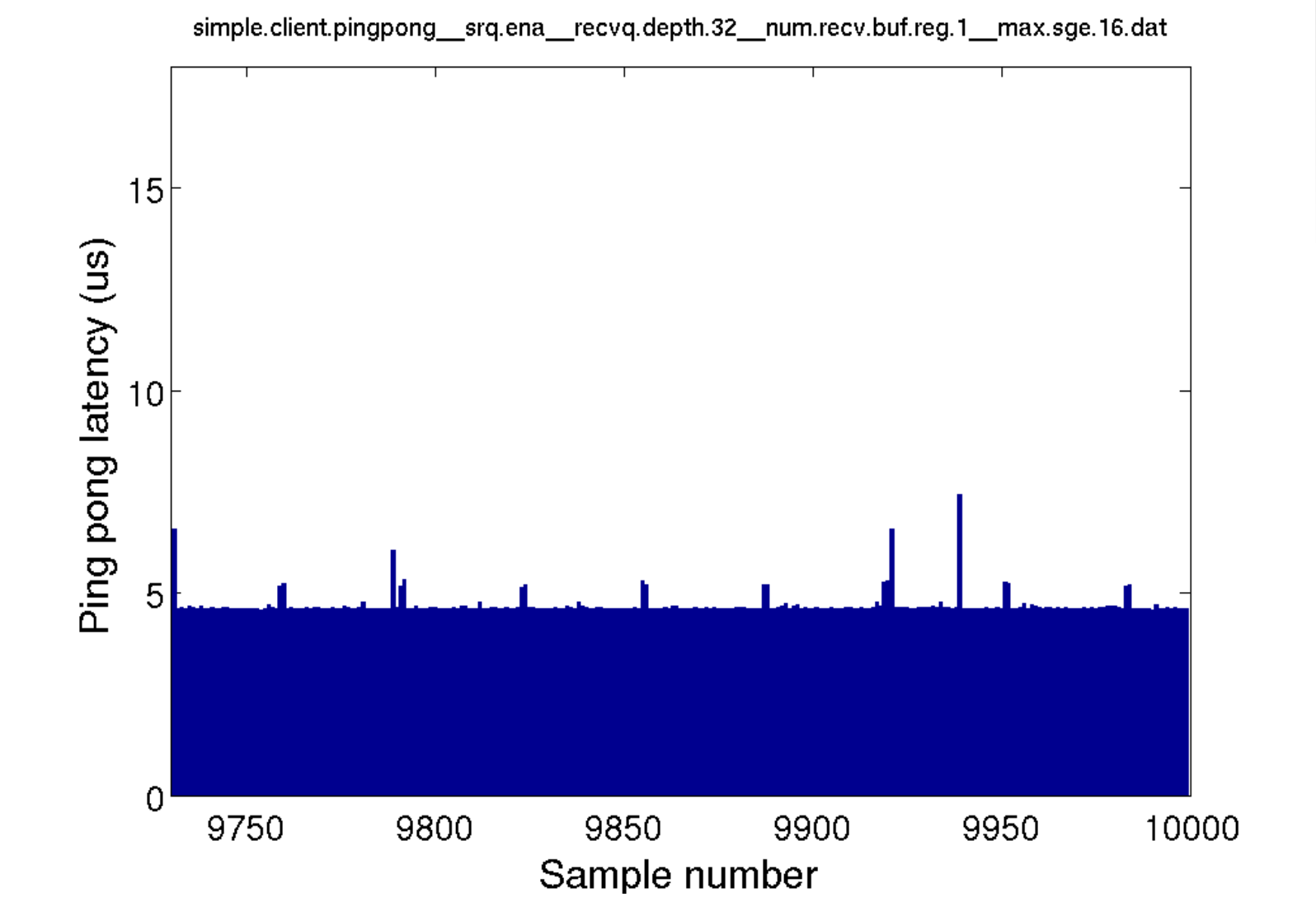
Maximum SGE in receive request

N = 32 max_sge = 8 thru 15



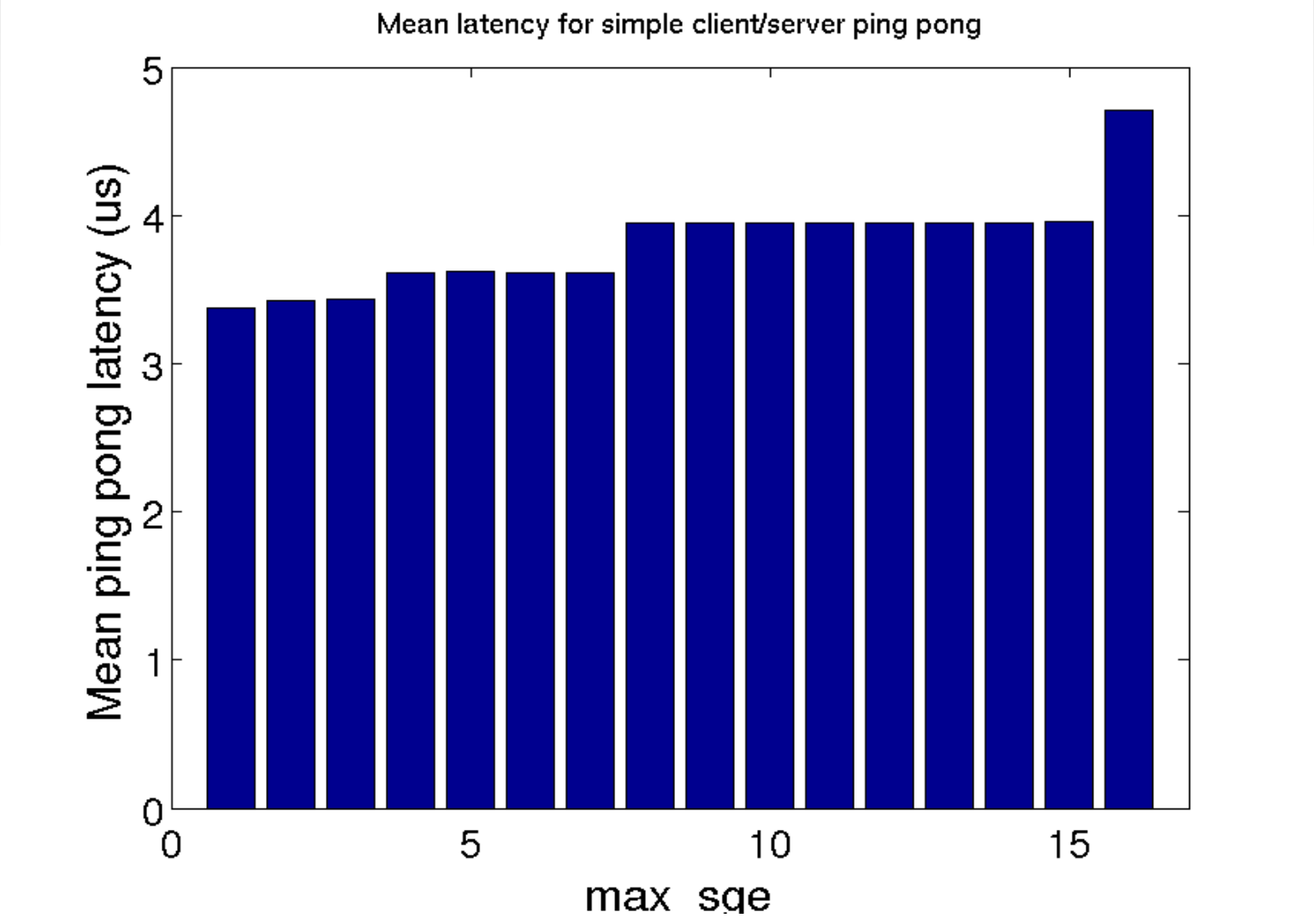
Maximum SGE in receive request

N = 32 max_sge = 16



Maximum SGE in receive request

N = 32



Impact on RAMCloud

Standard performance characteristics (3 replicas)

End-to-end test	max_sge = 8	max_sge = 1
read100	5.5 us	4.9 us
read1K	7.1 us	6.6 us
read10K	10.4 us	9.9 us
read100K	46.7 us	46.2 us
read1M	429.8 us	439.8 us
write100	14.9 us	14.0 us
write1K	18.9 us	17.9 us
write10K	37.4 us	36.9 us
write100K	244.2 us	244.2 us
write1M	2.4 ms	2.3 ms

Q & A