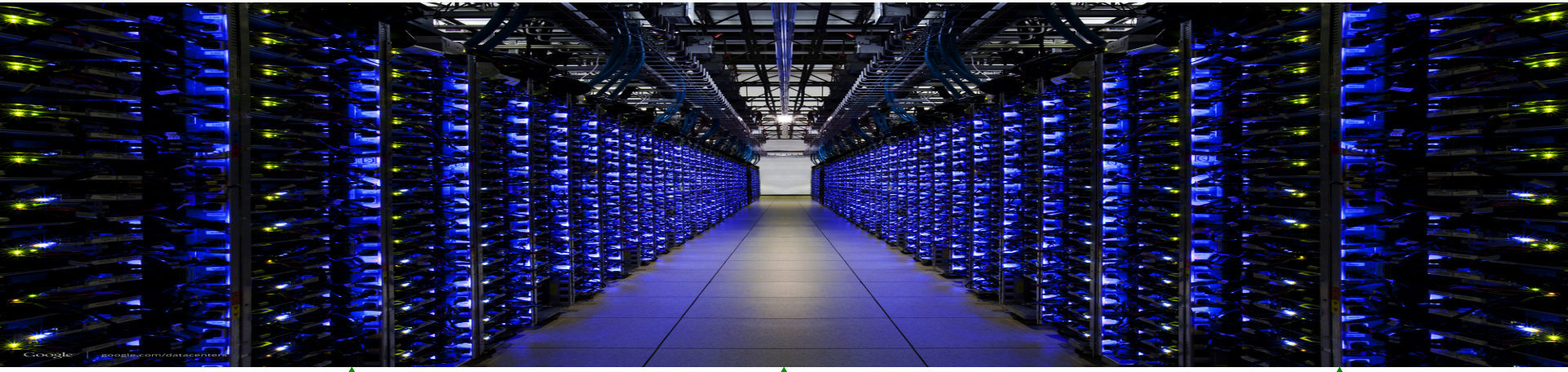
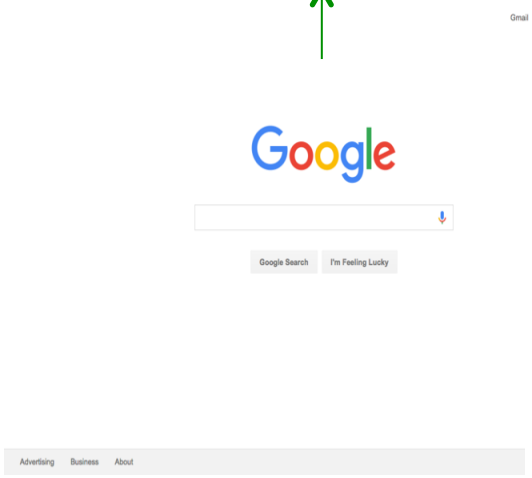
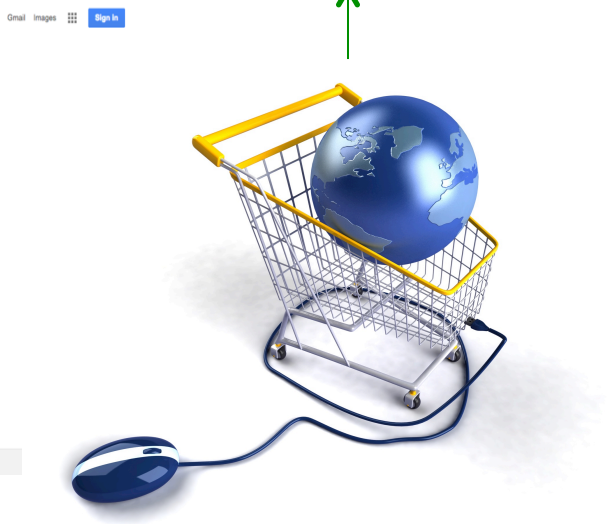
Deconstructing the Tail at Scale Effect Across Network Protocols

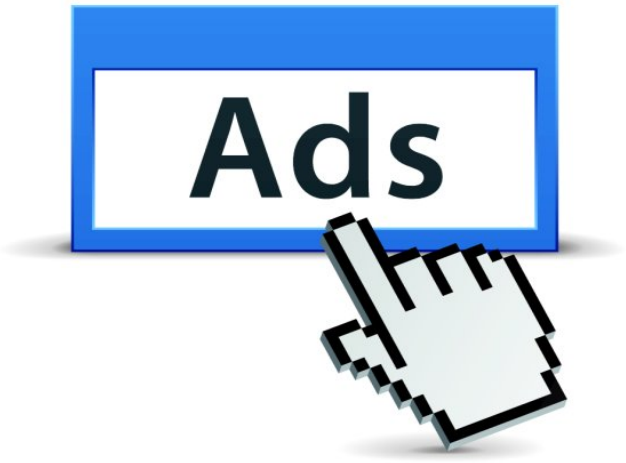
Akshitha Sriraman, Sihang Liu, Sinan Gunbay, Shan Su,
Thomas F. Wenisch
University of Michigan, Ann Arbor

Online Data Intensive Applications (OLDI)



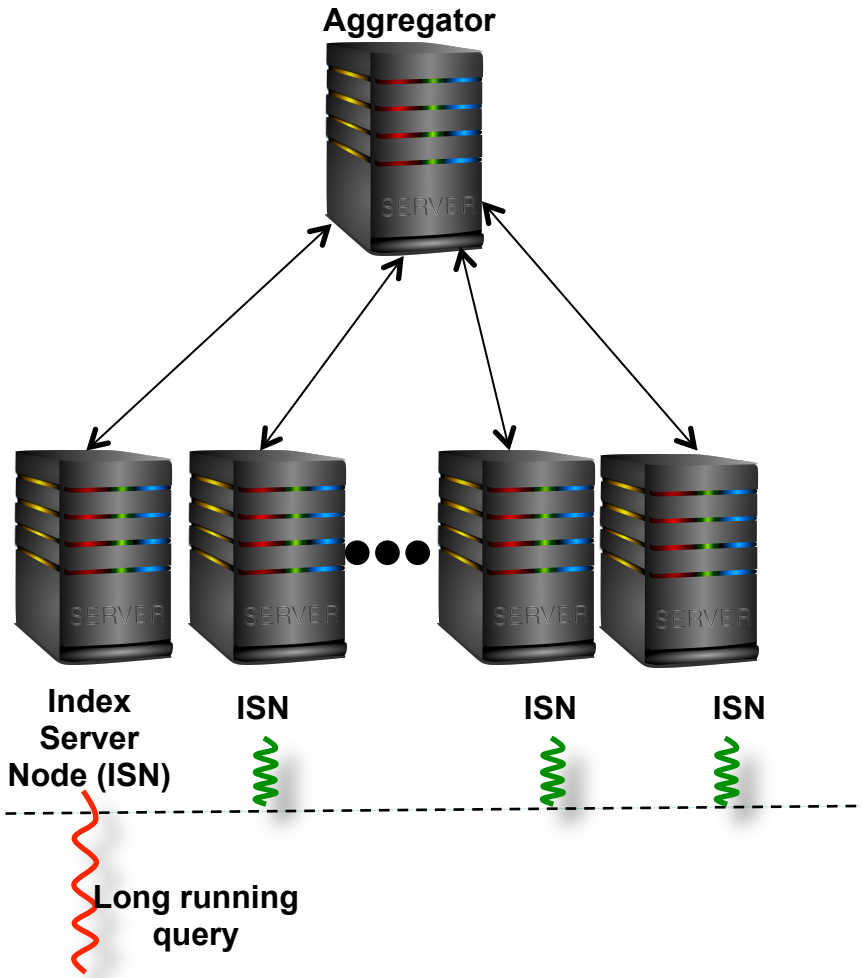


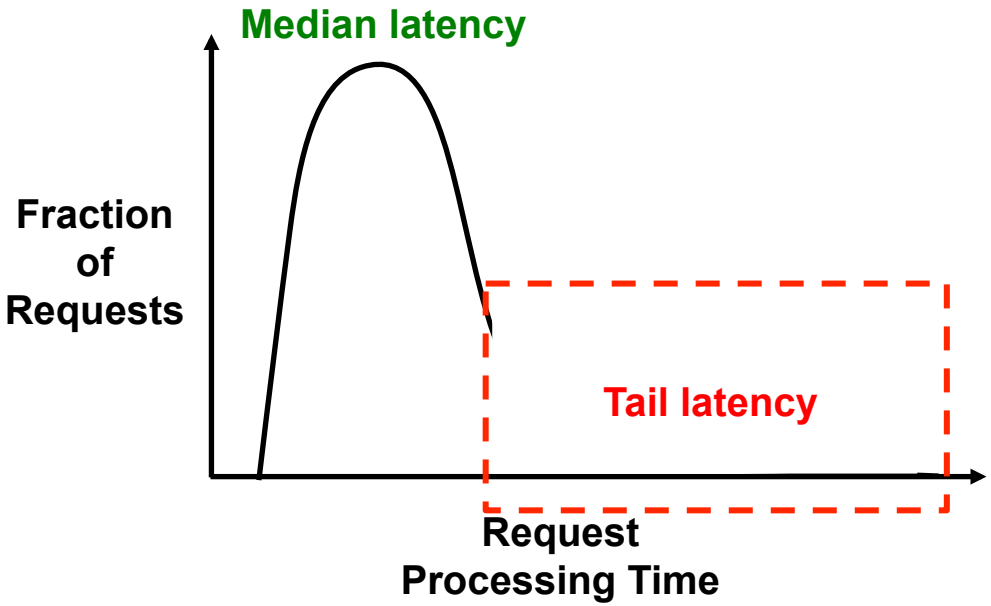




Process TBs of data with O(ms) request latency

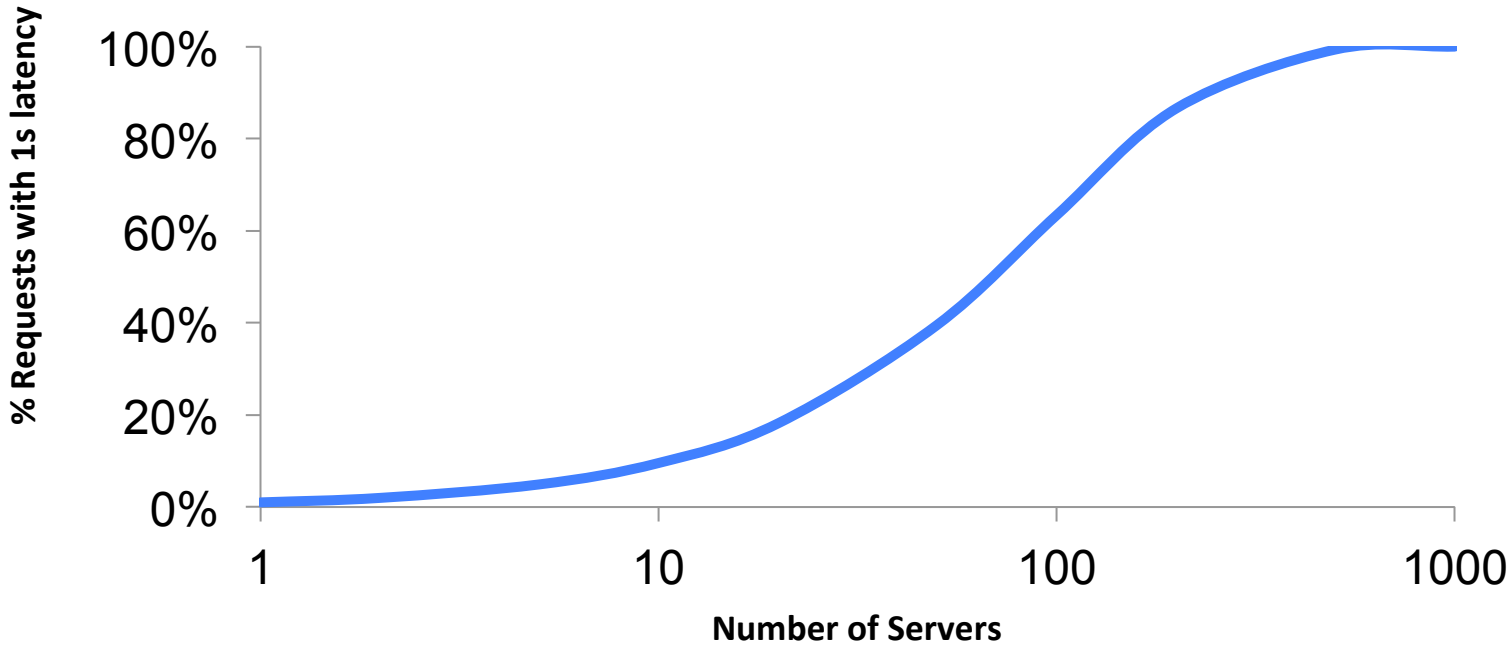
Tail latency and its significance





95th, 99th percentile tail latencies are critical

The tail at scale [Dean & Barroso '13]



- Latency > 1s:
  - **63%** of requests at 100-node scale
  - **99%** at 500-node scale

What causes tail latency? [Dean '13]

- Global resource sharing
- Background daemons
- Queuing
- Garbage collection
- Maintenance activities

- **NETWORK**

Conventional wisdom

- Prioritizing network flows: latency sensitivity
  [Hong '12, Zats '12, Zhu '14, Wilson '11, Vamanan '12]
  - TCP-IP: Approximate fair sharing
  - Flow scheduling to meet soft real-time deadlines

- Reducing network congestion [Alizadeh '10, Alizadeh '12]
  - Switch queuing delays: congestion notification

Tail latencies exist even in the absence of extrinsic parameters

Our contributions

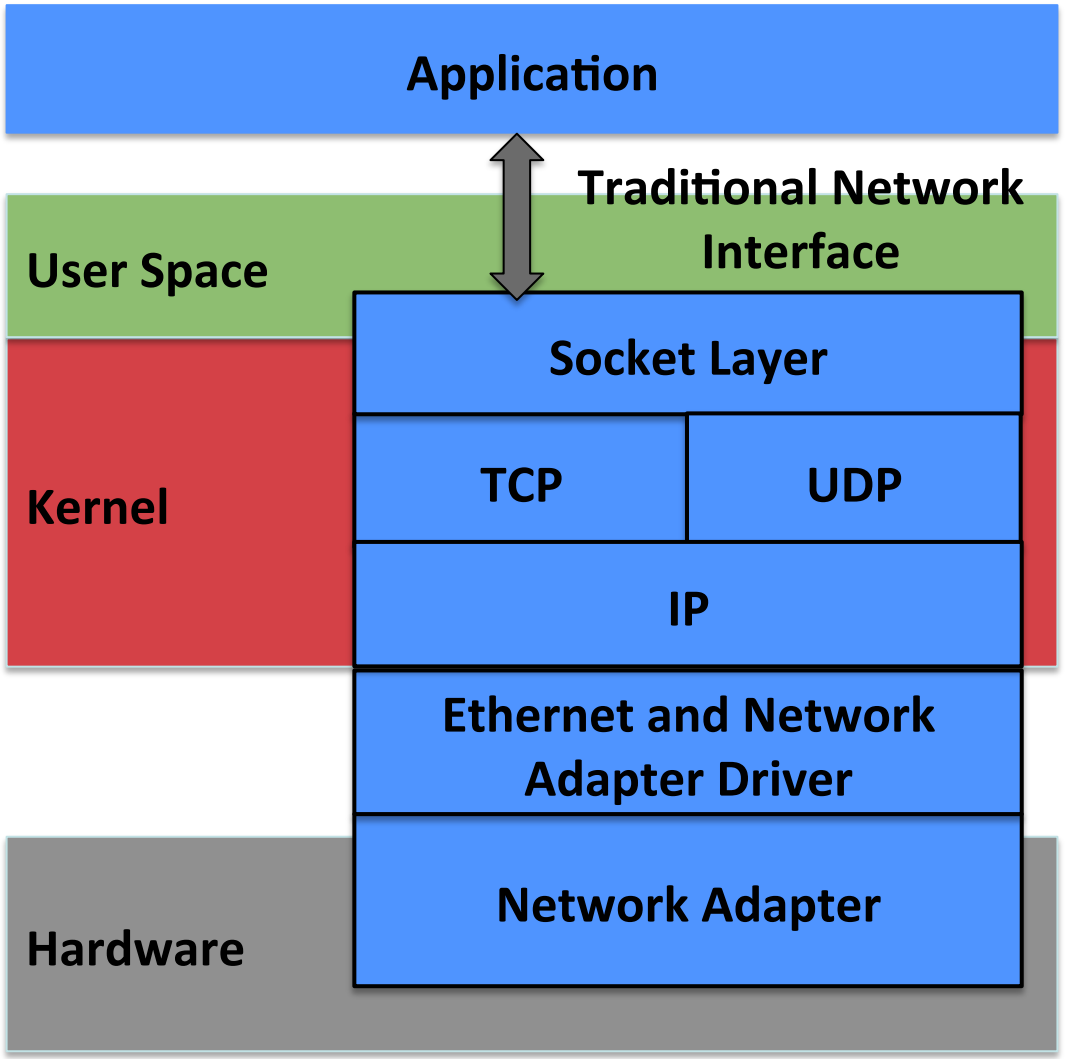
- Deconstruct tail latencies: network protocols
  - TCP, UDP, Remote Direct Memory Access (RDMA)
- Identify extreme tails in common protocols
  - 110x median latency
- Discover surprising source for extreme tails
  - Process of elimination

Rest of this talk

- Network protocols: data path
  - TCP-IP and UDP-IP
  - RDMA
- Identifying the cause for extreme tail latencies
  - Process of elimination

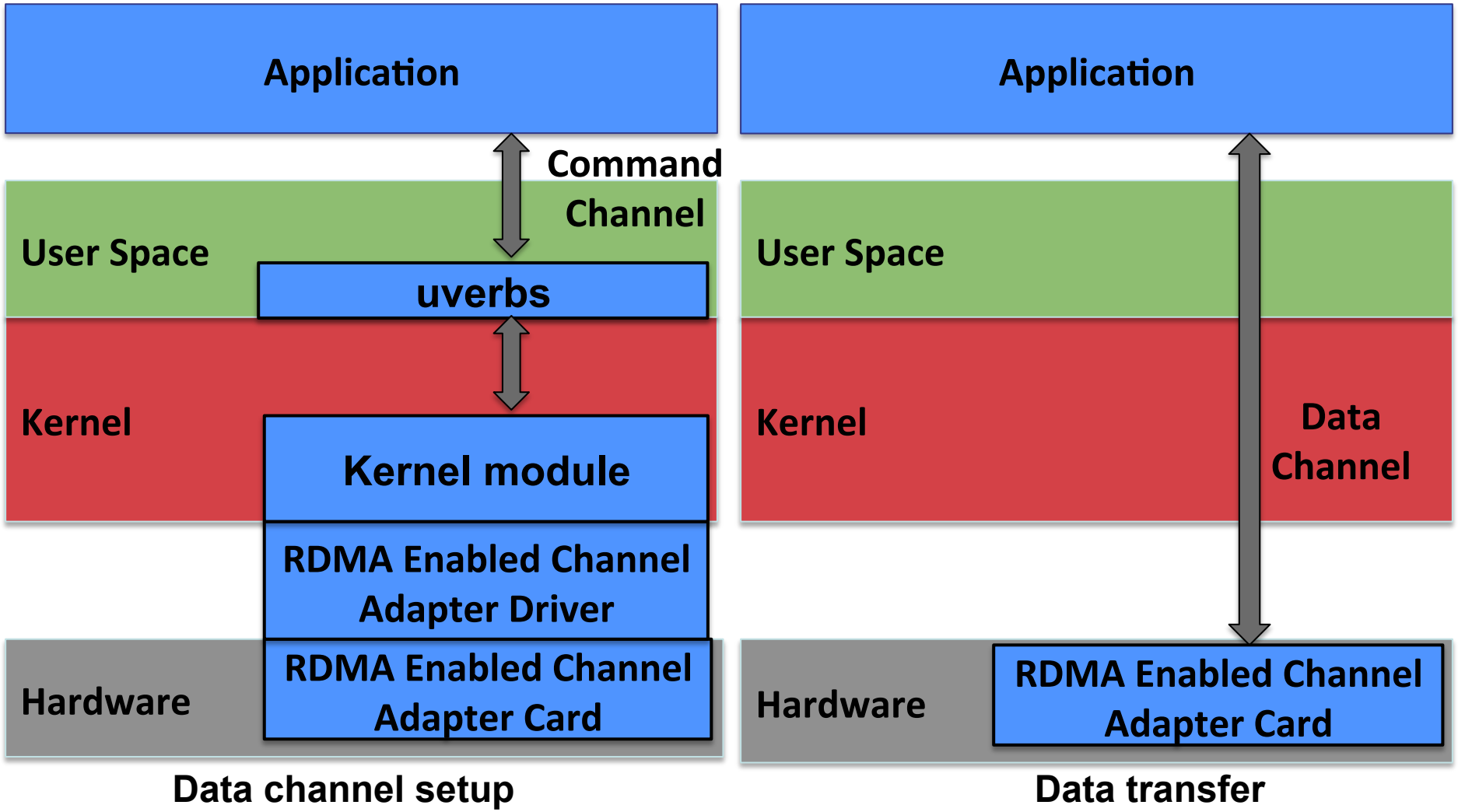
TCP-IP/UDP-IP background

- TCP-IP Vs. UDP-IP:
  - Reliable delivery
  - Error correction
  - Message ordering



TCP/UDP data transfer requires OS support

RDMA background

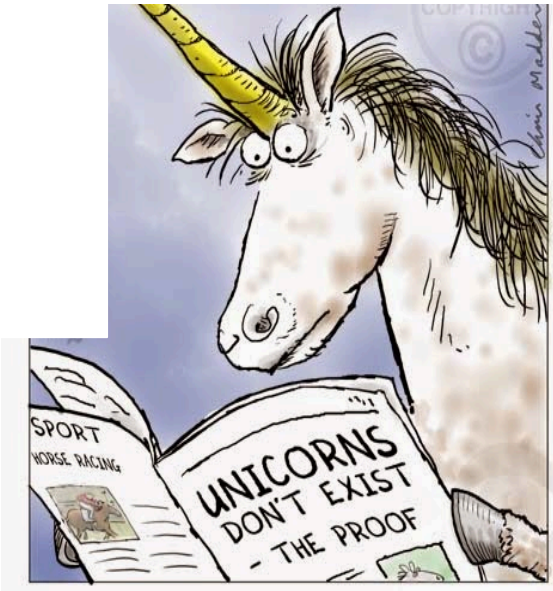


RDMA exchange bypasses OS & is non-blocking

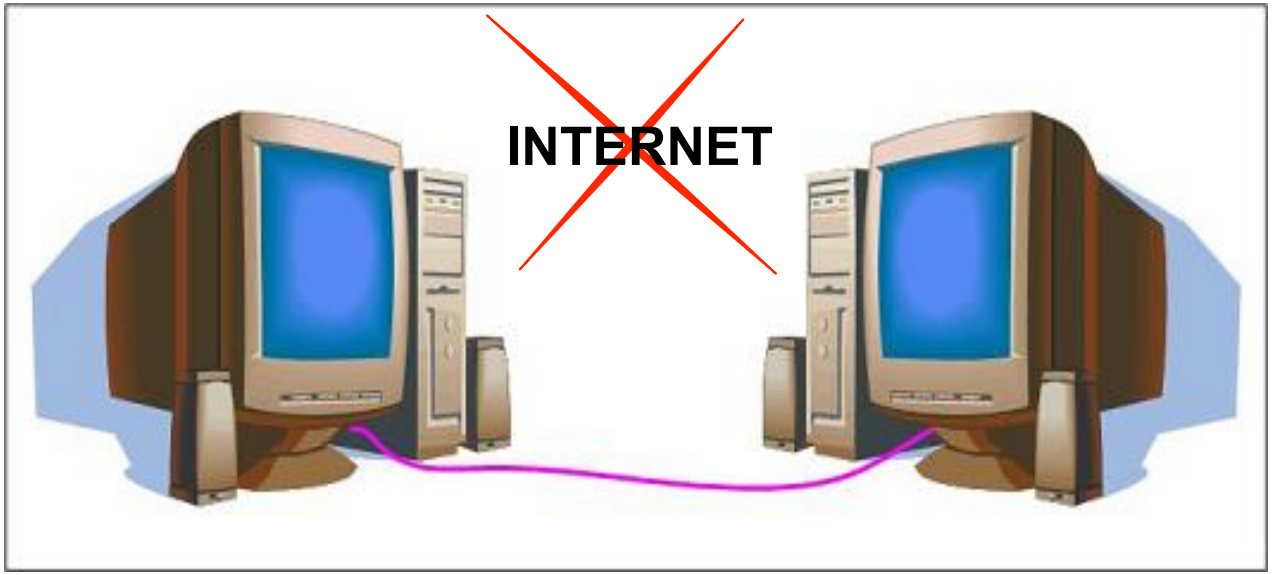
What is the culprit?

- Extrinsic network parameters?
- Bandwidth saturation?
- Blocking system calls?
- Additional TCP protocol operations?
- OS bottlenecks?

Only proof can turn anyone into a culprit



Our system



What is the culprit?

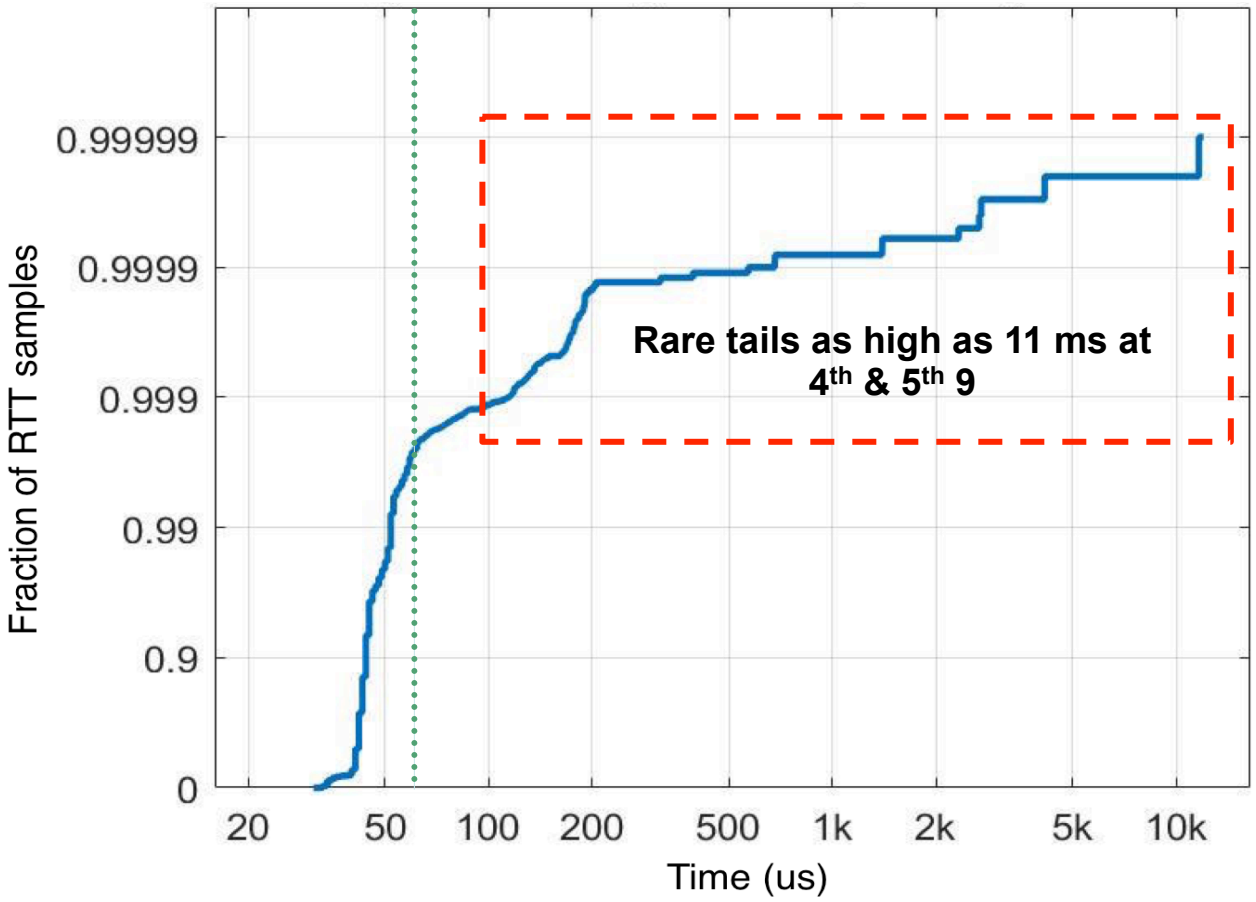
~~1. Extrinsic parameters~~

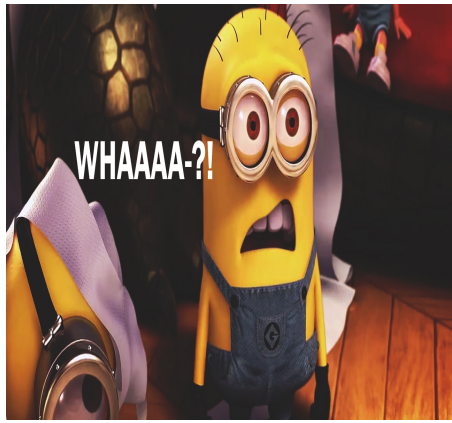
2. Bandwidth saturation
3. Blocking calls
4. TCP-IP complexities
5. Non-OS bottlenecks

- Isolated network of 2 Linux machines
  - NICs are directly attached via Ethernet cabling

Eliminates effect of extrinsic network parameters

TCP-IP: single server-client pair

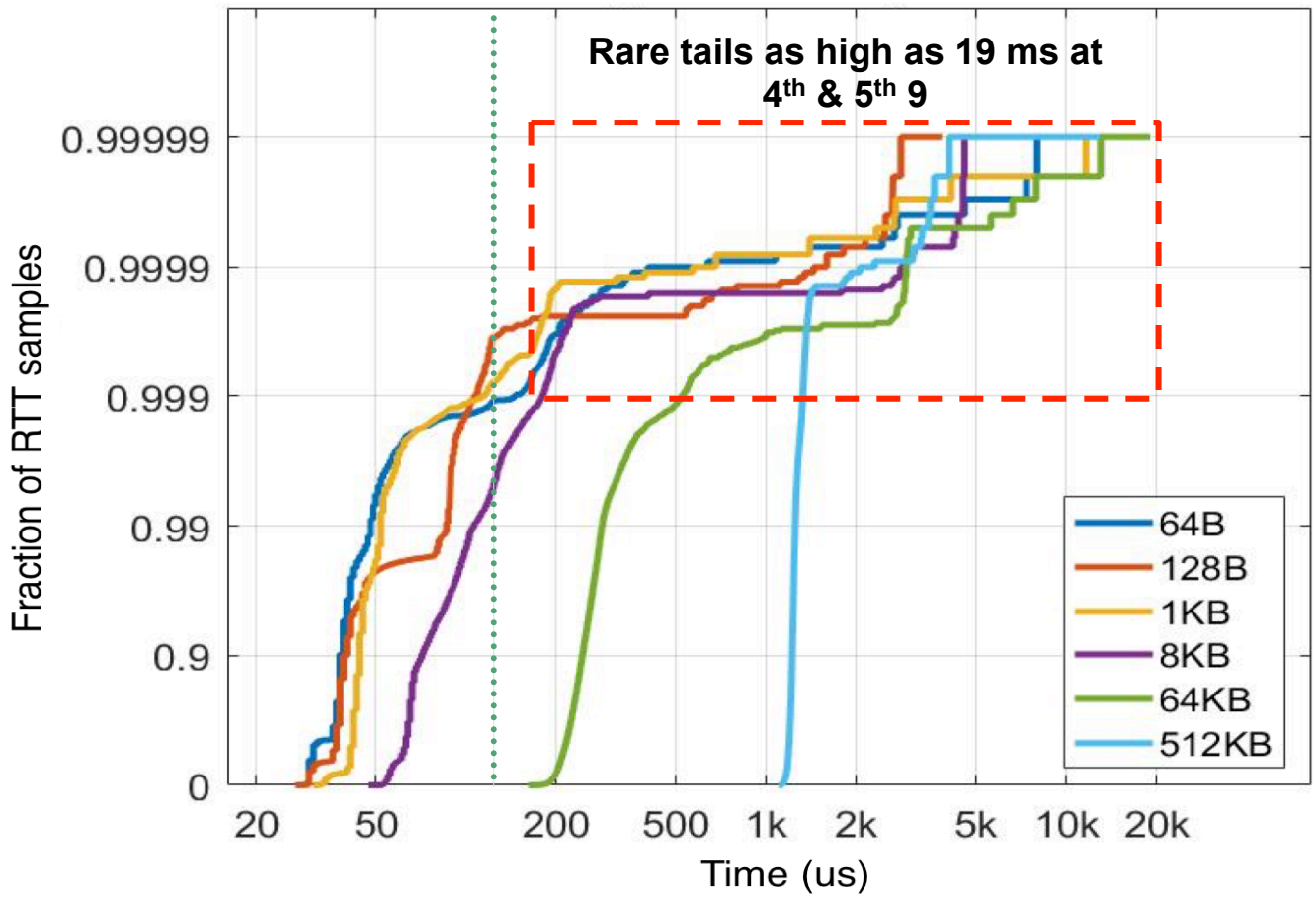




RTT for 100K exchanges of size 1KB each

Baseline TCP-IP exchanges exhibit extreme tails

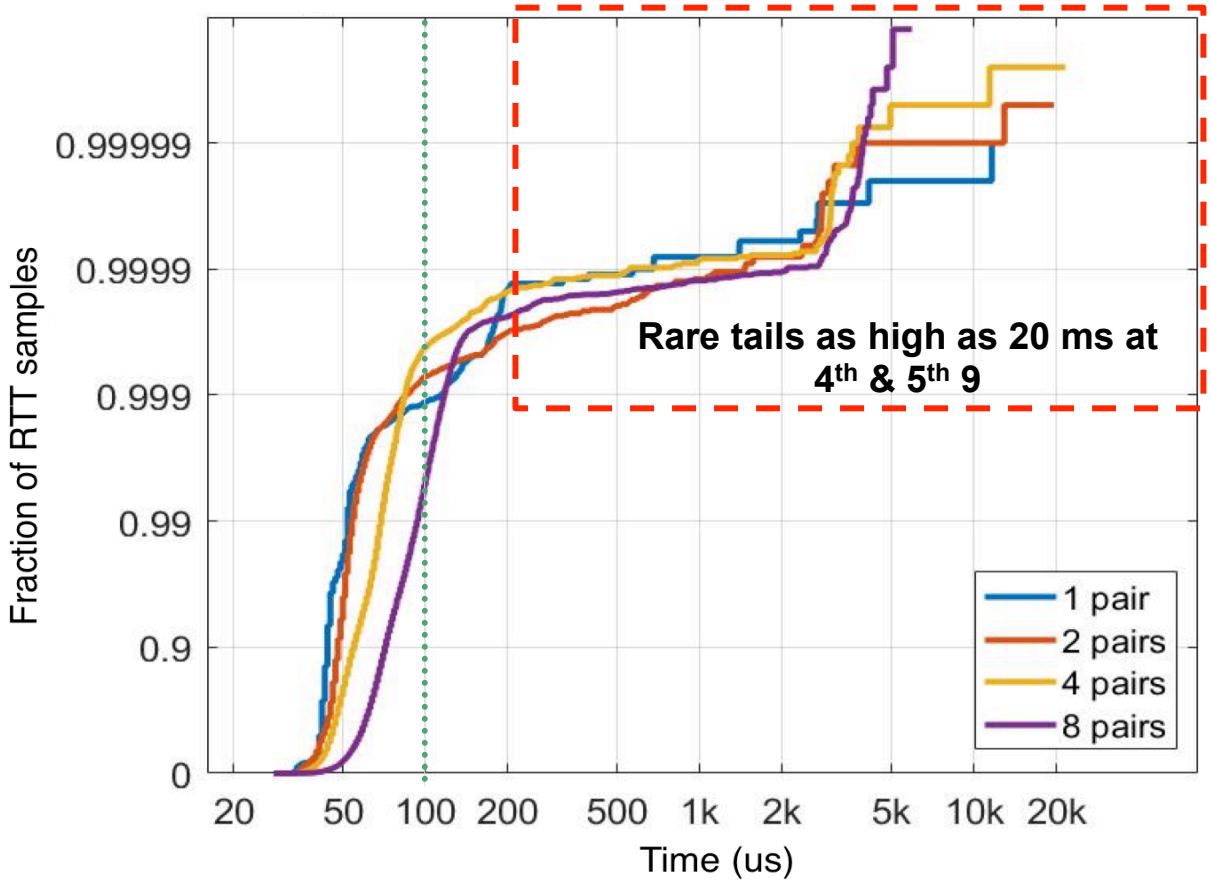
Increasing the payload size



RTT for 100K exchanges of sizes 64B-512KB each: single TCP-IP server-client pair

NO significant increase in baseline latency tails

Physical link multiplexing



RTT for 100K exchanges: 1-8 simultaneous TCP-IP server-client pairs

NO significant increase in baseline latency tails

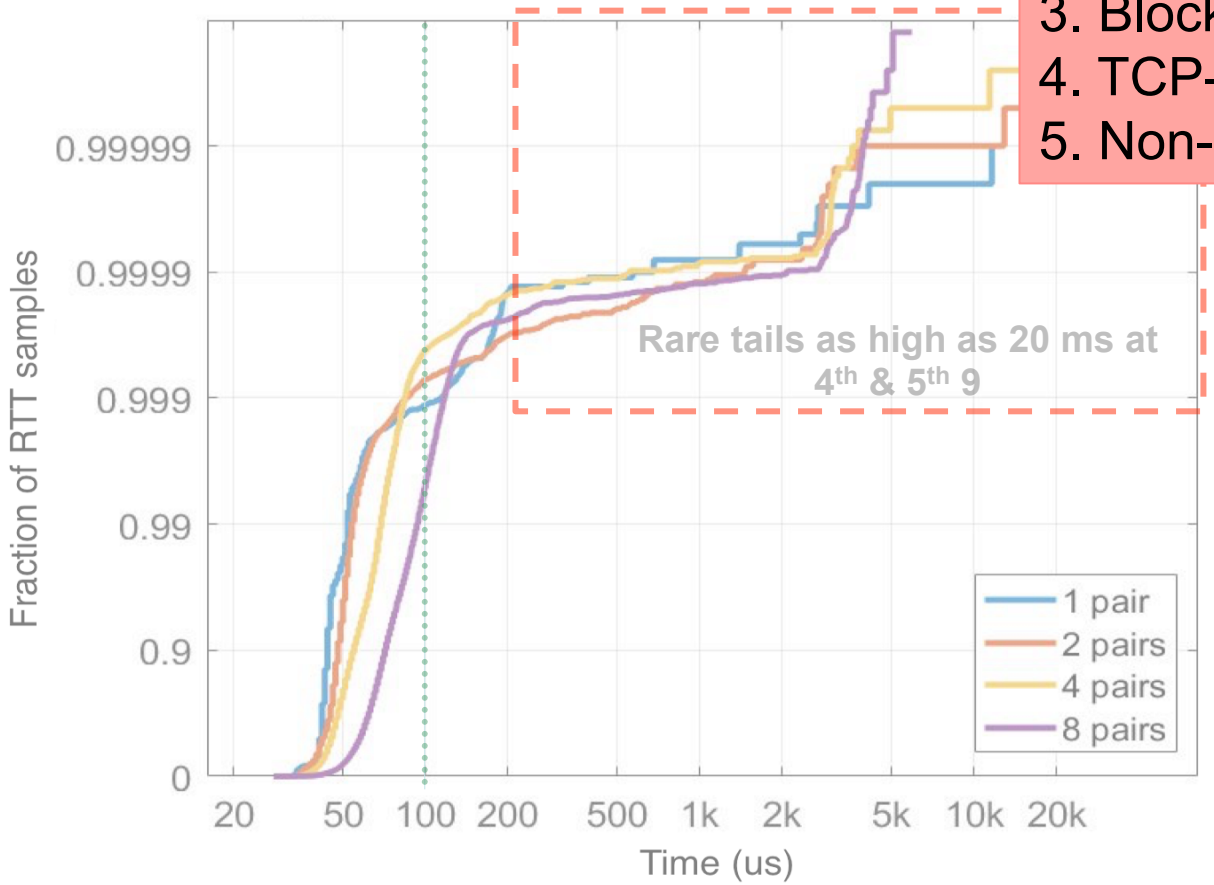
Physical link multiplex

What is the culprit?

~~1. Extrinsic parameters~~

~~2. Bandwidth saturation~~

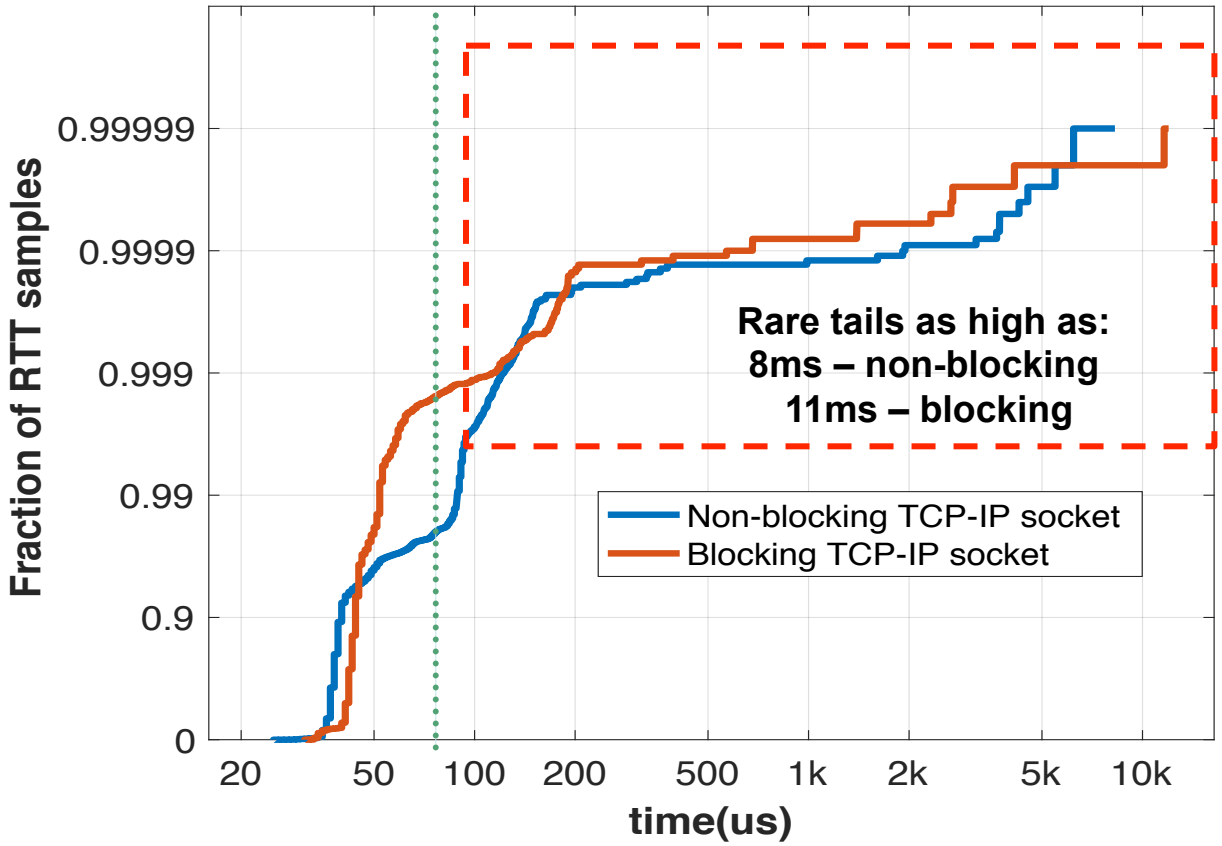
3. Blocking calls
4. TCP-IP complexities
5. Non-OS bottlenecks



RTT for 100K exchanges: 1-8 simultaneous TCP-IP server-client pairs

NO significant increase in baseline latency tails

Non-blocking system calls



RTT for 100K 1KB exchanges: single TCP-IP server-client pair

NO difference in latency tails exhibited by synchronous & asynchronous TCP-IP sockets

Non-blocking system c

What is the culprit?

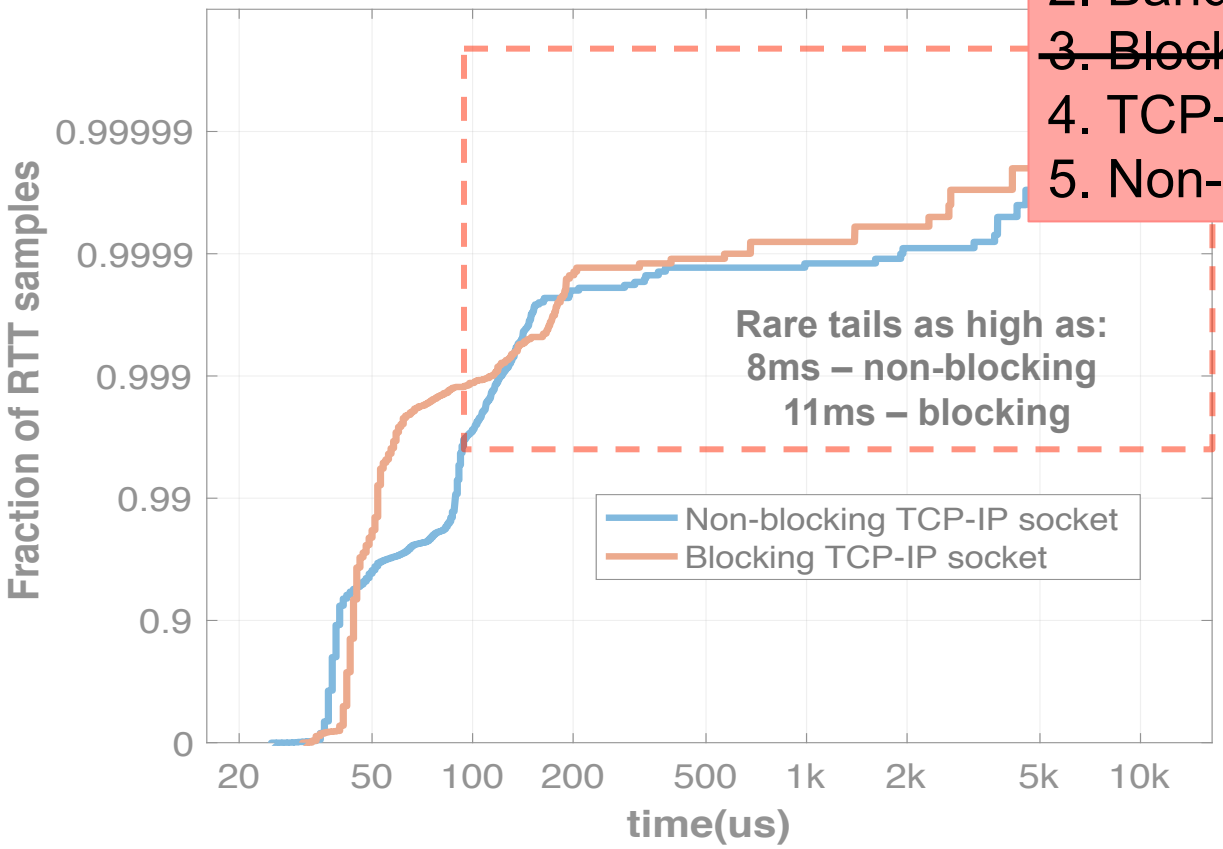
~~1. Extrinsic parameters~~

~~2. Bandwidth saturation~~

~~3. Blocking calls~~

4. TCP-IP complexities

5. Non-OS bottlenecks



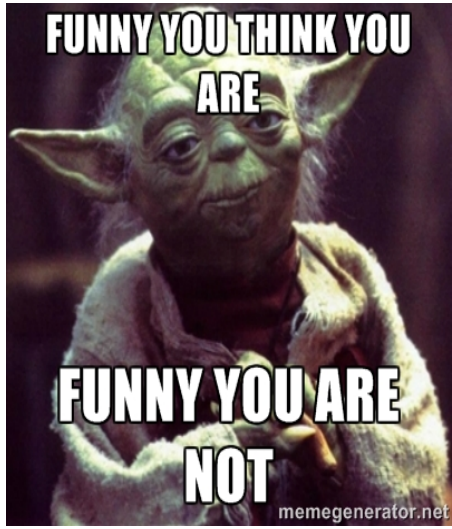
RTT for 100K 1KB exchanges: single TCP-IP server-client pair

NO difference in latency tails exhibited by synchronous & asynchronous TCP-IP sockets

TCP-IP complexities

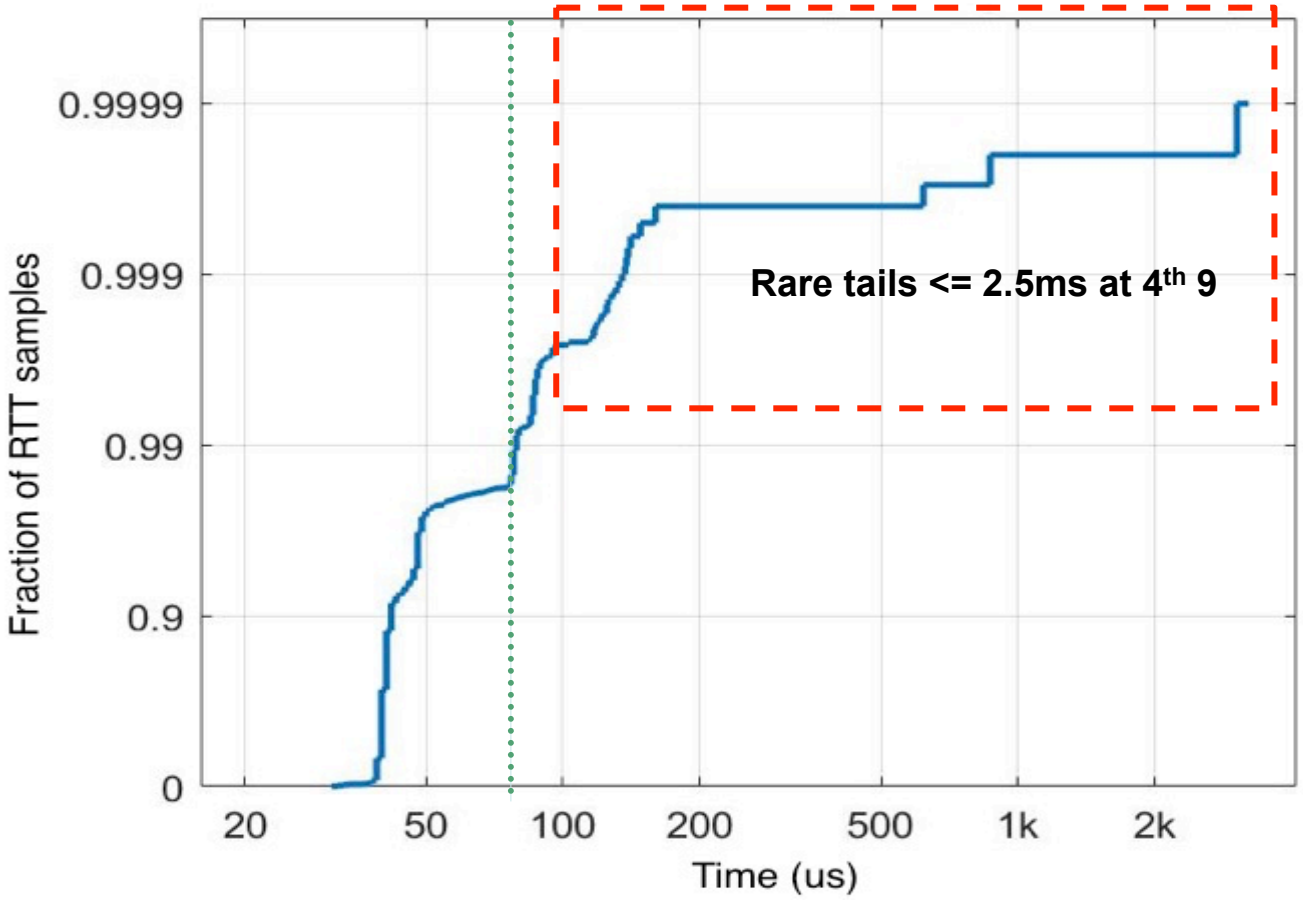
- Reliable delivery, flow control, error checking

"Hi, I'd like to hear a TCP joke."
"Hello, would you like to hear a TCP joke?"
"Yes, I'd like to hear a TCP joke."
"OK, I'll tell you a TCP joke."
"Ok, I will hear a TCP joke."
"Are you ready to hear a TCP joke?"
"Yes, I am ready to hear a TCP joke."
"Ok, I am about to send the TCP joke. It will last 10 seconds, it has two characters, it does not have a setting, it ends with a punchline."
"Ok, I am ready to get your TCP joke that will last 10 seconds, has two characters, does not have an explicit setting, and ends with a punchline."
"I'm sorry, your connection has timed out.
...Hello, would you like to hear a TCP joke?"



Naïve UDP-IP will not exhibit extreme tails?

UDP-IP: single server-client pair



RTT for 100K 1KB exchanges: single UDP-IP server-client pair

TCP-IP complexities do NOT induce tails

UDP-IP: single server-clie

What is the culprit?

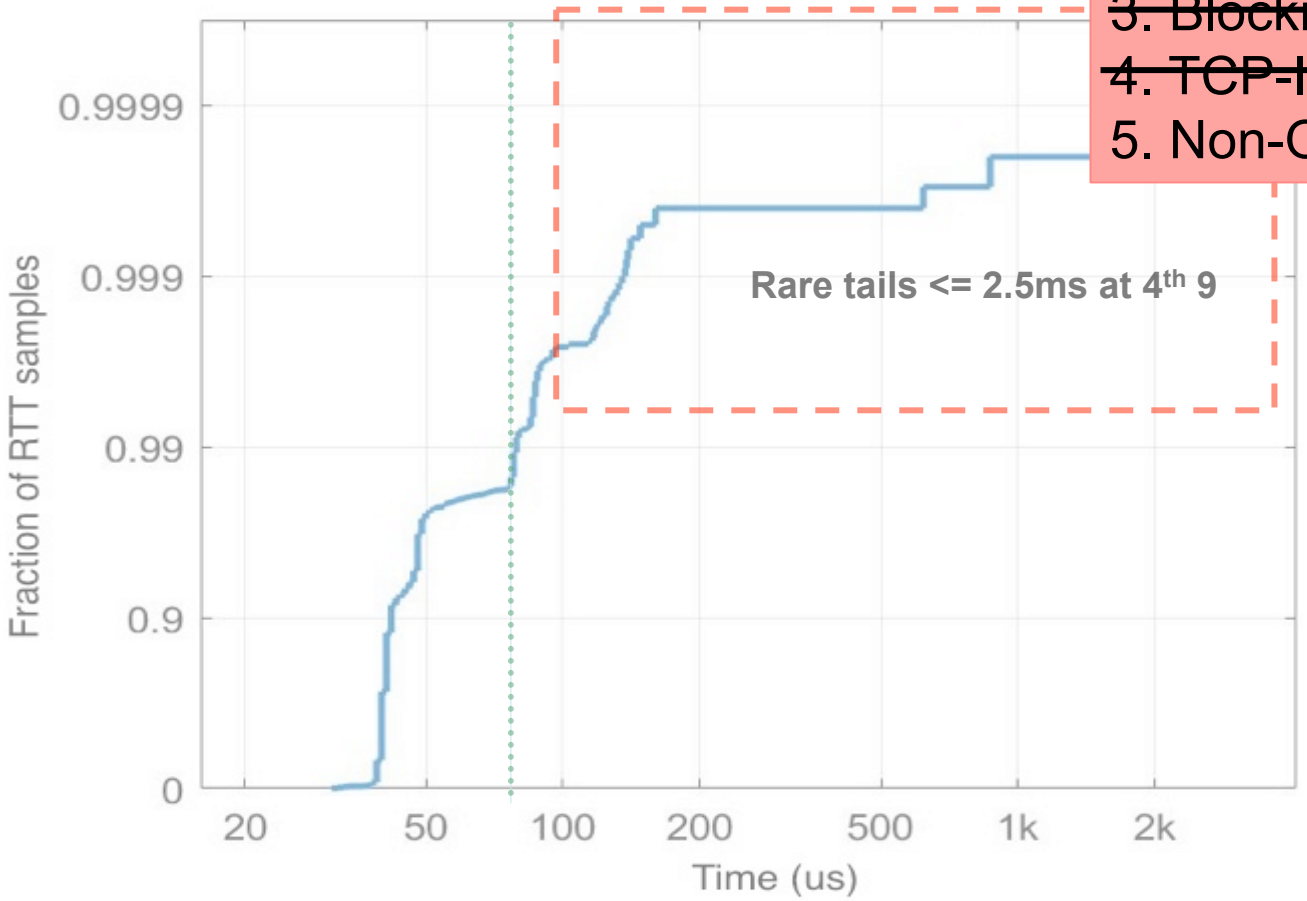
~~1. Extrinsic parameters~~

~~2. Bandwidth saturation~~

~~3. Blocking calls~~

~~4. TCP-IP complexities~~

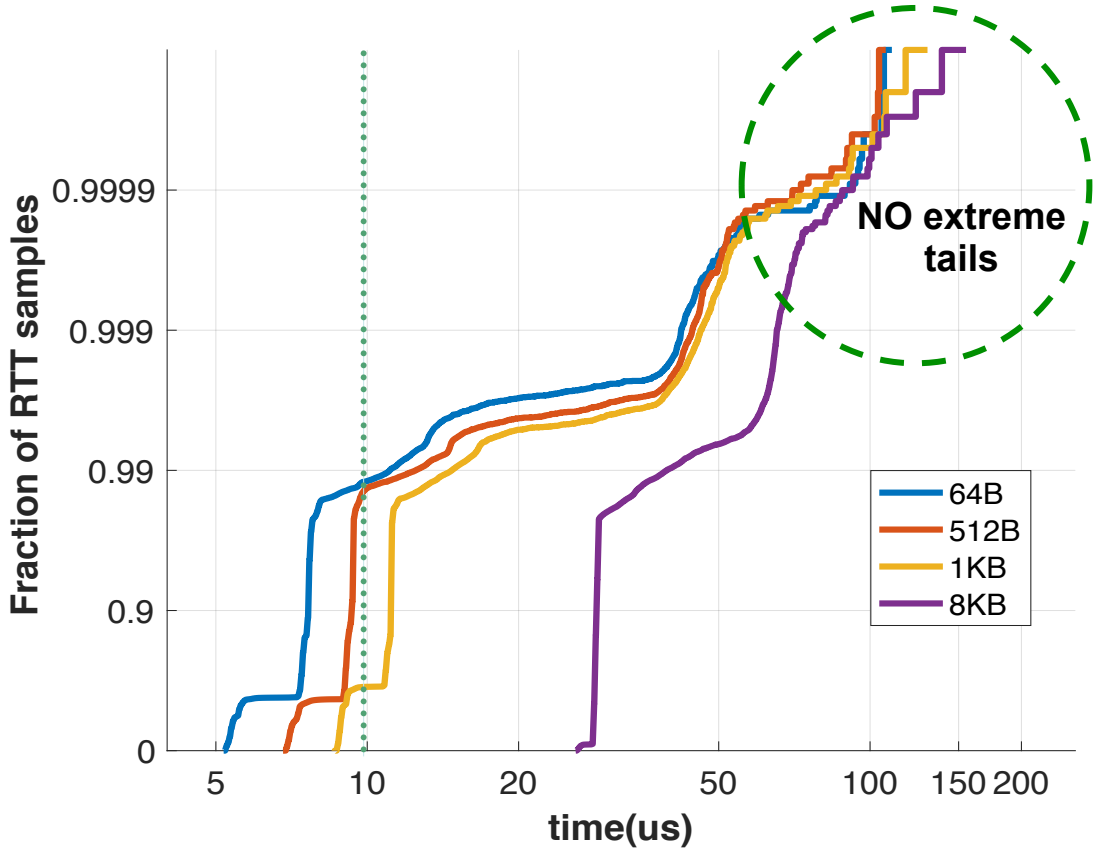
5. Non-OS bottlenecks



RTT for 100K 1KB exchanges: single UDP-IP server-client pair

TCP-IP complexities do NOT induce tails

RDMA single server-client pair



RTT for 100K RDMA reads of sizes 64B-8KB: single RDMA server-client pair

Non-OS factors do NOT induce extreme tails

RDMA single server-clien

What is the culprit?

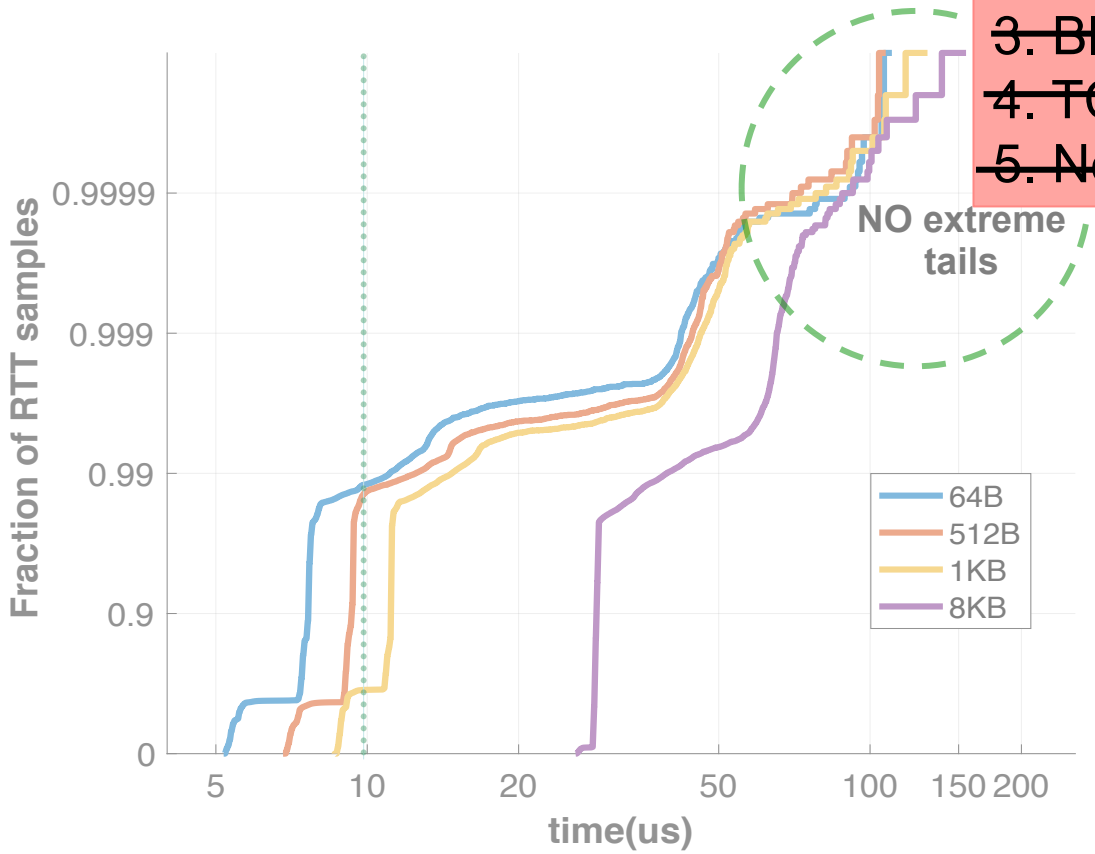
~~1. Extrinsic parameters~~

~~2. Bandwidth saturation~~

~~3. Blocking calls~~

~~4. TCP-IP complexities~~

~~5. Non-OS bottlenecks~~



RTT for 100K RDMA reads of sizes 64B-8KB: single RDMA server-client pair

Non-OS factors do NOT induce extreme tails

What is the culprit?

- Extrinsic network parameters?
- Bandwidth saturation?
- Blocking system calls?
- Additional TCP-IP complexities?
- Non-OS bottlenecks?

OS network stacks cause extreme tails!

Conclusion

- Source of extreme tails: OS protocol stack
  - Obsolete TCP/UDP protocol design
- Investigate individual components
  - TCP/UDP network protocol stack

Thank you!