BoF Discussion Agenda & Introductions – NVMeoF/TCP Offload

Opening Remarks

This panel explores the challenges, issues, and benefits of addressing NVMe over TCP deployments without compromise. The session will explore server, storage and I/O workload testing techniques, tools, methodology and approaches to show NVMe over Fabrics including TCP can be accelerated, while freeing up host CPU resources for other software defined workloads.

Introductions

- Greg Schulz – Independent Industry Analyst, Author, Consultant, Founder Server StorageIO™
- Bob Dugan – Director of Engineering at Chelsio Communications

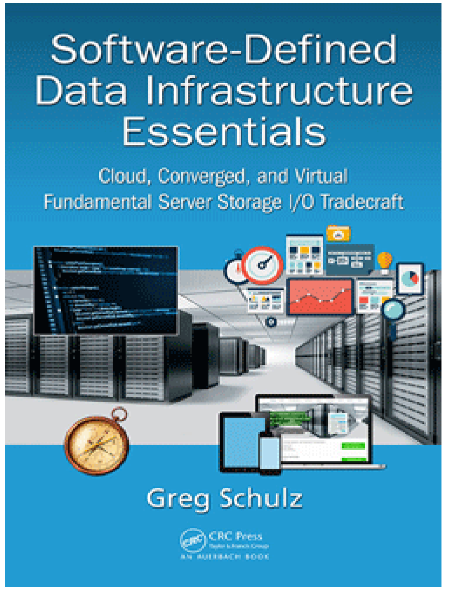
Brief Presentation and Perspectives

- Industry and Data Center Trends “Big Picture, Setting the Stage” – Greg Schulz
- Chelsio Perspectives Brief Presentation – Bob Dugan

Panel and Audience Q&A Discussion, Wrap-up

Industry & Data Center Trends – Greg Schulz StorageIO™

Greg has an Masters Software Engineering from University of St Thomas, worked as the customer in various IT organizations in roles from business applications to systems and data infrastructure. He has worked as a vendor, consulting analyst and author of several books including “Software-Defined Data Infrastructure Essentials” (CRC Press). Greg brings a diverse background with real world perspective across applications, data infrastructures, hardware, software, data protection, Performance and Capacity Planning as well as containers and clouds. Greg is a Microsoft MVP Cloud Data Center Management and previous ten-time VMware vExpert.


Software-Defined Data Infrastructure Essentials
Cloud, Converged, and Virtual Fundamental Server Storage I/O Tradecraft
Greg Schulz
CRC Press

StorageIO.com/book4

@StorageIO

- ✓ Continued shift to software defined data infrastructures (servers, storage, networks)
- ✓ Increased demand for compute resources (CPU, GPU, xPU and other offloads)
- ✓ Expanding focus from resource utilization to effectiveness and productivity
- ✓ NVMe & NVMe over Fabrics (NVMeoF) including NVMe over TCP
- ✓ Many I/O and storage performance problems are software and CPU problems
- ✓ Software needs hardware, hardware needs software, even serverless ;)

Everything is not the same, why treat everything the same with NVMe?

- In the datacenter (cloud, on-prem or edge), everything is not the same with different application workloads and characteristics, as well as resource needs and demands (hw, sw, network, compute, I/O, memory and storage). So why treat everything the same?

- Likewise everything is not the same when it comes to hw, sw, networks, including compute, I/O, memory and storage, along with software, how they are (or can) be used.

- Focusing on server (compute) and storage I/O networking connectivity, everything is not the same with different physical networks and fabrics as well as protocols and various components.

- With NVMe everything is not the same. From PCIe “x” slots to M.2, U.2 as well as NVMe over Fabrics including TCP among others. NVMe over TCP can be with software using standard Ethernet networking and host CPU cycles as an initiator or target, as well as with TCP Offload Engines.

- If everything is not the same, how do you know what to use when, where, why and how?

Everything is not the same, why treat everything the same with NVMe?

- If everything is not the same, how do you know what to use when, where, why and how?
- Answer is understanding the technology, workloads, resource needs, configuration, validation
- How to validate, verify, test and compare while being applicable to what you are trying to do?
  - ✓ What are you testing, where’s the focus, target, initiator, both?
  - ✓ What are you measuring and its applicability, as well as from where?
  - ✓ Target, initiator, network, speed of a device, all the above?
  - ✓ Metrics that matter: CPU, memory, I/O bandwith, IOPs/operations, response time/latency, errors, queues
  - ✓ Configuration: hw & sw config, with or without offload, number of devices (target, initiator)
  - ✓ Test configution: I/O size, activity rate, duration, size of target storage space (e.g. fit in cache, or larger)
  - ✓ Are you testing for marketing “hero” numbers, or validate something works as intended?

Part 2: Optimal Performance and OpEx for Enterprise and Cloud Ethernet Storage

BOF Session: Best Practices for NVMe/TCP Deployment

Presented by Bob Dugan

Chelsio Communications, Inc.

Agenda

- NVMe/TCP using TCP/IP and NVMe PDU Offload
  - Chelsio T6 – NVMe/TCP using TCP/IP Offload Engine (TOE)
  - NVMe/TCP (TOE) Layering – Closer View
  - Chelsio T7 – NVMe/TCP using TCP and NVMe PDU Offload
- Performance Benchmarks
  - NVMe/TCP (TOE) – BW & IOPs Test Configuration
  - Kernel NVMe/TCP (TOE) – CPU Savings
  - NVMe/TCP (TOE) – Bandwidth & IOPs
  - NVMe/TCP (TOE) – Latency Test Configuration
  - NVMe/TCP Latency Comparison
- Testing NVMe/TCP (TOE)
  - No Compromise Testing
- Conclusions
- Q&A and General Discussion

NVMe/TCP using TCP/IP Offload

NVMe/TCP using TOE – Highlights

- Extends NVMe over fabrics using TCP/IP at large scale
  - TOE allows scaling more effectively
  - Free ups CPU from network system overhead
  - Reduces congestion on the network
- NVMe/TCP using TOE first proof point
  - Chelsio 100GbE TOE
    - 12.66 µs delta latency between remote and local storage
    - 2.8 Million IOPs at 4K I/O size
    - Reduced host CPU by up to 55% vs host-based TCP/IP

Chelsio T6 Use Case: High-Performance, Scale-out Database

OVERVIEW

Database utilizes x86 servers

Disaggregated database storage using Ethernet block storage protocols

Throughput and latency comparable to local storage

FLEXIBILITY

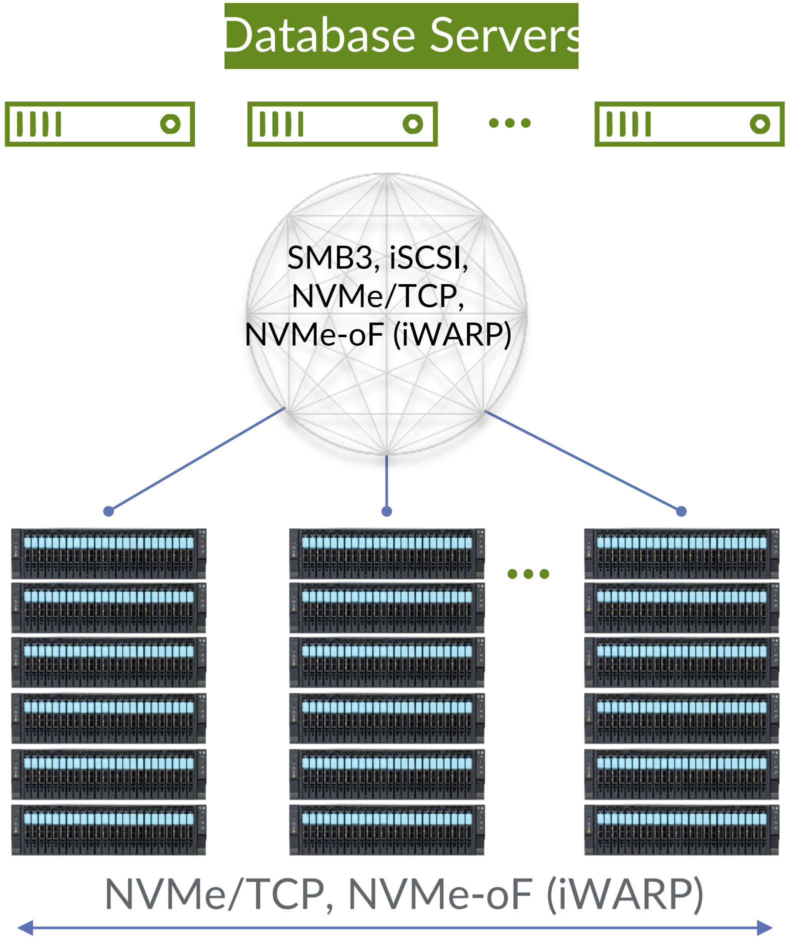
Scale storage separately from database servers

Simpler data management

COST SAVING BENEFITS

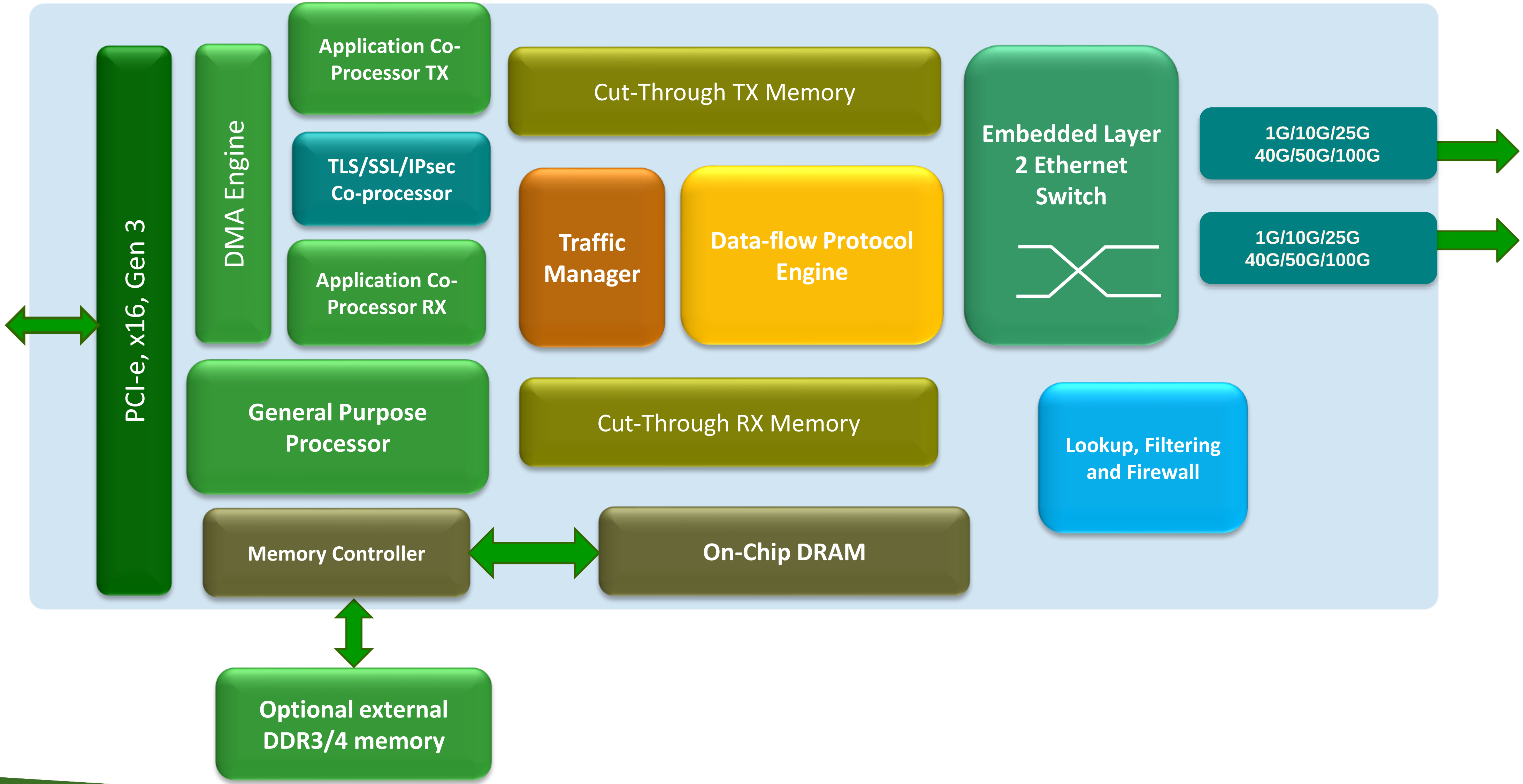
Higher storage utilization smaller/fewer database servers

Smaller footprint with database consolidation


Database Servers
•••
SMB3, iSCSI, NVMe/TCP, NVMe-oF (iWARP)
•••
NVMe/TCP, NVMe-oF (iWARP)

T6 supports iSCSI, SMBDirect, NVMe-oF (iWARP), & NVMe/TCP offload simultaneously

T6 Block Diagram


PCI-e, x16, Gen 3
DMA Engine
Application Co-Processor TX
TLS/SSL/IPsec Co-processor
Application Co-Processor RX
General Purpose Processor
Memory Controller
Optional external DDR3/4 memory
Cut-Through TX Memory
Traffic Manager
Data-flow Protocol Engine
Cut-Through RX Memory
On-Chip DRAM
Embedded Layer 2 Ethernet Switch
1G/10G/25G 40G/50G/100G
1G/10G/25G 40G/50G/100G
Lookup, Filtering and Firewall

Chelsio T7 Use Case:
Scale-out Cloud Storage

OVERVIEW

NVMe/TCP, NVMe-oF storage presented as large-scale NVMe drive

Throughput and latency comparable to local storage

FLEXIBILITY

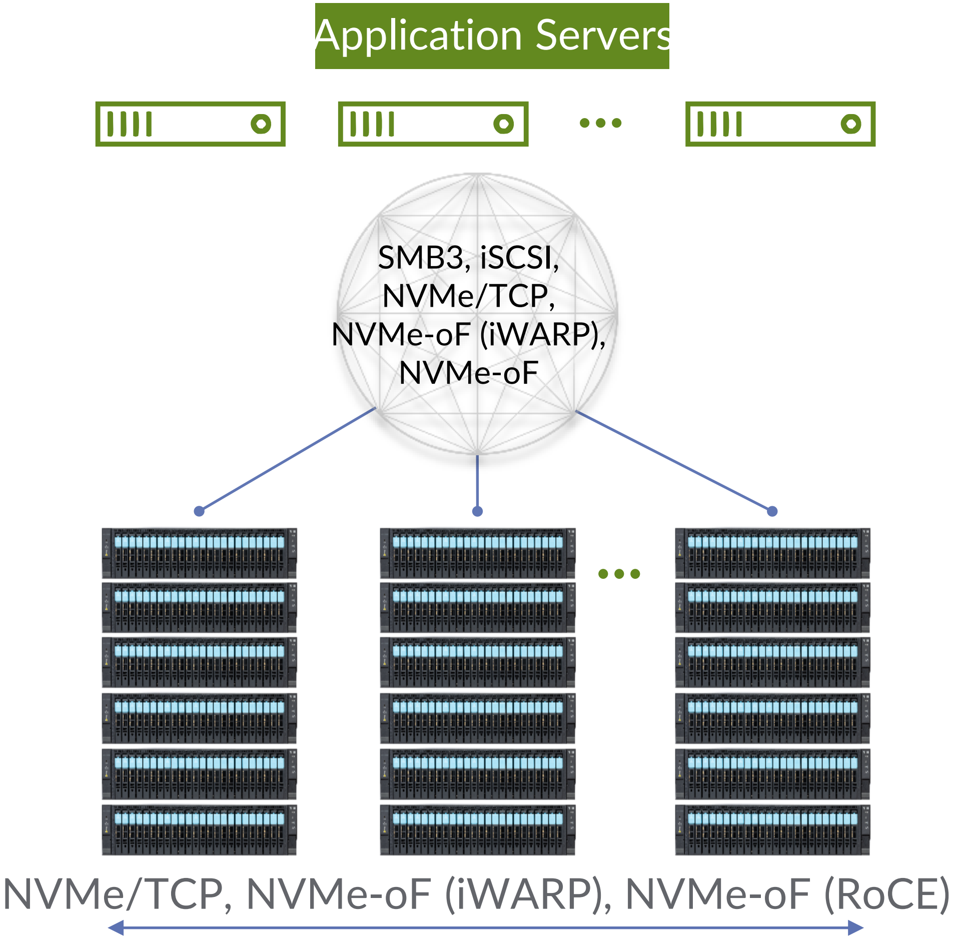
Scale storage separately from application servers

Simpler data management

COST SAVING BENEFITS

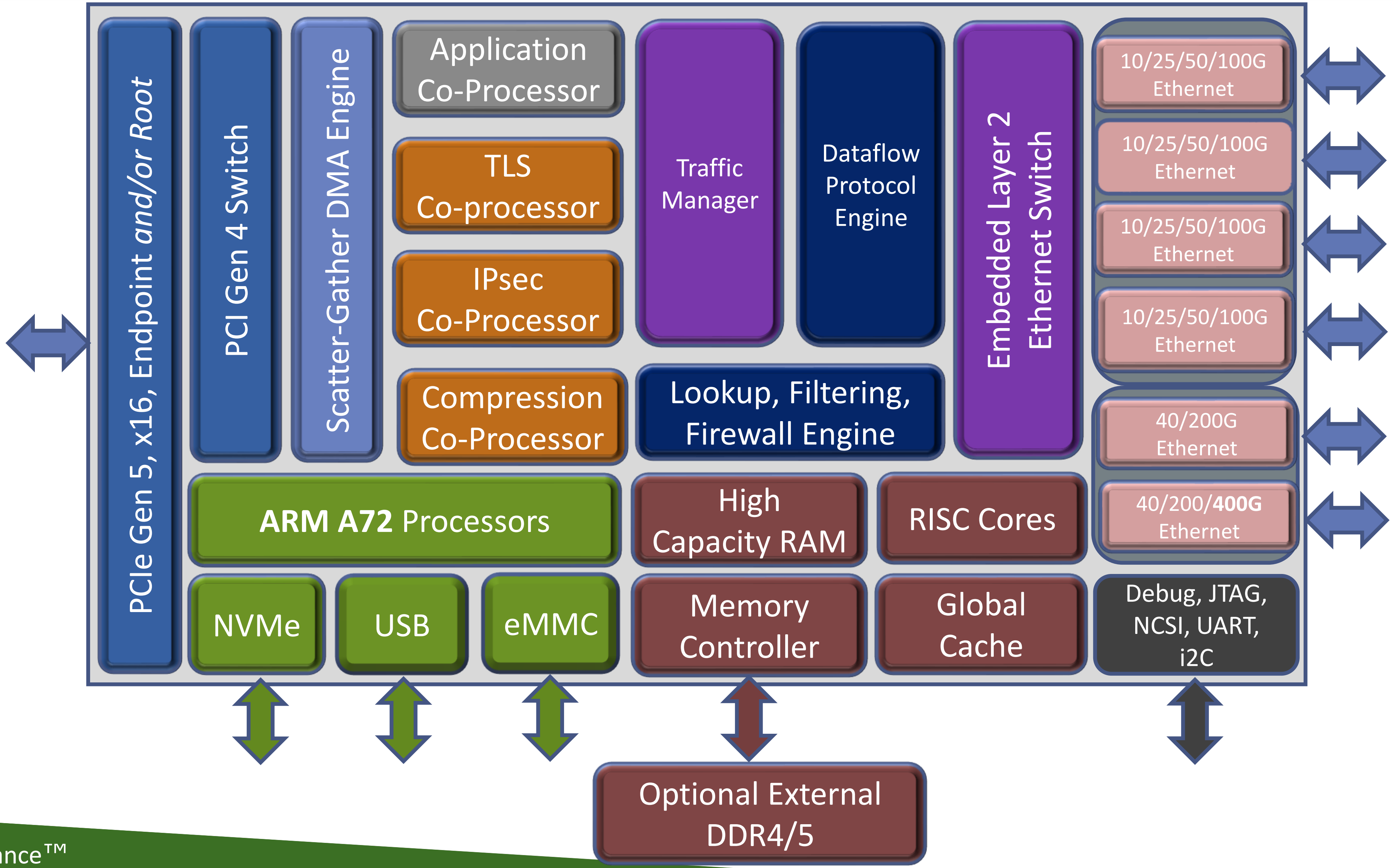
Higher storage utilization using smaller/fewer application servers

Maximum CPU cycles for tenant applications

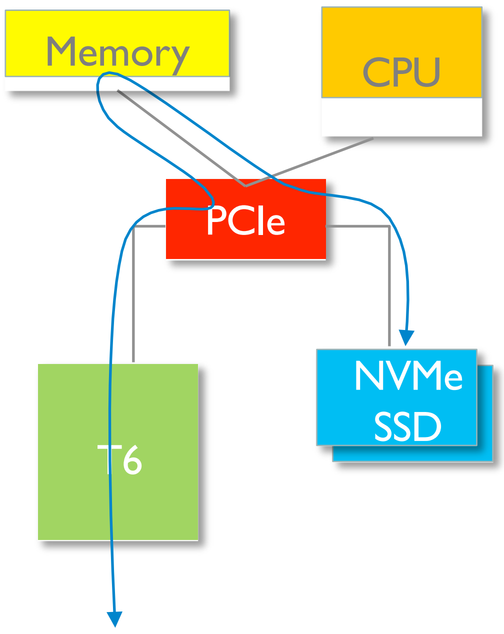

Application Servers
• • •
SMB3, iSCSI, NVMe/TCP, NVMe-oF (iWARP), NVMe-oF
• • •
NVMe/TCP, NVMe-oF (iWARP), NVMe-oF (RoCE)

T7 supports iSCSI, SMBDirect, NVMe-oF (iWARP), NVMe-oF (RoCE), & NVMe/TCP offload simultaneously

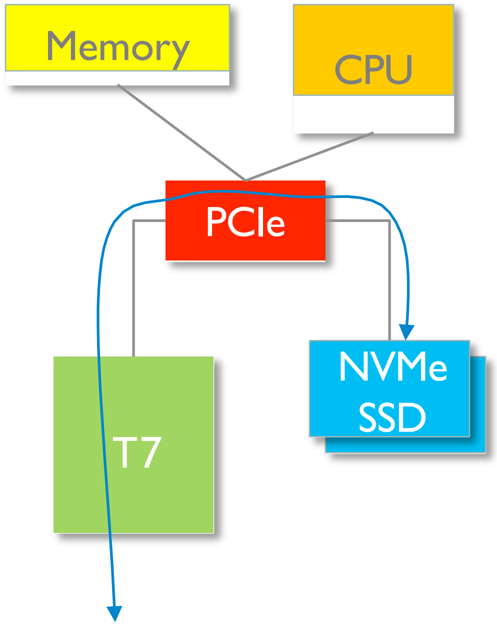
T7 DPU Block Diagram



NVMe/TCP Offload Evolution



- DMA to/from Host Memory
- CPU Handles NVMe Header/Data Transfer

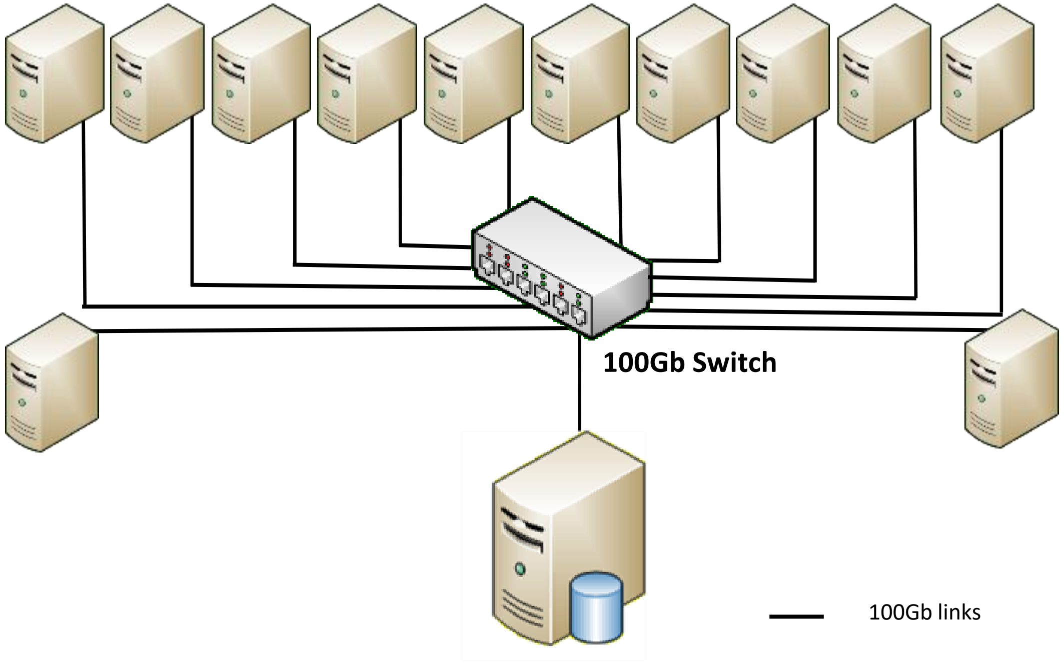


- Data Peer-to-peer DMA through PCIe
- No Host Memory Use

Data Path: ↔

Performance Benchmarks

NVMe/TCP (TOE) – BW & IOPs Test Configuration



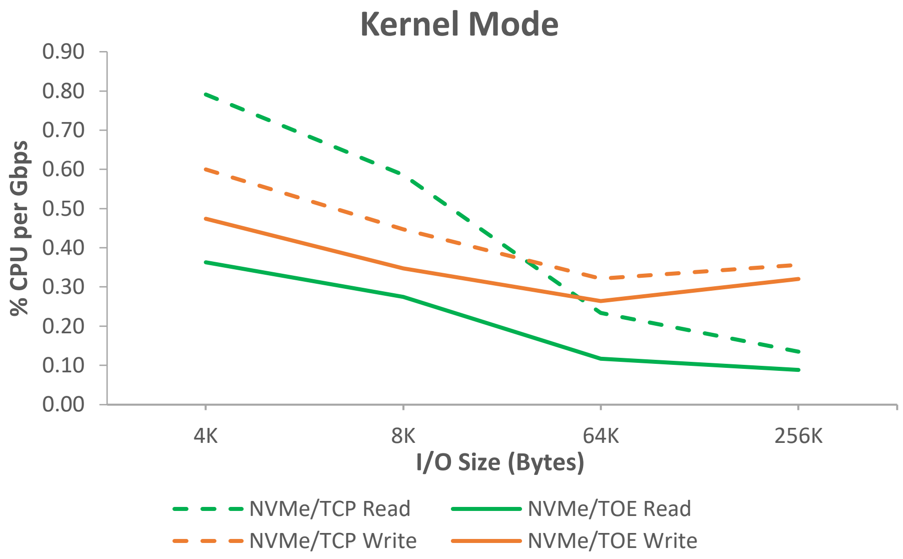
Hosts

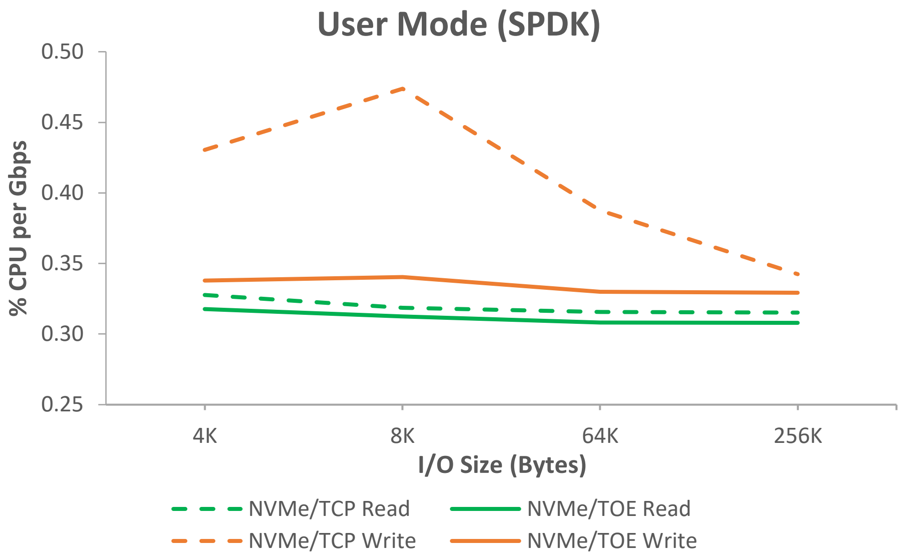
- 1 Intel Xeon CPU E5-1620 v4
- 4 cores (HT disabled) @ 3.50GHz
- 32GB of RAM
- Chelsio T62100-CR (2 x 100G ports)
- RHEL 8.6
- Kernel NVMe/TCP
- 2 connections per device
- MTU 1500

Target

- 2 Intel Xeon CPU E5-2687W v4
- 24 cores (HT disabled) @ 3.00GHz
- 128GB of RAM
- Chelsio T62100-CR (2 x 100G ports)
- RHEL 8.6 (5.4.100 kernel)
- 12 Null Block Devices (1 device/host)
- MTU 1500

NVMe/TCP (TOE) – CPU Savings





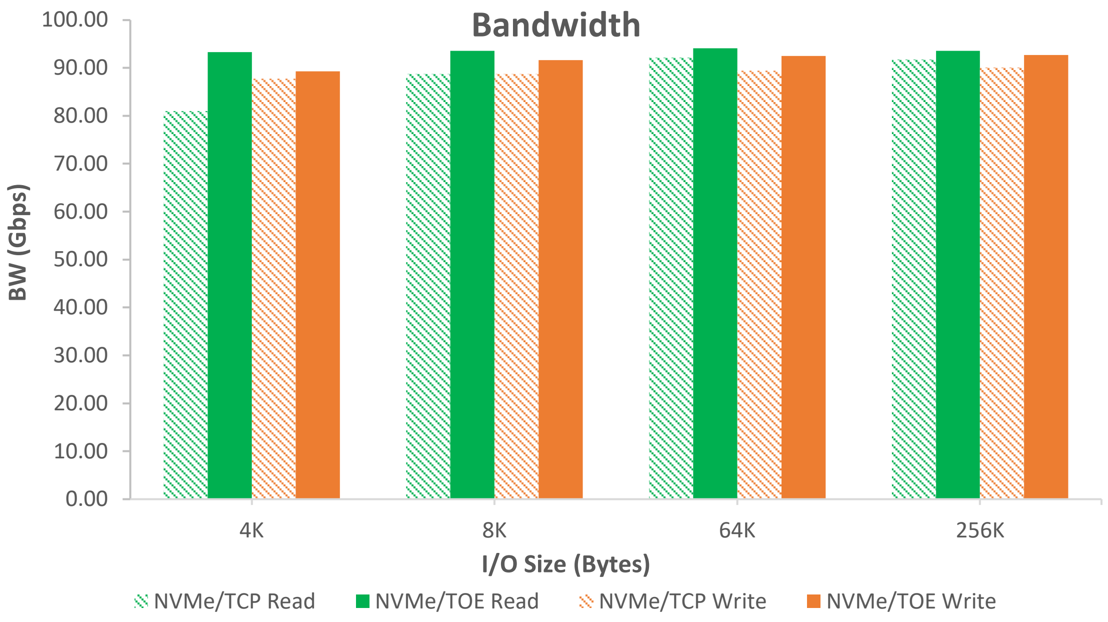
Setup

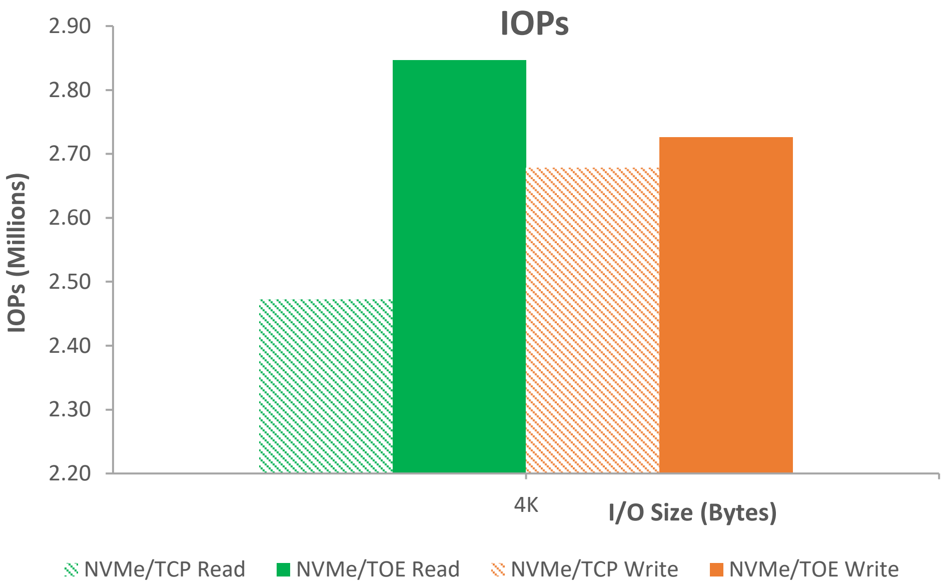
7 CPU Cores used for SPDK Application, to allow CPU headroom for other storage applications.

Summary

- Up to 55% savings in %CPU per Gbps with Chelsio TOE compared to Kernel TCP/IP.
- Up to 28% savings in %CPU per Gbps with Chelsio TOE compared to SPDK TCP/IP.

Kernel NVMe/TCP (TOE) – Bandwidth & IOPs

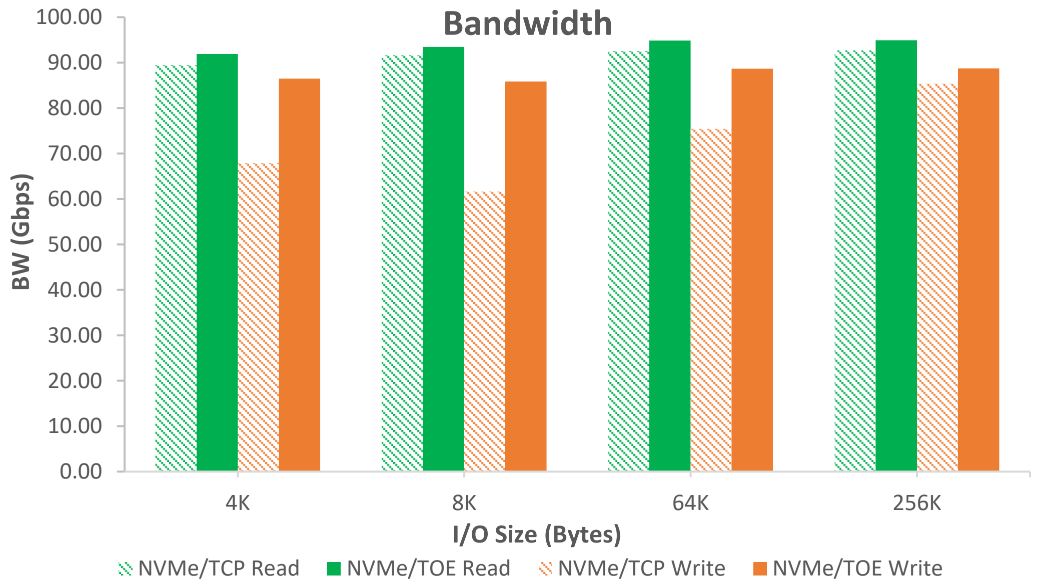


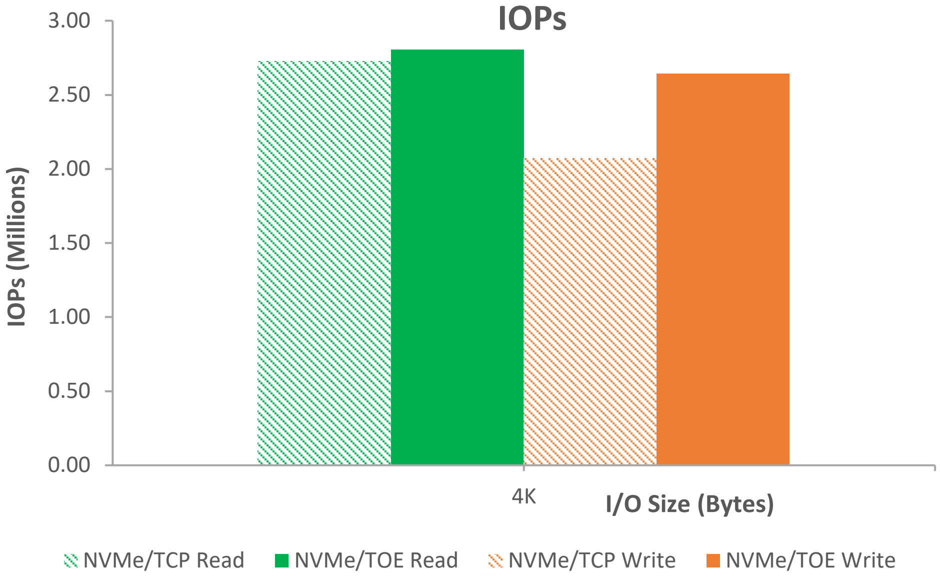


Summary – Kernel NVMe/TCP (TOE) Target

- Line Rate throughput of 94 Gbps
- 2.8 Million IOPs at 4K I/O size

SPDK NVMe/TCP (TOE) – Bandwidth & IOPs

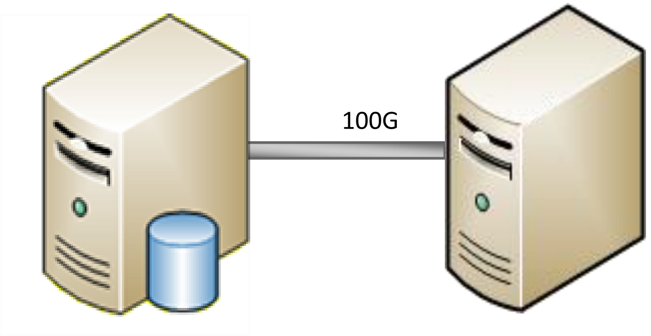




Summary – SPDK NVMe/TCP (TOE) Target

- Line Rate throughput of 94 Gbps
- 2.8 Million IOPs at 4K I/O size

NVMe/TCP (TOE) – Latency Test Configuration



Host

- 1 Intel Xeon CPU E5-1620 v4
- 4 cores (HT disabled) @ 3.50GHz
- 32GB of RAM
- Chelsio T62100-CR (2 x 100G ports)
- RHEL 8.6
- Kernel NVMe/TCP
- MTU 1500

Target

- 2 Intel Xeon CPU E5-2687W v4
- 24 cores (HT disabled) @ 3.00GHz
- 128GB of RAM
- Chelsio T62100-CR (2 x 100G ports)
- RHEL 8.6 (5.4.100 kernel)
- 1 Micron 9100 MAX 2.4TB PCIe NVMe SSD
- MTU 1500

NVMe/TCP Latency Comparison

With & Without Offload / Kernel Space & SPDK

	Read*			Write*		
Target	**Local**	**Remote**	**Delta**	**Local**	**Remote**	**Delta**
Kernel NVMe/TCP	108.24	130.99	22.75	23.79	45.29	21.5
Kernel NVMe/TOE	108.24	126.68	18.44	23.79	40.77	16.98
SPDK NVMe/TCP	108.24	129.51	21.27	23.79	45.17	21.38
SPDK NVMe/TOE	108.24	121.79	13.55	23.79	36.45	12.66

Summary

- NVMe/TCP (TOE) latencies with T6 are very close to those of local disk.
- Kernel based TOE gives ~20% latency improvements over Host-Based TCP/IP.
- SPDK based TOE gives ~40% latency improvements over Host-Based TCP/IP.
- Small changes in performance in large scale networks with high frequency and/or large volumes of traffic have a big impact!

*Latency tests used Queue Depth (QD) = 1

No Compromise Testing

- Robust testing covering functional, conformance, interoperability and stress provides stable protocol offload for NVMe/TCP.
  - Tools include fio, iozone, Dbench, SPDK fio plugin tools
  - Disks include RAMdisk, SSDs, NVMe disks
  - Adding UNH IOL test suite InterACT for conformance & plugfests for interoperability

- Performance
  - Tools include fio, iozone, Dbench, SPDK fio plugin tools
  - No compromise – delivering workload performance while maintaining interoperability
  - Performance measurement using fio providing Bandwidth, IOPs, Latency, Jitter
  - Most importantly CPU Usage .. obtained with mpstat

- Chelsio's TOE tested in usual networking scenarios which include
  - Netperf, Iperf, Netpipe, Sockperf tools
  - Applications: nfs, scp, ssh, rsh, cifs, http
  - Network related: MTU, VLANs, IP Alias, Bonding, Nagle, Pause, congestion algos

Conclusions

- Chelsio T6 TOE improves NVMe/TCP performance and scalability by
  - Allowing more host CPU cycles for application software stacks
  - Reducing host CPU costs

- Chelsio T7 TOE and NVMe PDU offload further improves NVMe/TCP performance
  - Further reducing host CPU overhead to support growth and/or do more with less
  - Further boosting server-storage I/O performance (IOPs, response time, throughput)
  - Enabling remote storage performance similar to local storage and NVMe-oF-based performance

Q&A and General Discussion

Contact info

Greg Schulz: greg@unlimitedio.com

Bob Dugan: bobdugan@chelsio.com

Please take a moment to rate this session