Western Digital

Nanosecond Scale Storage: Ultrafast SSDs and Persistent Memory Applications of Emerging NVMs

Zvonimir Z. Bandic

Next-Gen. Platform Technologies

Western Digital

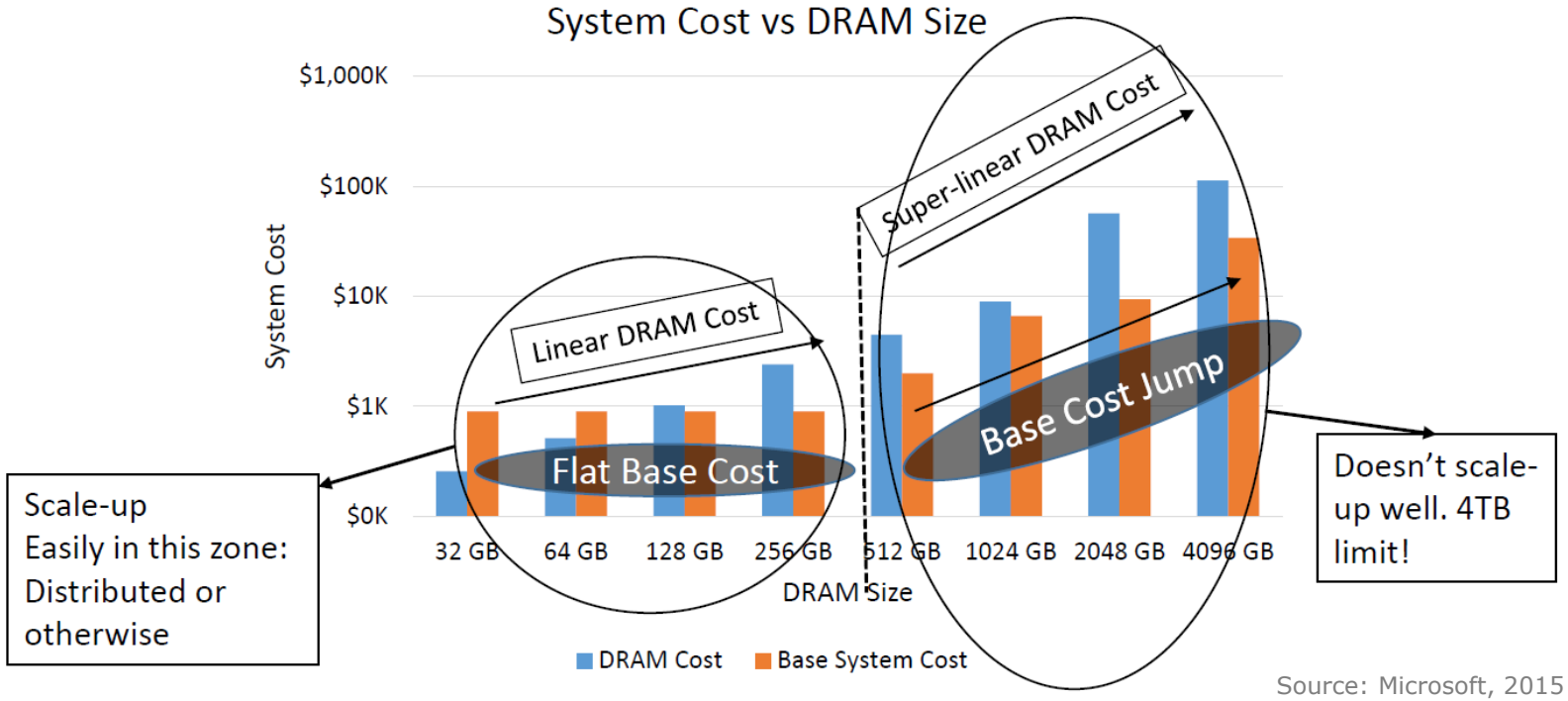
Acknowledgments

- Qingbo Wang
- Filip Blagojevic
- Md Kamruzzaman
- Martin Lueker-Boden
- Dejan Vucinic
- Damien LeMoal
- Cyril Guyot
- Steffen Hellmold

Agenda

- **What are emerging NVM?**
- **Programming models**
  - CPU memory
  - Fast block storage
- **Prototyping and performance**
- **Large scale deployment**
  - RDMA networking
  - NVMe over fabrics
- **Conclusions**

DRAM System Scaling Challenges



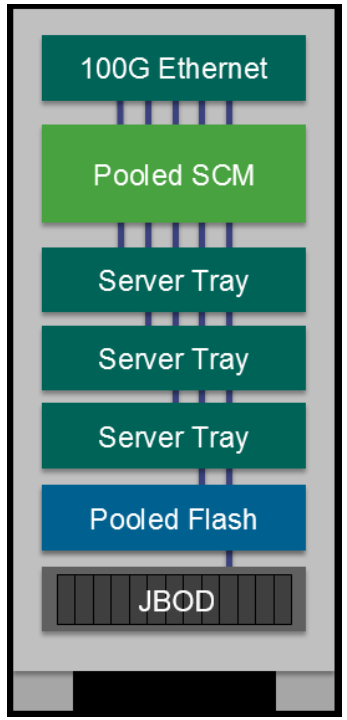
Source: Microsoft, 2015

- DRAM is expensive, and does not scale well beyond 4TB per node
  - And at that point system cost and DRAM cost are prohibitively high
- Big Data analytics, in-memory DB, HPC all require a lot of memory, not just in a single node, but rack/data center level

Data Centric Compute Architectures

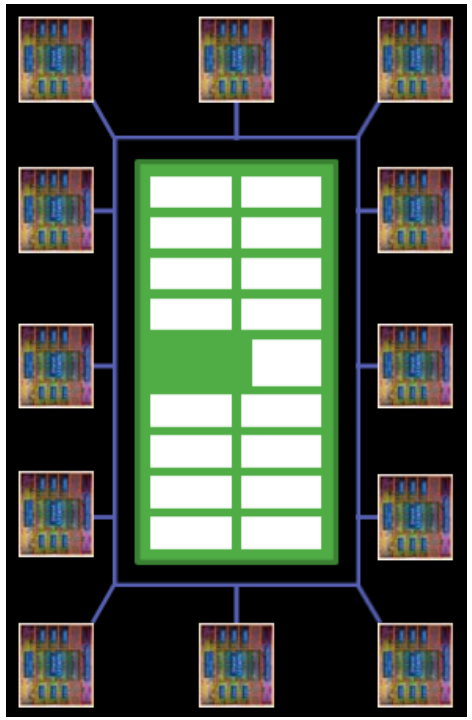
Data Center

Rack Scale Architecture

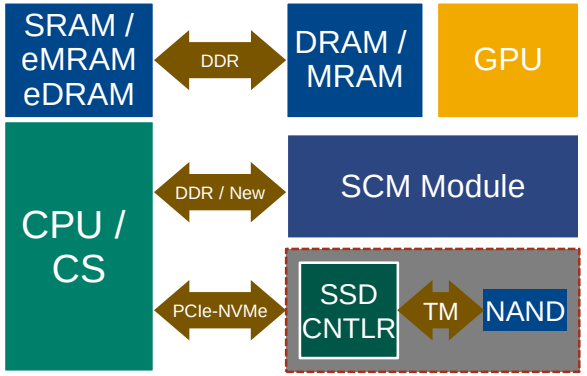


Data Center

Cheap CPUs Around PB of SCM



Client Compute



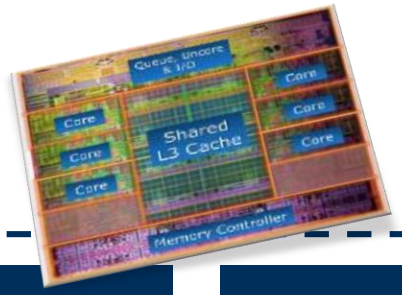
SCM complements DRAM for compute intensive clients

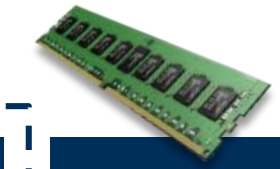
Mobile

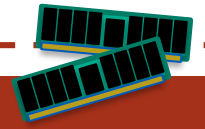
Large memory requirements for Virtual Reality

TIME

Extending Storage to 250ns Latency





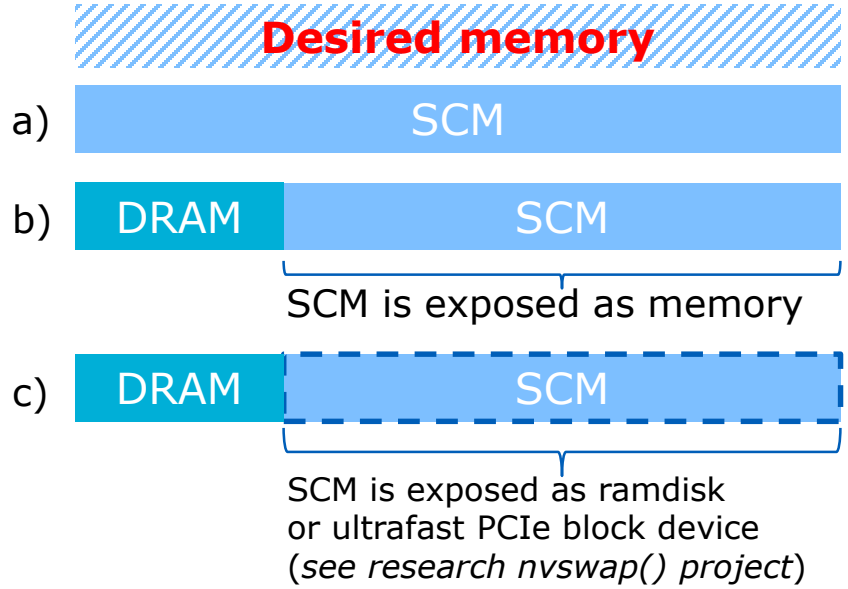


	Core Register	Core L1 Cache	Core L2 Cache	Shared L3 Cache	DRAM	Storage Class Memory	Flash	HDD
Size	64KB	256KB	2-4MB	16-128GB	128GB-1TB	512GB-4TB	4-16TB
Speed	1ns	3-10ns	10-20ns	50-100ns	250-5,000ns	100,000ns-2,000,000ns	5-10,000,000ns
Cost				100x	20-25x	5x	1x

Source: Western Digital estimates

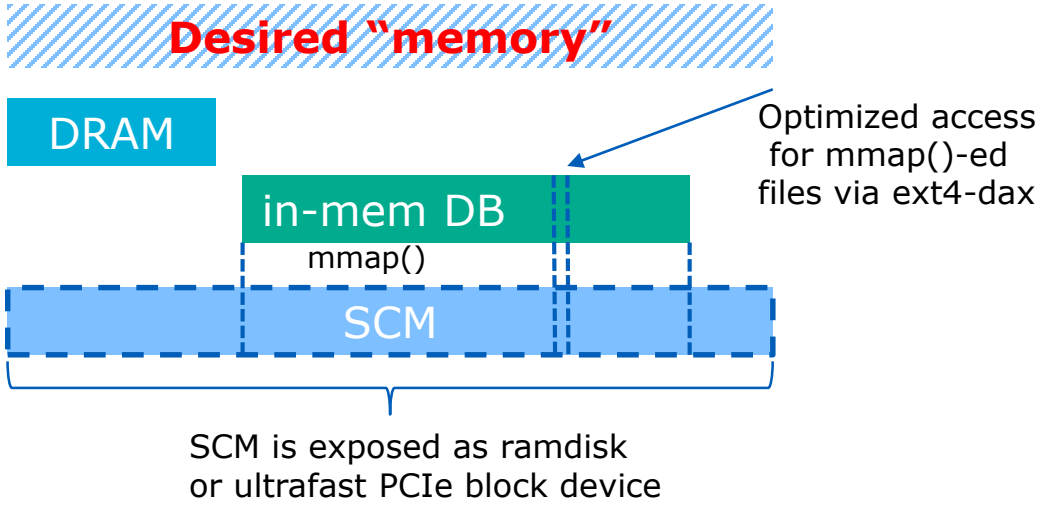
Memory Architecture Models

SCM as memory



- Type b) can be implemented via customized memory controller (Intel) or new programming model
  - For example, new data structure types
  - Requires significant rewriting of OS/applications
- Type c) requires OS improvements – such as rewriting swap() or coming up with new memory architectures, but less impact on applications

SCM as storage



- Best understood model, with direct application for large in-memory DB (Oracle, IBM) and web-analytics on "Big Data"
- OS changes are needed
  - E.g. already ready in Linux and Windows

Where Should We Attach Non-volatile Memory?

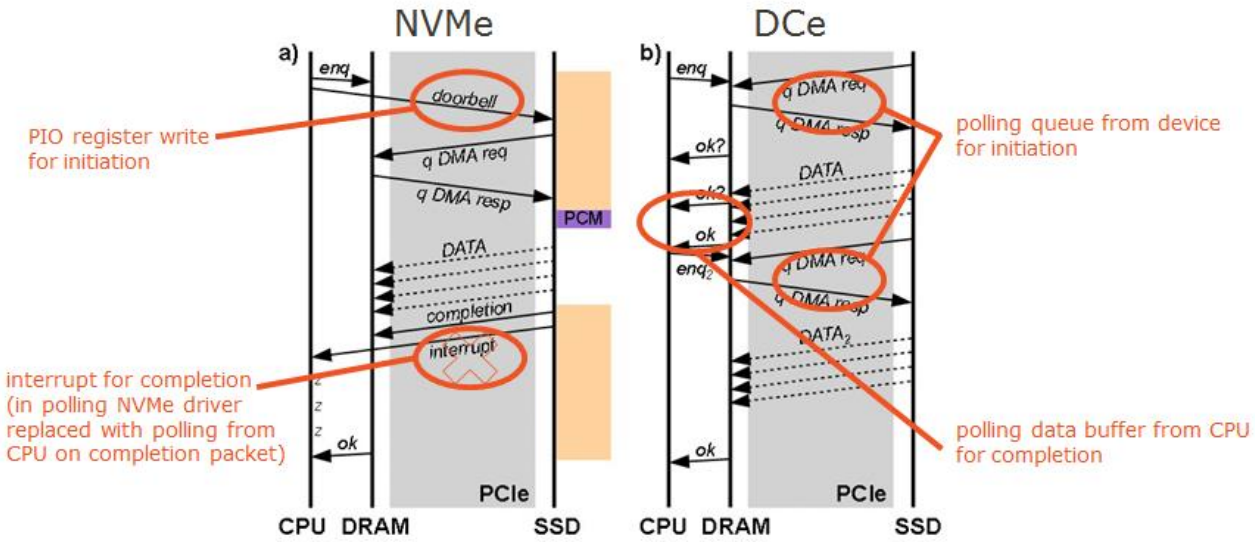
	CPU bus - parallel	CPU bus - serial	Serial peripheral bus
Physical interface	DIMM	DIMM/other	PCIe
Logical interface	Non-standard DDR4, NVDIMM-P	CCIX, OpenCAPI 3.1, Rapid-IO, gen-Z	NVMe, DC express*
Pros	- Low latency - High bandwidth - power proportional - coherent through memory controller	- High bandwidth - Significant pin reduction - Higher memory bandwidth to CPU - Coherent through memory controller, or in some cases can even present lowest point of coherence	- Standardized - CPU/platform independent - Latency low enough for storage - Easy RDMA integration - Hot pluggable
Cons	- CPU memory controller has to implement specific logical interface - Not suited for stochastic latency behavior - Not hot pluggable - BIOS needs change	- CPU memory controller has to support - May have higher power consumption	-Higher latency (~1us)

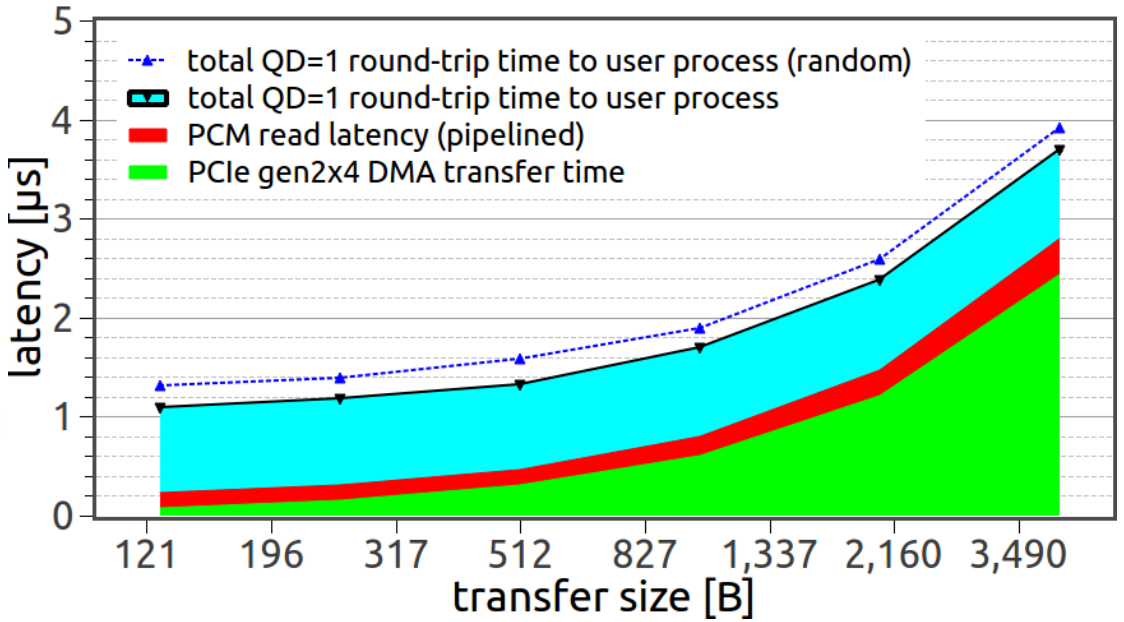
PCIe Attached Non-volatile Memory Block Device

Shown at FMS 2014

Western Digital Innovation

- NVMe
  - Polling and polling driver work
- DC Express
  - New, leaner PCIe storage protocol
  - Minimizes number of packets per command

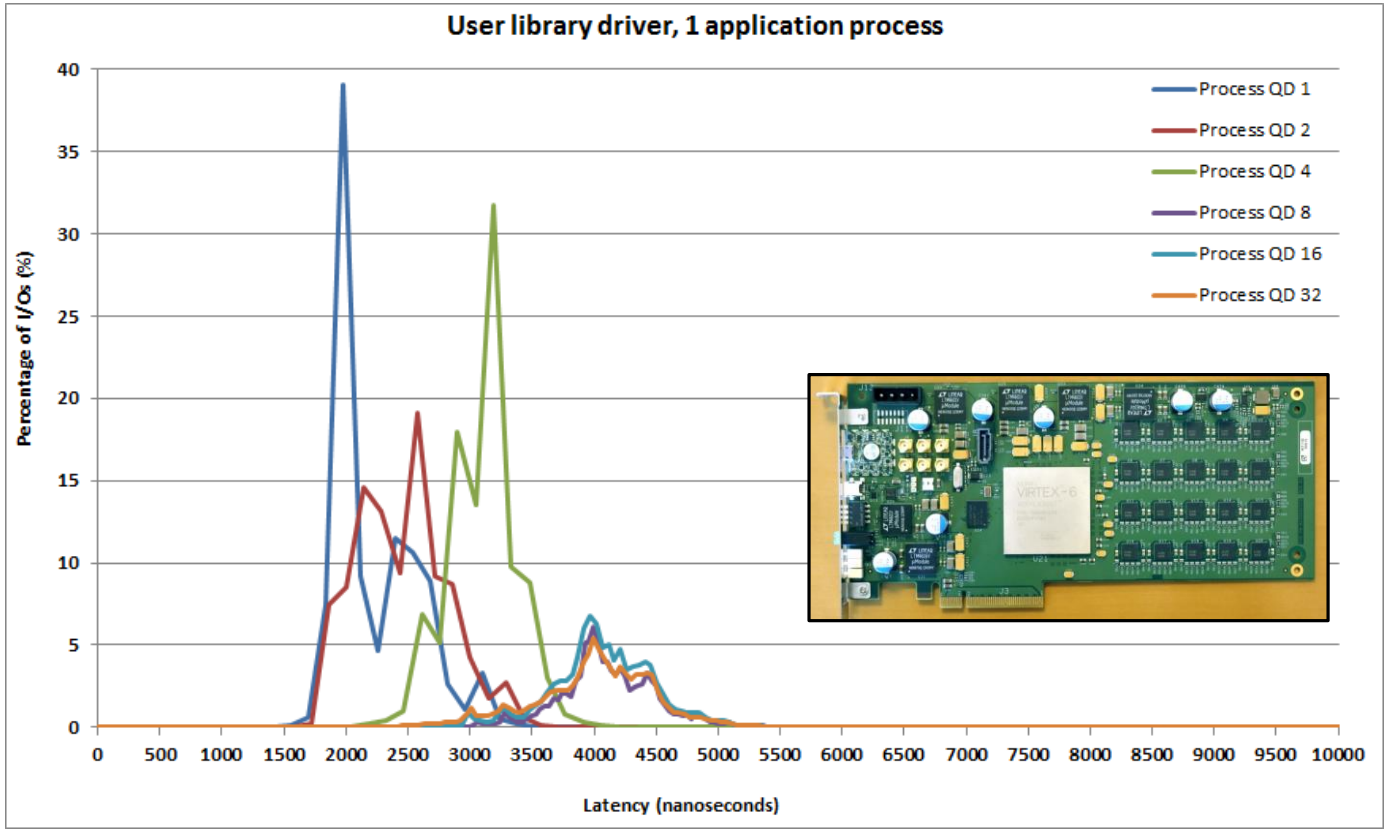




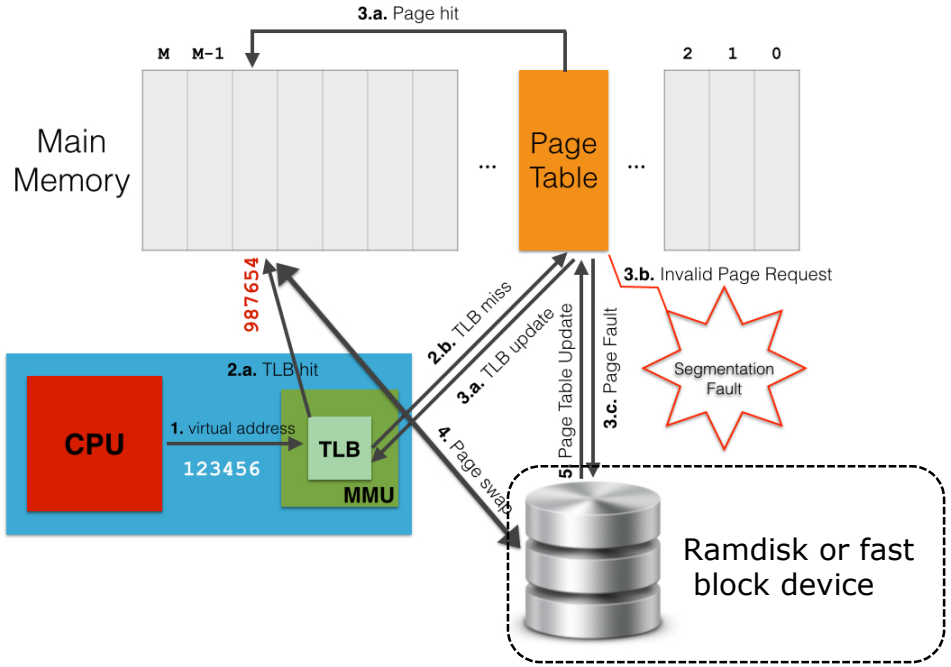
DC Express Prototype Device Performance

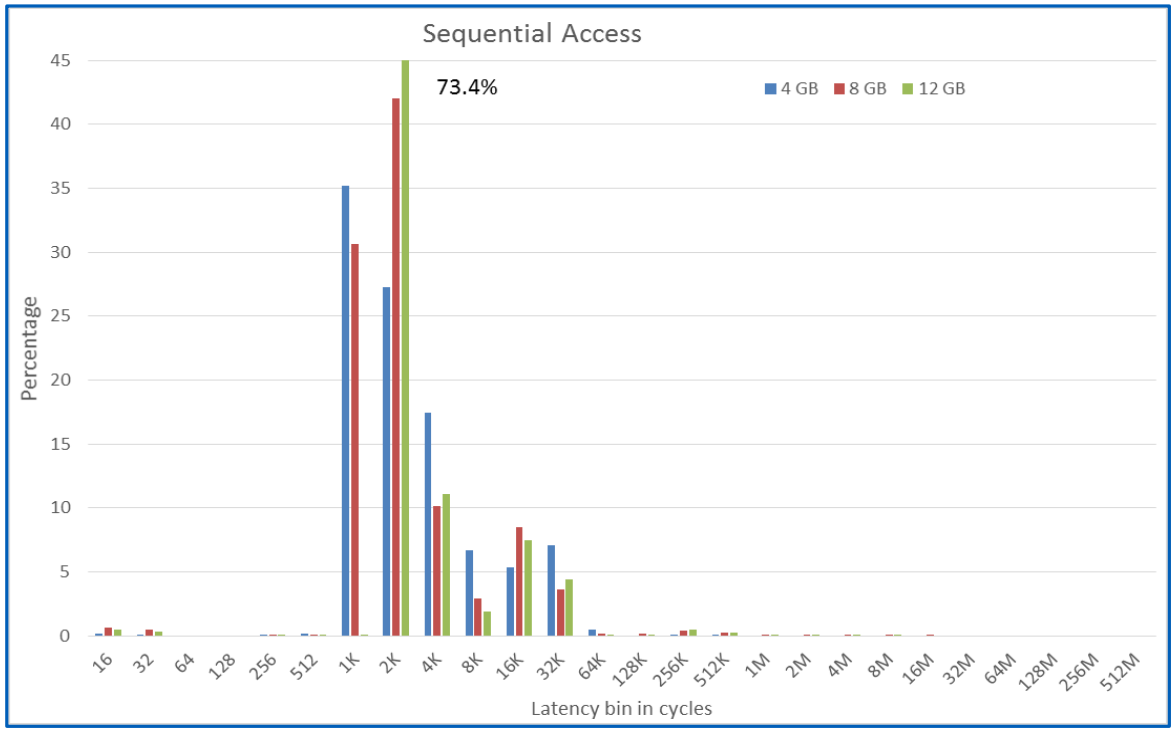
Demonstrated new, low-latency interface technology – DC express in 2014

- FPGA prototype devices demonstrated on Flash Memory Summit in 2014
- Proprietary low latency technology from HGST
  - DC express: No doorbells, no completions
- User library proprietary DC express driver
  - QD=1 performance is 1.8us for 512B (vs. NVMe of ~4.3us)
  - At high QDs, 99.9% of IOs complete within 5.5 us



DRAM+SCM as Memory – Using Swap()

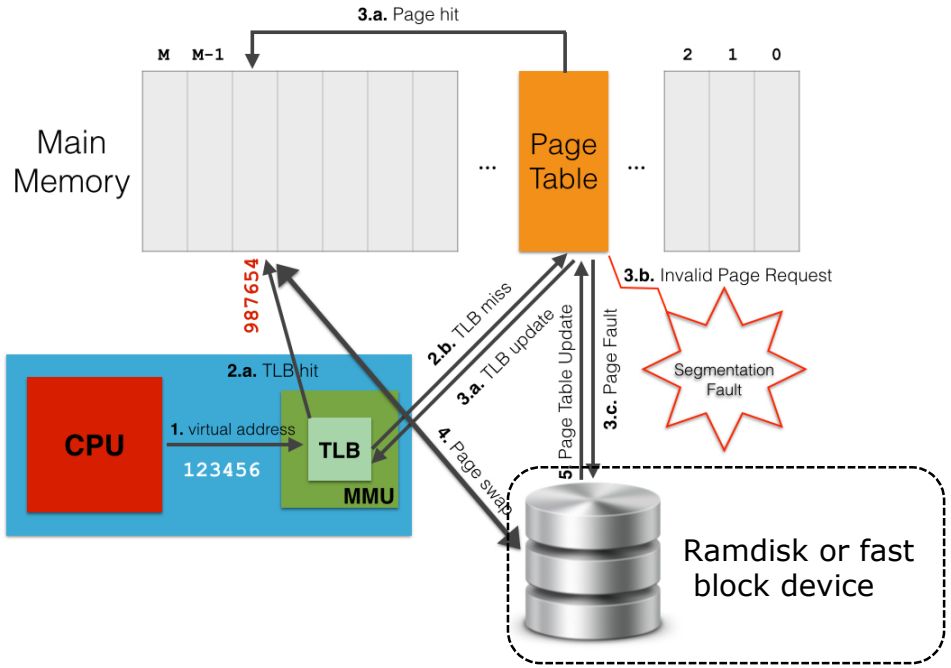


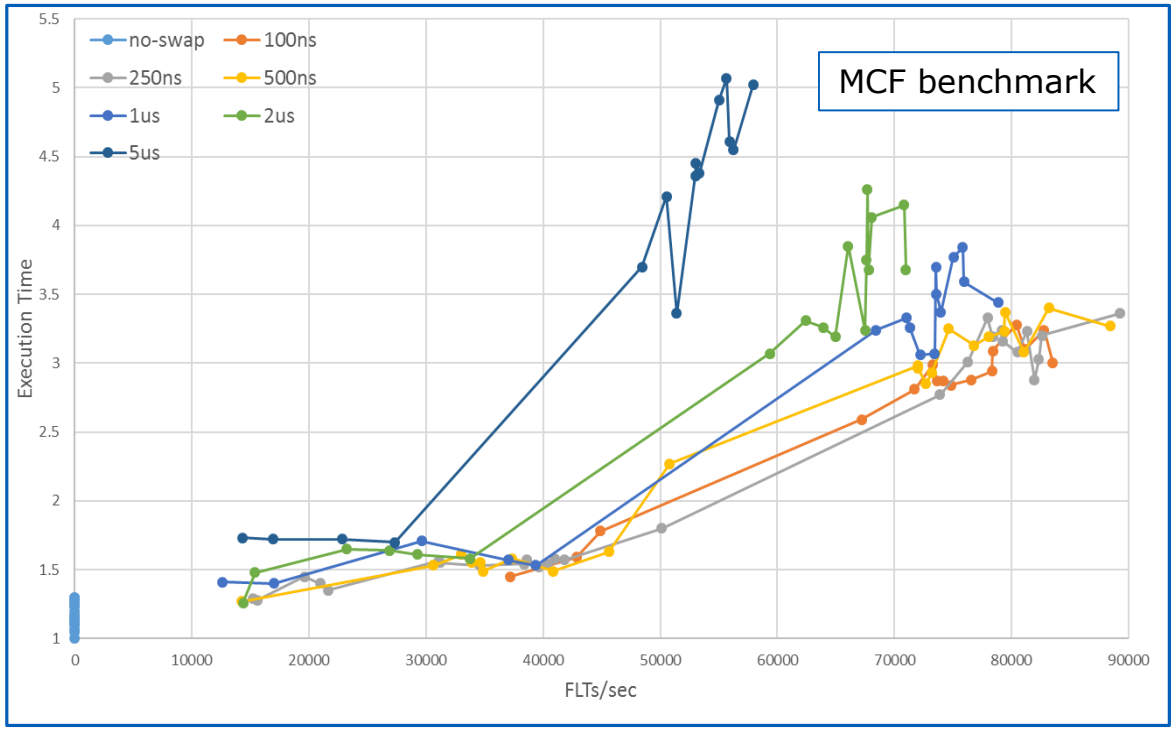


We have studied in latency observed by application

- In most cases between 1K-32K cycles (300ns-10us) assuming DRAM-like swap() performance
- We have also observed very long tail behavior – up to 16M cycles (corresponding to 4 ms), which is a consequence of reactive behavior of Linux swap() – and can be improved

DRAM+SCM as Memory – Using Swap()



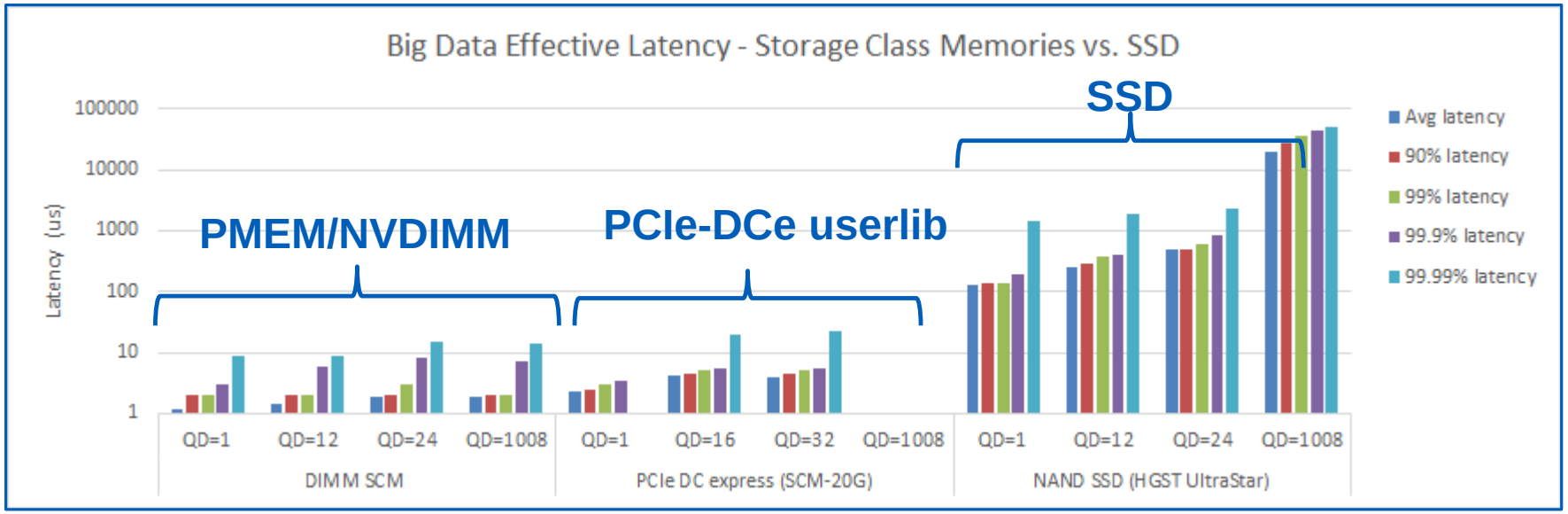


We have also studied what happens if ramdisk device has additional slowdown due to the technology itself – in the range between 0 and 5 us

- Clearly see increased execution time as latency increases
- Also, faster swap devices can process more page faults (as expected)
- Slowdown difference between 100ns and 500ns is not dramatic – at least for MCF benchmark

Using SCM as Storage – Comparison Between PMEM and Block*

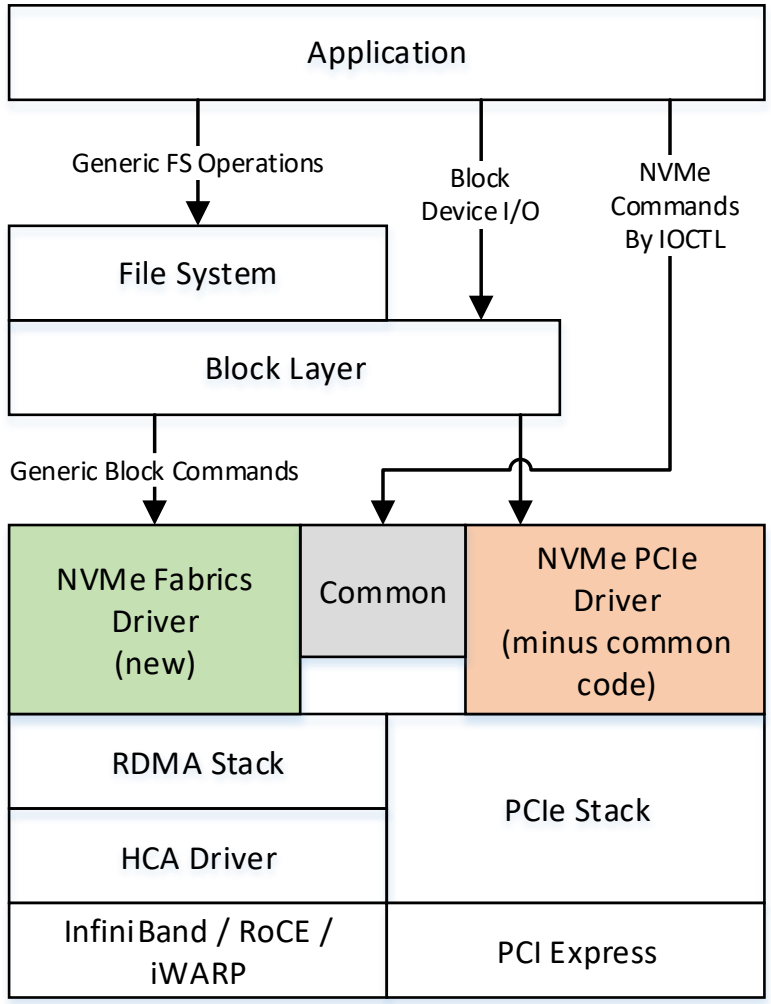
Includes file system overhead



- We compare latencies of file system level access to DRAM vs. DC express PCIe prototype card vs. standard SSD
- Tremendous potential of SCM technology is getting DRAM-like performance at storage level persistence and even APIs/access models
- PCIe block device is obviously slower, however due to file system overhead and QoS impact of operating system, the difference is not dramatic
  - PCIe device may be very interesting in the early phase of the market, prior to standardization

NVMf Host Software Architecture

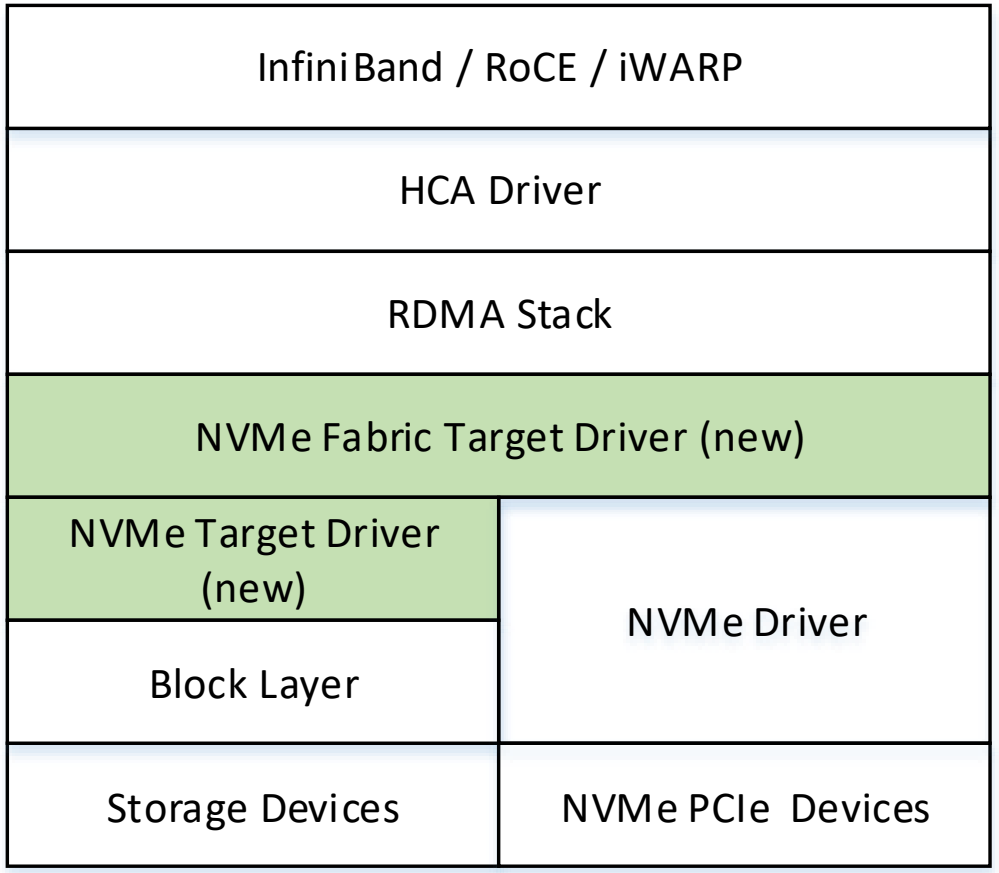
- Common code was extracted from NVMe PCIe driver
- NVMe Fabrics driver is new, incorporated into NVMe driver
- Other driver, stack, and FS modules are unmodified
- RoCE, iWARP, Infiniband – all supported



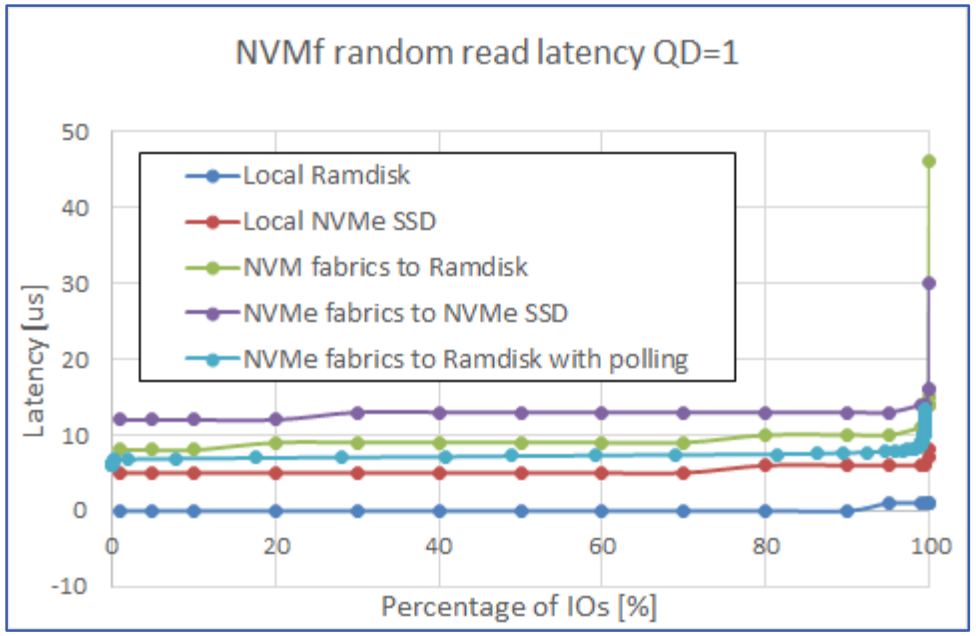
NVMe over Fabrics Controller Architecture

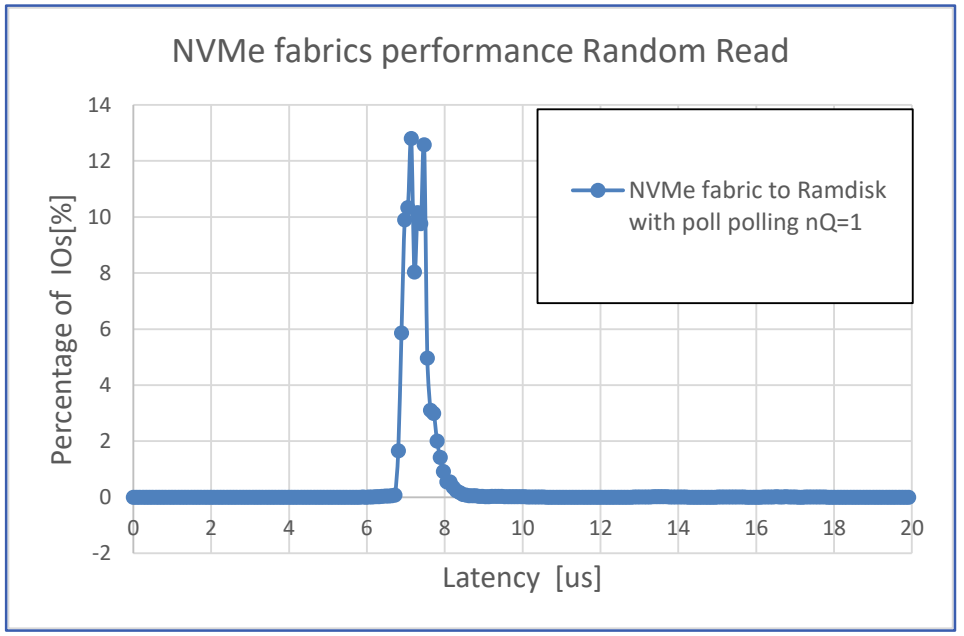
- Target devices include
  - RAM disk
  - NVMe device
  - Other NVM SATA/SAS devices

RoCE, iWARP, Infiniband – all supported



Performance Measurements (with polling)

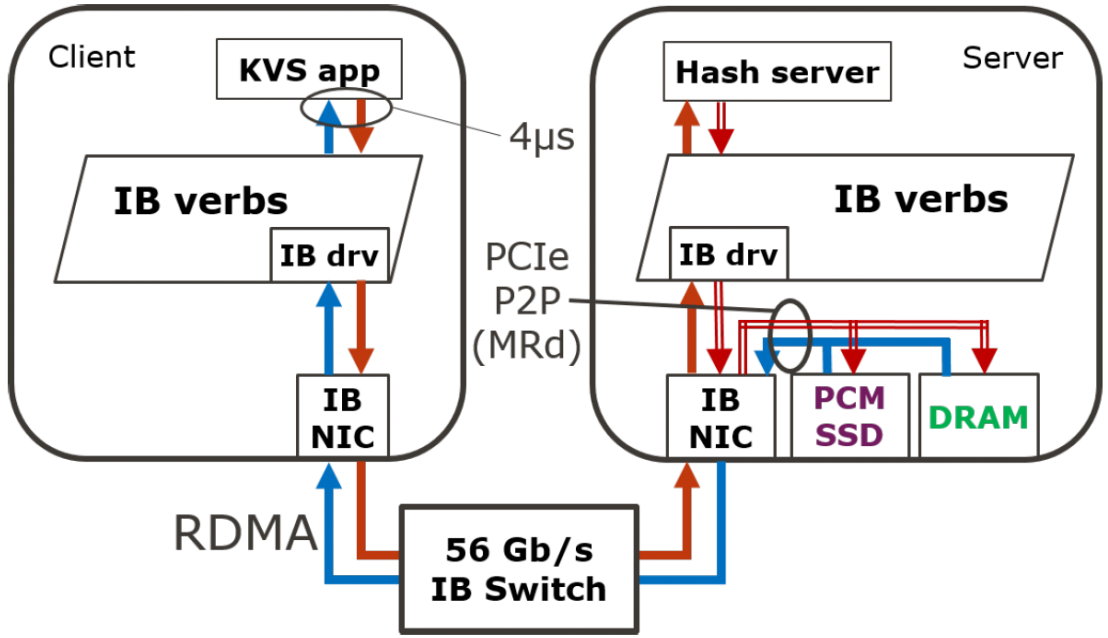


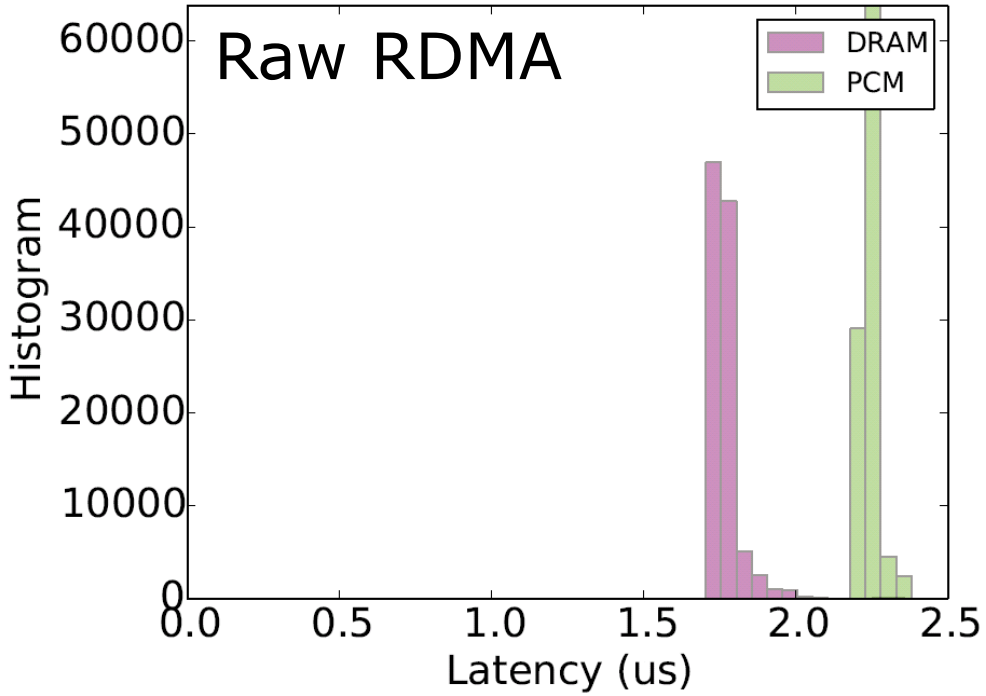


- Over Infiniband
- Added polling on the host side
  - On the controller side the Ramdisk driver always executes synchronously
- Latency (end-to-end) is 8 us:
  - Network latency contribution is <7 us

Remote eNVM Has Performance Similar to Remote DRAM

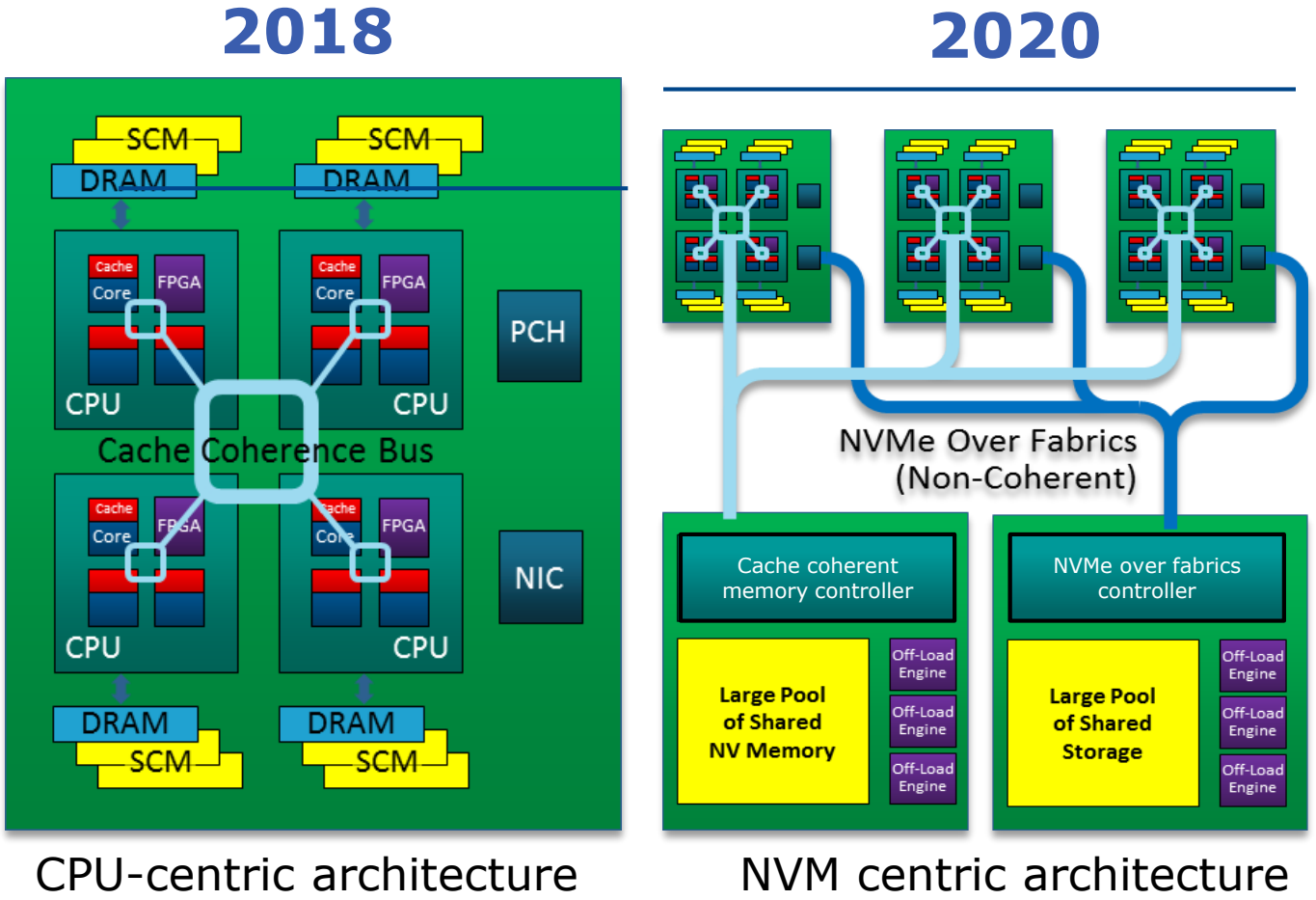
Raw RDMA access to remote PCM via PCIe peer2peer is 26% slower than to DRAM





Memory, Storage Fabric Standardization

- Transition from CPU to non-volatile memory centric architectures
- Industry standards efforts ongoing
  - NVDIMM-P / NVDIMM-N
  - Gen-Z
  - CCIX
  - OpenCAPI
  - Rapid-IO



CPU-centric architecture

NVM centric architecture

Open industry standards key to broad-based nanosecond-class storage adoption and capital investments

Conclusions

- Emerging NVMs programming models
  - CPU attached memory
  - Ultrafast block device
- Both models have their pros and cons, primarily related to interface standardization
- Putting persistent memory on the network
  - Need of fast fabrics – i.e. RDMA Ethernet
  - Protocols for block storage: NVMe over fabrics
- Network latency needs to be similar or better than the latency of underlying persistent memory resource
  - Network is a new bottleneck
  - Memory fabrics will be required: Gen-Z, OpenCAPI etc.
- More standardization is required
  - Simplification of memory interface options
  - Standardization of memory fabrics


Western Digital®

Western Digital

A Storage Solutions Leader

- In a strong strategic position to lead global evolution of broad-based and changing storage industry
- Broad storage portfolio, including HDDs, SSDs, embedded and removable flash memory, and storage-related systems
- 13,500+ active patents worldwide
- Vertically integrated business model to maximize operational efficiency
- Consistent profitable performance, strong free cash flow

Broadest Portfolio of Products & Solutions

Client Solutions

Client Devices

Data Center Devices & Solutions

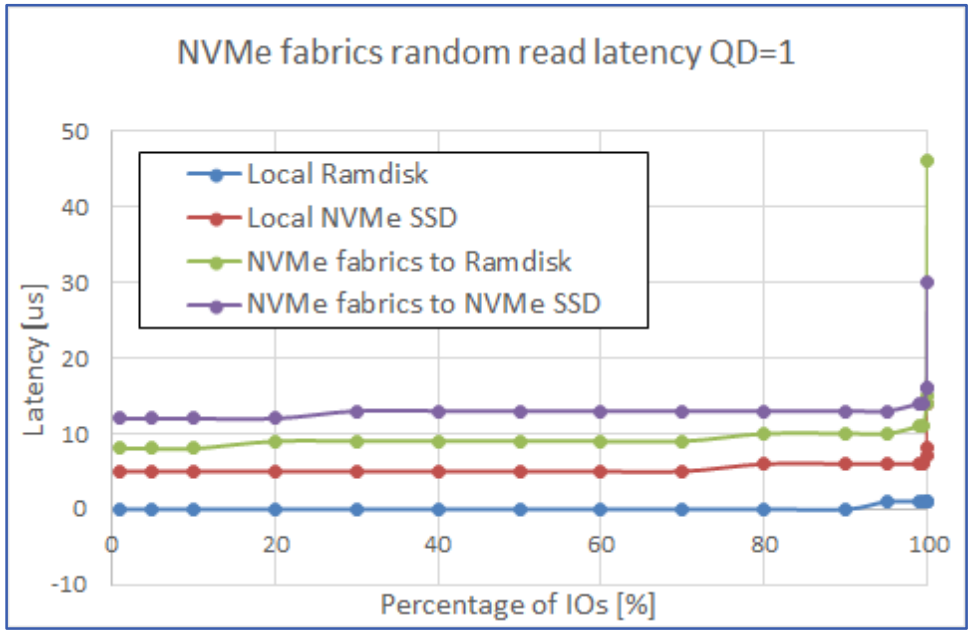
Western Digital.

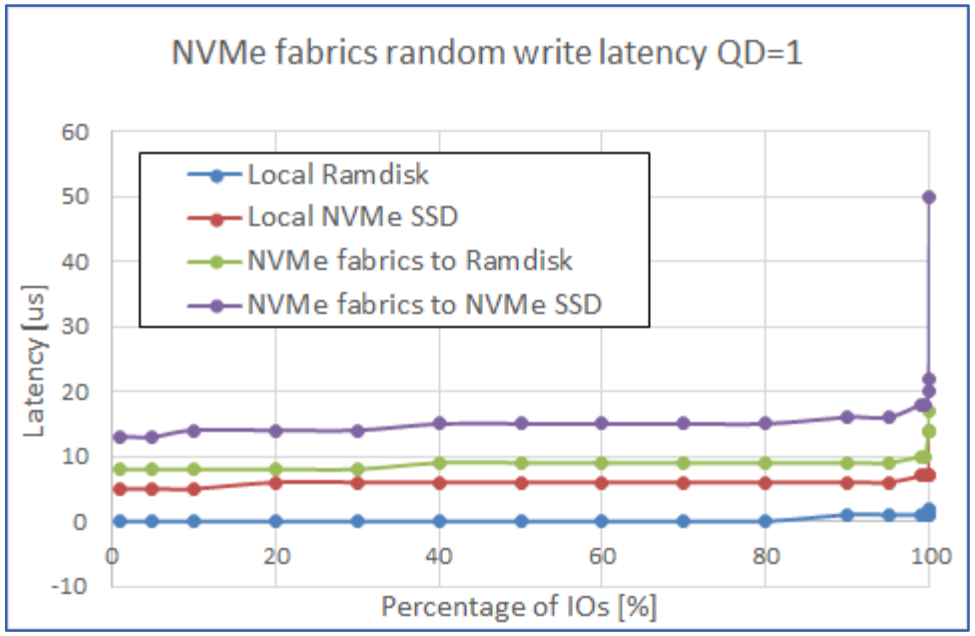
SanDisk®

WD

HGST
a Western Digital brand

Performance Measurements





- Over Infiniband
- 13 us latency at QD=1 for random reads
  - Sub-10 us network contribution
- Further improvements
  - Polling library should remove 3 us from the local device
  - 2-3 us additional improvement in network contribution should be possible

Western Digital®

Thank You