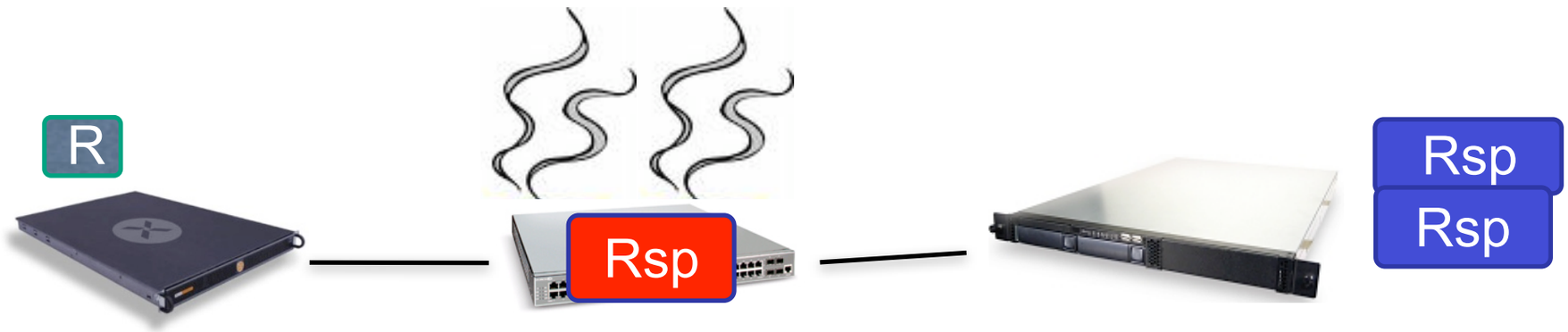
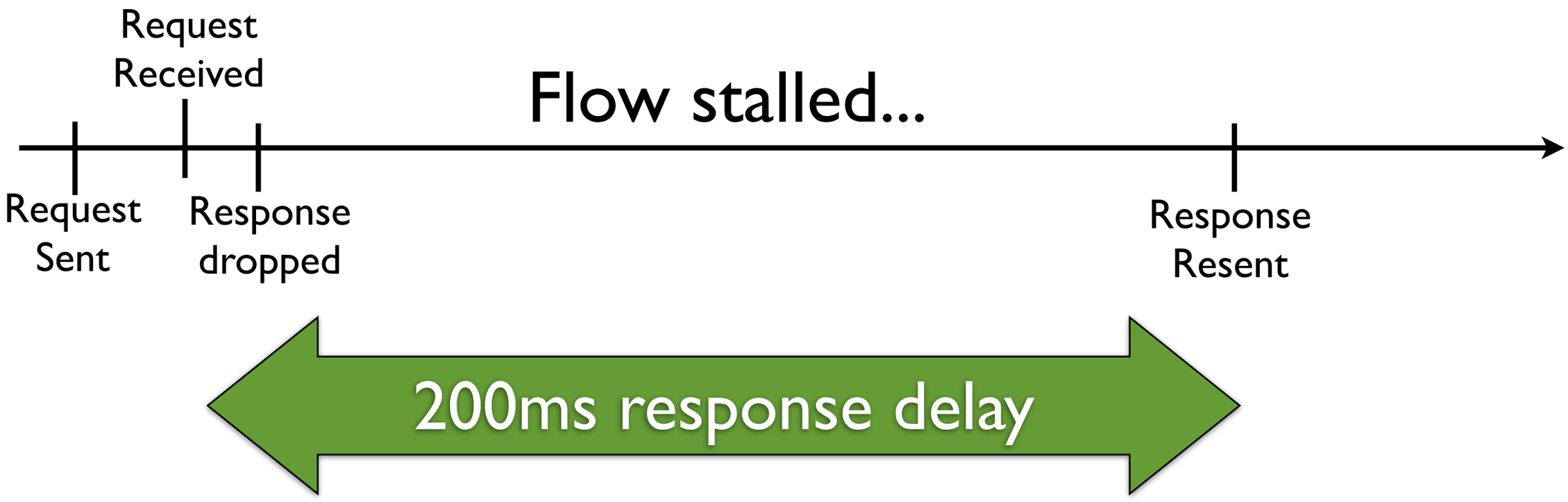
Safe and Effective Fine-grained TCP Retransmissions for Datacenter Communication

Vijay Vasudevan, Amar Phanishayee, Hiral Shah, Elie Krevat
David Andersen, Greg Ganger, Garth Gibson, Brian Mueller*

Carnegie Mellon University, *Panasas Inc.

Datacenter TCP Request-Response





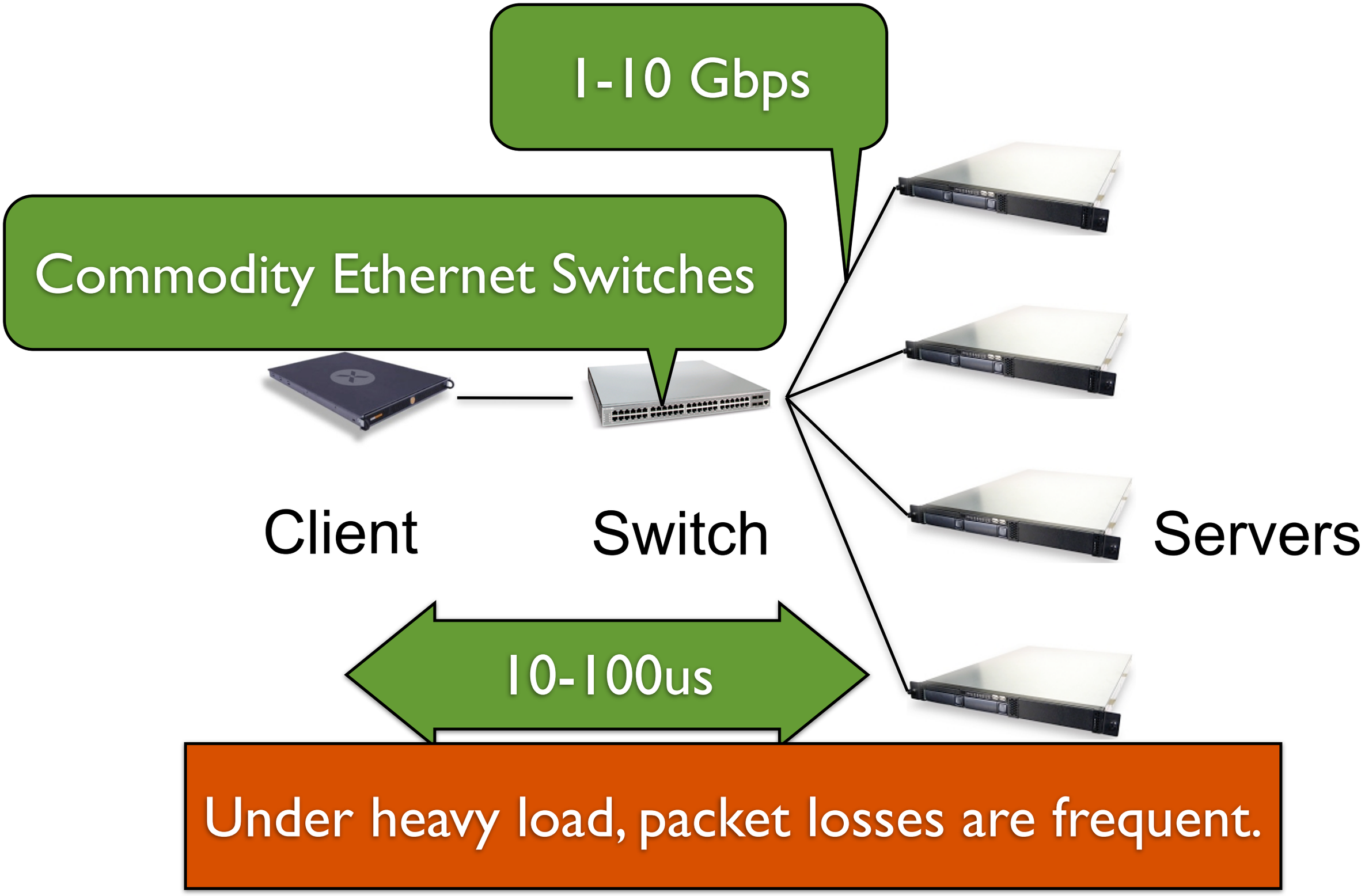
Applications Sensitive to 200ms TCP Timeouts

- "Drive-bys" affecting single-flow request/response
- Barrier-Sync workloads
  - Parallel cluster filesystems (Incast workloads)
  - Massive multi-server queries (e.g., previous talk)
    - Latency-sensitive, customer-facing

Main Takeaways

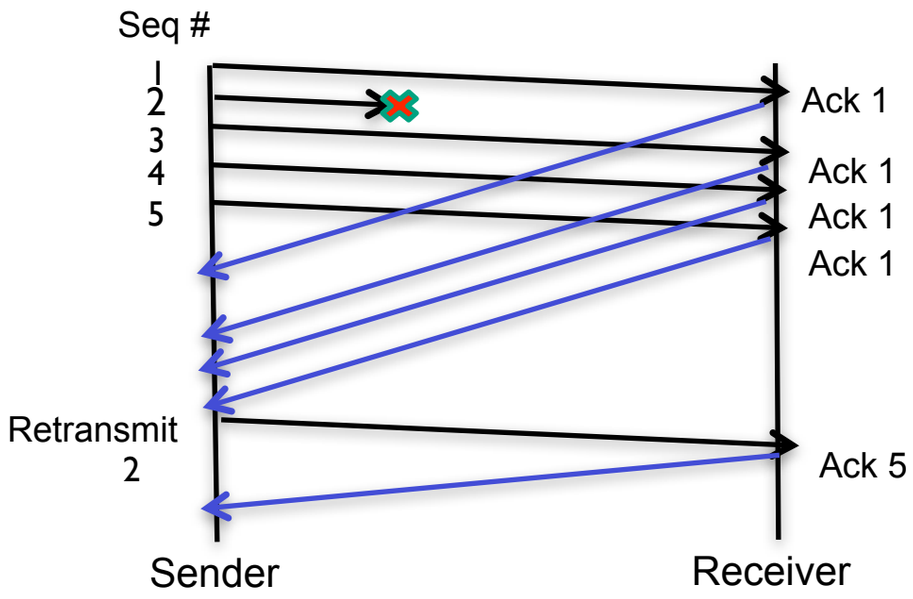
- Problem: 200ms TCP timeouts can cripple datacenter apps
- Solution: Enable microsecond retransmissions
- Can improve datacenter app throughput/latency
- Safe in the wide-area

The Datacenter Environment



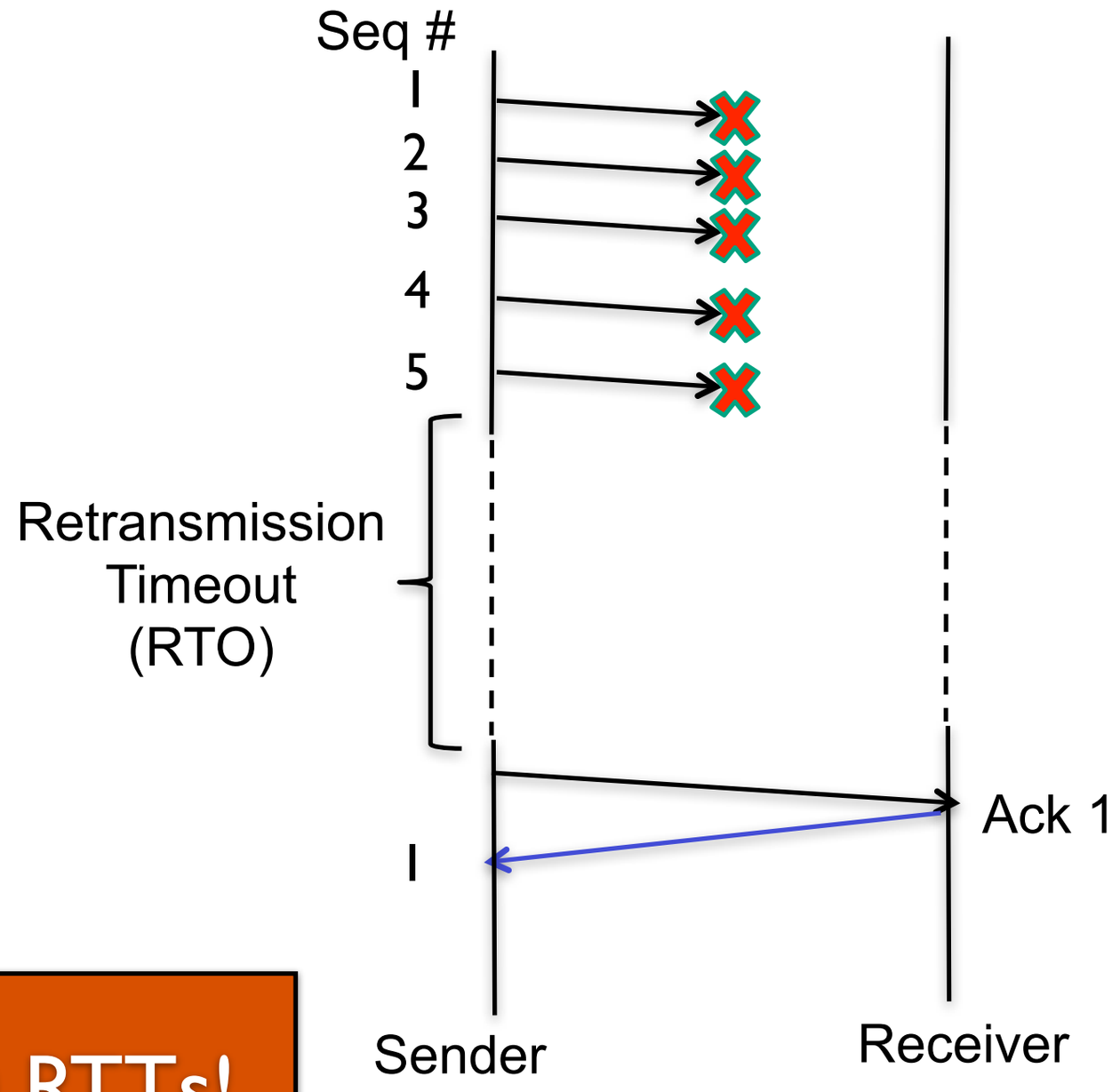
TCP: Loss Recovery Comparison

Data-driven recovery is super fast (us)



minRTO	200.0ms
DC Latency	0.1ms

Timeout driven recovery is painfully slow (ms)



1 TCP Timeout lasts 1000 RTTs!

RTO Estimation and Minimum Bound

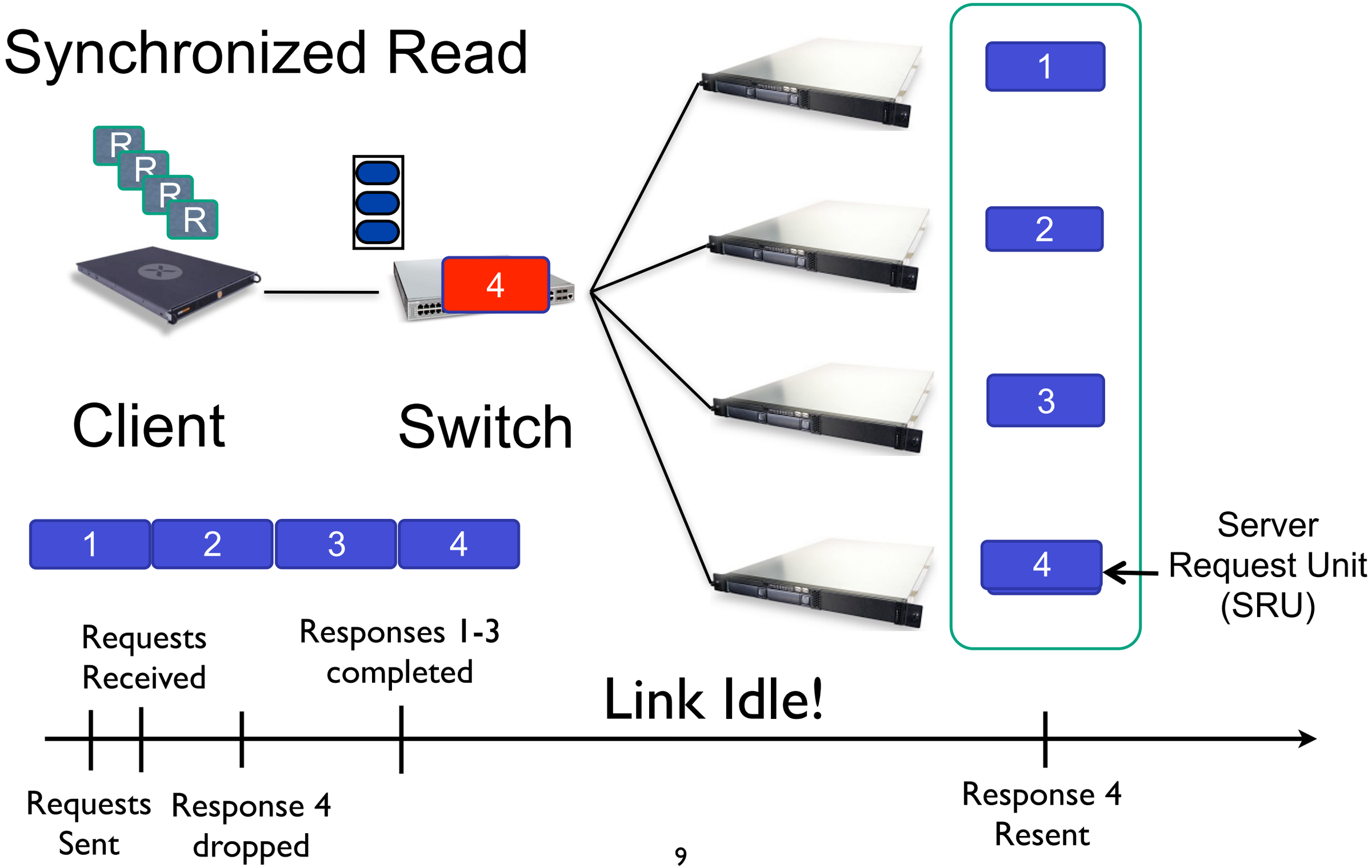
- Jacobson's TCP RTO Estimator
  - RTO = SRTT + 4*RTTVAR
- Minimum RTO bound = 200ms
  - Actual RTO Timer = max(200ms, RTO)

The Incast Workload

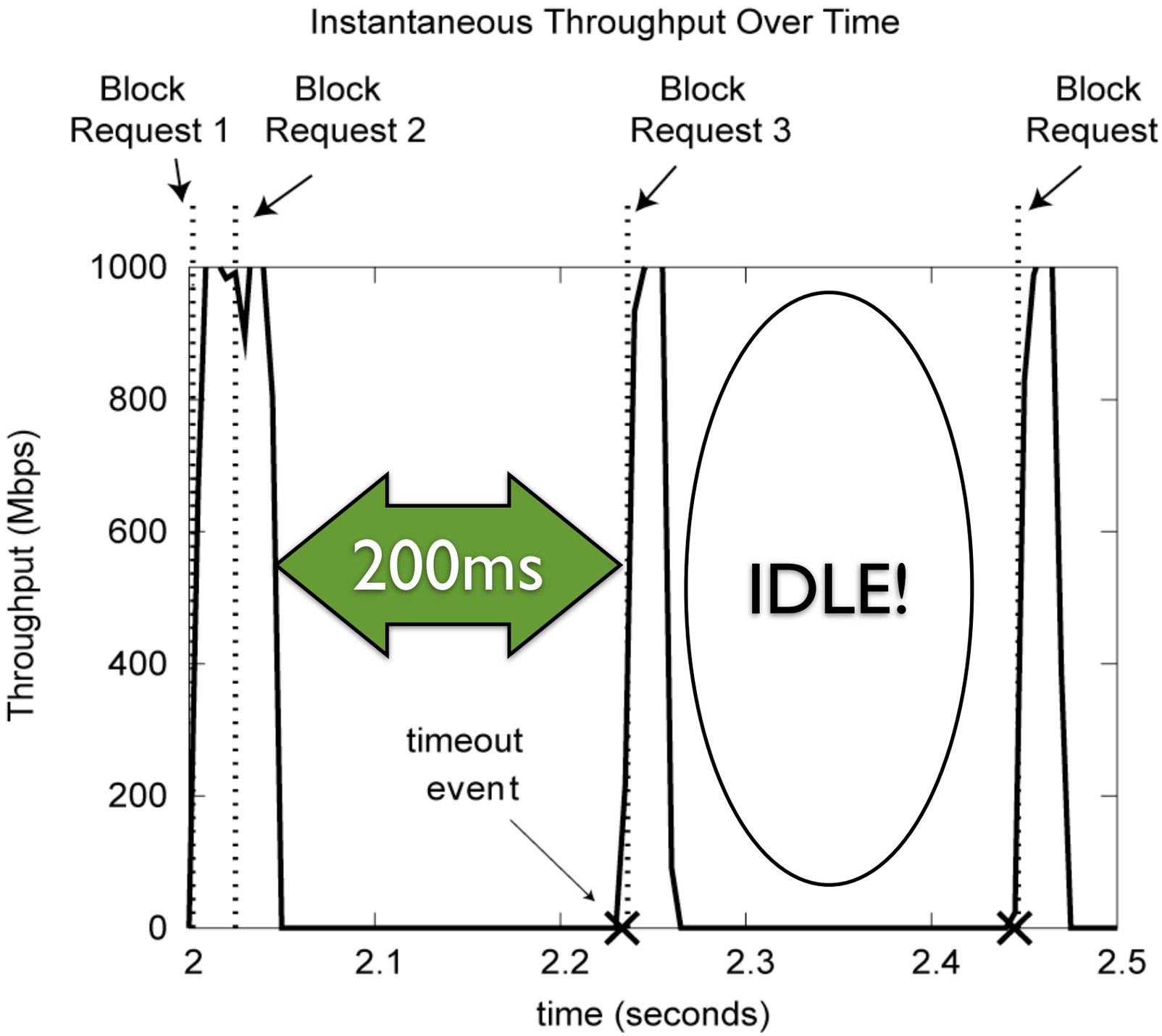
Synchronized Read

Client now sends next batch of requests

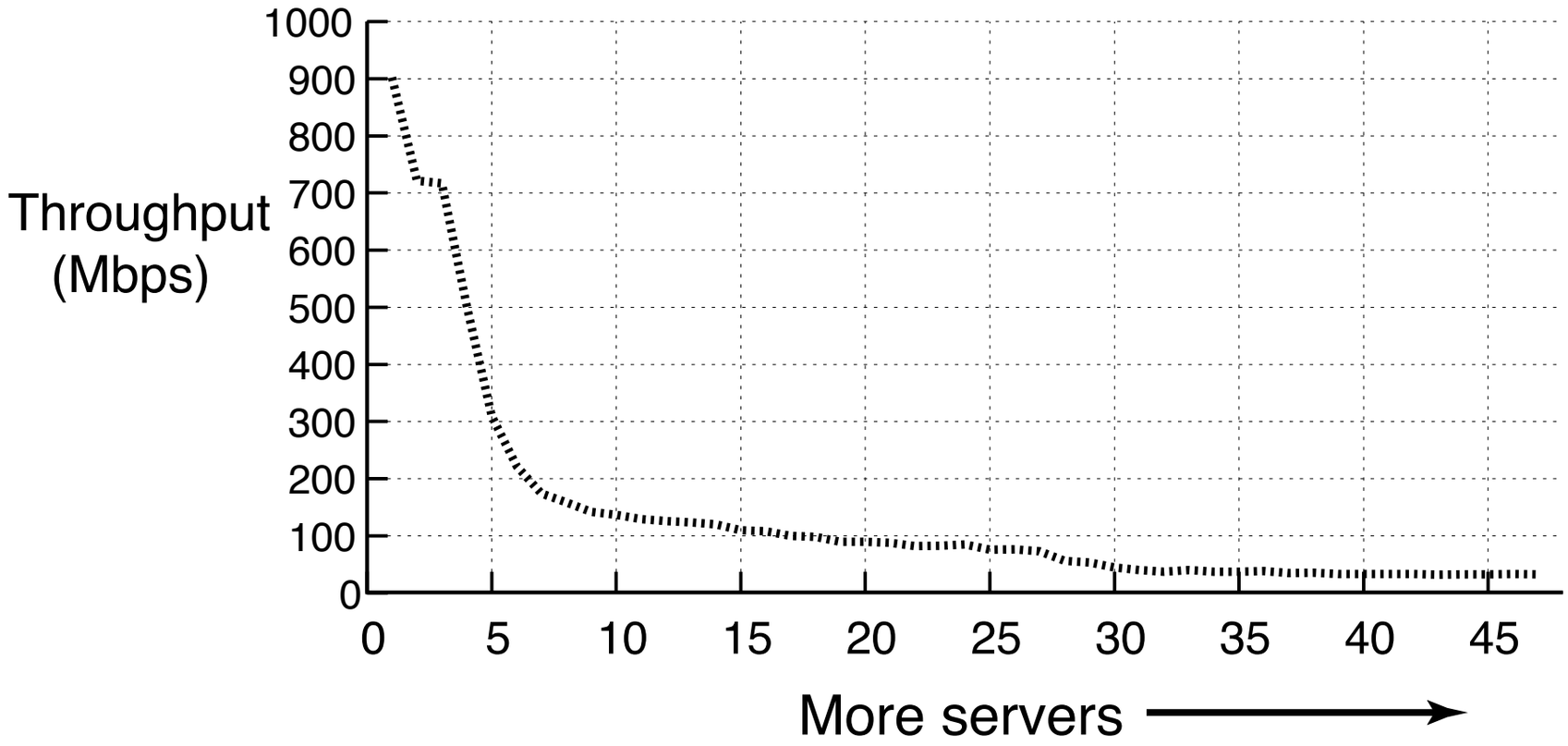
Incast Workload Overfills Buffers



Client Link Utilization



200ms Timeouts Cause Throughput Collapse



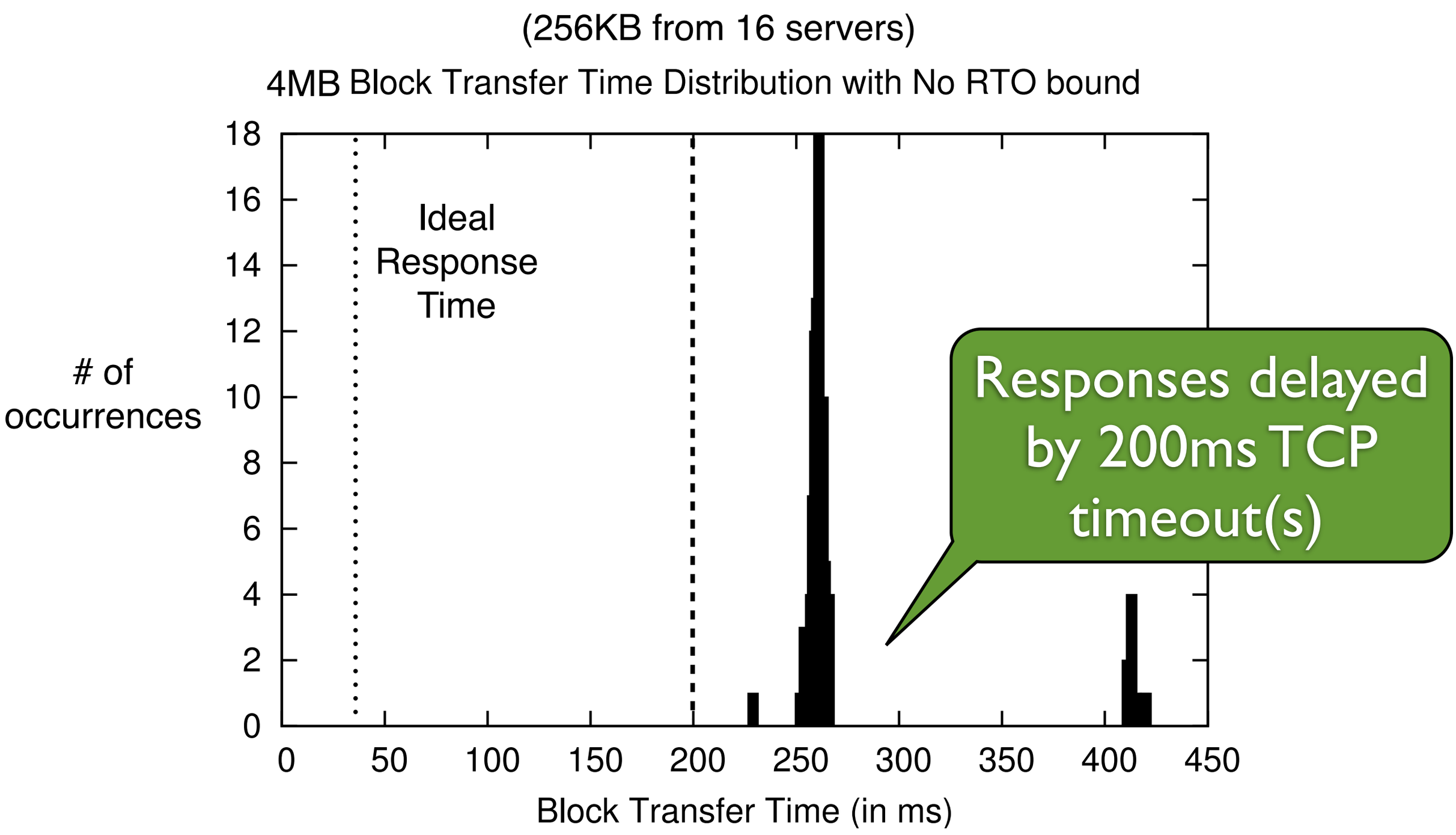
Cluster Environment
1Gbps Ethernet
100us Delay
200ms RTO
S50 Switch
1MB Block Size

- [Nagle04] called this Incast; provided app-level workaround
  - Cause of throughput collapse: 200ms TCP Timeouts
- Prior work: Other TCP variants did not prevent TCP timeouts. [Phanishayee:FAST2008]

Latency-sensitive Apps

- Request for 4MB of data sharded across 16 servers (256KB each)
- How long does it take for all of the 4MB of data to return?

Timeouts Increase Latency



Outline

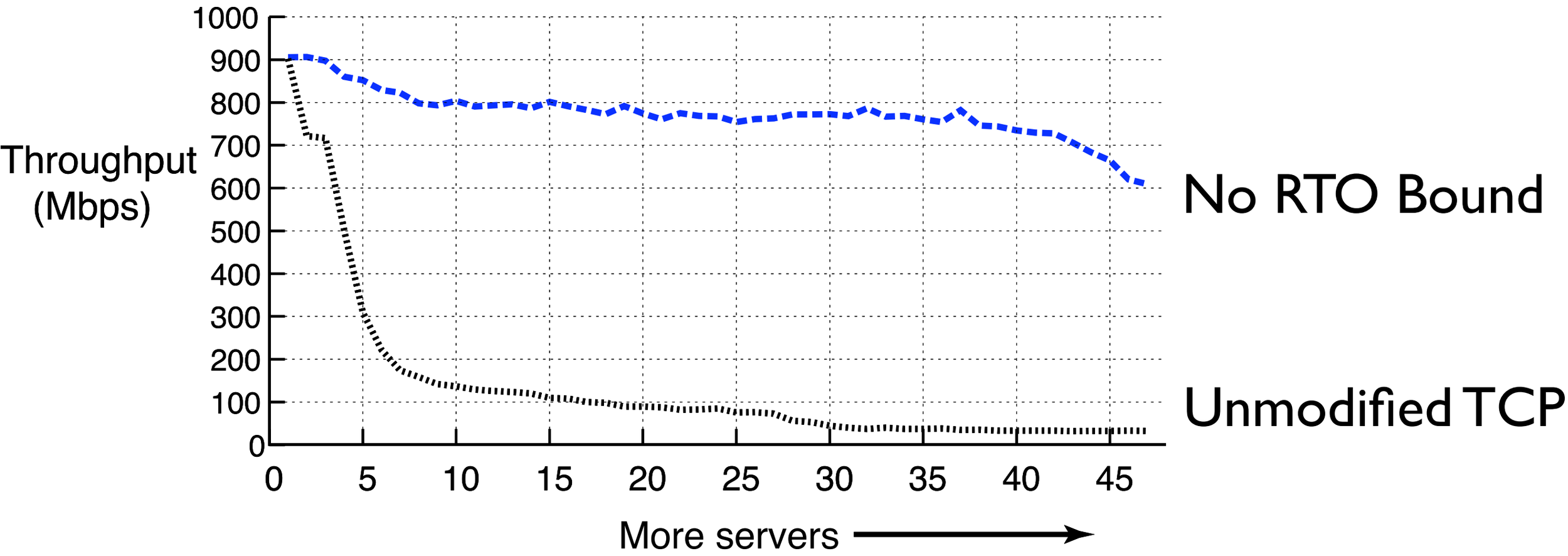
- Problem Description, Examples
- Solution: Microsecond TCP Retransmissions
- Is it safe?

Eliminate minRTO

✓ Simple one-line change in Linux

- Does not change RTT measurement granularity
- Still uses low-resolution, 1ms kernel timers

Eliminating the RTO bound helps

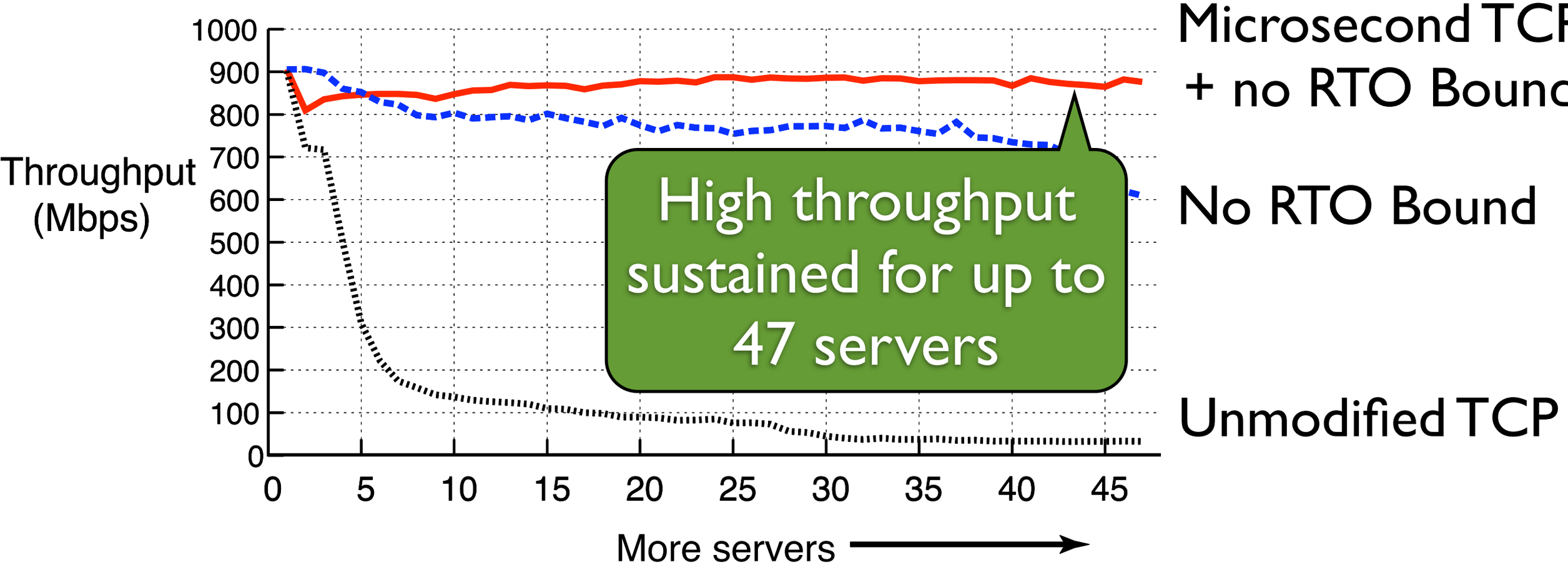


Millisecond retransmissions not enough

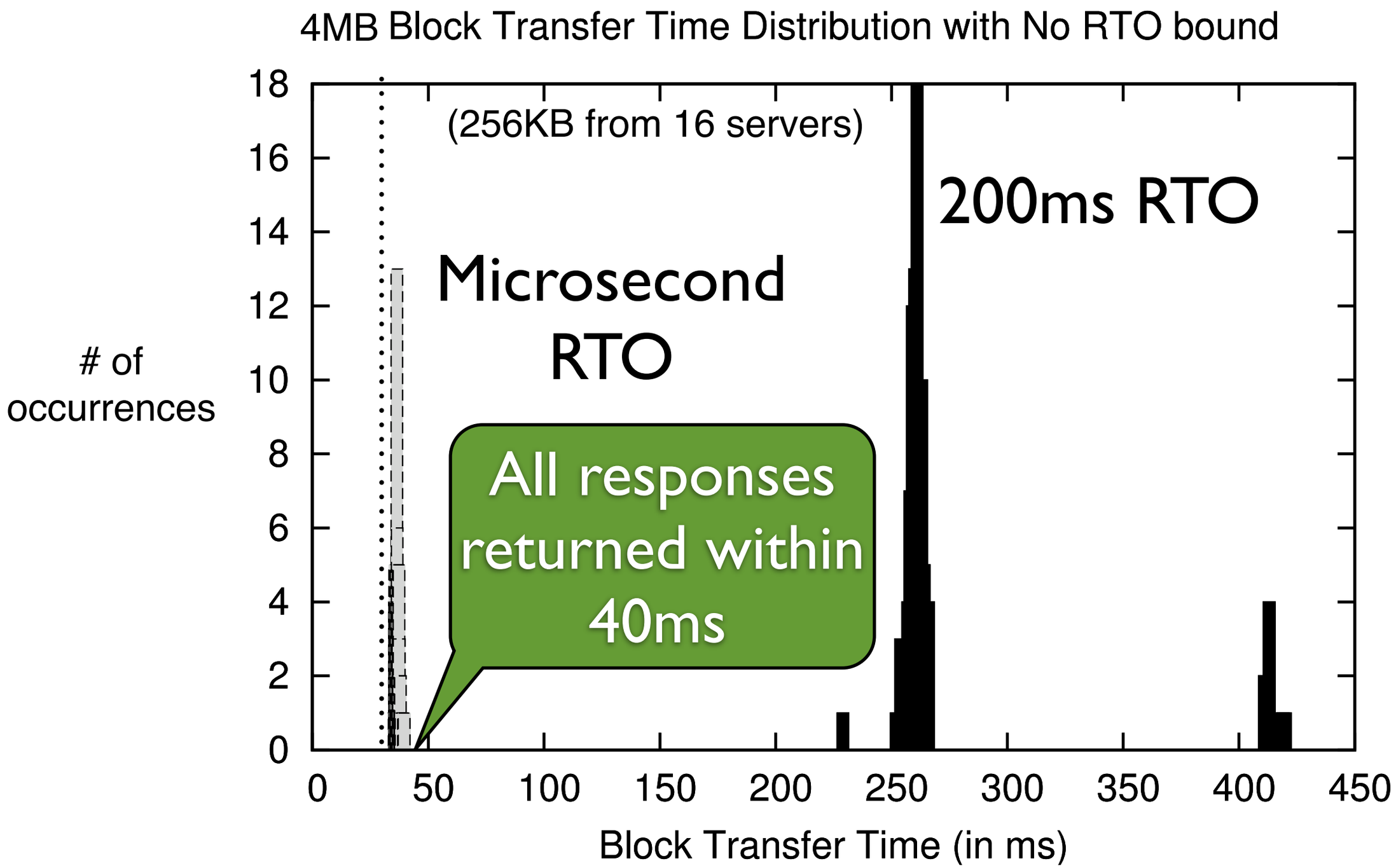
Requirements for Microsecond RTO

- TCP must track RTT in microseconds
  - Modify internal data-structures
  - Timestamp option (backwards compatible)
- Efficient high-resolution kernel timers
  - Use HPET for efficient interrupt signaling

Microsecond timeouts are necessary



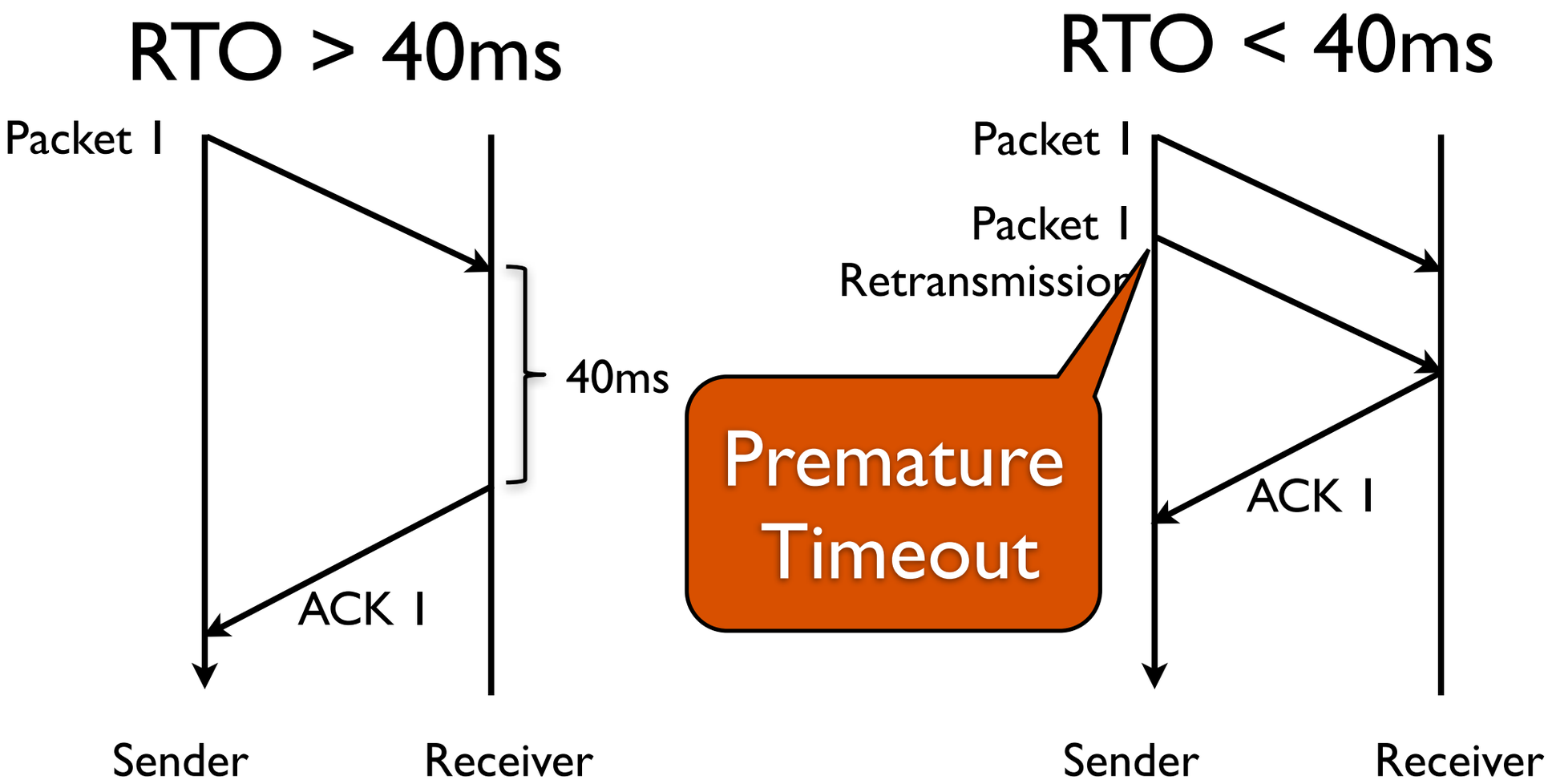
Improvement to Latency



Outline

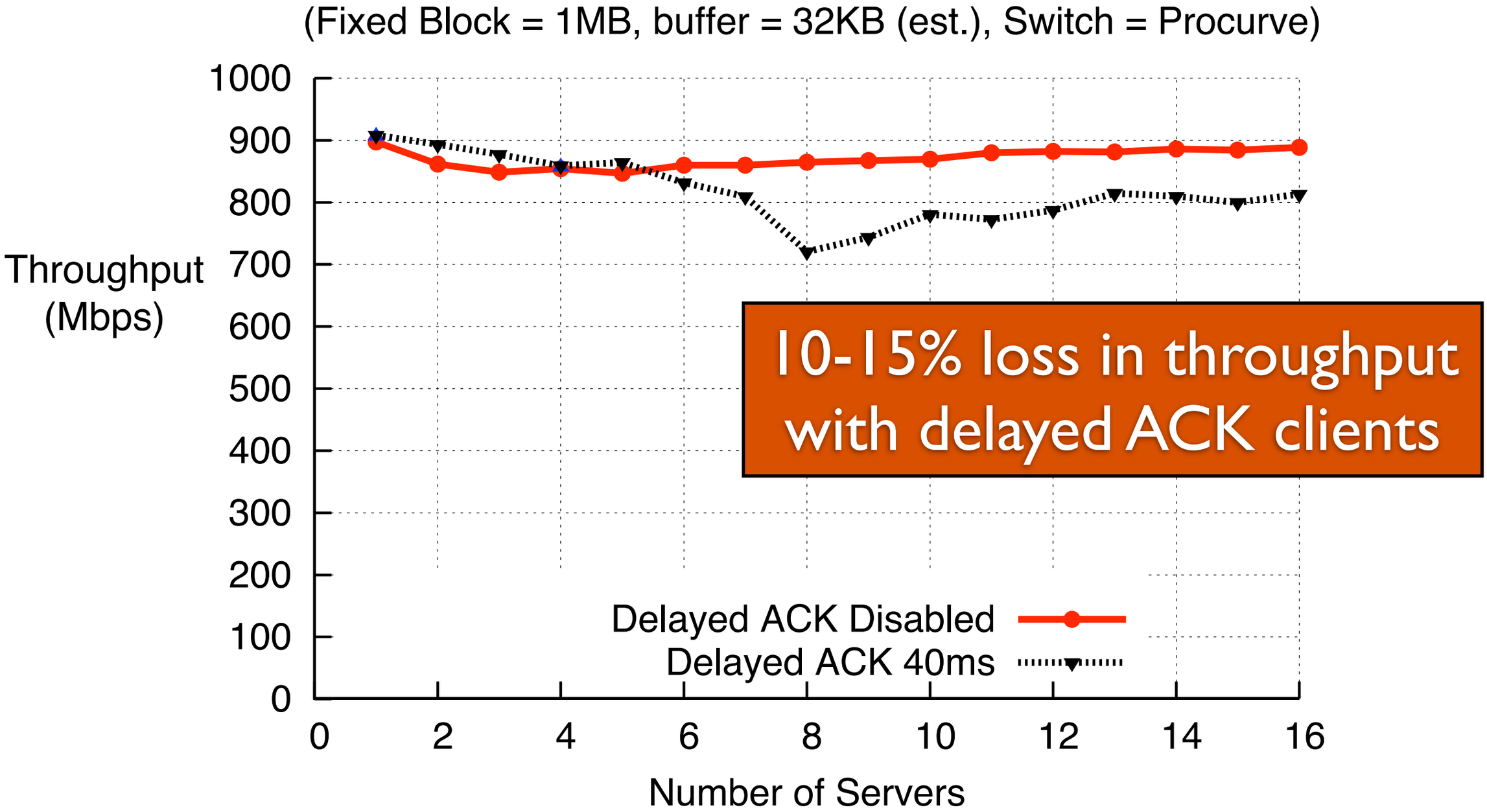
- Problem Description, Examples
- Solution: Microsecond TCP Retransmissions
- Is it safe?
  - Interaction with Delayed ACK
  - Performance in the wide-area

µs RTO and Delayed ACK



RTO on sender triggers
before delayed ACK on receiver

Impact of Delayed ACKs



Is it safe for the wide area?

- Potential Concerns:
  - Stability: Could we cause congestion collapse?
  - Performance: Do we often timeout unnecessarily?
- Stability preserved
  - Timeouts retain exponential backoff
  - Spurious timeouts *slow* rate of transfer
- Performance: spurious timeouts vs. timely response
  - No optimal RTO estimator [Allman99]

Spurious timeouts less harmful

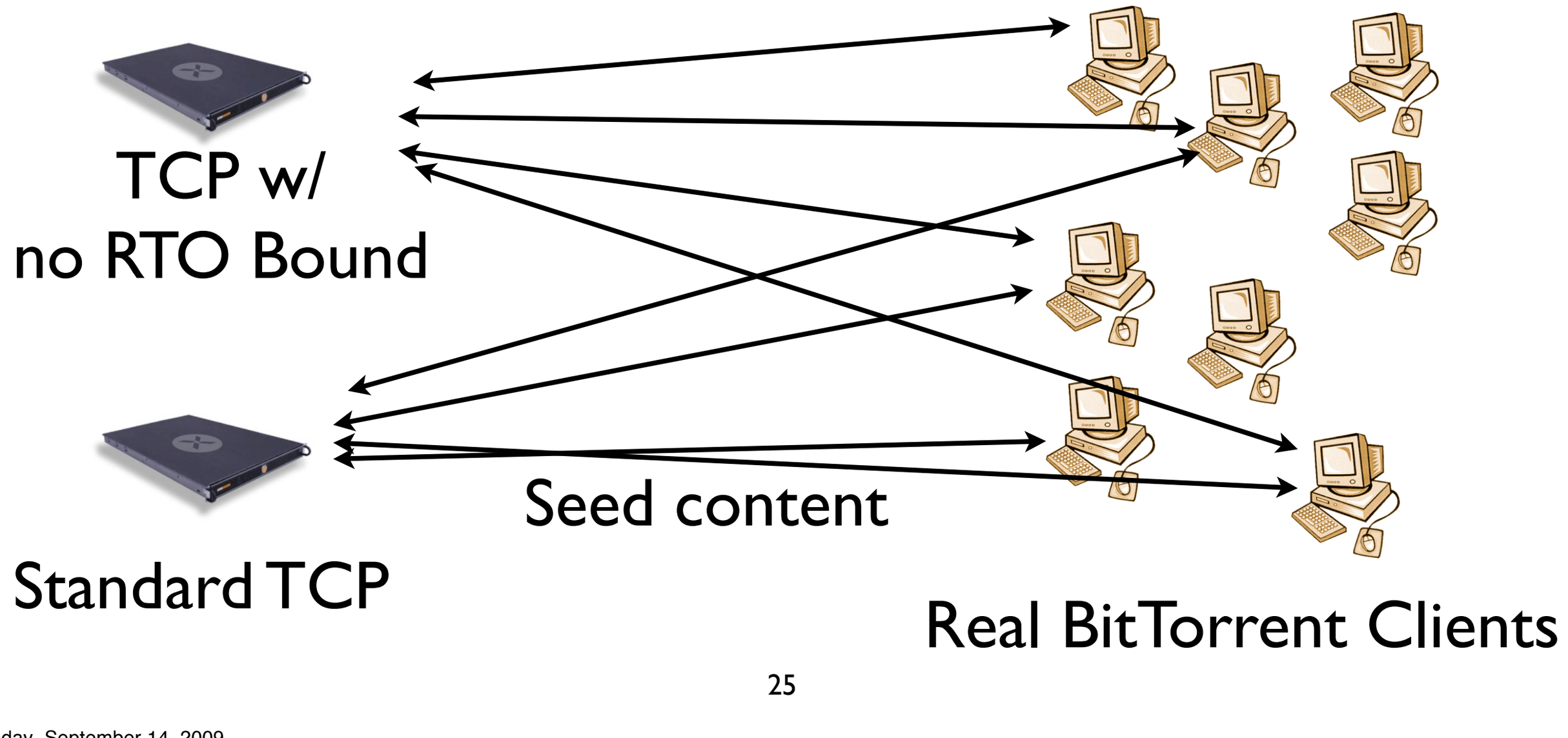
Today's TCP has mechanisms to:

1. Detect spurious timeouts
   - Using TCP timestamp option
2. *Recover* from spurious timeouts
   - Forward RTO (F-RTO)

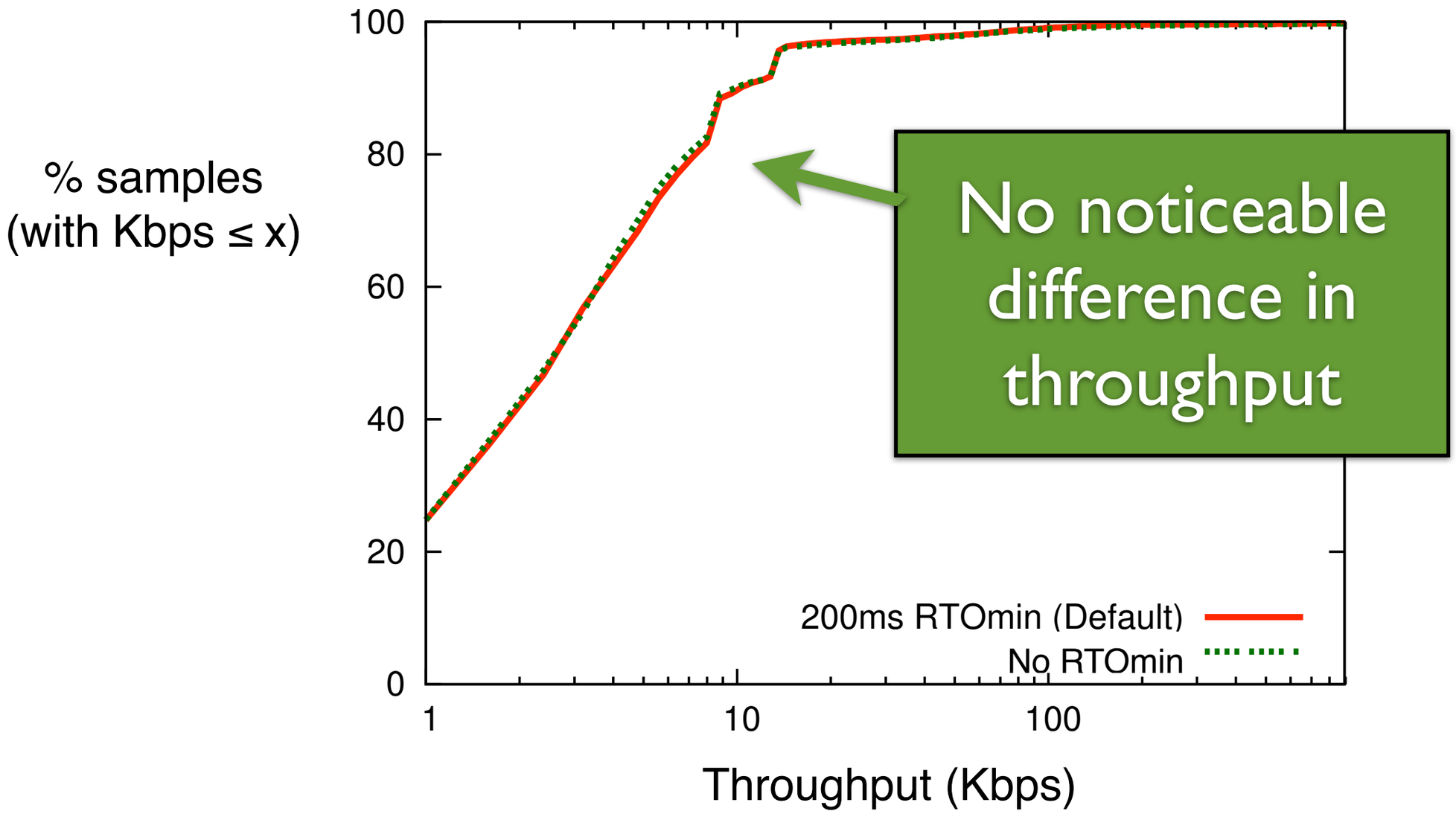
Both implemented widely!

Wide-area Performance Without minRTO

- Do microsecond timeouts harm wide-area throughput?



Wide-area Results



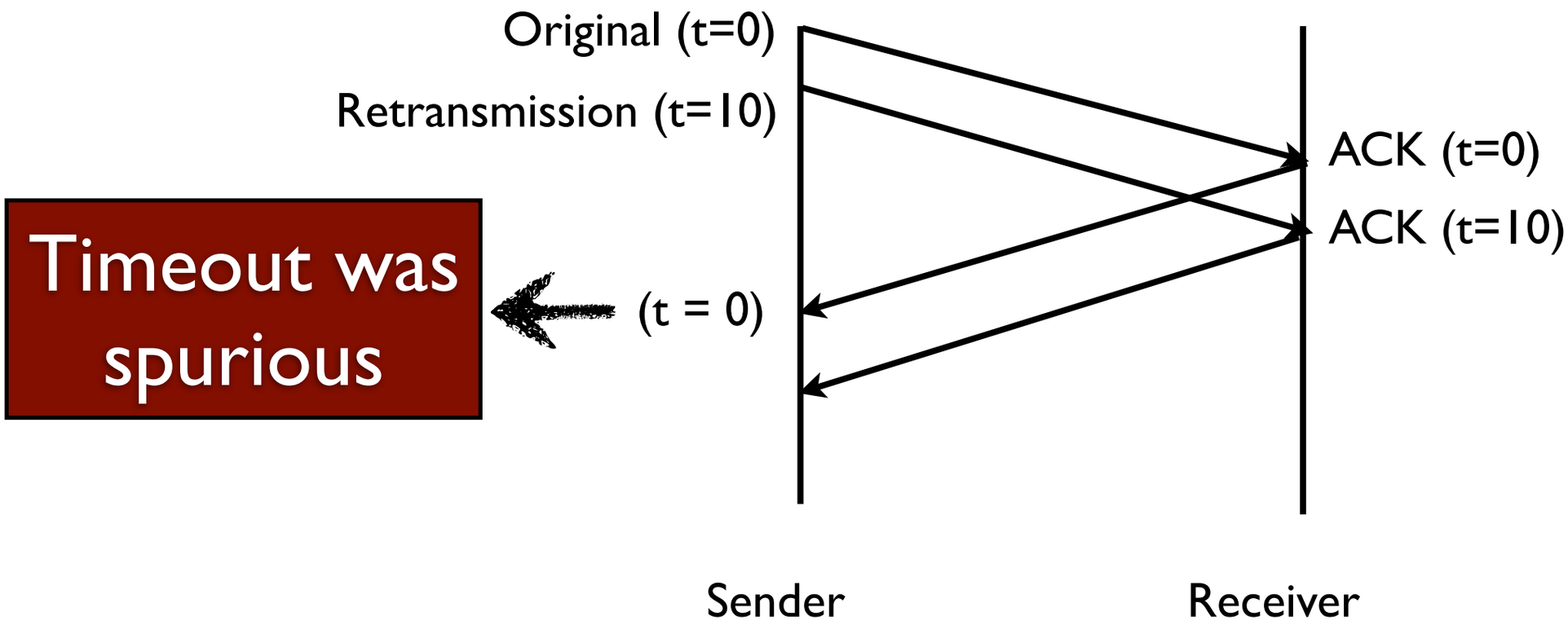
- Few total timeouts (spurious or legitimate)

Conclusion

- Microsecond RTOs can help datacenter application response time and throughput
- Safe for wide-area communication as well
- Linux patch available:
  - http://www.cs.cmu.edu/~vrv/incast/

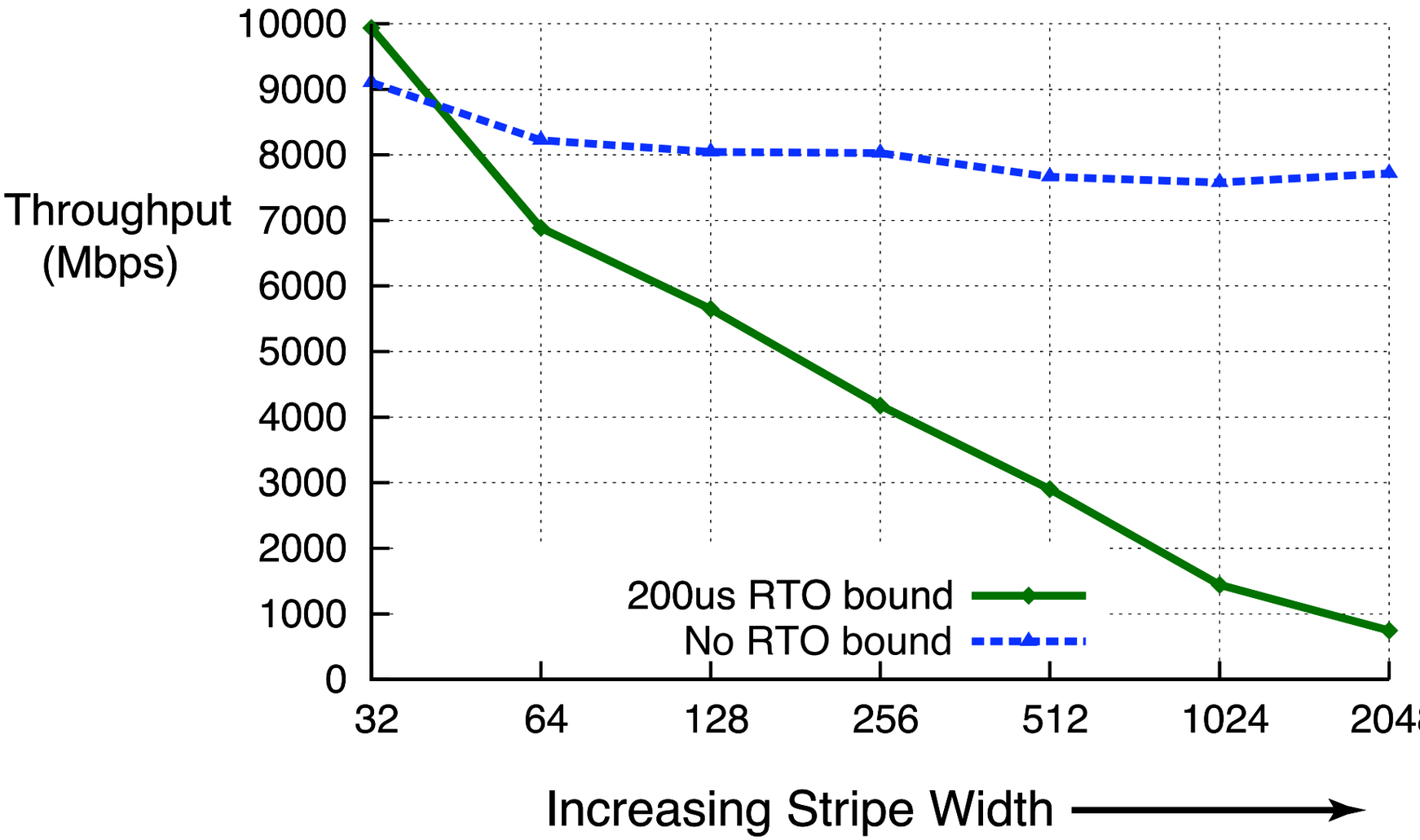
Reasons for Relief in 2009

- No ACK ambiguity with TCP timestamp option



- Forward RTO Recovery (RFC4138) more widely deployed (standard in Linux)

The Need for Microsecond Timeouts



Simulation Environment
10Gbps Ethernet
20us Delay
40MB Block Size

- Future datacenters: More bandwidth, less delay, more servers
- Retransmissions should not be bounded

Control Parameter