SDC
STORAGE DEVELOPER CONFERENCE
SNIA SANTA CLARA, 2016

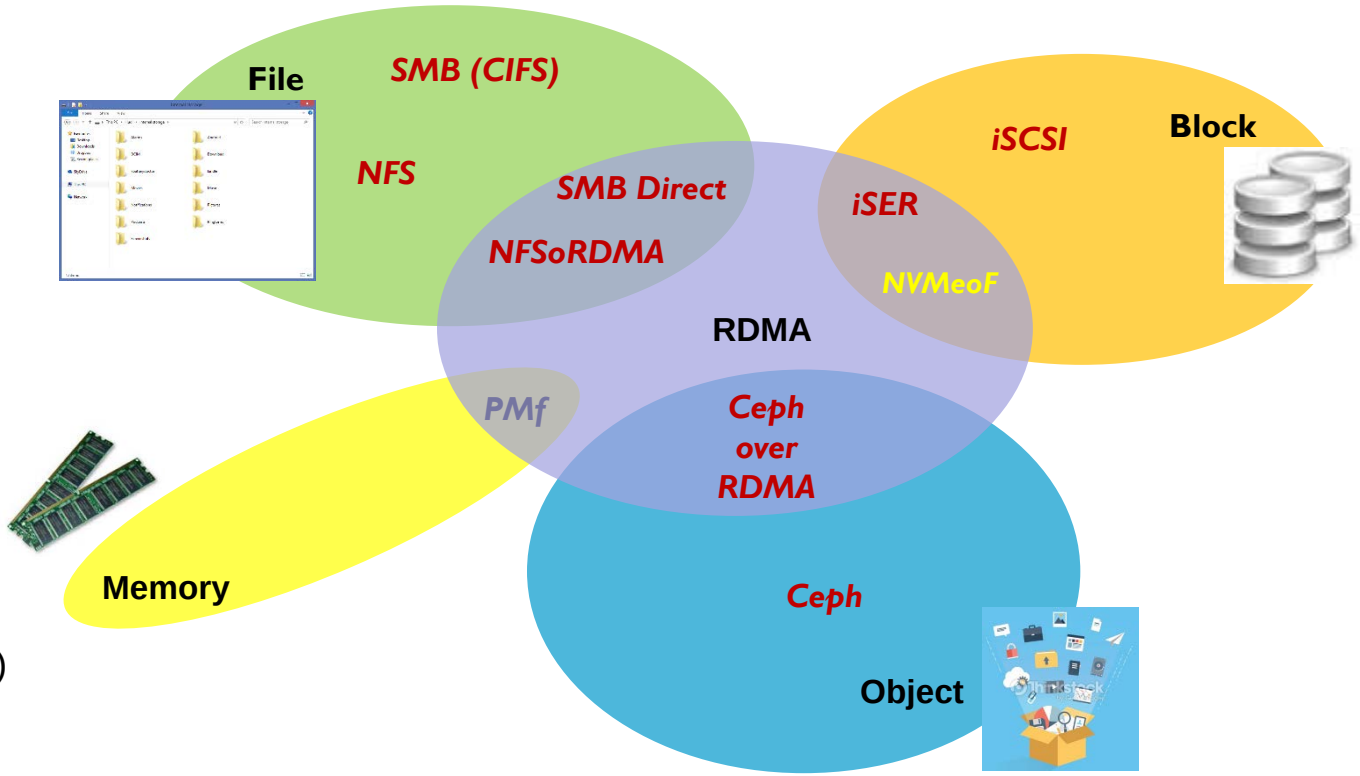
Accelerating Flash Storage with Open Source RDMA

Rob Davis & Idan Burstein

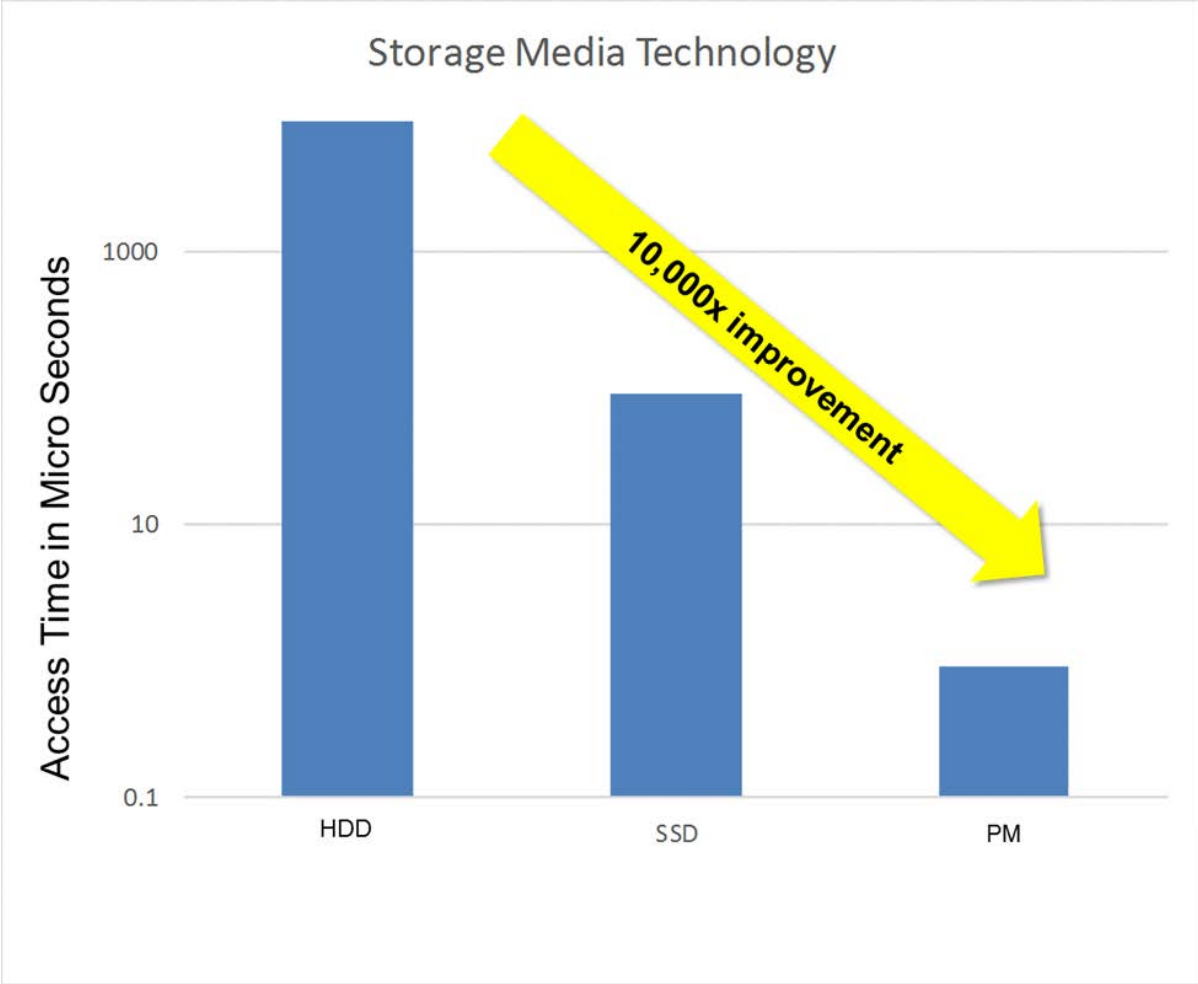
Mellanox Technologies

Open Source Flash Storage Solutions

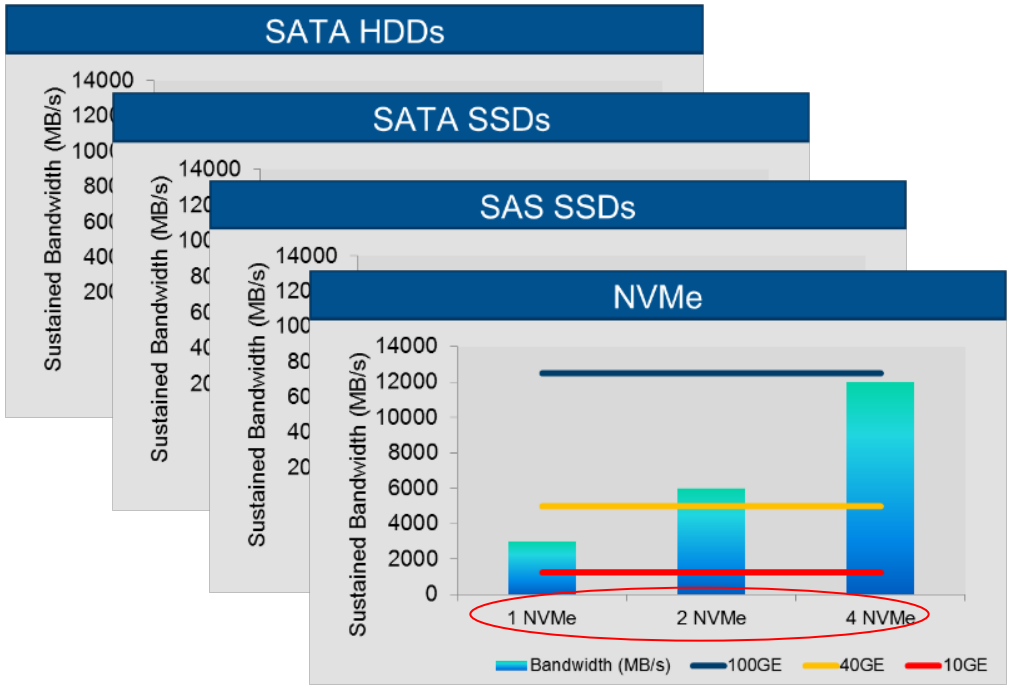
- Pure Bandwidth – up to 100Gb/s
  - Flash over Block, File and Object
- RDMA
  - RoCE, iWARP, InfiniBand
  - iSER
  - SMB Direct, NFSoRDMA
  - Ceph over RDMA
- Non-Volatile Memory (NVM)
  - NVMe over Fabrics (NVMeoF)
  - PMf (3D-XPoint)

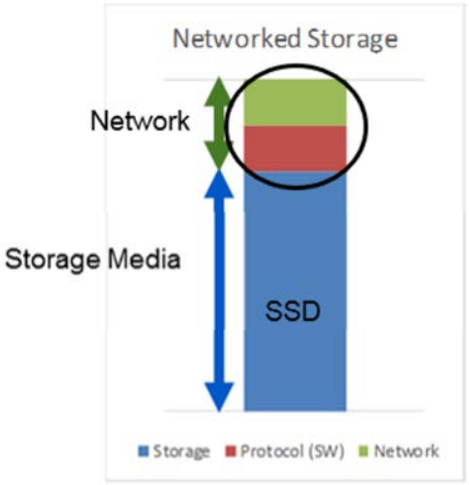


Why Should We Care About RDMA



Faster Storage Needs a Faster Network





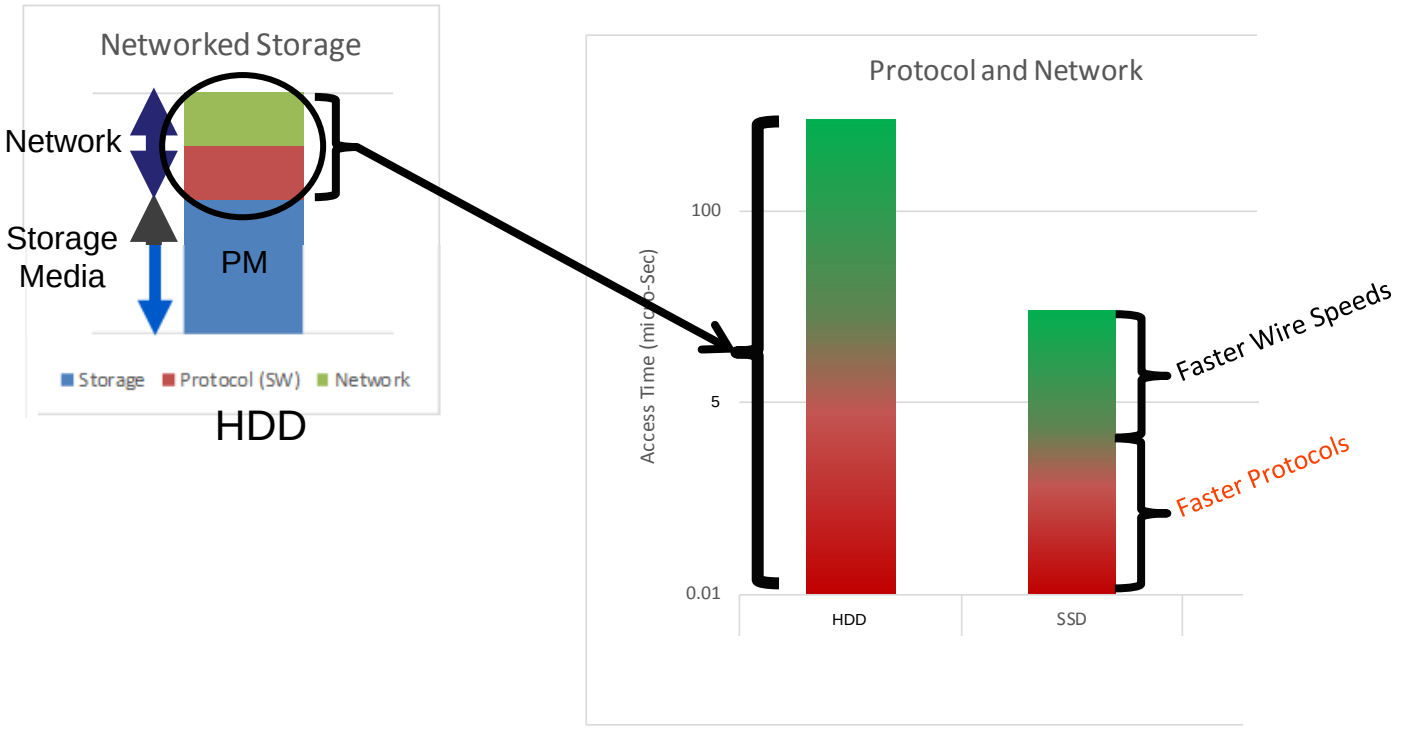
Faster Wires are Here Today

End-to-End 25, 40, 50, 100Gb Ethernet

100Gb InfiniBand

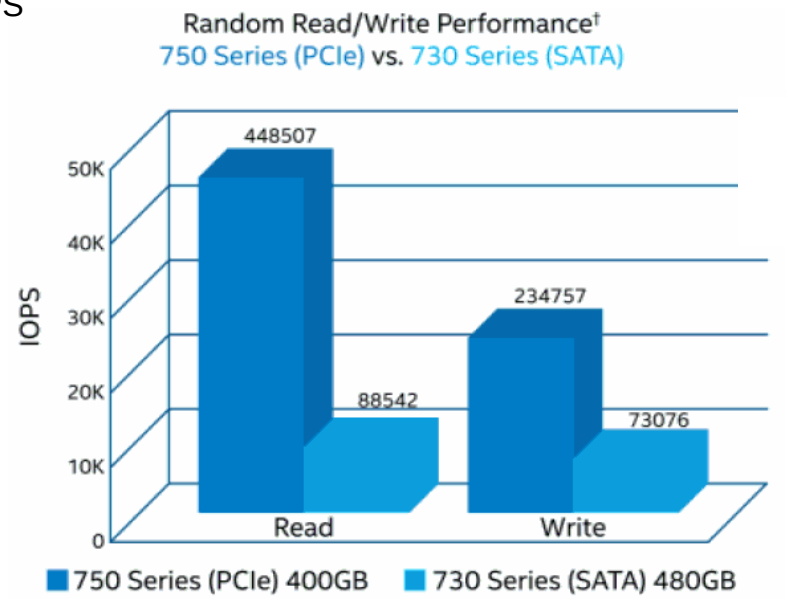
Gen4 PCIe and 32GbFC

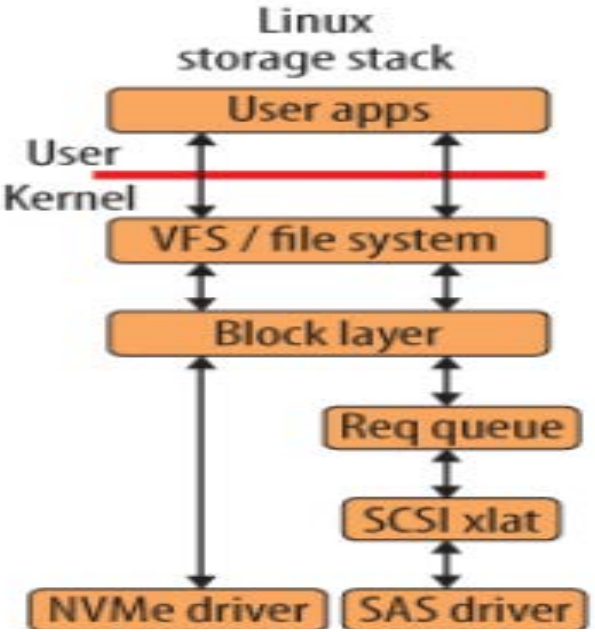
Faster Wires Only Solve ½ the Problem



Faster Protocol: NVMe

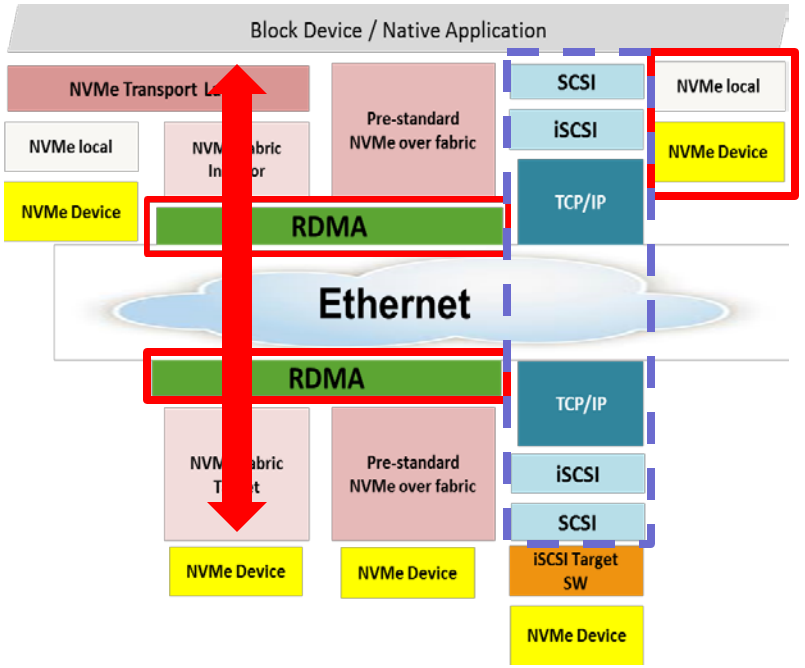
- NVMe: Optimized for flash and next-gen NV-memory
  - Traditional SCSI interfaces designed for spinning disk
  - NVMe bypasses unneeded layers
- NVMe Flash Outperforms SAS/SATA Flash
  - 2x-2.5x more bandwidth, 40-50% lower latency, Up to 3x more IOPS



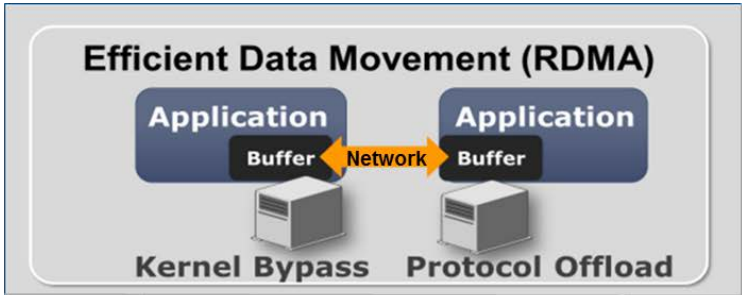


Faster Protocol: NVMf

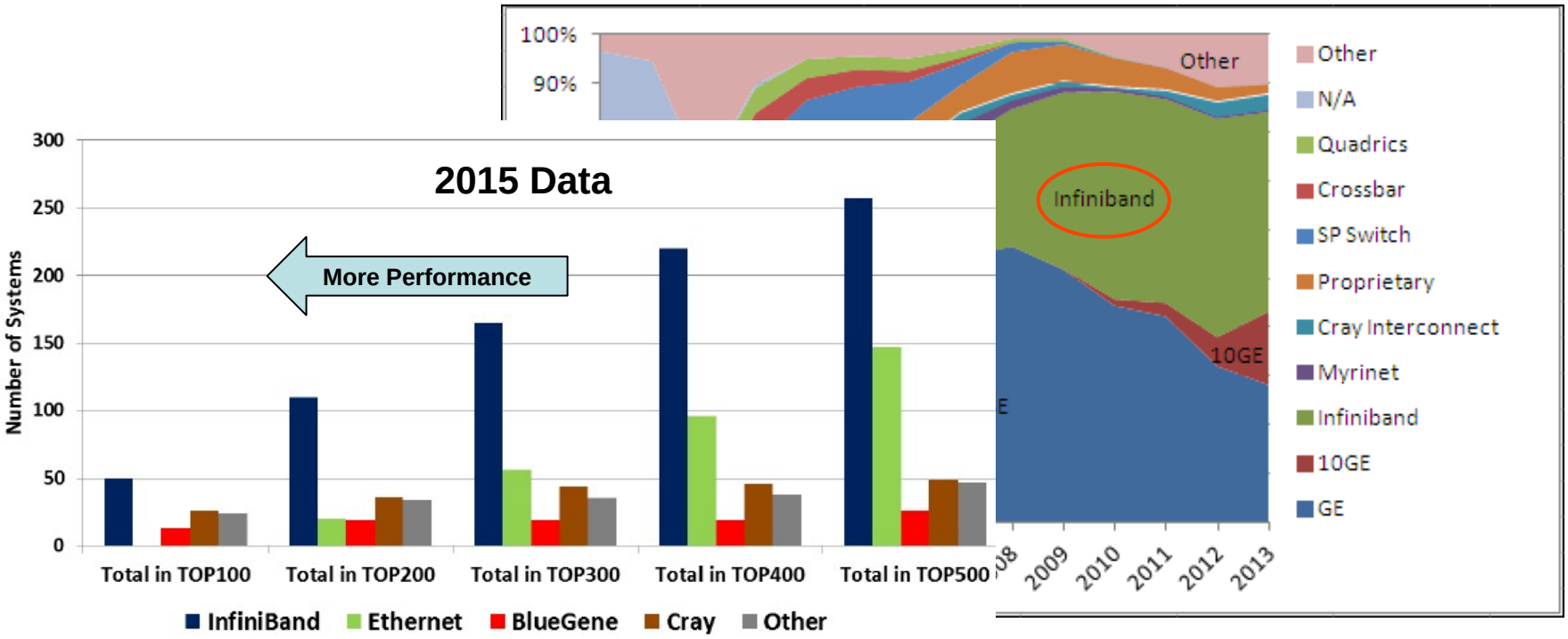
- The idea is to extend the efficiency of the local NVMe interface over a fabric
  - Ethernet or IB
  - NVMe commands and data structures are transferred end to end
- Relies on RDMA for performance
  - Bypassing TCP/IP



What is RDMA?

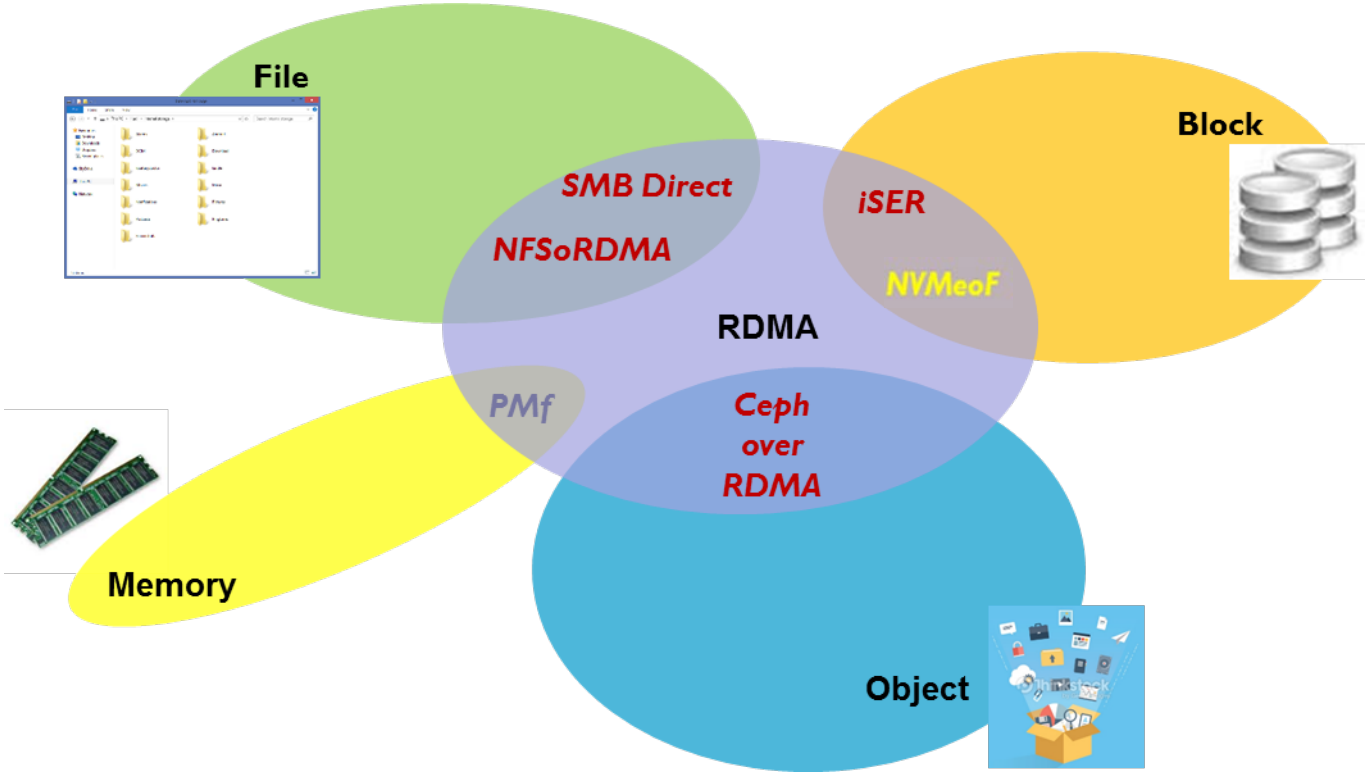


RDMA barrowed from HPC

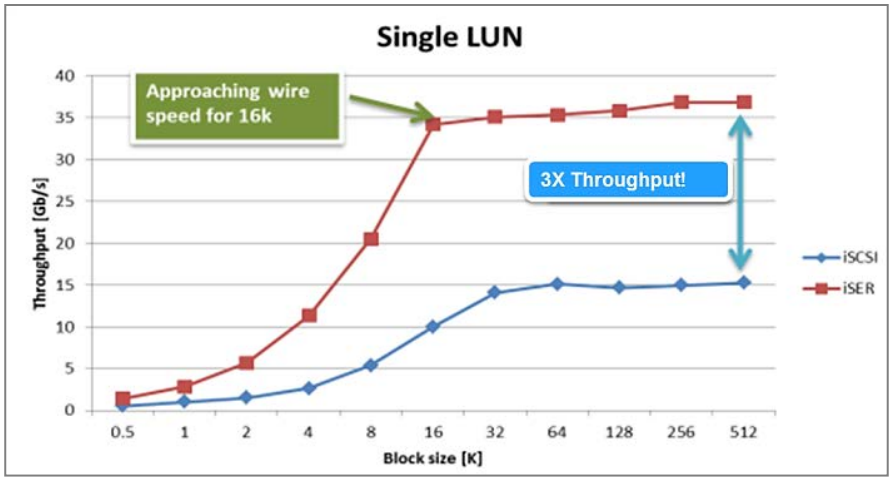


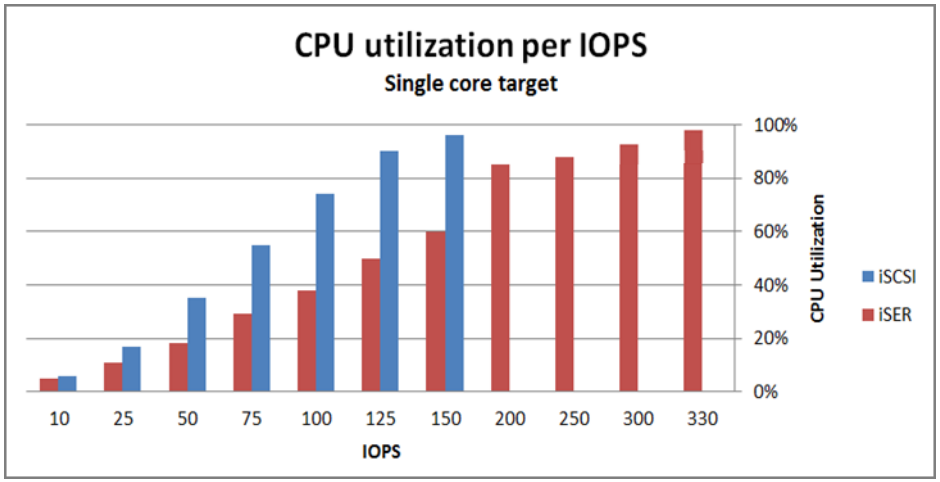
Open Source RoCE RDMA Storage Solutions Performance

- RoCE Storage Protocols
  - iSER
  - SMB Direct
  - Ceph over RDMA
- Non-Volatile Memory (NVM)
  - NVMe over Fabrics (NVMeoF)
  - PMf (3D-XPoint)



iSER Ethernet RoCE RDMA Efficiency

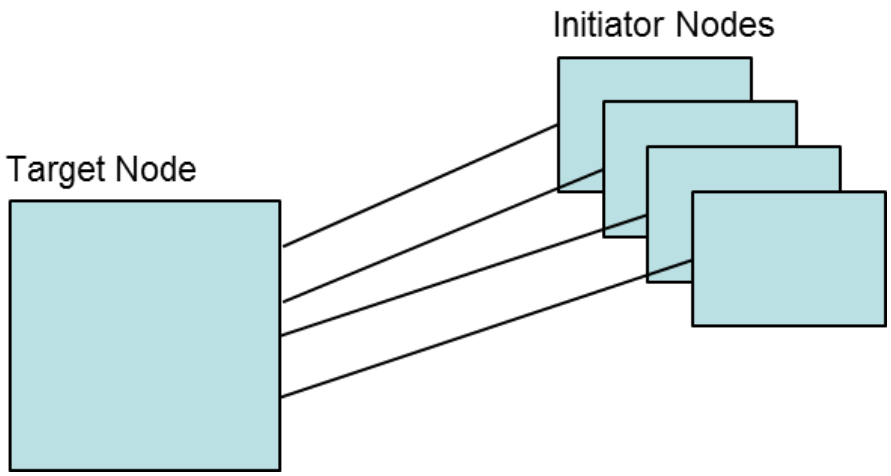




Higher Bandwidth and IOPS with Less CPU Utilization than iSCSI

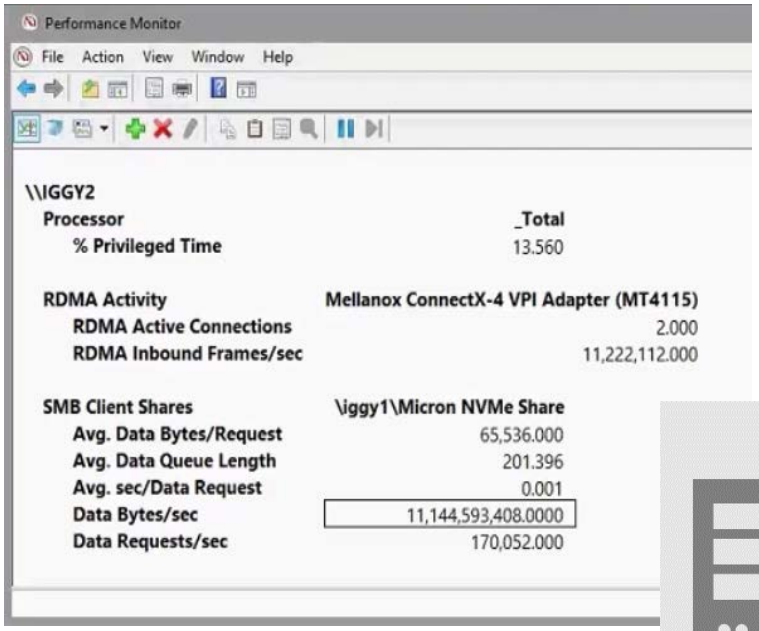
iSER RoCE Performance Demo at FMS

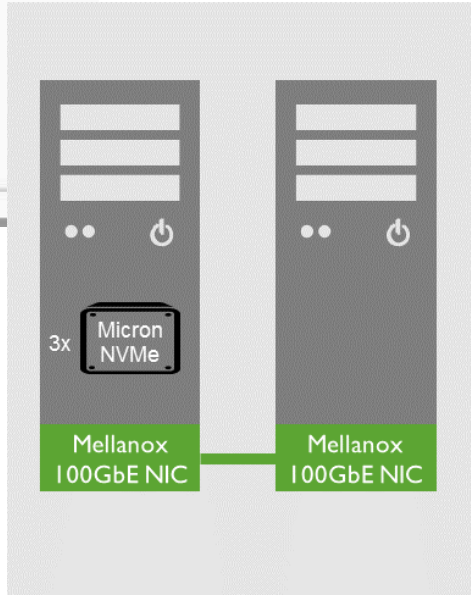
- Target node
  - Dual-socket x86 server
  - 4x40GbE NICs
  - iSER LIO target
  - 20xPM953 NVMe drives
- Initiators
  - Dual-socket x86 server
  - 1x40GbE NIC
- Performance
  - 2.1M – 4K Random Read
  - 17.2GB/s – 128K Seq Read



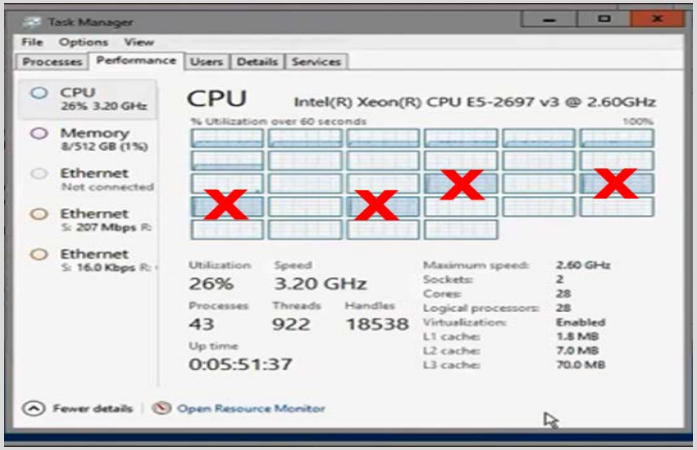
SMB Ethernet RoCE File Performance

- SMB3 using 100Gbps RDMA
- Storage Spaces using NVMe SSDs
- Over 11GB/sec over one NIC port
- 1ms latency with SMB3 storage
- Less than 15% CPU utilization





SMB RoCE File Demo at Microsoft Ignite



2X Better Bandwidth

Half the Latency

33% Lower CPU

See the demo: https://www.youtube.com/watch?v=u8ZYhUjSUoI

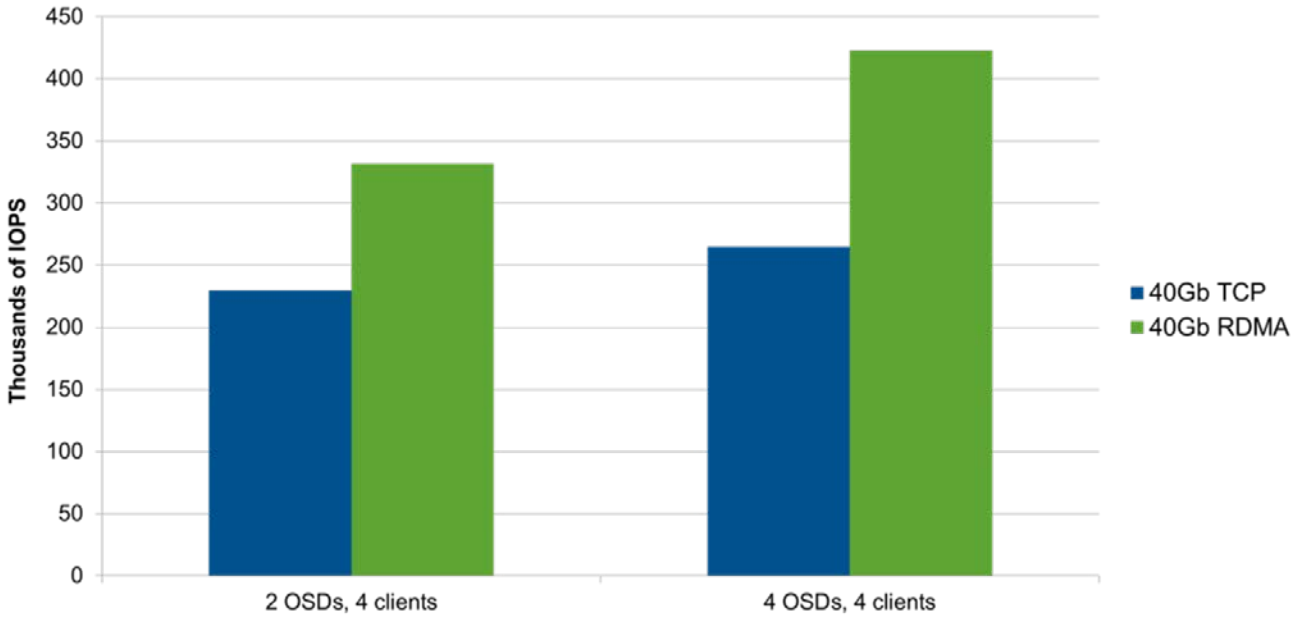
Microsoft

- Without RDMA
  - 5.7 GB/s throughput
  - 20-26% CPU utilization
  - 4 cores 100% consumed by moving data

- With Hardware RDMA
  - 11.1 GB/s throughput at half the latency
  - 13-14% CPU utilization
  - More CPU power for applications, better ROI

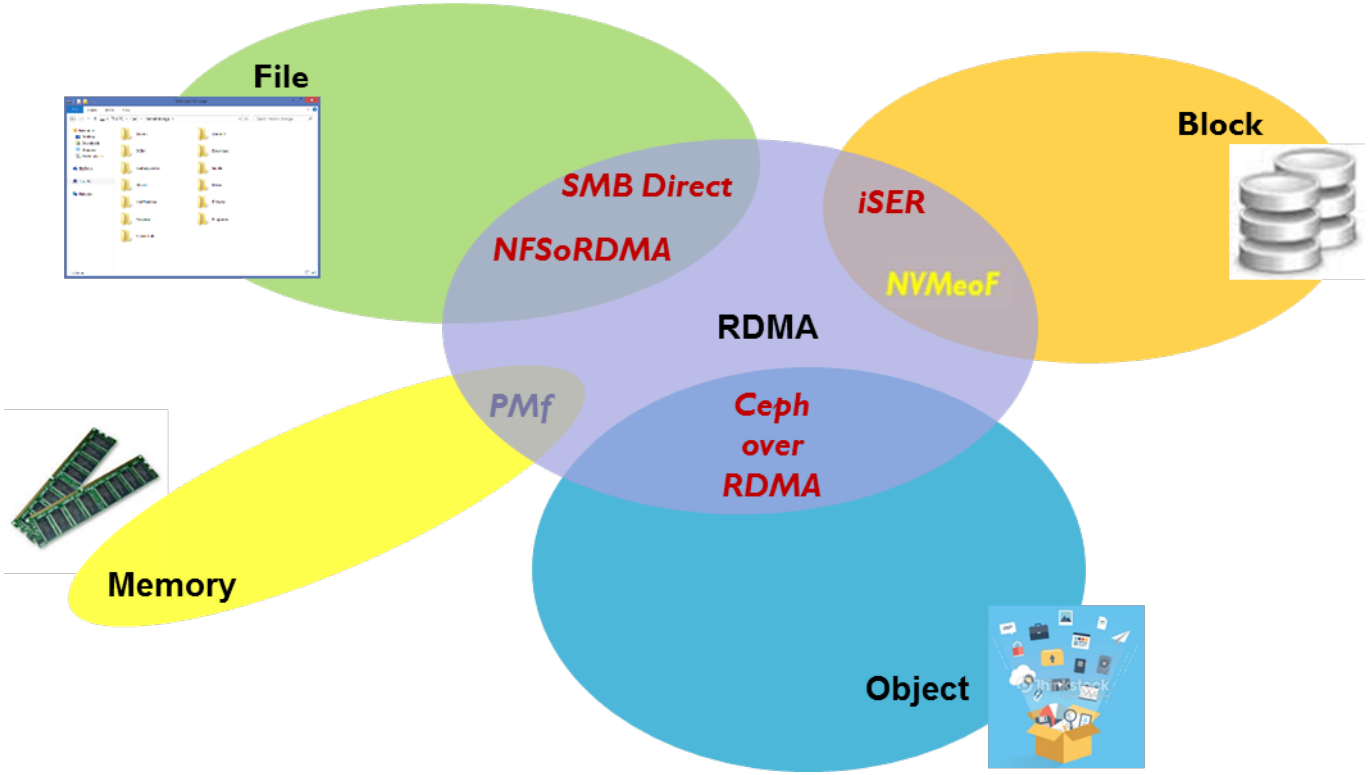
Object Storage with RoCE on Ceph

- RDMA is implemented in Ceph *Hammer Release* as Beta
- Tests show performance 30-40% better with RDMA



Open Source RoCE RDMA Storage Solutions Performance

- RoCE Storage Protocols
  - iSER
  - SMB Direct
  - Ceph over RDMA
- Non-Volatile Memory (NVM)
  - NVMe over Fabrics (NVMeoF)
  - PMf (3D-XPoint)

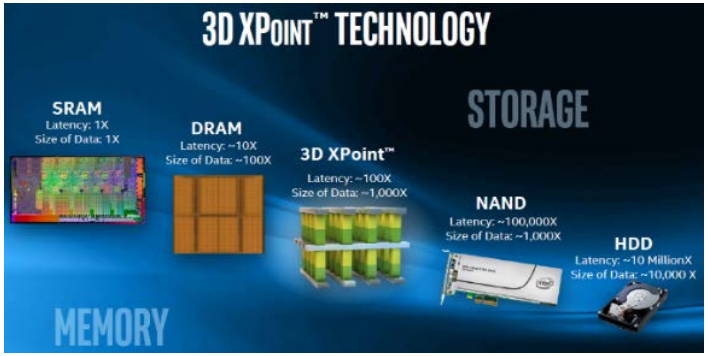


NVMEoF Version 1.0 Open Source Drivers

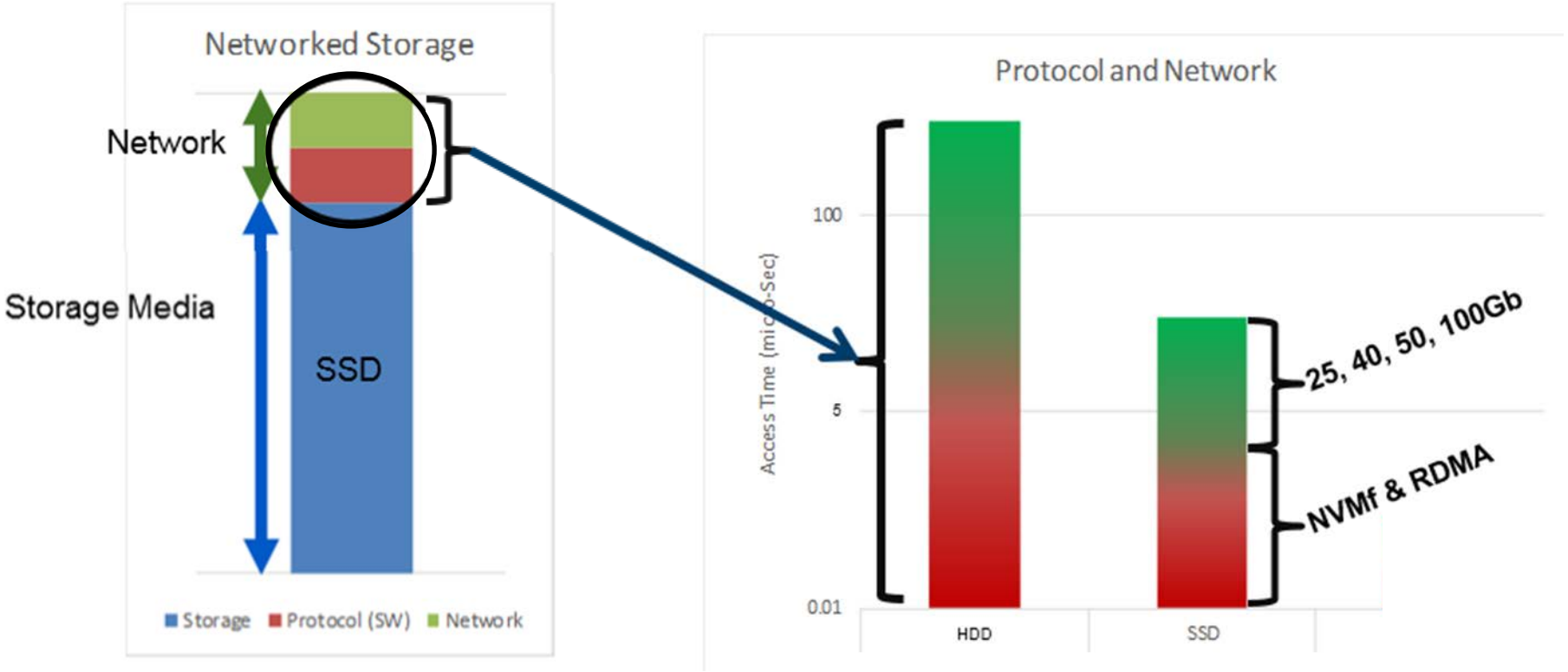
- Mellanox
- Intel
- HGST
- EMC
- Apeiron Data Systems
- Broadcom Corporation
- Chelsio Communications, Inc
- Excelero
- Hewlett Packard Enterprise
- Kazan Networks
- Kenneth Okin Consulting
- Mangstor
- NetApp
- Oracle America Inc.
- PMC
- Qlogic Corporation
- Samsung
- SK hynix Inc.

Persistent Memory in Storage

- Storage with memory performance
  - Large Write Latency Improvements over Flash
  - Byte Addressability
  - E.g. 3dxpoint, NVDIMM, NVRAM, RERAM
- Emerging Eco-system for Direct Attach Storage
  - SNIA NVM Programming Model TWIG
  - Memory mapping of the storage media
    - PMEM.IO, DAX changes in file system stack
- Next step is Remote Access
  - SNIA NVM PM Remote Access for High Availability



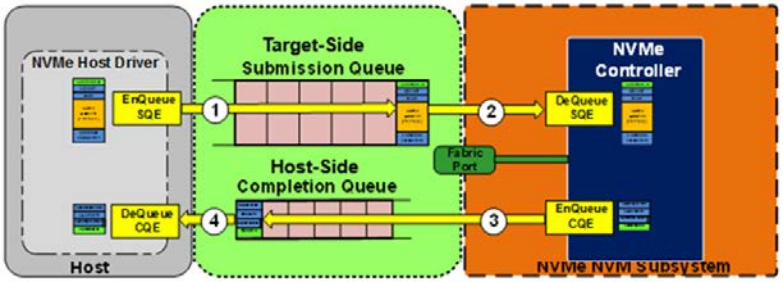
RDMA and Persistent Memory

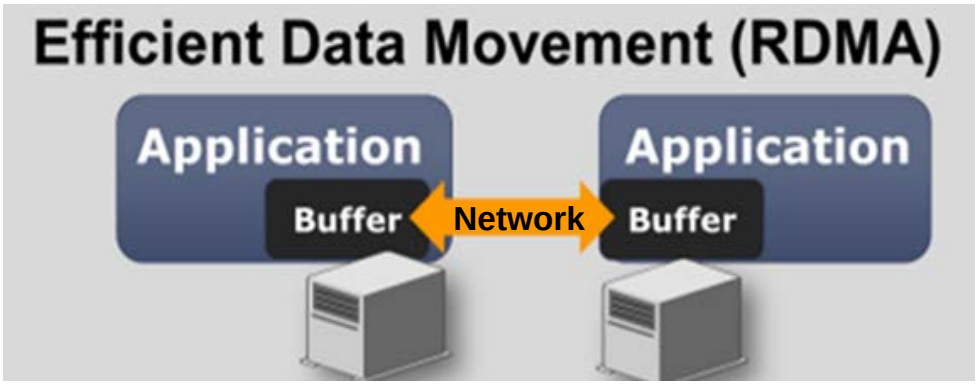


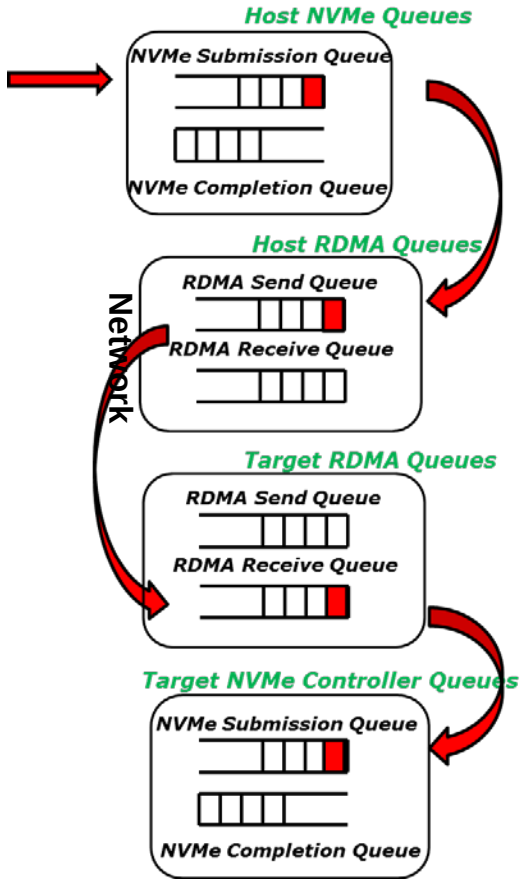
Protocol Deep Dive

- NVMe over Fabrics
- On going work for remote persistent memory access

NVMe and NVMeoF Fit Together Well

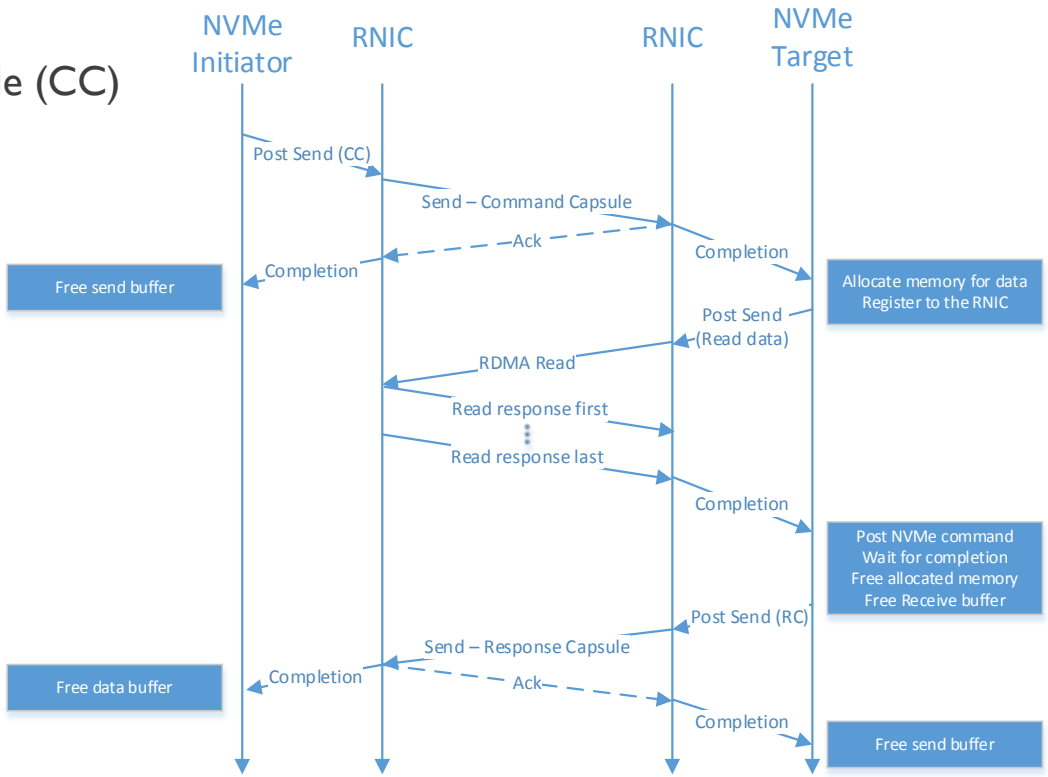






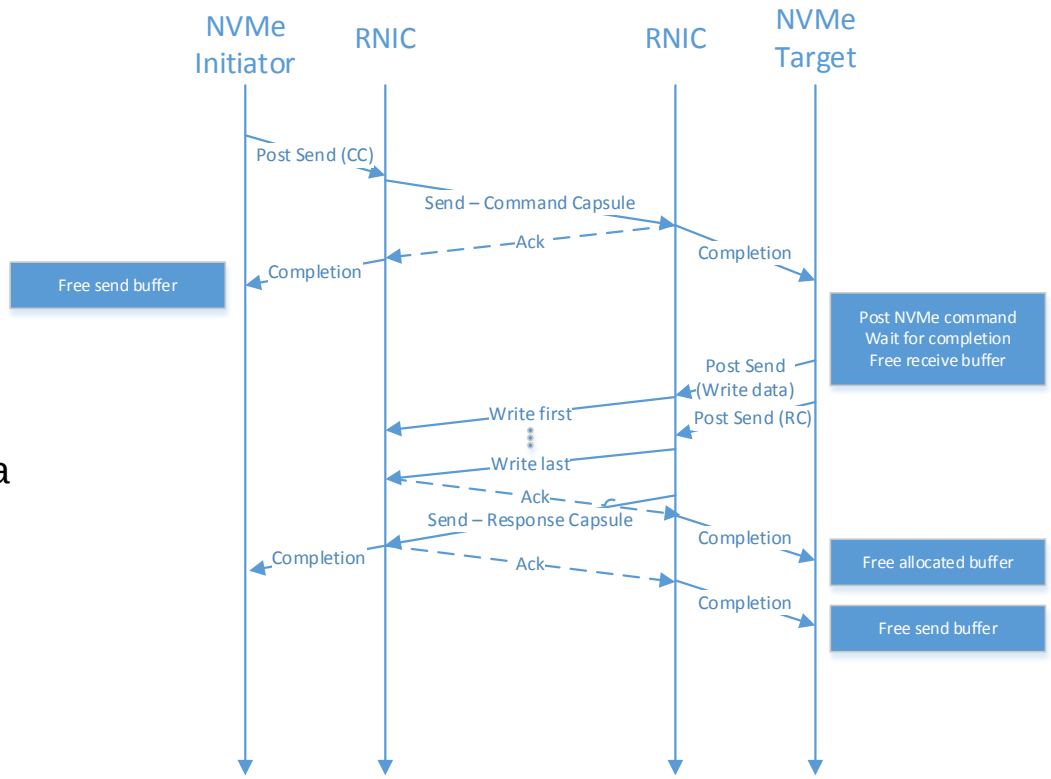
NVMe-OF IO WRITE

- Host
  - Post SEND carrying Command Capsule (CC)
- Subsystem
  - Upon RCV Completion
    - Allocate Memory for Data
    - Post RDMA READ to fetch data
  - Upon READ Completion
    - Post command to backing store
  - Upon SSD completion
    - Send NVMe-OF RC
    - Free memory
  - Upon SEND Completion
    - Free CC and completion resources



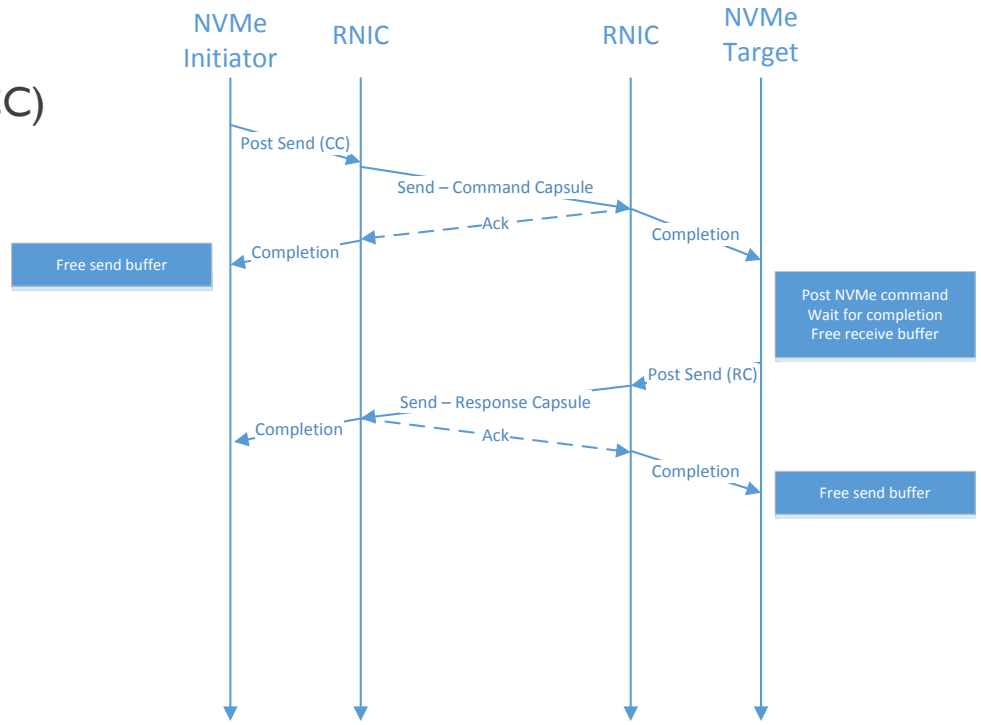
NVMe-OF IO READ

- Host
  - Post SEND carrying Command Capsule (CC)
- Subsystem
  - Upon RCV Completion
    - Allocate Memory for Data
    - Post command to backing store
  - Upon SSD completion
    - Post RDMA Write to write data back to host
    - Send NVMe RC
  - Upon SEND Completion
    - Free memory
    - Free CC and completion resources

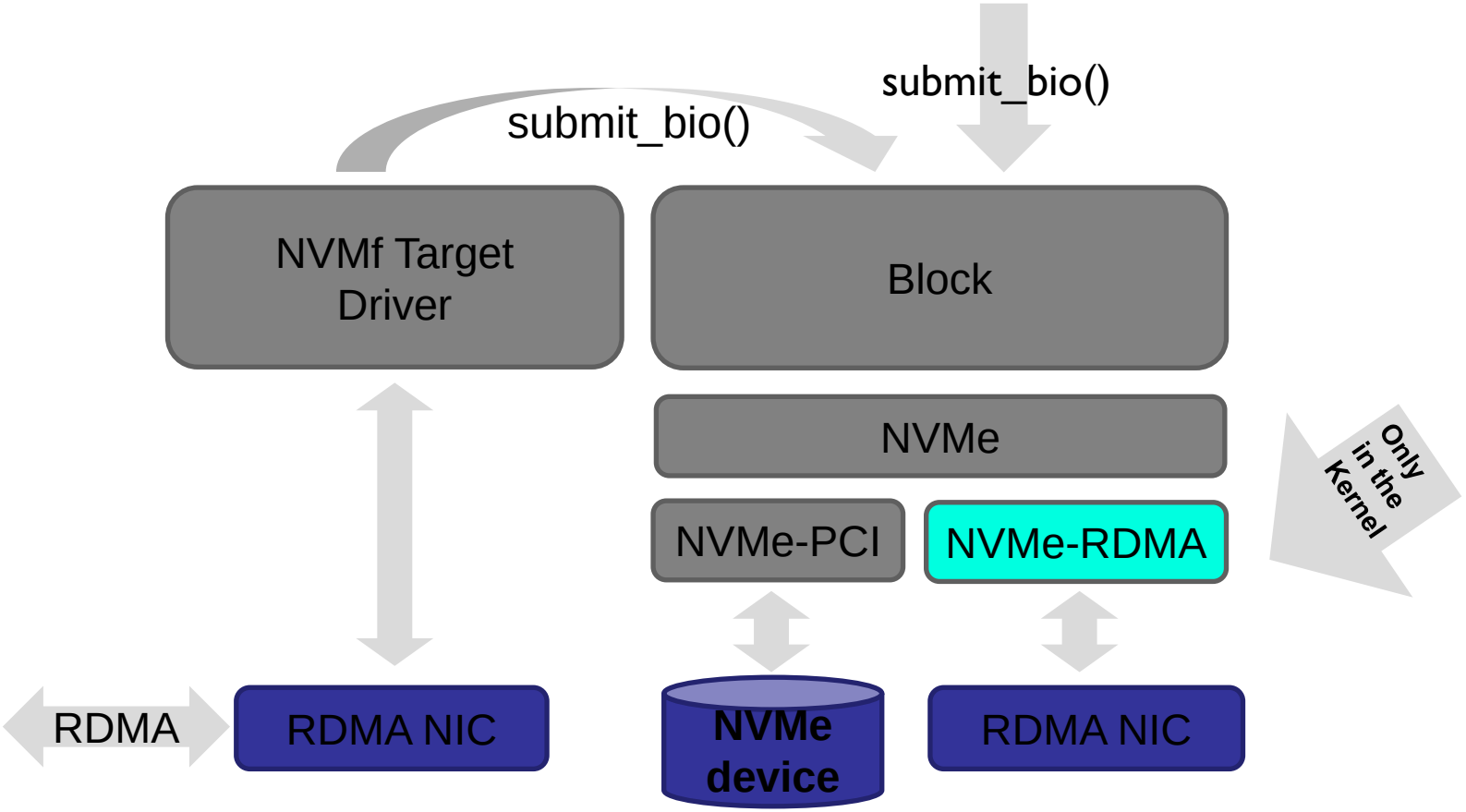


NVMe-OF IO WRITE IN-Capsule

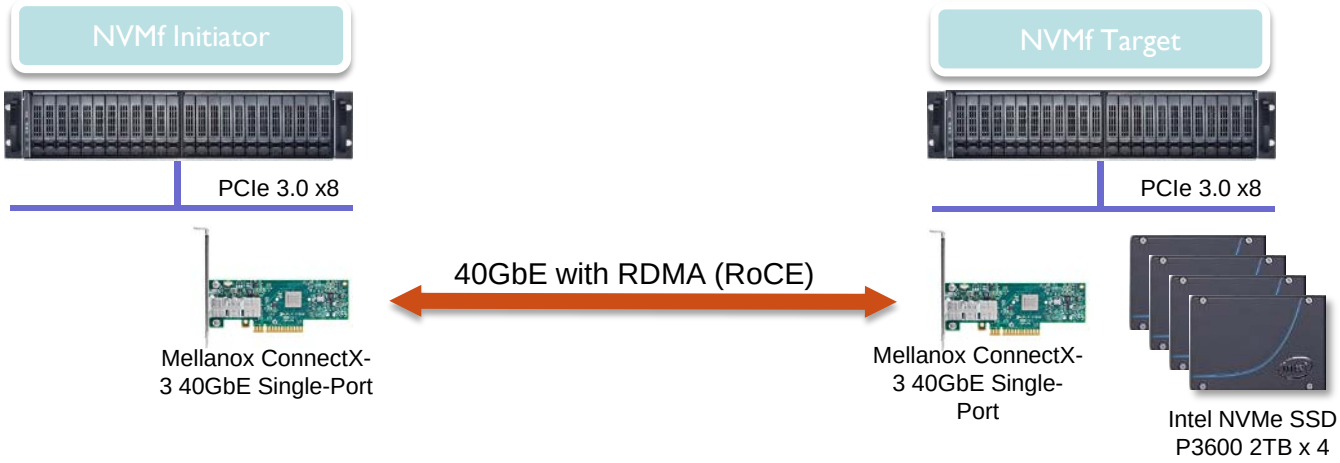
- Host
  - Post SEND carrying Command Capsule (CC)
- Subsystem
  - Upon RCV Completion
    - Allocate Memory for Data
  - Upon SSD completion
    - Send NVMe-OF RC
    - Free memory
  - Upon SEND Completion
    - Free CC and completion resources



Linux 4.8 NVMf Design



SPDK NVMe over Fabrics demo configuration



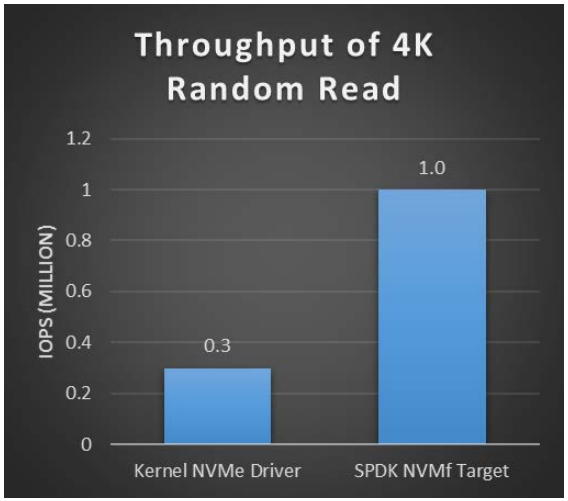
NVMf Initiator

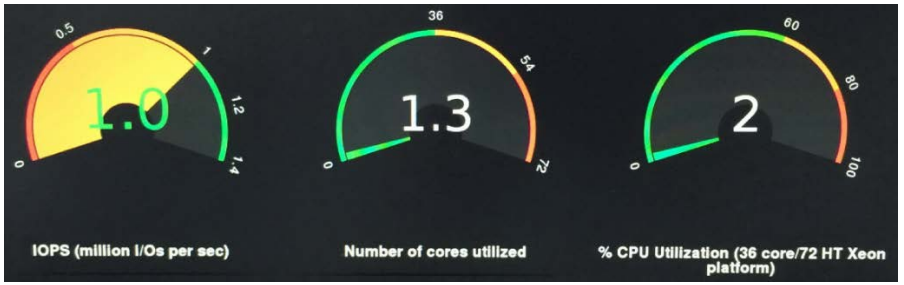
- CPU: Intel Xeon E5-2699 v3 2.3GHz, 18 cores, x2
- Memory: 8 x 8GB DDR4-2133 DIMM
- Network: Mellanox ConnectX-3 40GbE Single-Port
- OS: CentOS 7.2 Kernel 3.10
- Benchmark: FIO 2.2.9
- Intel also tested this with ConnectX-4 Lx using 2x25GbE

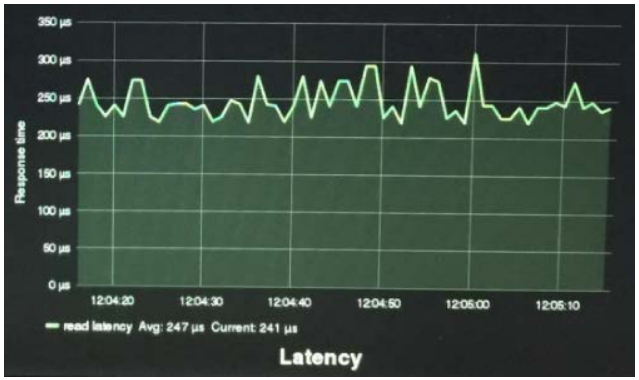
NVMf Target

- CPU: Intel Xeon E5-2699 v3 2.3GHz, 18 cores, x2
- Memory: 8 x 8GB DDR4-2133 DIMM
- Storage: Intel SSD DC P3600 Series 2.5" 2TB, x4
- Network: Mellanox ConnectX-3 40GbE Single-Port
- OS: CentOS 7.2 Kernel 3.10
- SPDK: http://spdk.io

SPDK NVMe over Fabrics demo performance





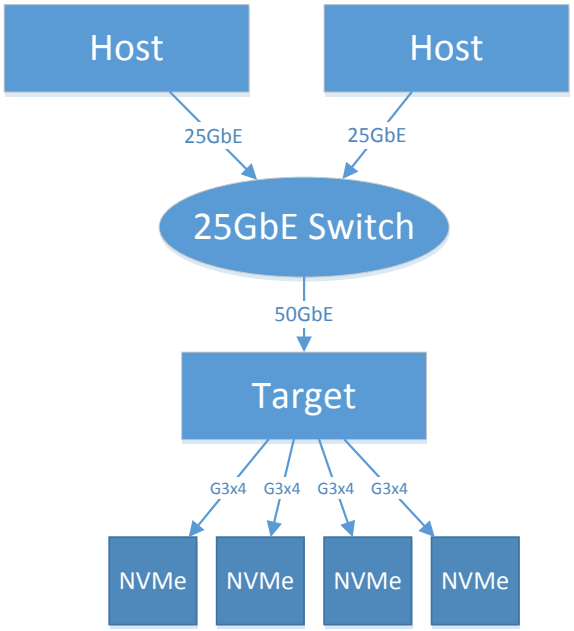


- Throughput of NVMf with SPDK can reach 1.0M IOP
- Only 1.3 CPU cores utilized, which is 2% CPU utilization

Mellanox RDMA fabric can greatly improve CPU efficiency and optimize application latency

Mellanox NVMe over Fabrics Performances Evaluation

- Topology
  - Two compute nodes
    - ConnectX4-LX 25Gbps port
  - One storage node
    - ConnectX4-LX 50Gbps port
    - 4 X Intel Nvme device (P3700/750 series)
  - Nodes connected through switch



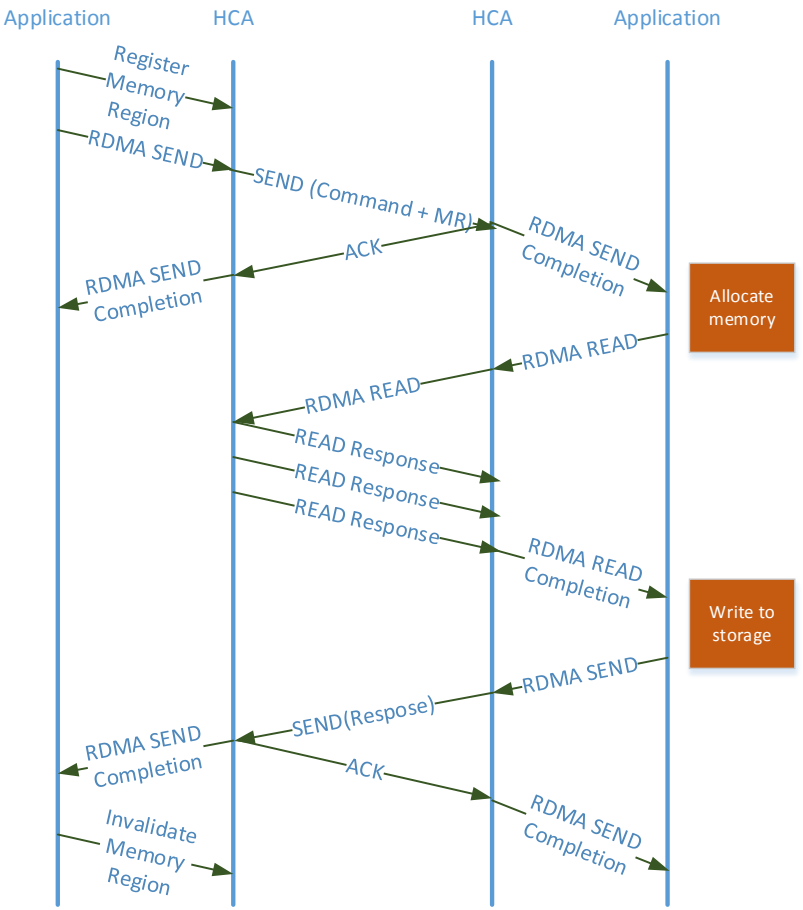
Added latency
~11us

	Bandwidth (Target side)	IOPS (Target side)	Num Online cores	Each core utilization
BS = 4k, 16 jobs, IO depth = 64	5.2GB/sec	1.3M	4	50%

Remote PMEM Access

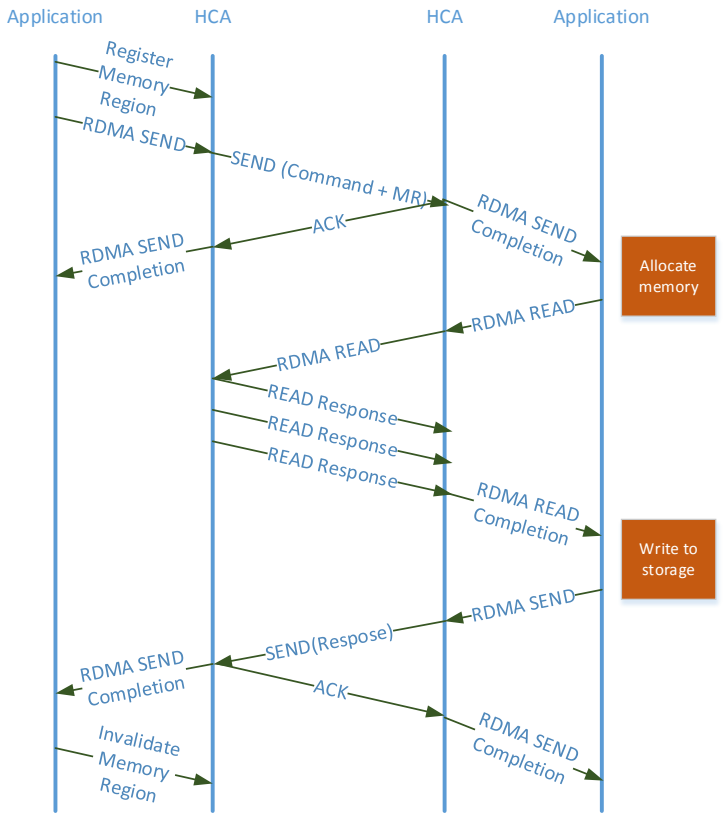
Background – RDMA Based Storage Protocols

- Exchange model highlights (pull model):
  - Control commands/completions are transferred with SEND
  - Data transfer is being initiated by the storage target application due to memory scalability
- The reason for the design of the protocols:
  - Latency of storage device is orders of magnitude higher than network latency
  - Storage is not byte addressable
  - Scalability of intermediate memory
- Examples for storage protocols
  - SMB-Direct
  - NFS/RDMA
  - iSER
  - SRP
  - NVMe/Fabrics



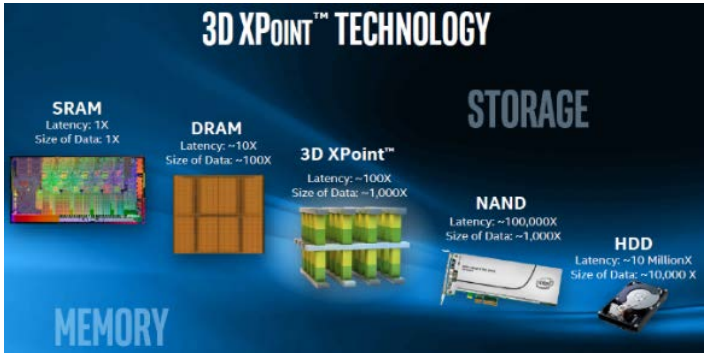
Latency Breakdown for Accessing Storage

- Traditional storage latency are >10usec
- Round trip time (RTT) and interrupt overheads are in the same order of magnitude at the best case
- Persistent Memory Non-Volatile memory technologies are reducing these to <1usec
- Spending time on RTT and interrupts becomes unacceptable



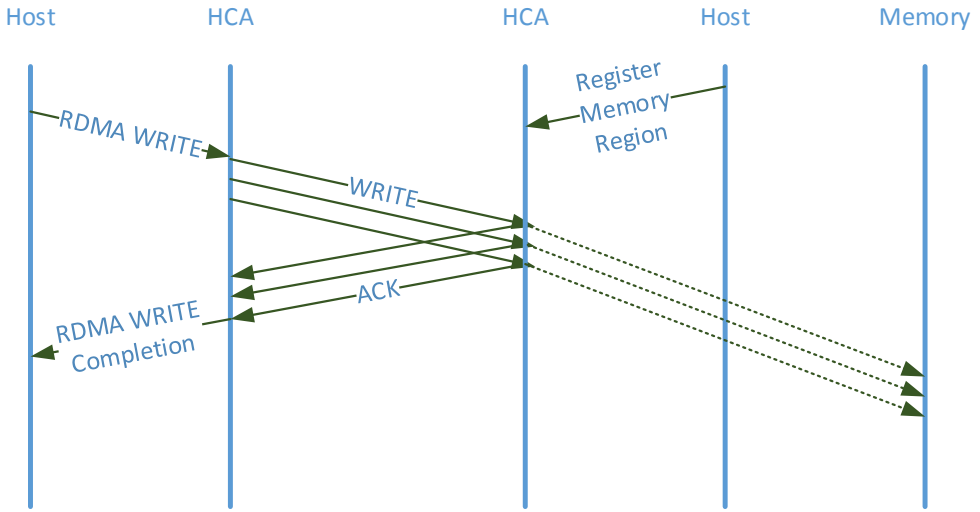
Attributes of Persistent Memory

- Byte addressable over the coherent / non-coherent BUSes
- Latency of access is compatible with the BUSes requirements
- Therefore:
  - These attributes enables the Persistent Memory to be exposed to RDMA access
  - Reliability extensions should be provided at the protocol level to verify persistency

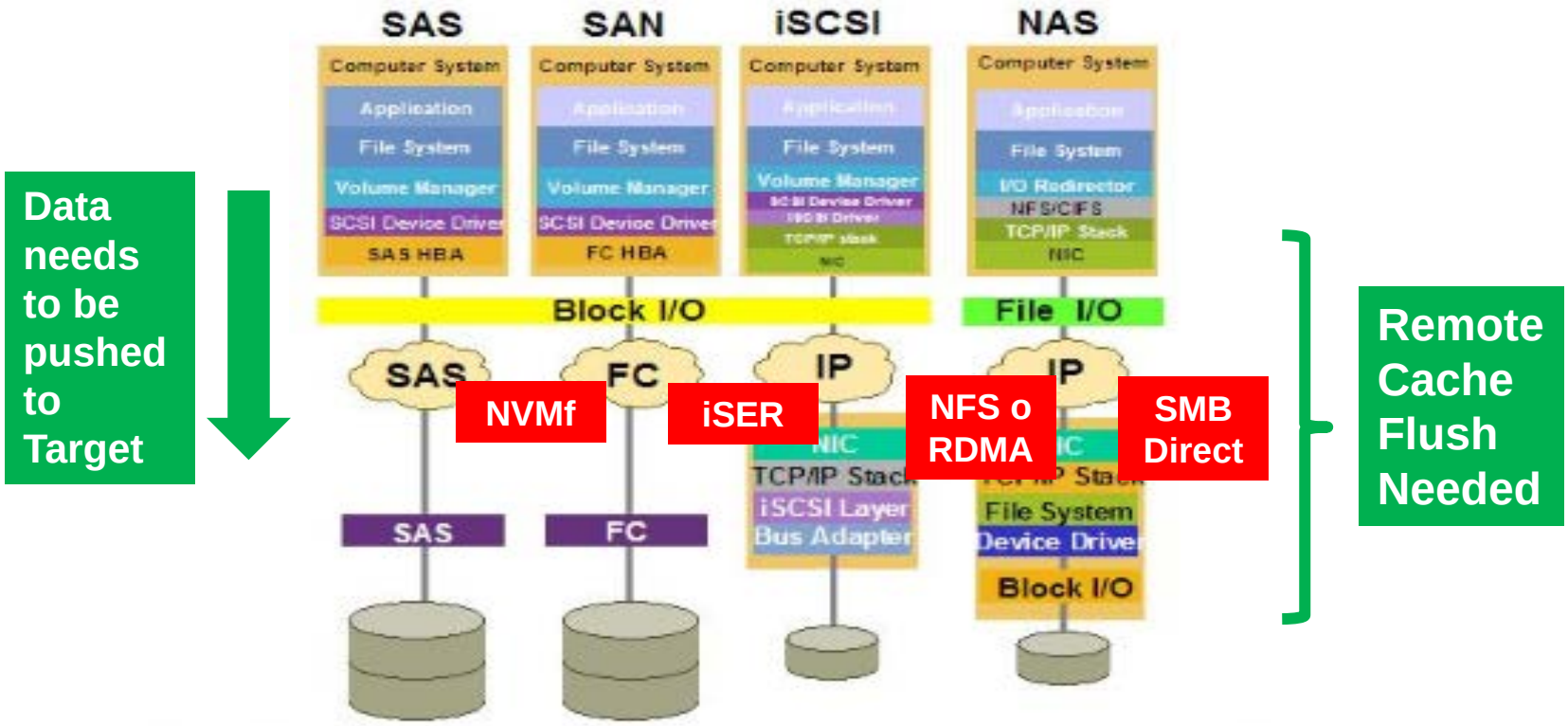


RDMA WRITE

- RDMA Acknowledge (and Completion)
  - Guarantee that Data has been successfully received and accepted for execution by the remote HCA
  - Doesn't guarantee data has reached remote host memory
- Further Guarantees Implemented by ULP



Remote PM Extensions Must be Implemented in Many Places



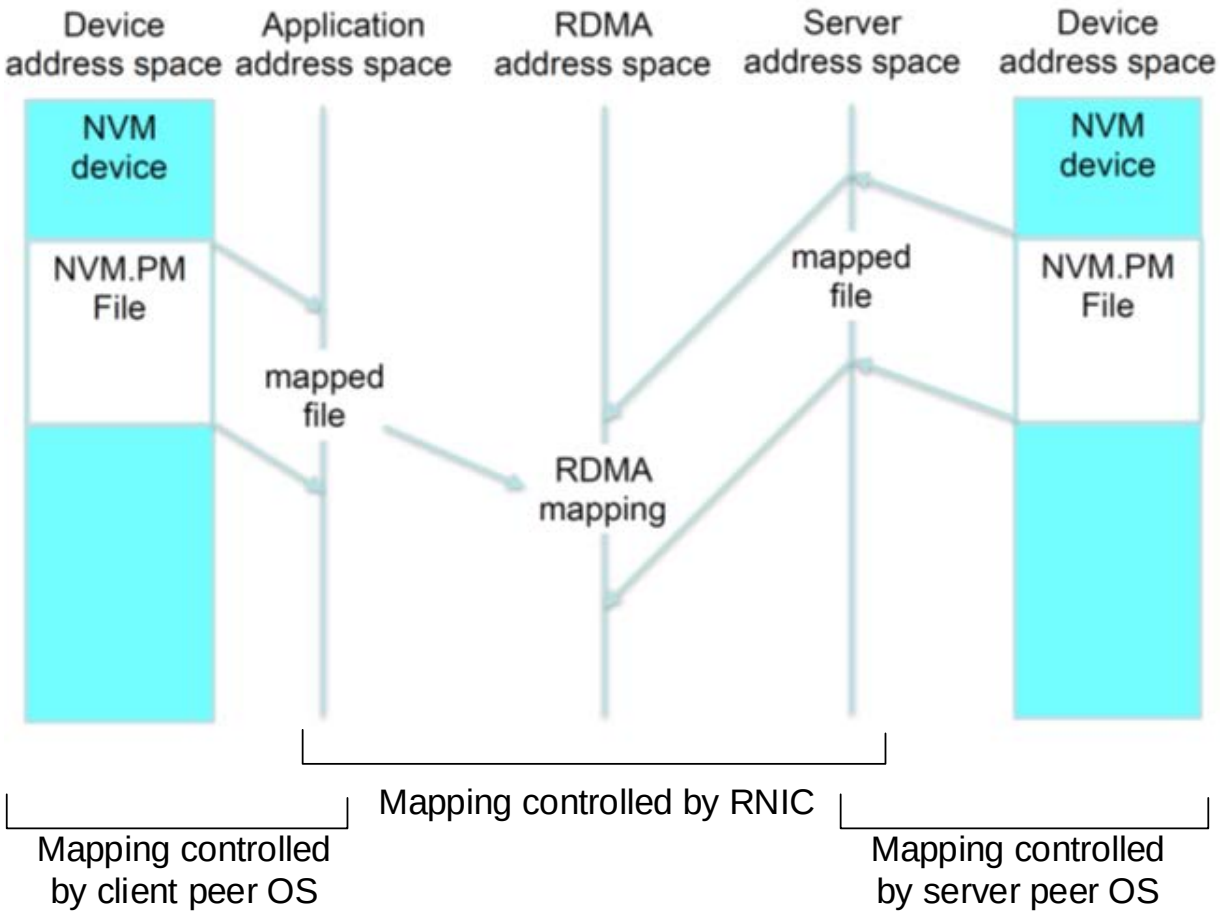
SNIA NVM Programming Model

- Accelerate the availability of software that enables NVM (Non-Volatile Memory) hardware.
  - Hardware includes SSD's and PM
  - Software spans applications and OS's
- Create the NVM Programming Model
  - Describes application visible behaviors
  - Allows API's to align with OS's
  - Exposes opportunities in networks and processors

NVM.PM.FILE Basic Semantics

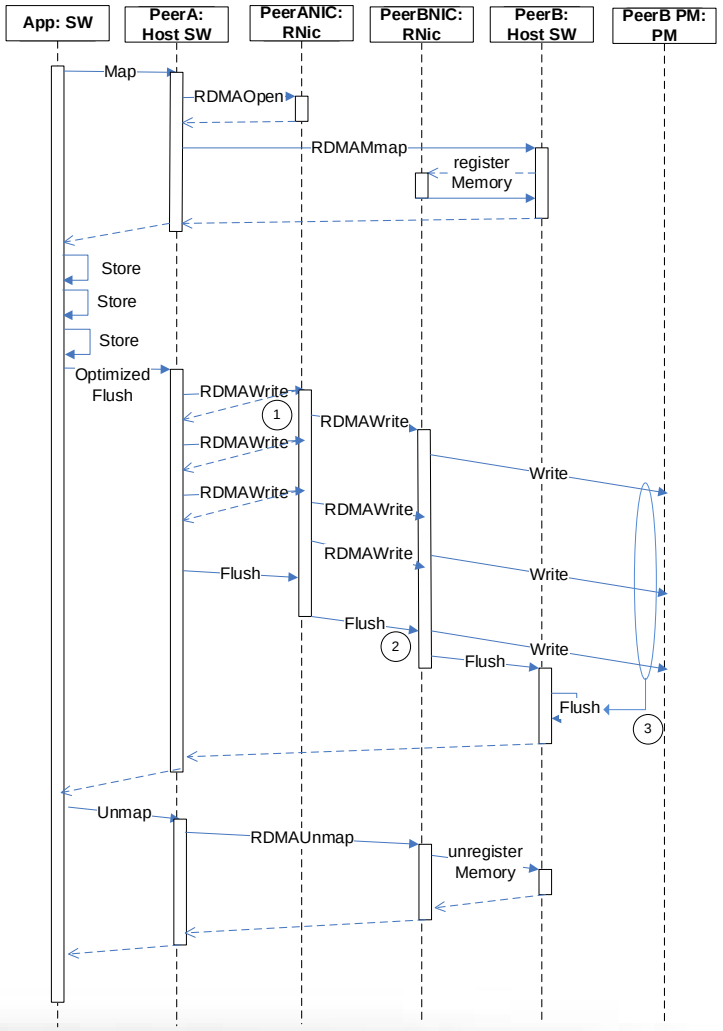
- MAP
  - Associates memory address with a file descriptor
  - Specific address may be requested
- SYNC
  - Commit the read/write transactions to persistency
  - Types
    - Optimize flush
    - Optimize flush and verify

Use Case: RDMA for PMEM High Availability



Use Case: RDMA for PMEM High Availability

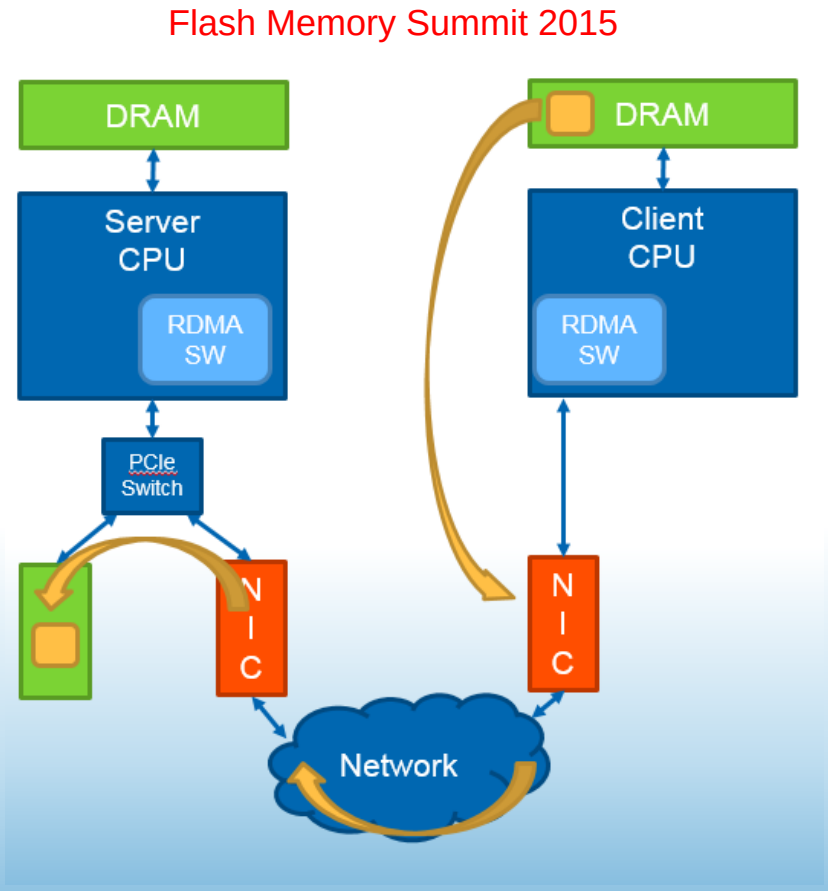
- MAP
  - Memory address for the file
  - Memory address + Registration of the replication
- SYNC
  - Write all the “dirty” pages to remote replication
  - FLUSH the writes to persistency
- UNMAP
  - Invalidate the registered pages for replication



Examples for POCs

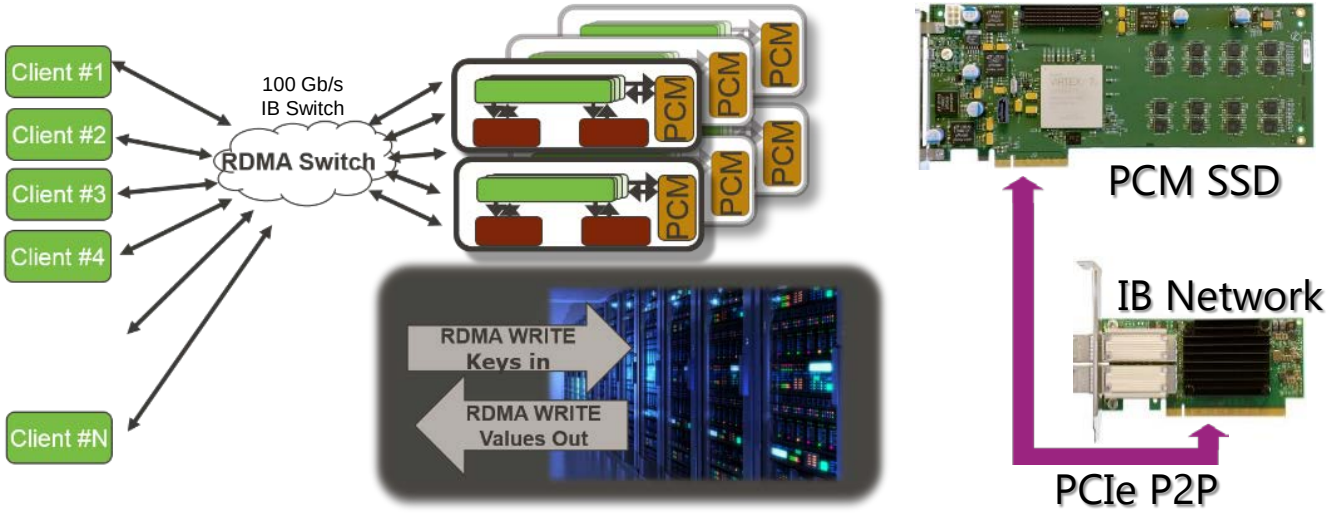
Peer-Direct NVRAM over RDMA Fabrics Proof of Concept

- Development platform to enable testing of remote memory transactions over RDMA fabrics to non-volatile storage
  - Mellanox RDMA HCA
  - PMCS NVRAM Card
  - PMCS PCIe Switch
- IO transactions bypass host CPU on server using Peer-Direct
  - Reduced server load and DRAM bandwidth
- 4us latency for 4KB IO from client to server non-volatile memory over RDMA connection
  - Network latency no longer a don't-care for remote block IO transactions



HGST FMS Demo

Key-Value Store fetch from Non-Volatile Storage in ~ 2 µs,
comparable to cutting-edge DRAM systems



Open Source RDMA Software

- http://linux-iscsi.org/wiki/ISCSI_Extensions_for_RDMA
- http://docs.ceph.com/docs/master/releases/
- https://www.samba.org/ (Windows SMB)
- https://www.kernel.org/doc/Documentation/filesystems/nfs/nfs-rdma.txt
- git://git.infradead.org/nvme-fabrics.git
- https://community.mellanox.com/docs/DOC-2283

Conclusions

- By adding RoCE RDMA network technology storage performance, whether Block, File, or Object, can be enhanced dramatically.
- By adding RDMA support you can future proof your network for next generation NVM storage technologies
- The software that powers RoCE RDMA technology is available through open source

Thanks!

robd@mellanox.com
idanb@Mellanox.com