Demystifying Storage Configurations for Optimal Performance

Steven Johnson

Performance Scientist

Sun Microsystems, Broomfield CO

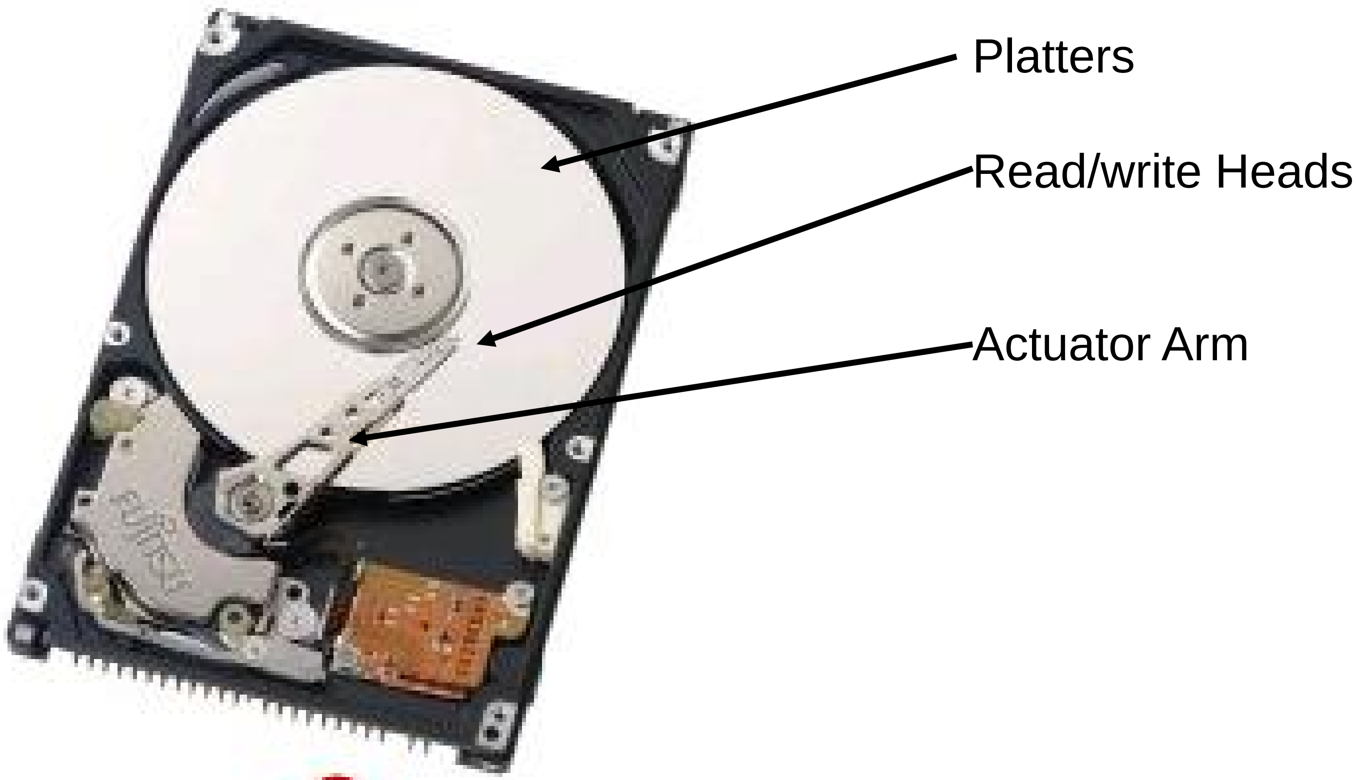
Agenda

- Overview of Disk Drive Performance
- Compare FC/SAS drives to SATA drives
- Optimal Array configurations for throughput and response times
  - Understanding the Impact of Stripe Depth
    - RAID 1
    - RAID 5

Overview of Disk Drive Performance

I promise to be quick!

Internals to a disk drive



Internals to a disk drive

SDC
STORAGE DEVELOPER CONFERENCE
SNIA SANTA CLARA, 2008

Latency – Time for the data to fly under the head

Outer Diameter (OD)

Inner Diameter (ID)

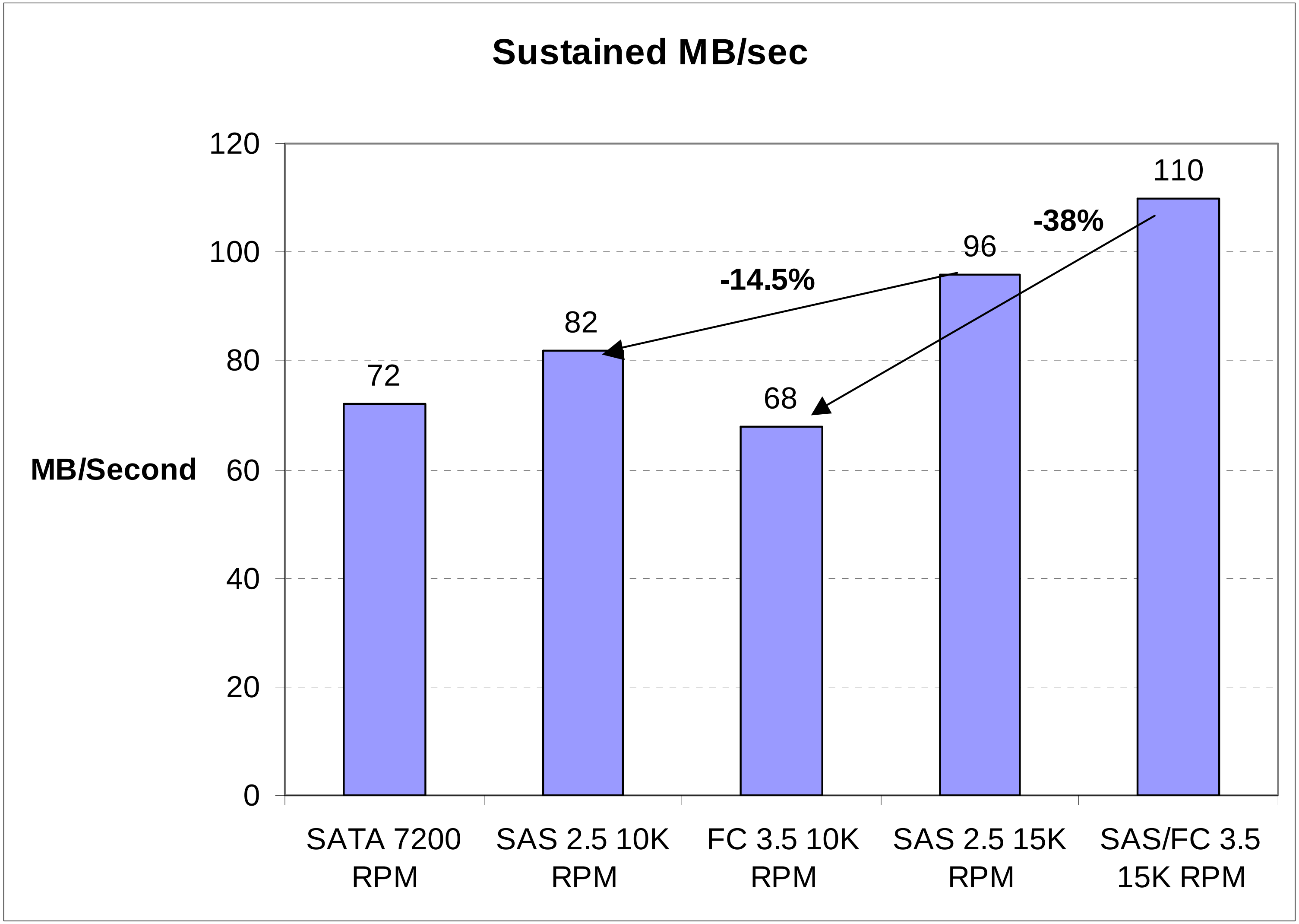
Seek - Move the Actuator arm

FUJITSU

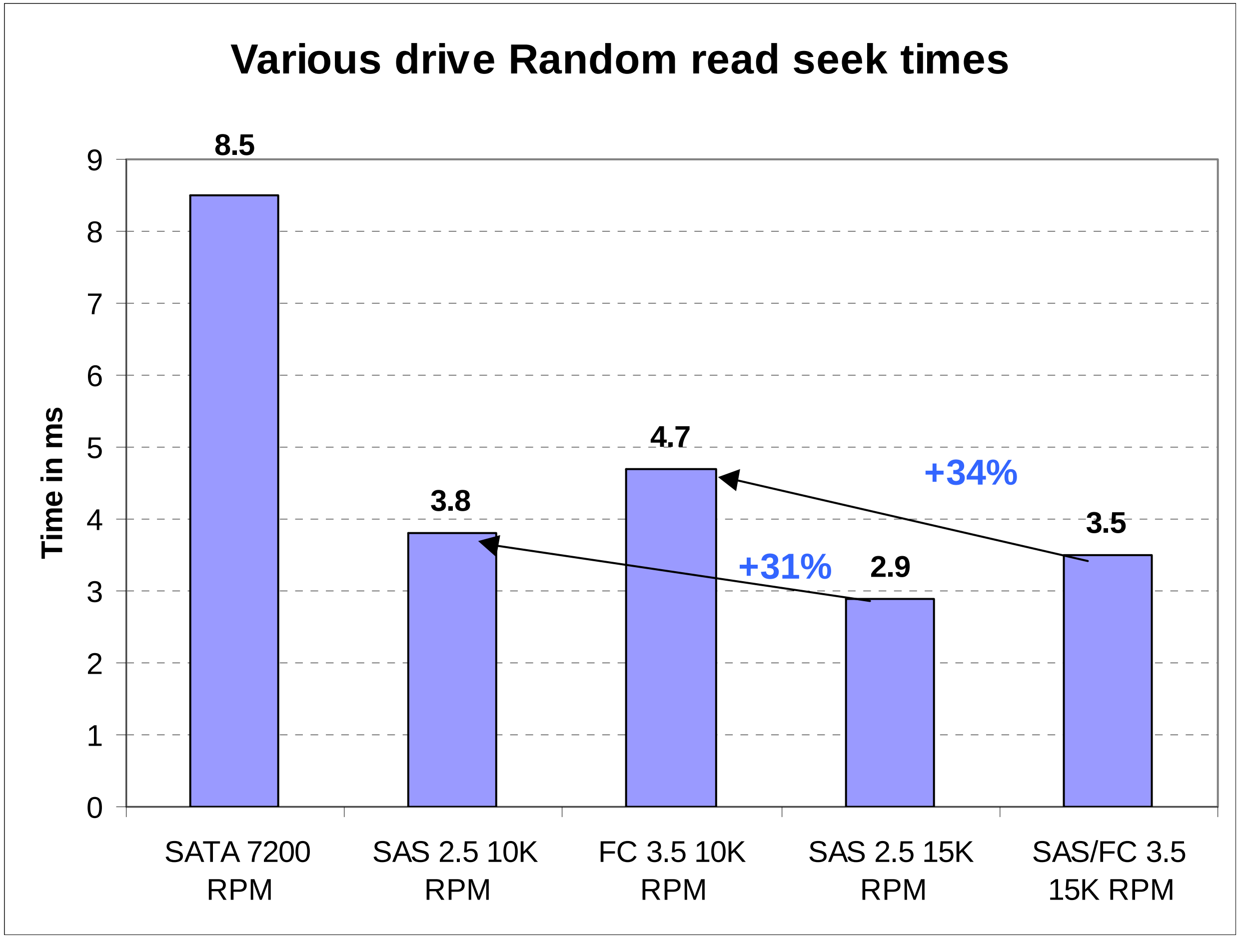
The next generation of drives

- SATA – approaching 1 TB capacities
  - Very inexpensive, very high capacity, slow
- 3.5 inch drives – Now 500 GBs
  - 10 and 15K versions, good capacity and transfer characteristics. Excellent performance
- 2.5 inch drives – currently largest is 146 GB
  - Lower capacity, very fast access, slightly less transfer rates compared to 3.5 inch.

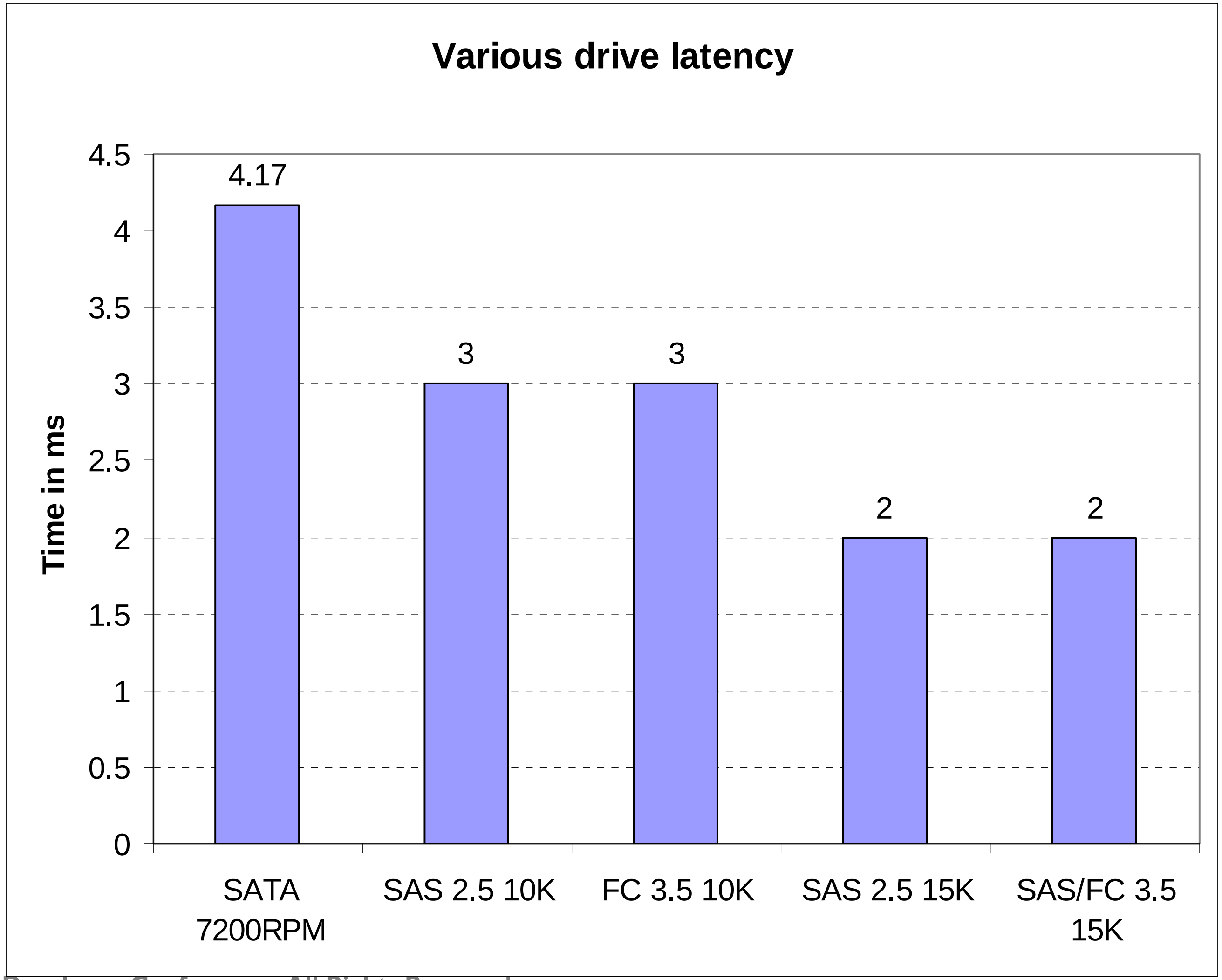
Sustained Data rates



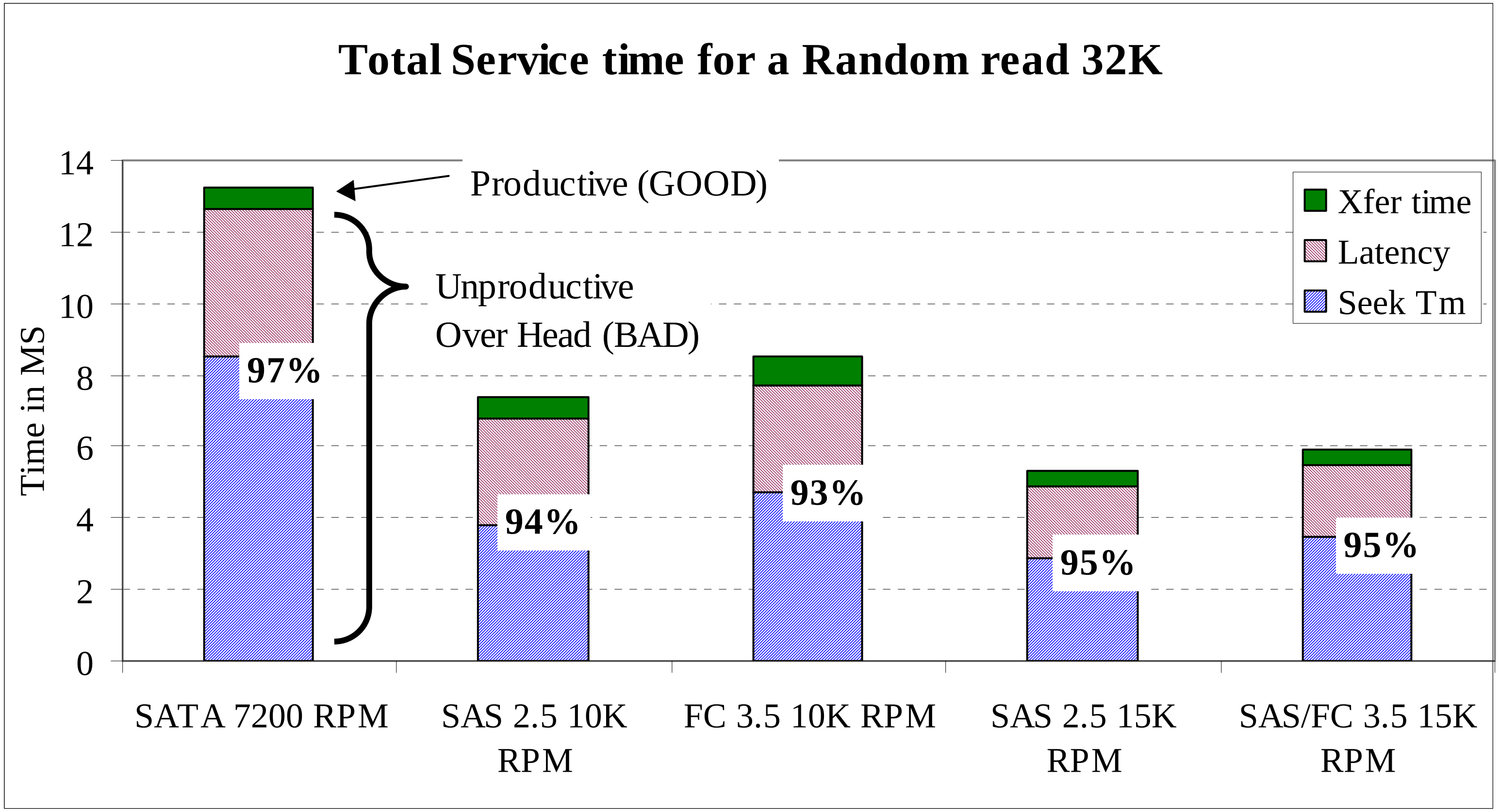
Seek times


Various drive Random read seek times
Time in ms
0
1
2
3
4
5
6
7
8
9
8.5
3.8
4.7
2.9
3.5
+34%
+31%
SATA 7200 RPM
SAS 2.5 10K RPM
FC 3.5 10K RPM
SAS 2.5 15K RPM
SAS/FC 3.5 15K RPM

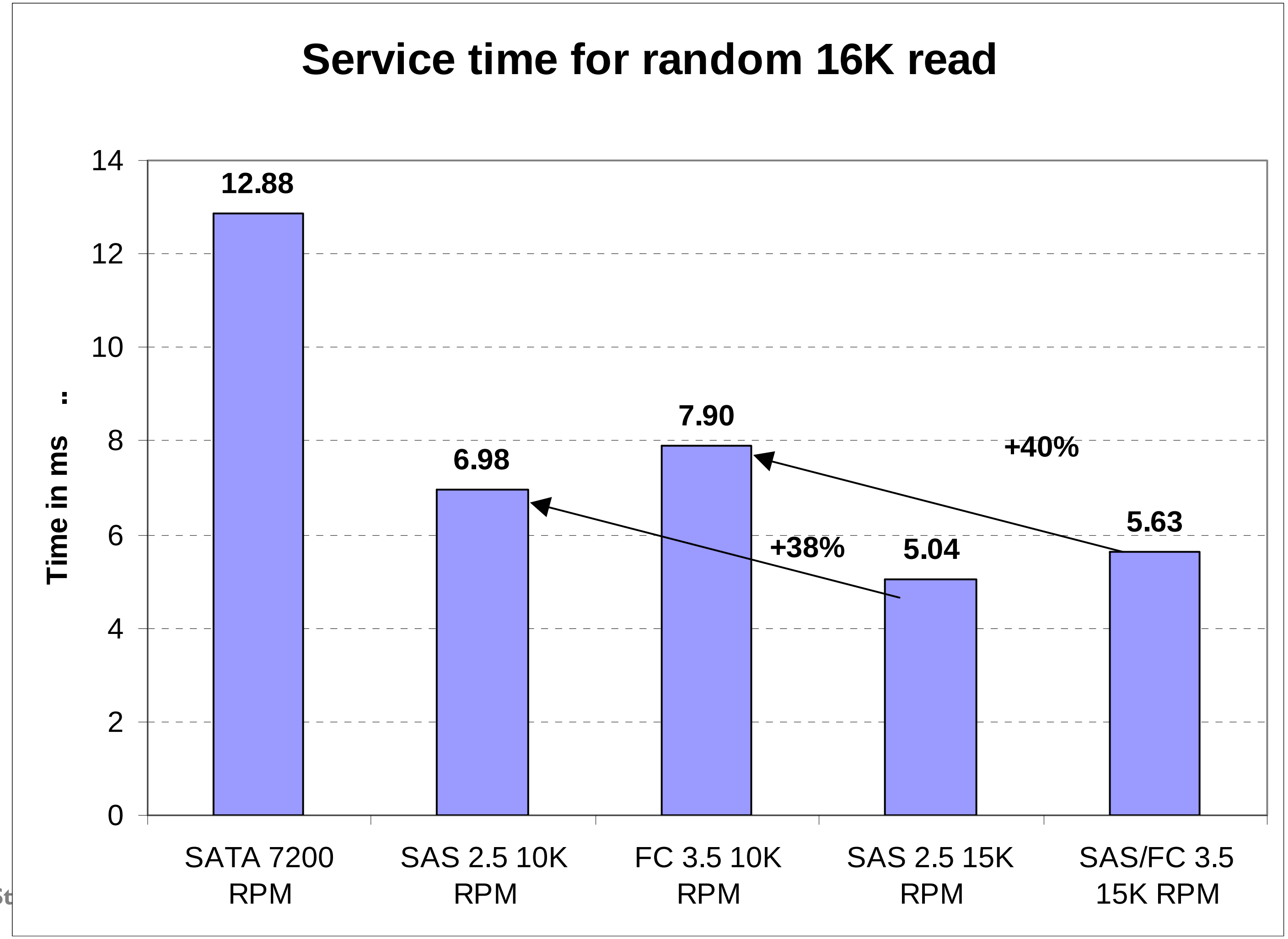
Drive Latencies



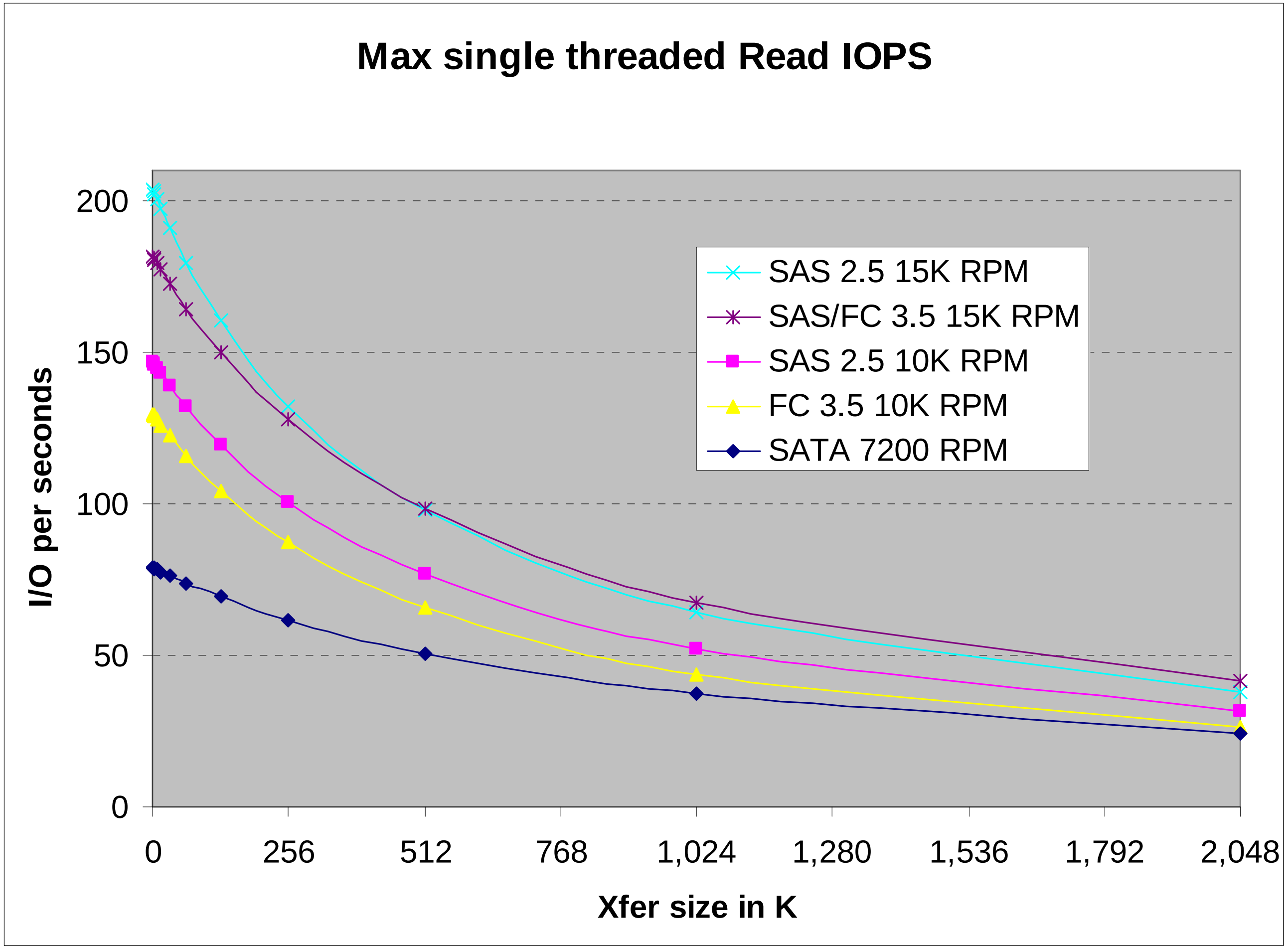
Break down of a 32K read



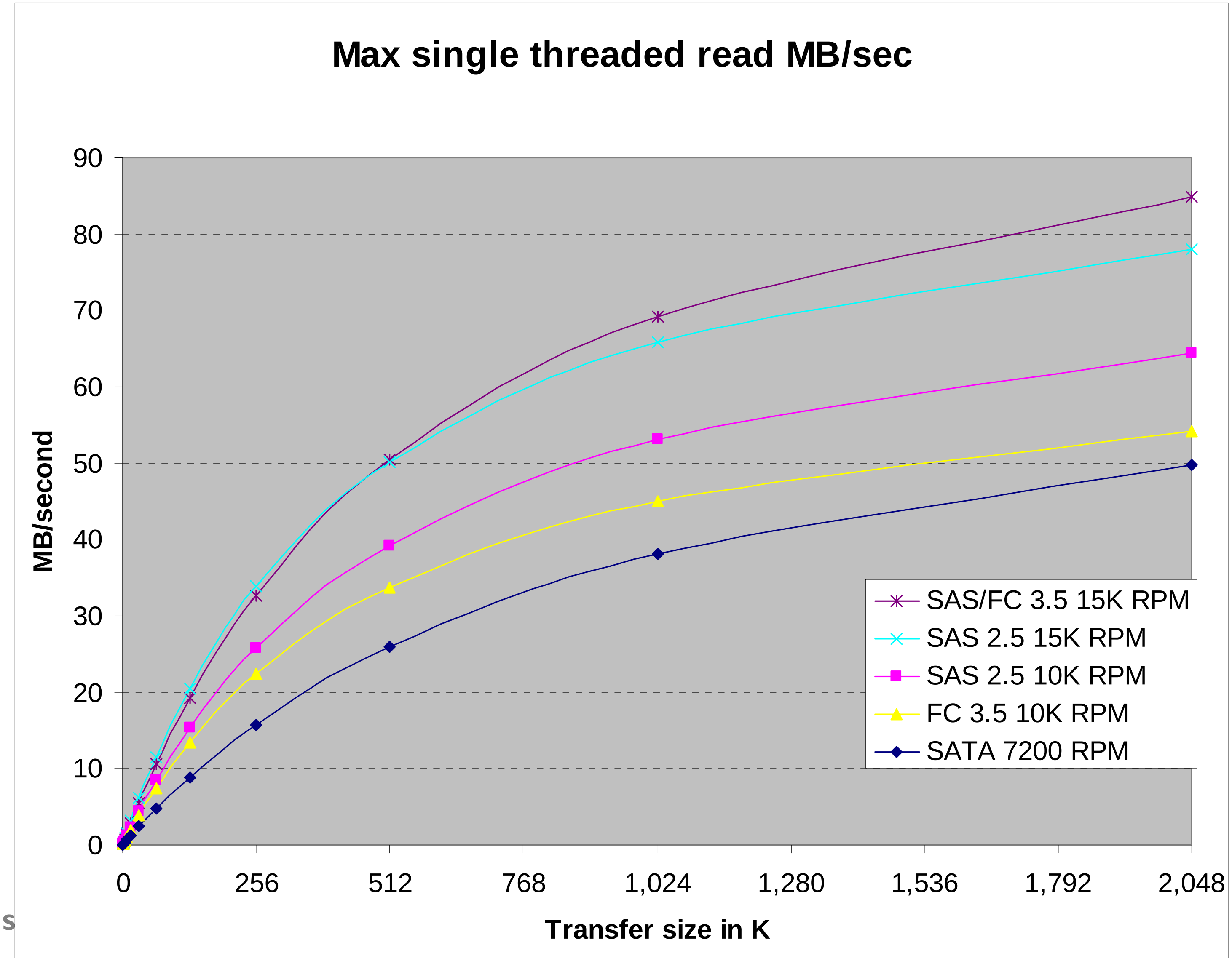
Drive Service times for 16K Random Read



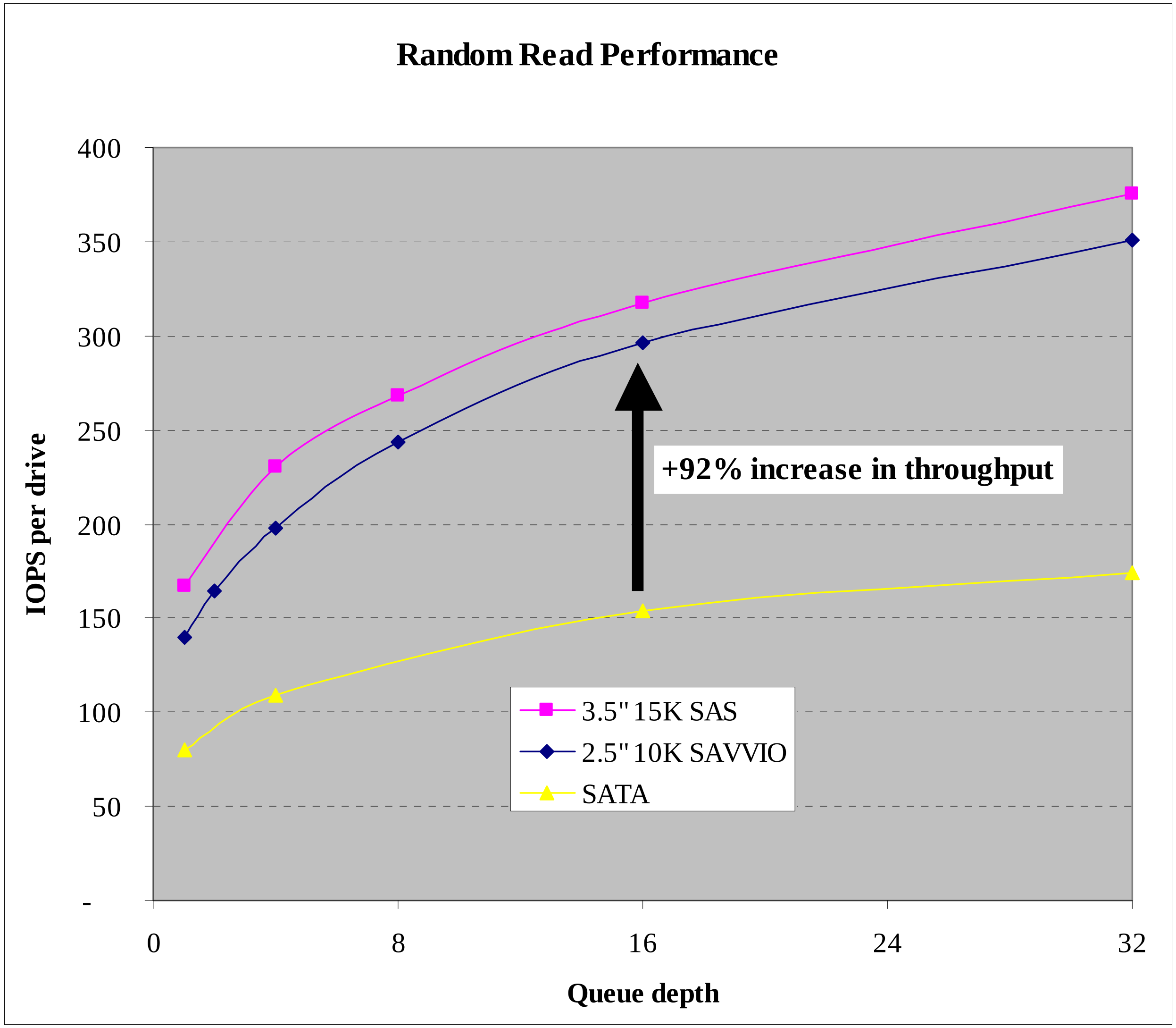
Theoretical IOs/Second vs Transfer size



Theoretical Drive Transfer speed vs Transfer size



Increased Queue depth per drive improves throughput



Optimal Array Configurations

Real Lab Data (but not a guarantee of performance)

Which is more efficient?

1 person with a Shovel?

10 people with a Spoon?

To move a pile of dirt

Array Striping

- Many performance problems start with Array Striping
- Folk lore states "thin wide stripes" provides optimal performance
- Older storage arrays had 16K default stripe size
- Such small stripe sizes kill our performance
- WHY?

Write Coalescing

- Most real workloads have “locality”
- Tend to read or write is the same general area
- Write back cache delay writes as much as possible
- Collect all the write data over time, then write it to disk in one IO
- Minimize the write penalty by effectively amortizing seek and latencies over more IOs

RAID 1 (Mirroring)

- Personally I believe there is only one choice: Go BIG! Assuming controller does partial stripe staging.
- Controller write cache will perform write coalescing
- As the data ages out, it will write to disk fairly efficiently

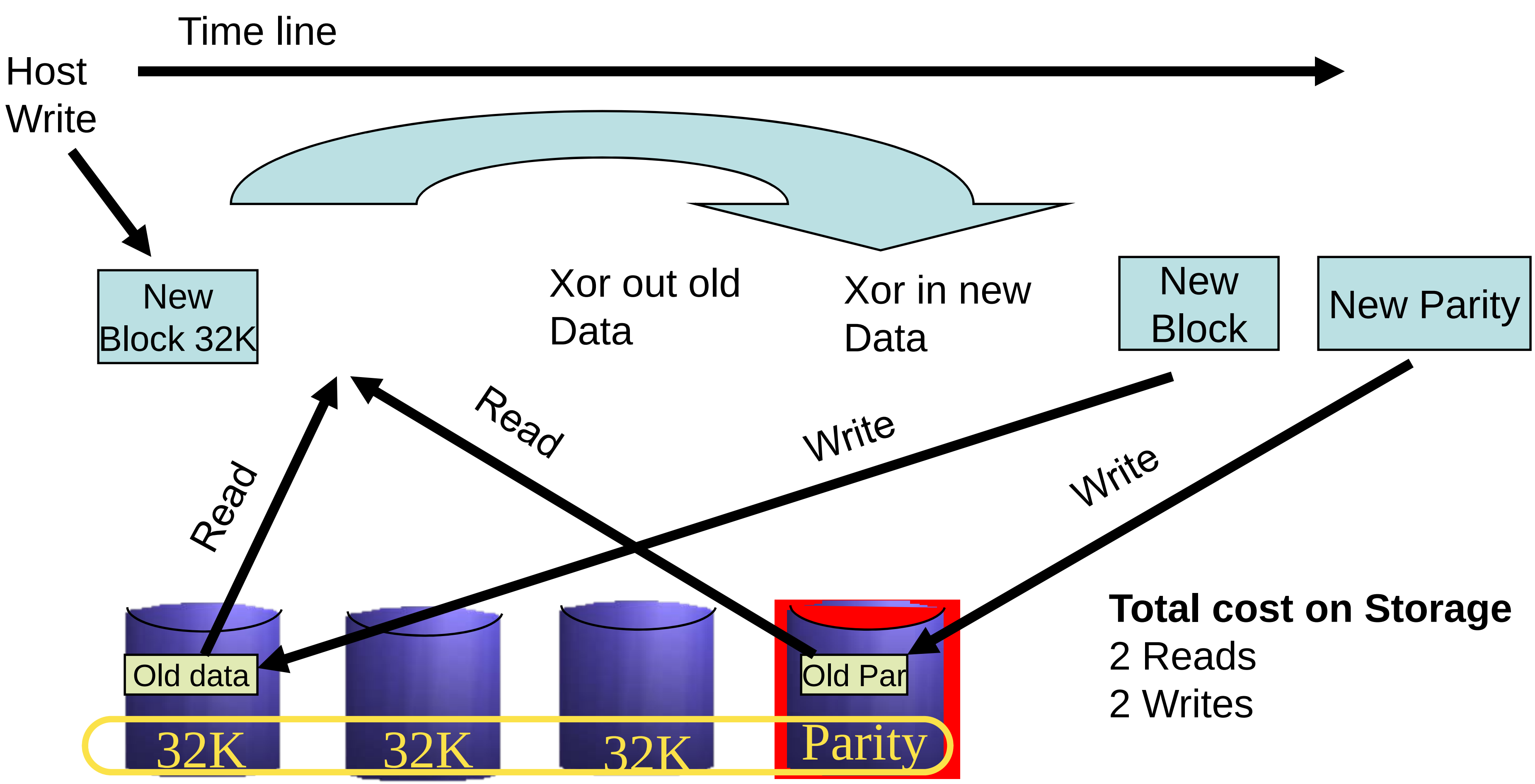
Solutions to the Raid 5 Penalty

- What is the Raid 5 Penalty?
- What is Parity on the fly
- What is Partial Stripe Update
- How parallelism hurts read/write performance
- Real lab data
- Why do we care? ~85% of customer configurations are RAID 5

The Raid 5 Write Penalty

- To do a random write, the array controller must do two reads and two writes for every block written
- Must read Old Data and Old Parity
- Then xOR out old data, xOR in new data
- Write New Data and New Parity
- Puts 4 times as much load on the system
- Write cache is designed to mask away this delay

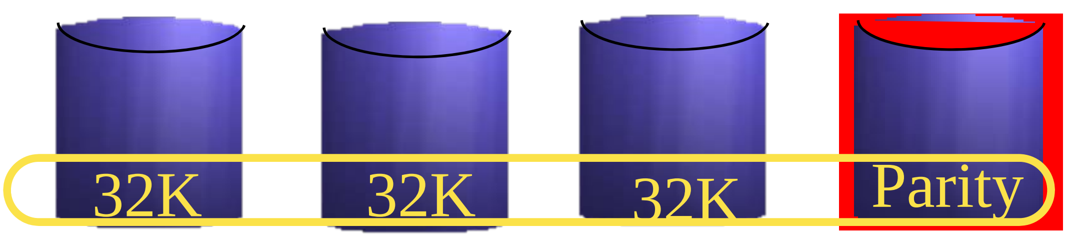
Raid 5 update write



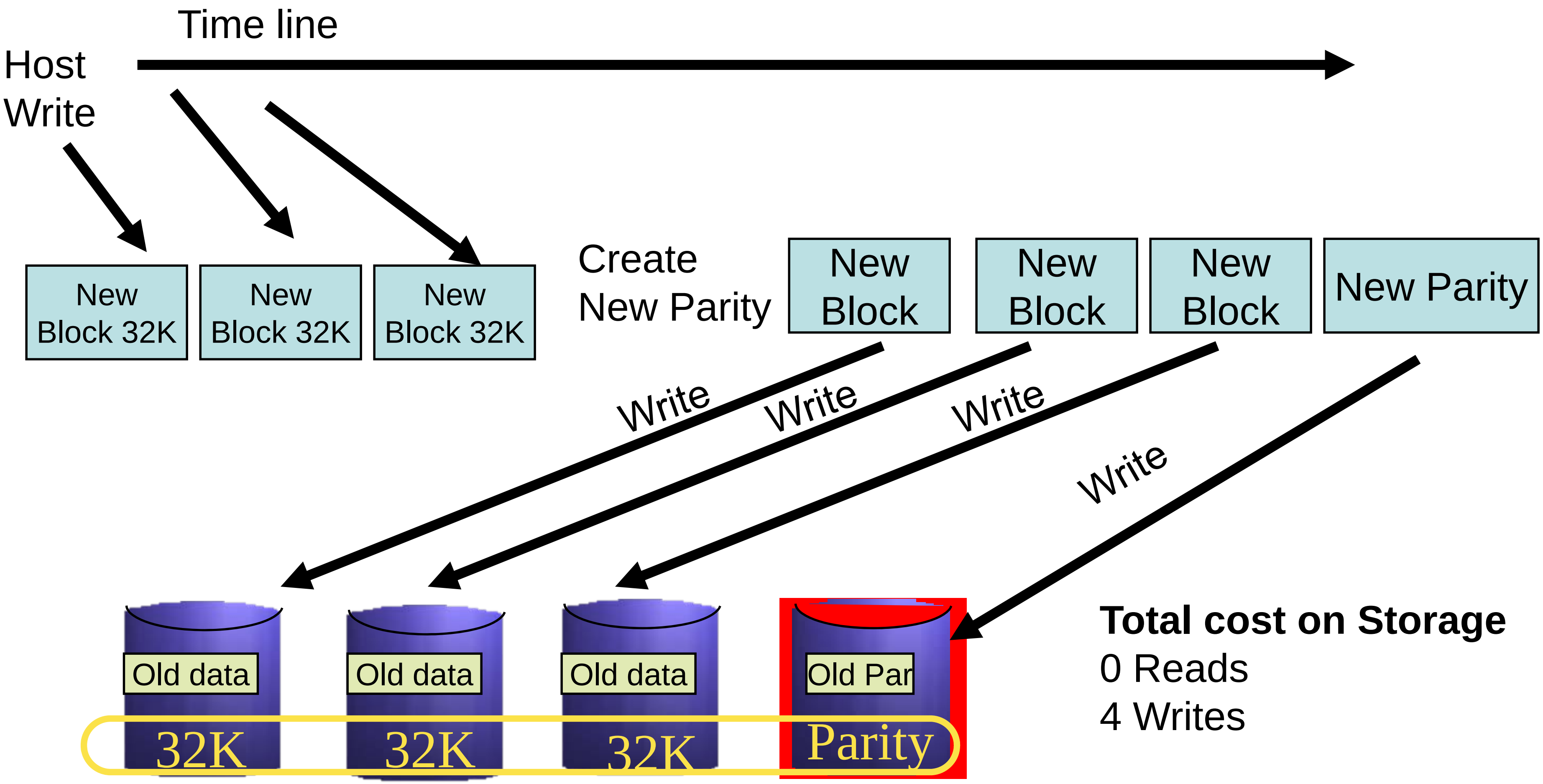
Total cost on Storage
2 Reads
2 Writes

Raid 5 and "Parity on the Fly"

- Term used for when we have a full stripe in write cache
- Able to calculate Parity with ZERO READS!
- Effectively no penalty, just a little more overhead



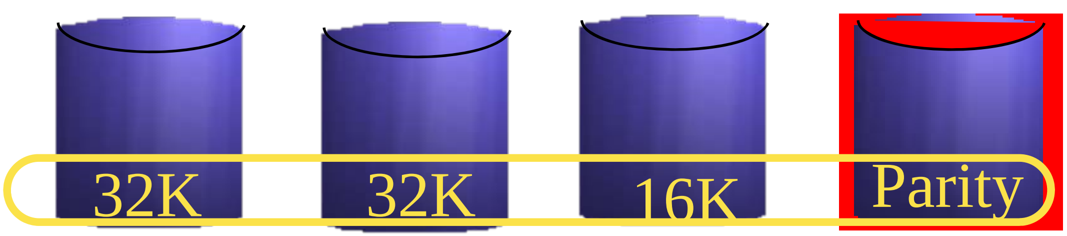
Raid 5 "Parity on the fly"



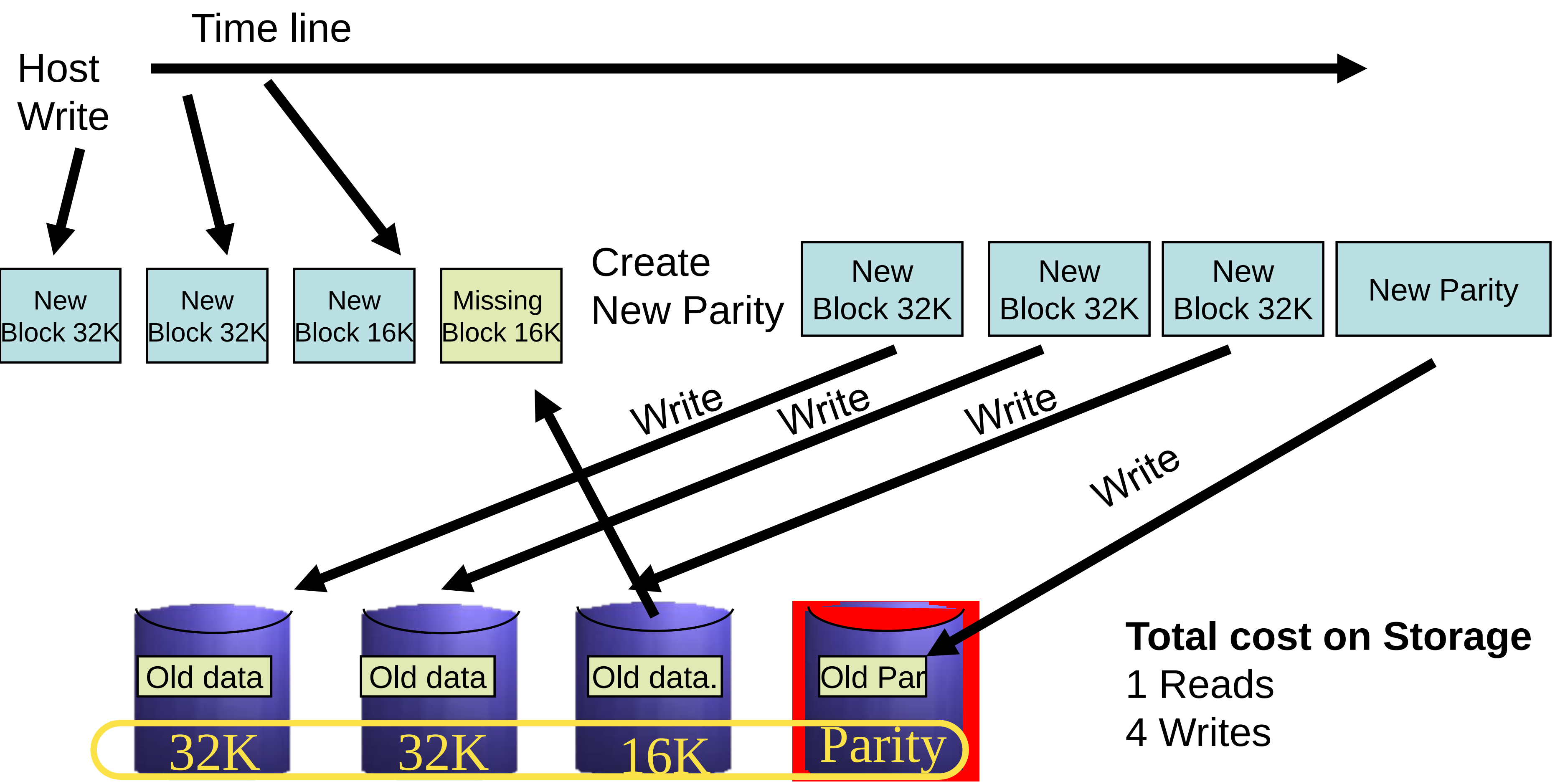
Total cost on Storage
0 Reads
4 Writes

Partial Stripe Update

- Current intelligent controllers now perform "partial stripe update"
- Calculates what is more efficient
  - Standard parity update (N reads and N writes)
  - Or fill out the stripe with missing data and do a "parity on the fly"
- All is designed to make Raid 5 more efficient



Raid 5 Partial Stripe

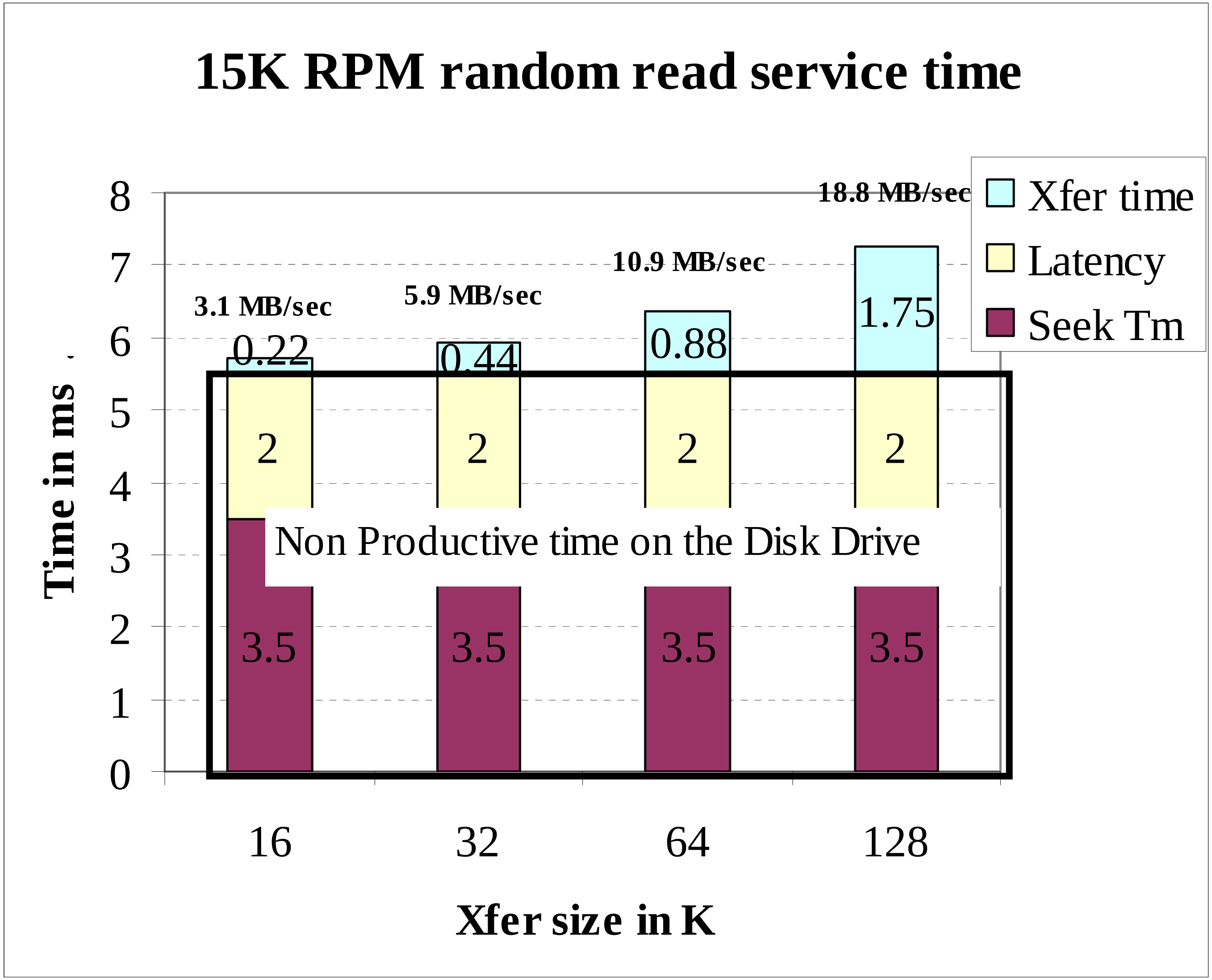


Total cost on Storage
1 Reads
4 Writes

Disk Drive Speed

- We have all heard of Moore's Law
  - Silicon density doubles every 12 months
- Similar behavior with Storage Capacity
  - Near doubling in capacity every 12-18 months
  - That increased density increases MB/sec
- Problem is mechanical access times are not keeping up
  - Access times only improve 7-8% each year

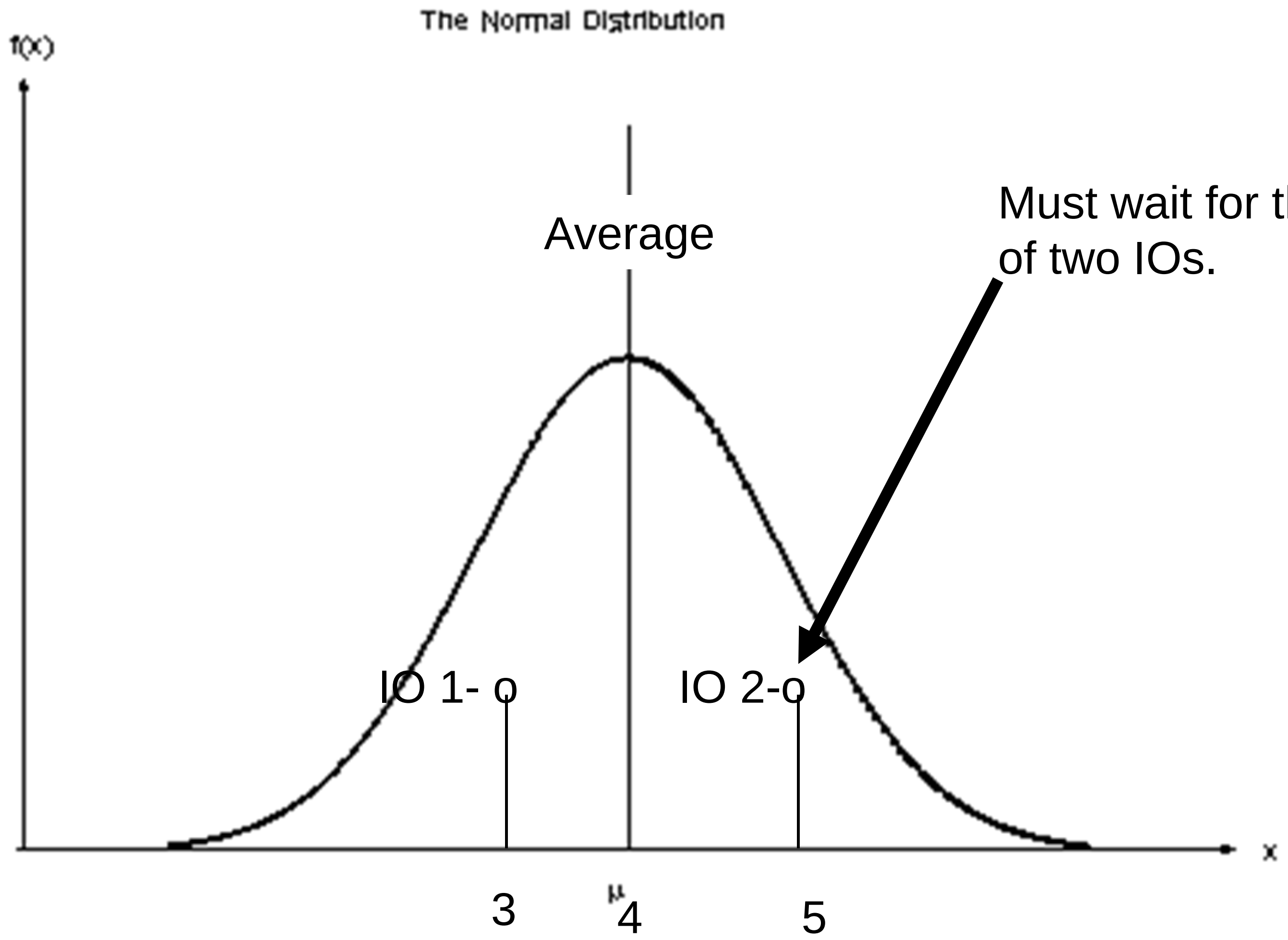
Drive efficiency Improves with Larger Transfer sizes



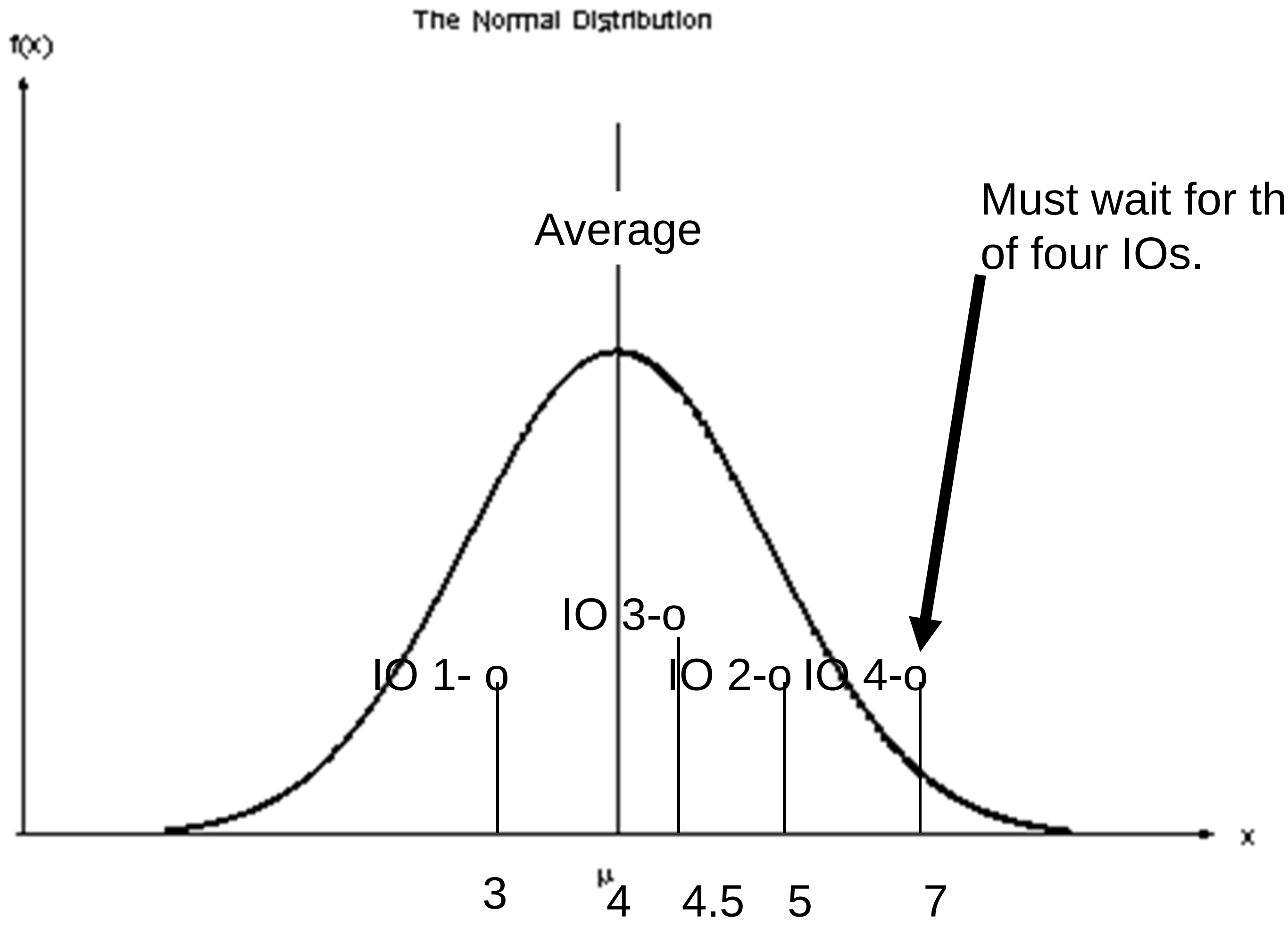
Efficiencies on drives

- By transferring more data per seek and latency will improve drive MB/second
- But why should we care?
- It's being done in Parallel – Its' going faster right?
- No – it is actually going ***<u>slower</u>***
- When we read from more than one drive, we are no longer working with averages!!

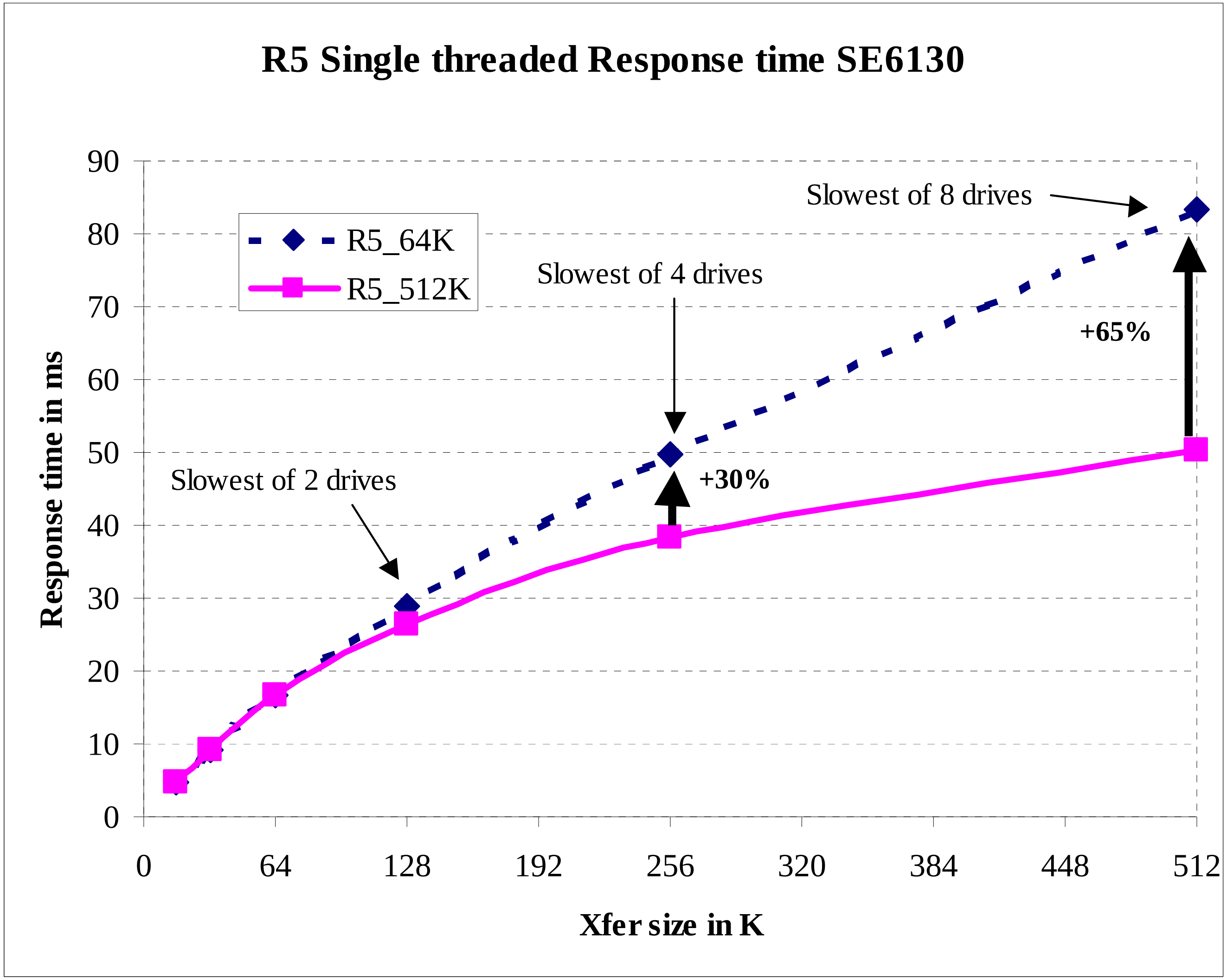
Each IO is not the same



Each IO is not the same



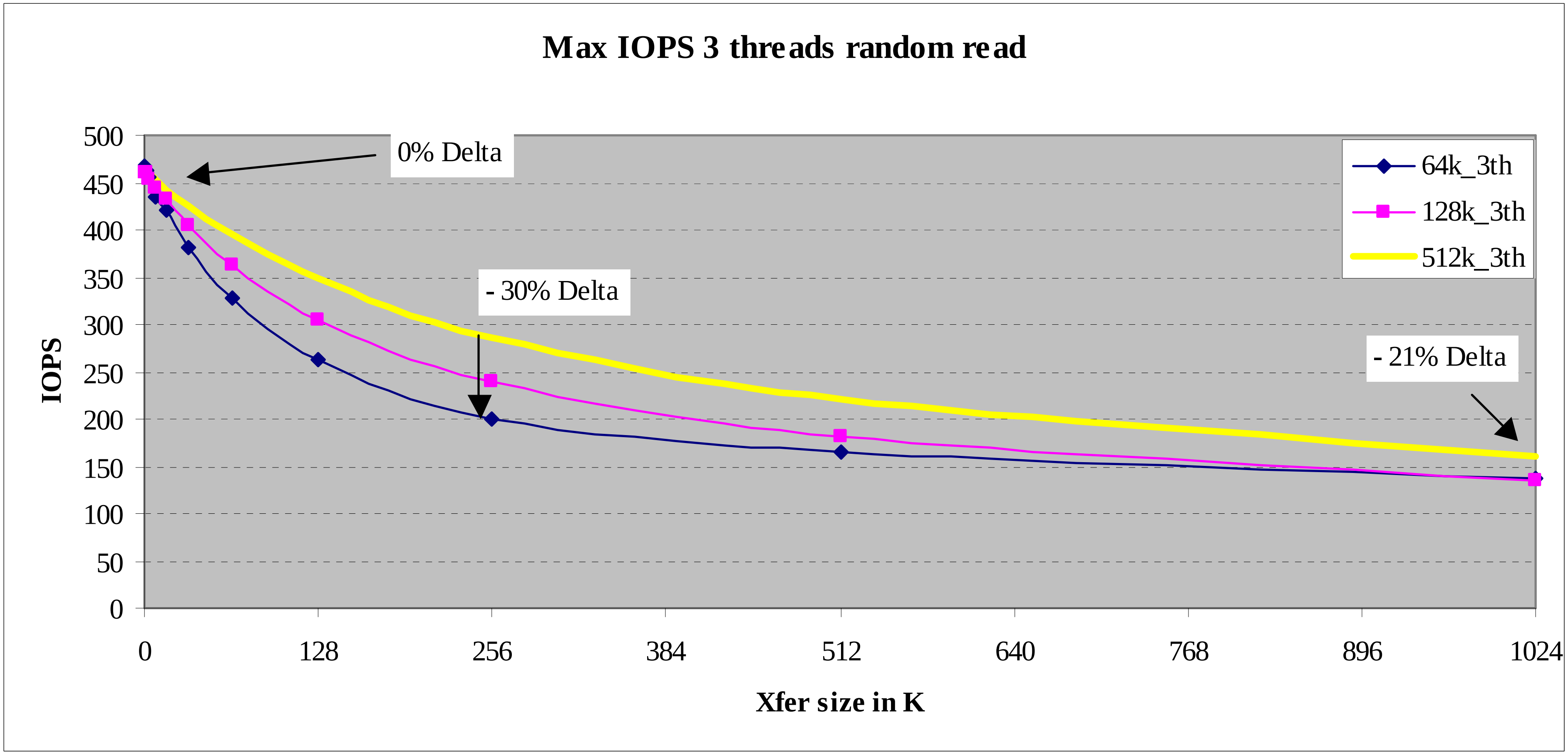
Best Case single threaded read



Stripe Depth Study

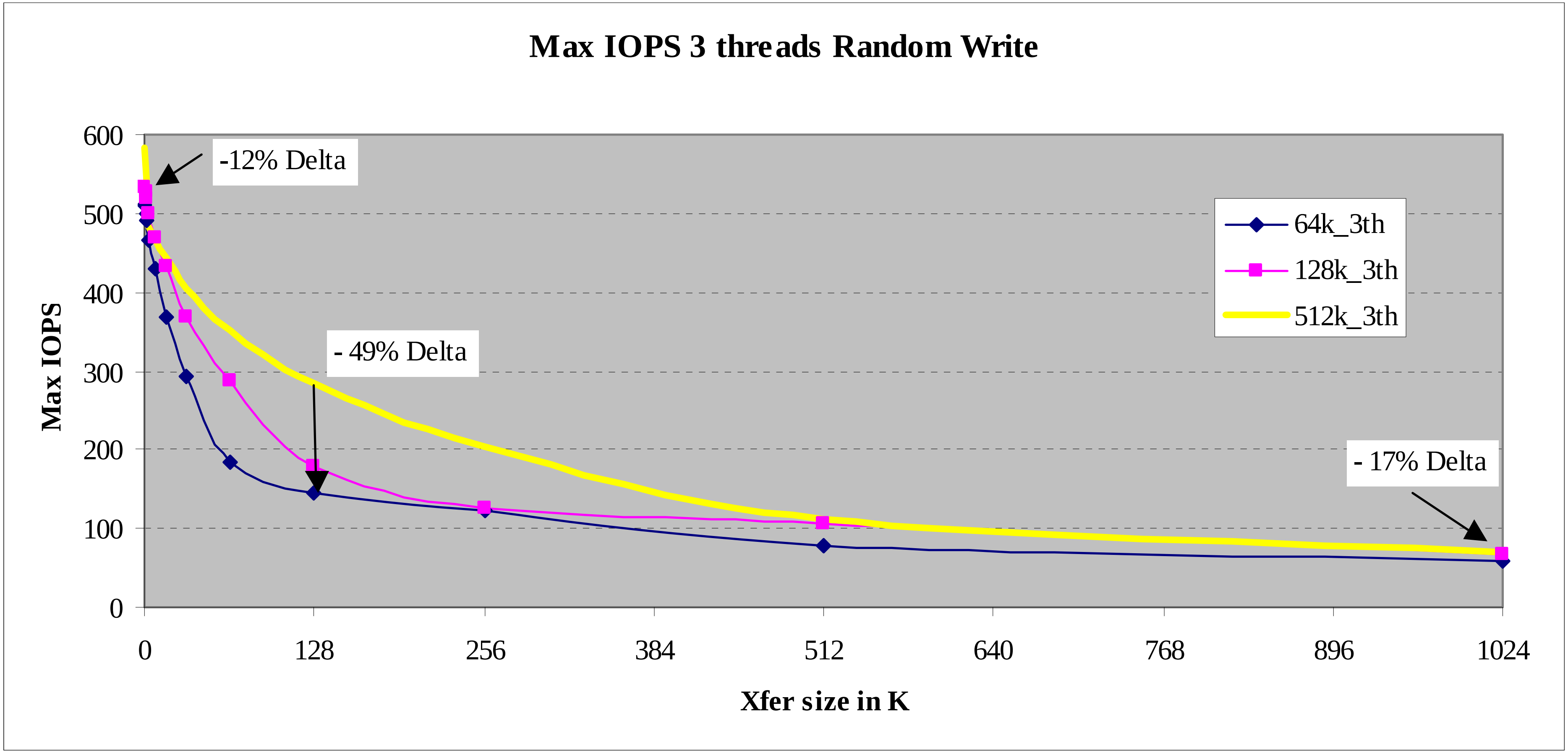
- Stripe depth on the **ST2530**
- Varied from 64K, 128K, and 512K on a **5+1 Raid 5** parity group
- Random reads and Random writes across a wide range of transfer sizes (512 bytes to 1 MB)
- Lets look at MAX IOPS from deep stripes

Only 3 threads against 6 drives Random Read MAX Throughput in IOPS

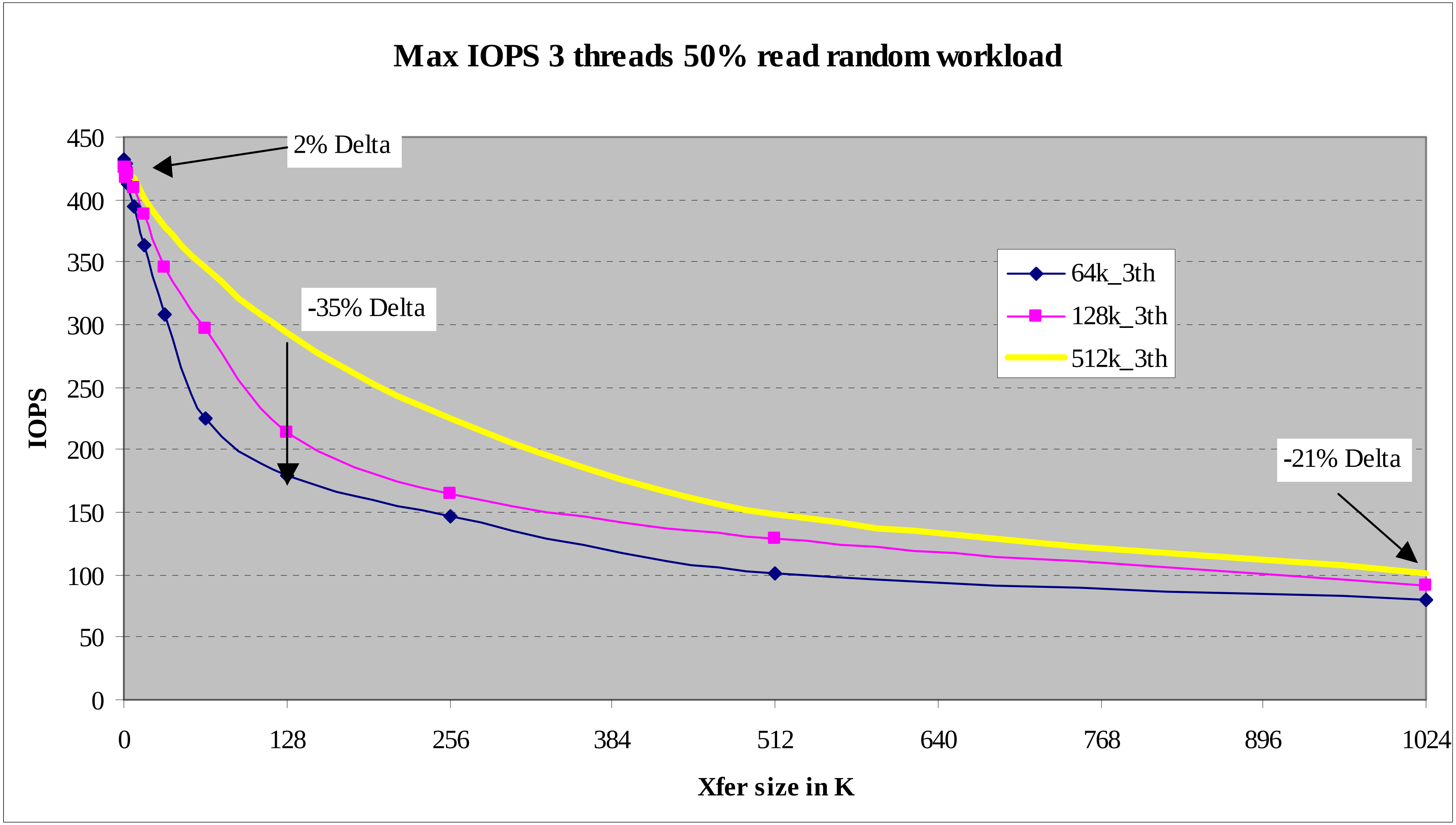


Random Writes MAX Throughput in IOPS

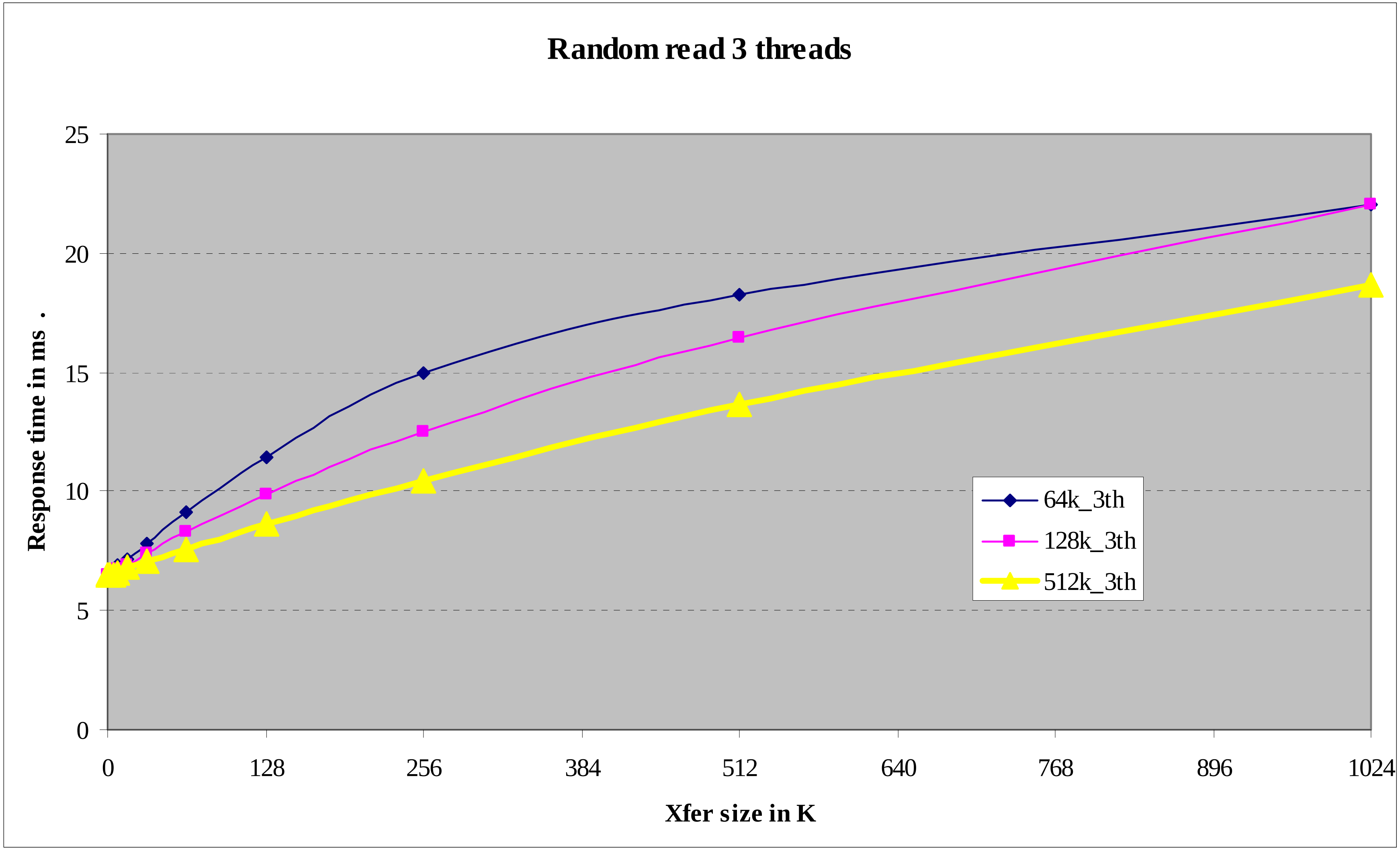
STORAGE DEVELOPER CONFERENCE
SNIA SANTA CLARA, 2008



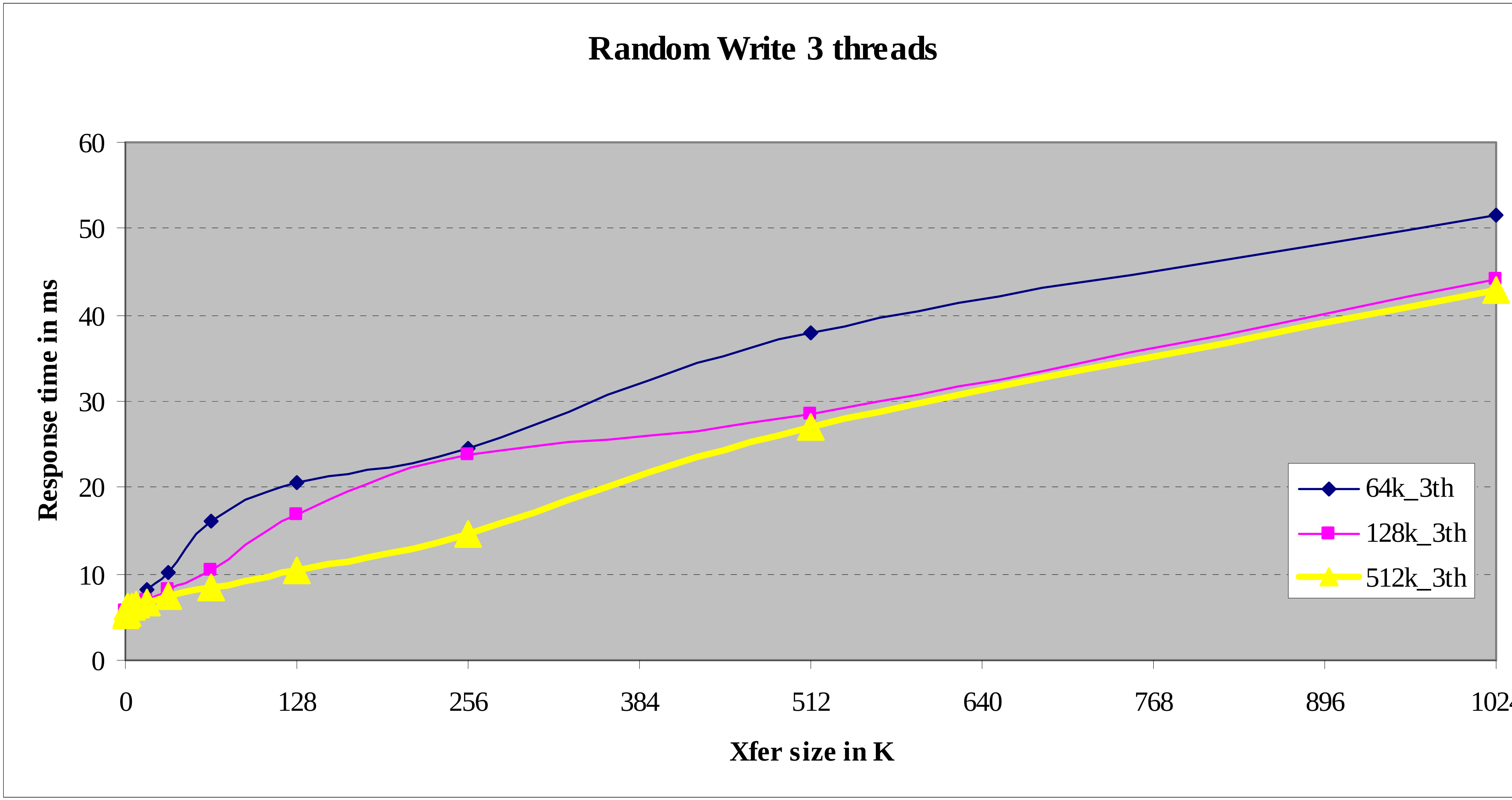
Mixed workloads 50% Write



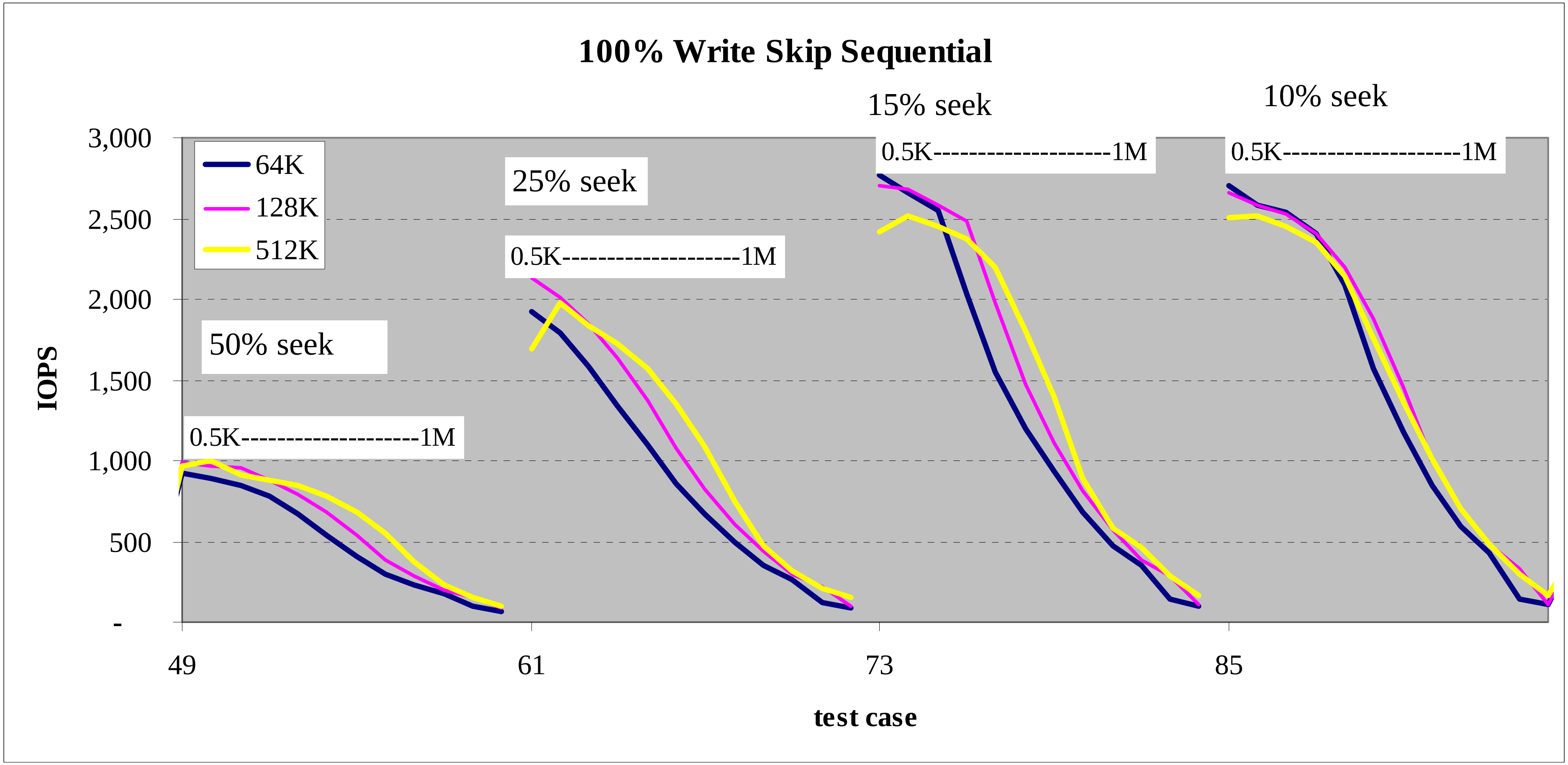
Read Response Time Improve



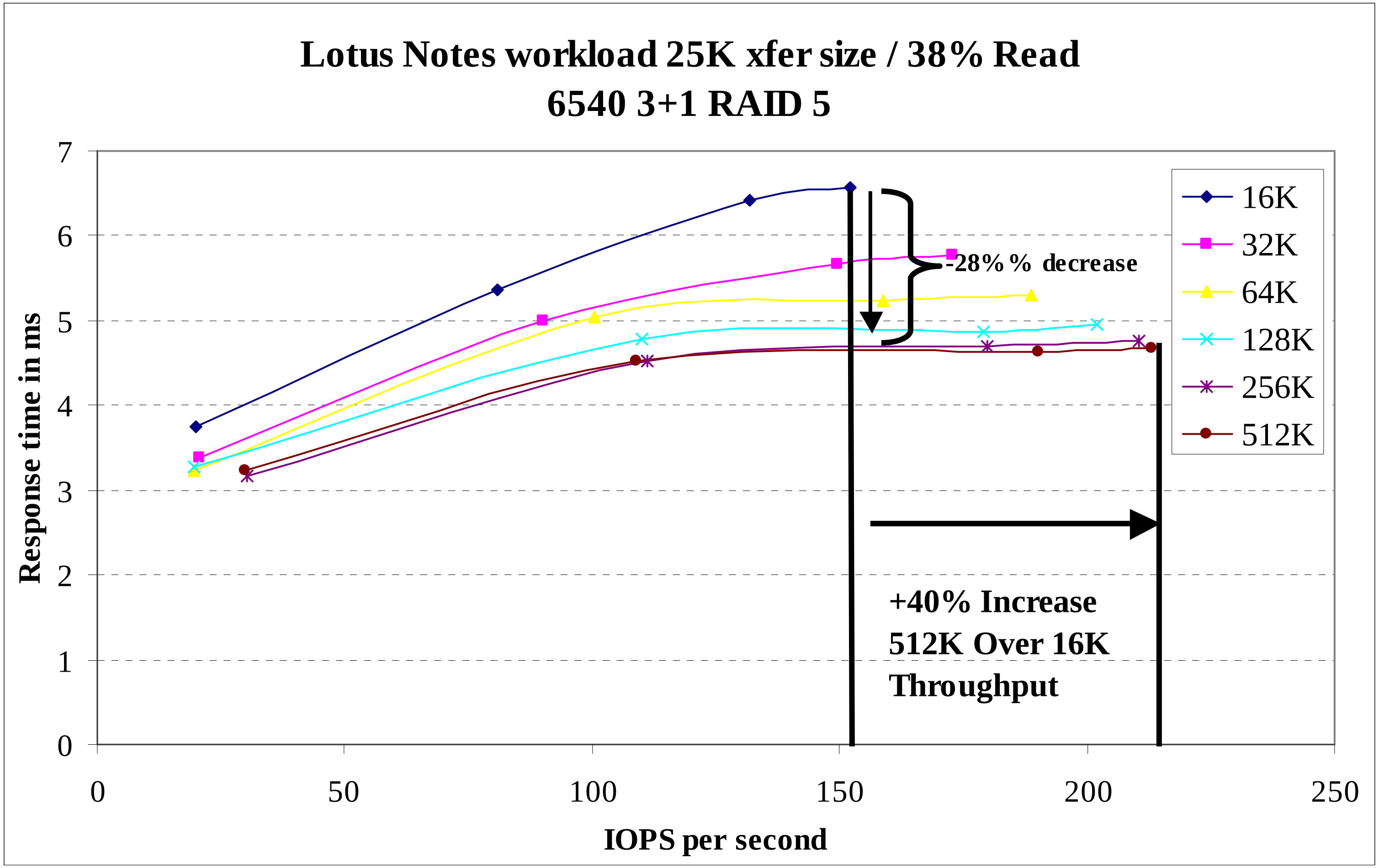
Better Write Response Times



Skip Sequential Workloads



Lotus Notes workload



Source: Sun StorageTek 6540/6140 Tuning and sizing guide for Lotus Notes/Domino environments by Tom Hanvey

In Summary

- Overview of SAS, FC, SATA
  - SATA is for very infrequently access data or very very sequential workloads
  - Transfer times are dropping rapidly with Moore's Law
- Stripe Depth Matters!
  - Breaking a system read or write into multiple IOs across multiple drives hurts Response Time and Throughput

Questions?

Steven Johnson

Performance Scientist

Sun Microsystems