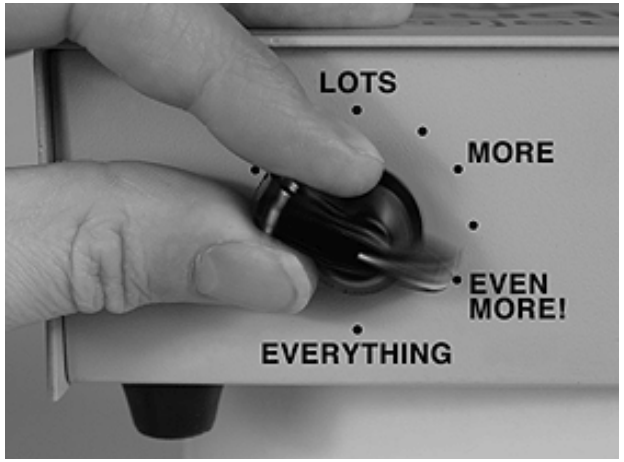
Get Software Out of the Way

Of High Performance Flash Arrays

Tom Matson
Business Development Manager, Storage, One Stop Systems

Application / Storage Trends

- More Data
- More Users
- More Flash
- More Network Bandwidth
- More Low Latency…
- More Feature Options
- More Complexity



What is the Answer?

- Must Be Easy
  - Just pick your favorite All Flash Array
  - Plug it in and all problems are solved!

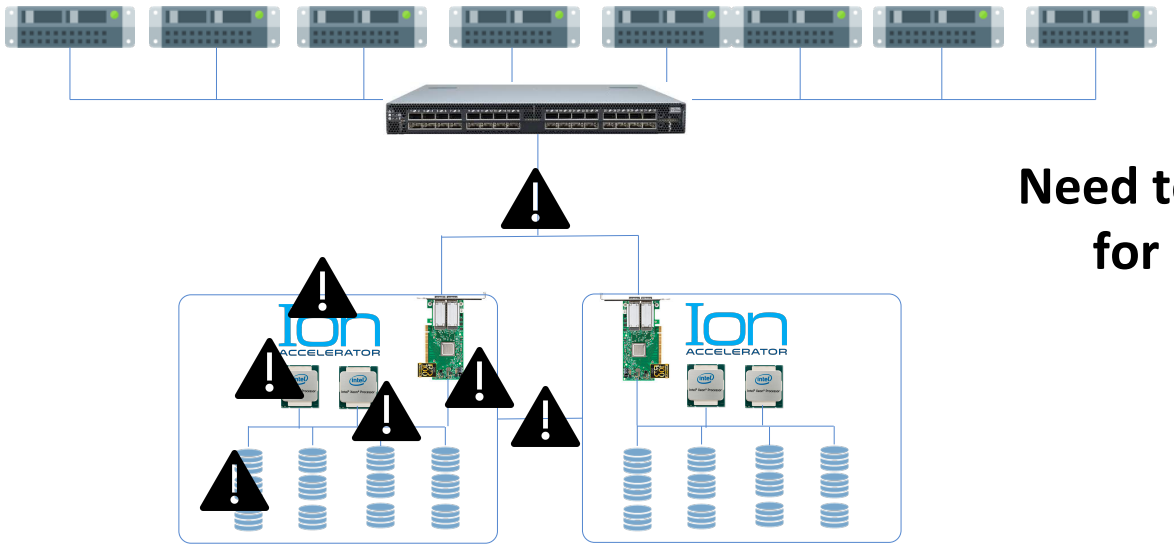
- But Wait a Minute
  - Every customer application mix is different
    - Data sets, read/write mix, latency requirements
  - And customization isn't usually an option

AFA Reality

- Design Points
  - Every vendor has a starting point for their architecture
  - Older ones were designed for SAS SSDs or even hard drives
  - Compromises made along the way
  - Legacy software features come along for the ride
  - Software builds up over time

- Upgrading to NVMe
  - Throughput and latency changes the equation
  - To achieve full performance, need to eliminate the bloat and balance the system

So Let's Dig In



Need to Eliminate All Bottlenecks for Balanced Performance

Potential Design Bottlenecks

- Initiator to Storage Network
  - 10 Gb/s Ethernet used to seem so fast!
  - With NVMe speed, needs to be much faster
  - 2 port EDR InfiniBand (100 Gb/s); 2 port 100 Gb/s Ethernet; 32 Gb/s FC (4 port)

- NVMe Drive Connection
  - Not all slots are created equal!
  - Many systems route x16 PCIe buses to 2 sections of drives
  - After first 4 drives, theoretically out of bandwidth!
  - Reality is that 6-8 drives are needed to saturate

Design Bottlenecks (cont)

- NVMe Drives
  - NVMe doesn’t mandate high performance!
  - Check the IOPS (R and W) and write endurance to match application requirements
  - Keep in mind the bandwidth limit of the internal PCIe bus connection

- Network Cards
  - Be careful that the internal PCIe routing doesn’t limit bandwidth
  - A x4 connection only supports 4 GB/s (32 Gb/s)
  - Dual port for performance and fault tolerance (4 port for FC performance)

Design Bottlenecks (cont)

- Processor
  - Watch out for how the PCIe links are committed
  - Integrated networking not always the way to go

- High Availability Link
  - HA with dual systems requires replication link
  - Link performance can have a huge impact, as data must be committed
  - Similar to network connection, don't limit to x4
  - Need method to throttle to minimize impact to application performance

Design Bottlenecks (cont)

- Software Features
  - Before NVMe, CPU had plenty of time to handle features
- NVMe Makes Everything Worse!
  - Less time spent waiting for data delivery
  - Overhead is more exposed
- Features Can Chew Up Processor Cycles and Increase Latency
  - Processor-intense compression
  - File systems
  - Encryption
  - Management layer
  - Snapshots

A Little Test

- read_u128_kb1024_0n1.out: read: IOPS=2847, BW=2848MiB/s
- read_u128_kb1024_0n1:1n1.out: read: IOPS=5690, BW=5691MiB/s
- read_u128_kb1024_0n1:1n1:2n1.out: read: IOPS=8525, BW=8526MiB/s
- read_u128_kb1024_0n1:1n1:2n1:3n1.out: read: IOPS=11.3k, BW=11.1GiB/s
- read_u128_kb1024_0n1:1n1:2n1:3n1:4n1.out: read: IOPS=12.4k, BW=12.2GiB/s
- read_u128_kb1024_0n1:1n1:2n1:3n1:4n1:5n1.out: read: IOPS=12.5k, BW=12.2GiB/s

- read_u128_kb1024_0n1:1n1:2n1:3n1:4n1:5n1:6n1.out: read: IOPS=14.5k, BW=14.2GiB/s
- read_u128_kb1024_0n1:1n1:2n1:3n1:4n1:5n1:6n1:7n1.out: read: IOPS=16.6k, BW=16.2GiB/s
- read_u128_kb1024_0n1:1n1:2n1:3n1:4n1:5n1:6n1:7n1:8n1.out: read: IOPS=18.6k, BW=18.2GiB/s
- read_u128_kb1024_0n1:1n1:2n1:3n1:4n1:5n1:6n1:7n1:8n1:9n1.out: read: IOPS=20.7k, BW=20.2GiB/s
- read_u128_kb1024_0n1:1n1:2n1:3n1:4n1:5n1:6n1:7n1:8n1:9n1:10n1.out: read: IOPS=22.7k, BW=22.1GiB/s
- read_u128_kb1024_0n1:1n1:2n1:3n1:4n1:5n1:6n1:7n1:8n1:9n1:10n1:11n1.out: read: IOPS=23.7k, BW=23.2GiB/s

Summary

- Data Delivery is the key to application performance today
- All Flash Arrays don’t all solve the same problem
- Balancing every path is critical
- Choose your flash solution wisely!