NVMe Performance
Local vs. Remote

Forum F-22

Oscar Pinto, Sr. Staff Architect
Ming Lin, Sr. Architect
Gunna Marripudi, Principal Architect

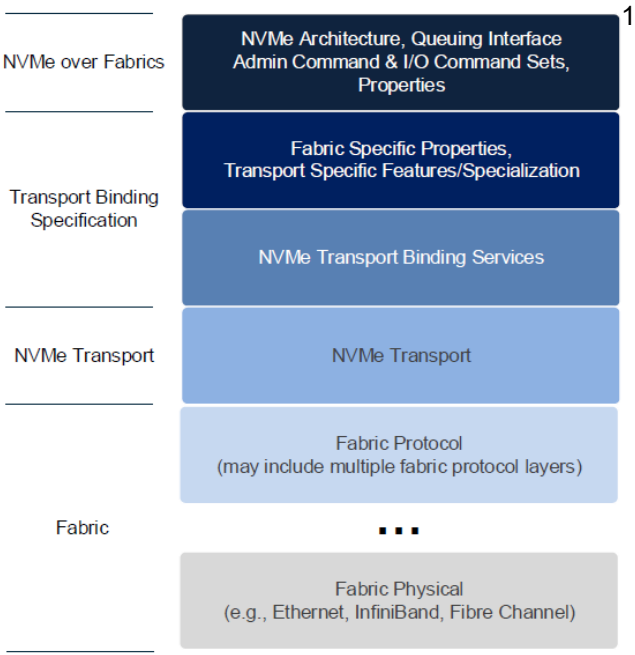
Samsung Semiconductor Inc.

Agenda

- NVMf Overview
- Test Configuration
- Performance Comparison
- Call for Action

NVMe over Fabrics Overview

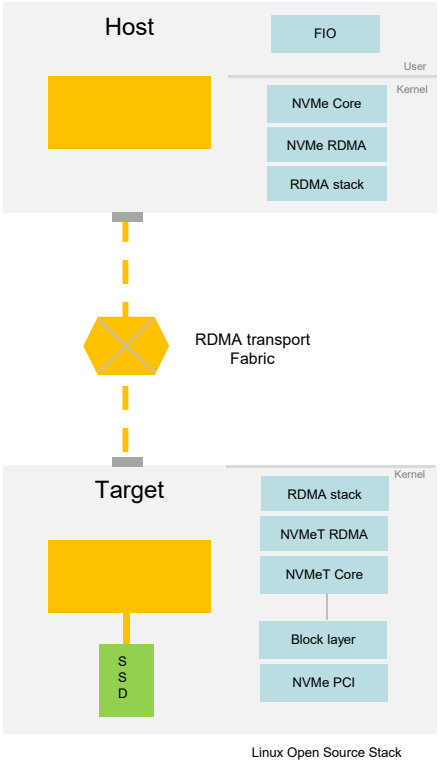
- Maintains consistency with base NVMe definition but for fabrics support[1]
- Support for multiple transport types
- Exposes NVMe parallelism to host
- Performance closer to local attached NVMe devices



[1] *http://www.nvmexpress.org/wp-content/uploads/NVMe_over_Fabrics_1_0_Gold_20160605.pdf*

NVMe over Fabrics Linux Stack

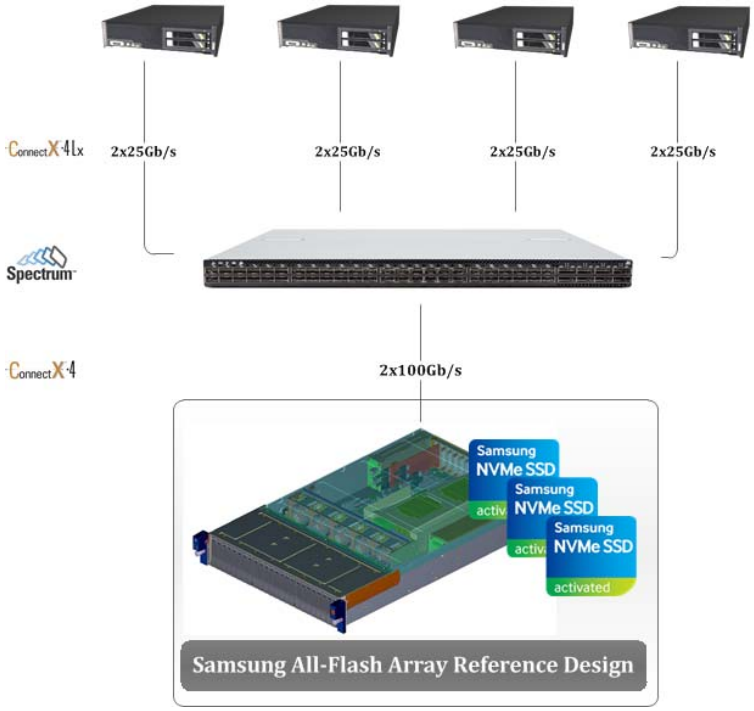
- Kernel stack
- Leverages OFED RDMA stack
- NVMe command processing in
  - Initiator: *nvme_core*
  - Target: *nvmet_core*
- NVMe transport binding in
  - Initiator: *nvme_rdma*
  - Target: *nvmet_rdma*



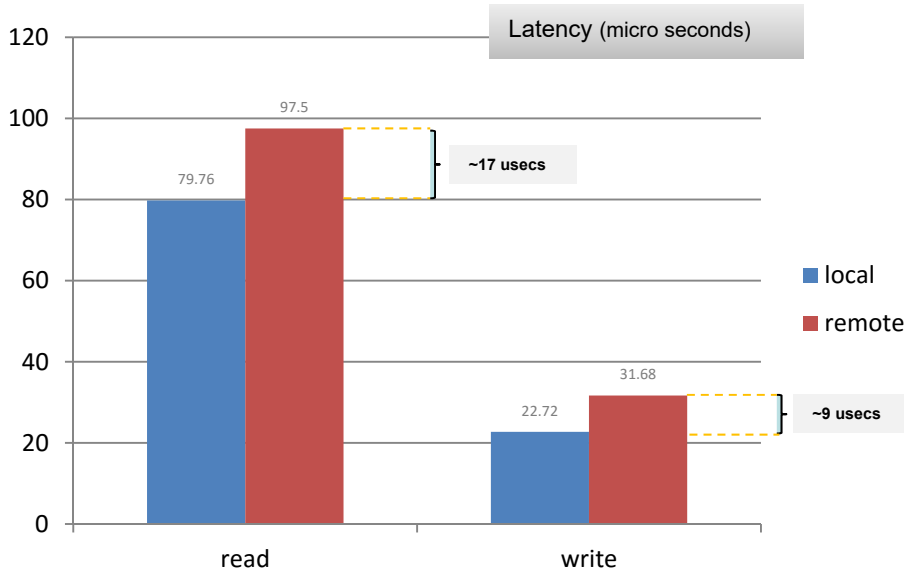
Linux Open Source Stack

Test Configuration

- Configuration
  - 1x NVMf target
    - 24x Samsung PM963 NVMe 2.5” 960GB SSDs
    - 2x 100Gb/s ConnectX®-4 EN
  - 4x initiator hosts
    - 2x25Gb/s each
  - Ubuntu 14.04.4 LTS Linux 4.7.0-rc2 kernel
  - Open Source NVMf kernel drivers



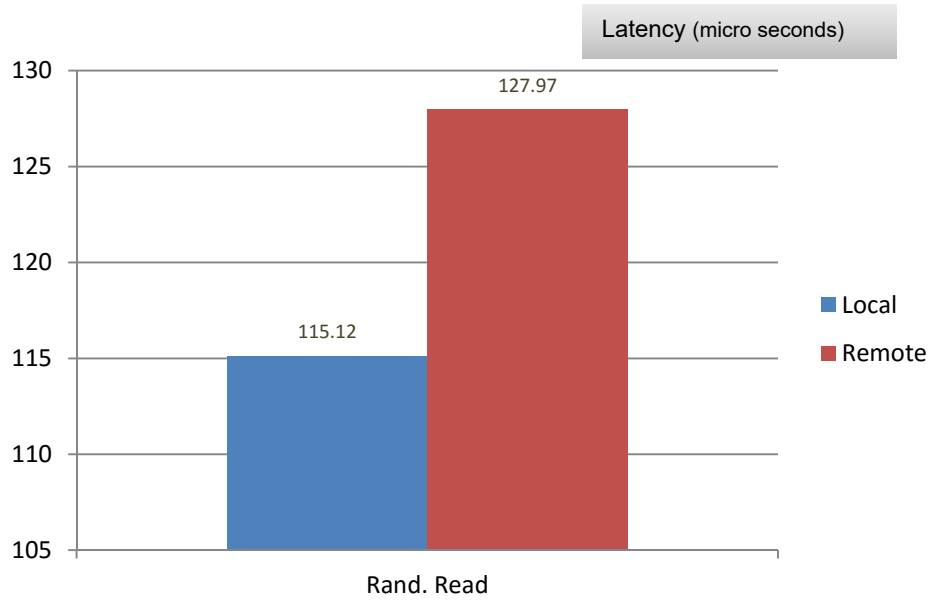
Latency Comparison



Random IO at QD1, 1 job

- Round-trip delta: Reads ~17usecs; Writes ~9usecs

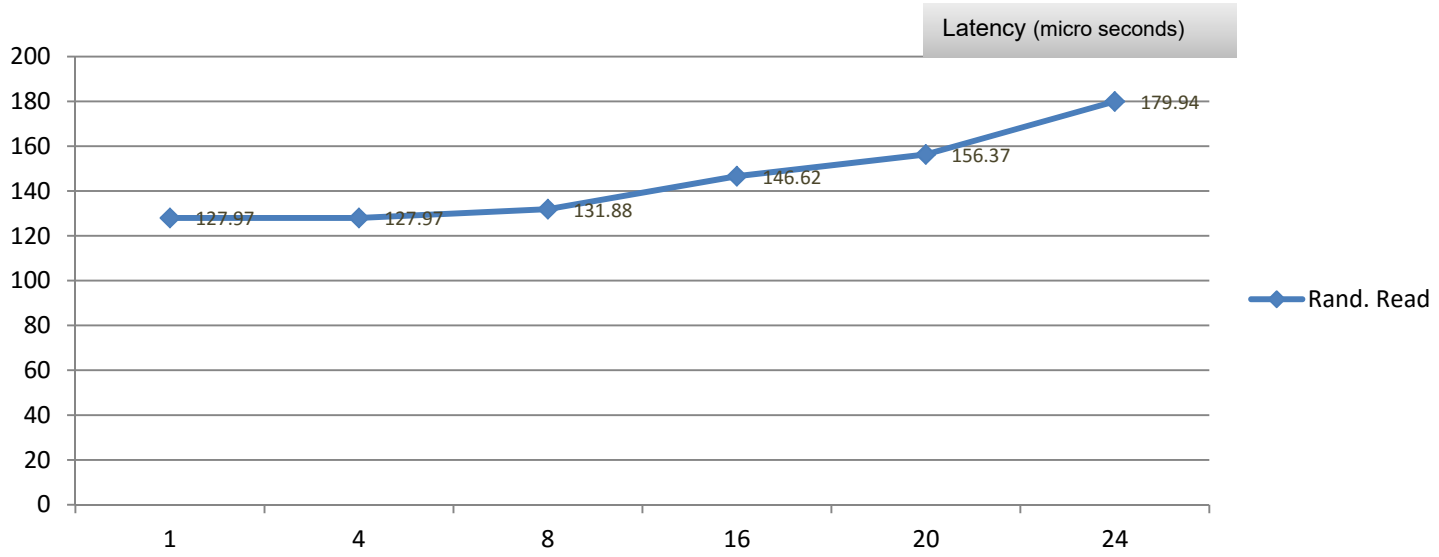
Latency Comparison - Loaded



Random IO at QD16, 2 jobs

- Performance delta: ~12 usecs

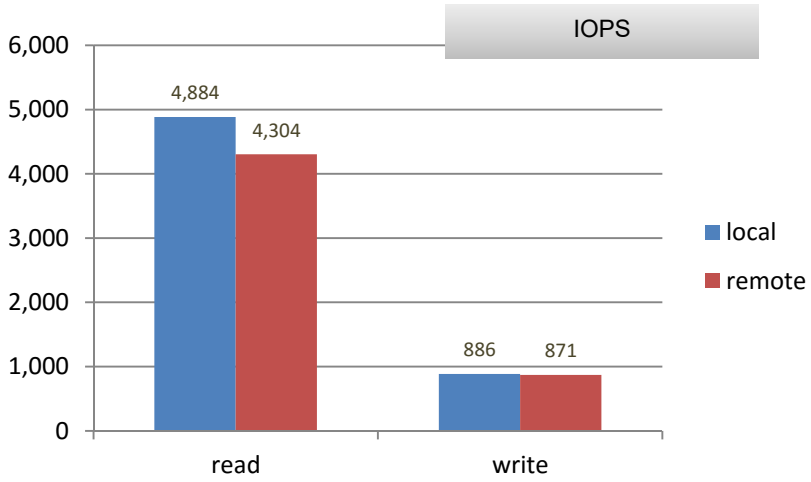
Latency Comparison - Scale

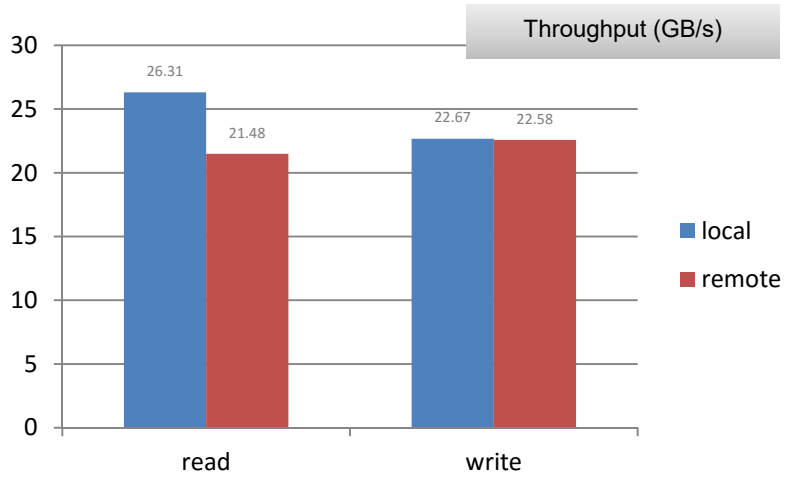


Random IO QD16

- NVMf latency scales from 1 to 24 drives

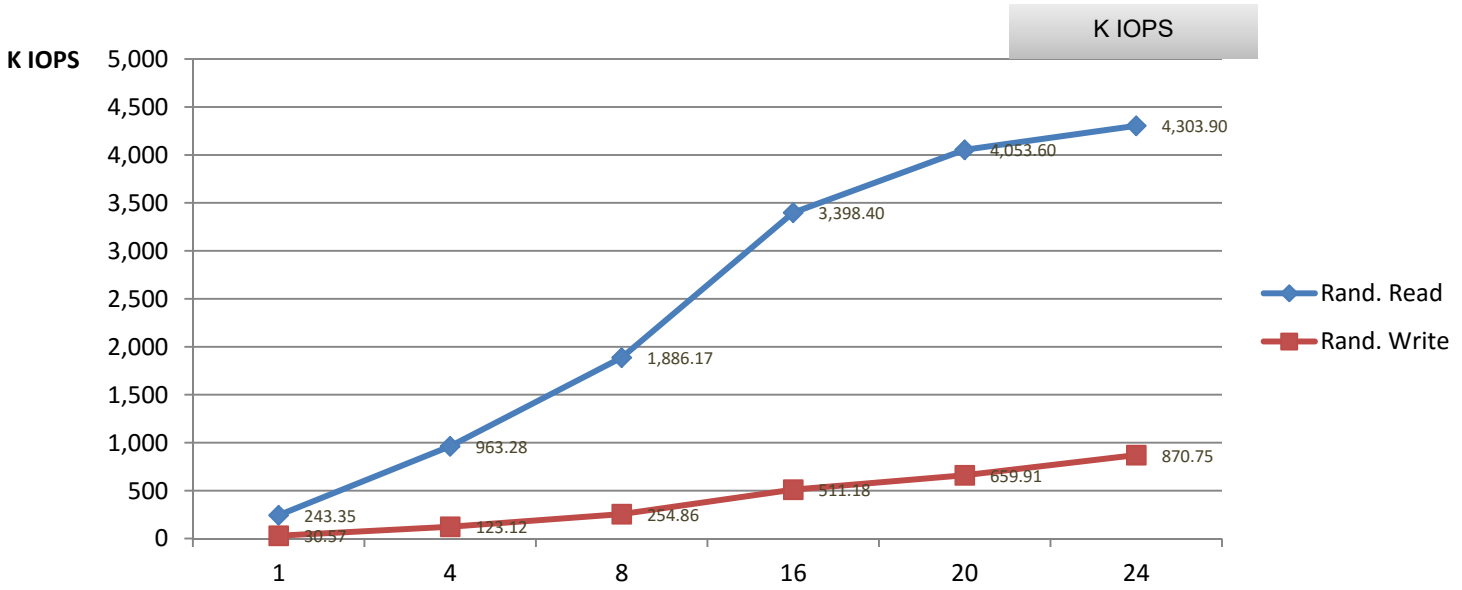
Performance (24 SSDs)





- High aggregate NVMf performance: 4.3M IOPS & 21.5GB/s throughput
- Further optimizations needed for performance to scale

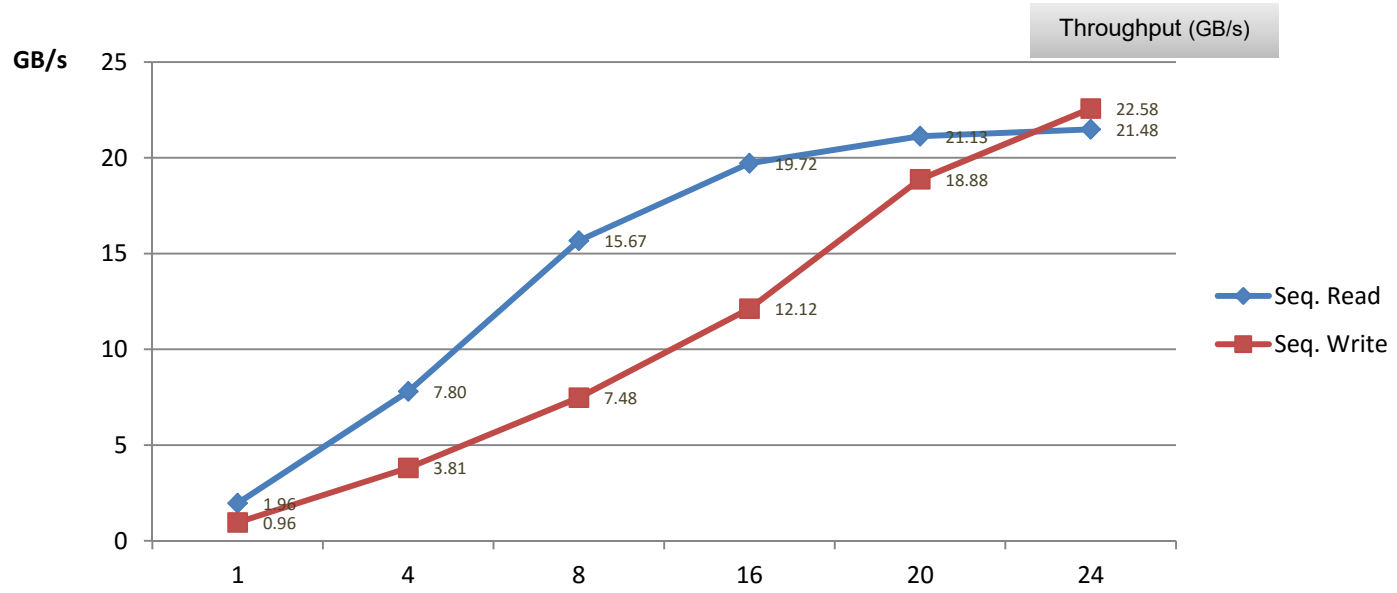
IOPS Scaling



Random IO at QD16

- NVMf stack scales linearly for Random IO

Throughput Scaling



Sequential IO at QD16

- NVMf stack scales linearly for Sequential IO

Summary: NVMe Local vs. Remote

Performance Delta		1-drive	24-drive
Latency	**Read**	11%	15%
	Write	On par	On par
IOPS	**Read**	10%	12%
	Write	On par	2%
Throughput	**Read**	On par	18%
	Write	On par	On par

Call for Action

- Further performance Analysis & tuning of Linux Open Source stack
- Transport binding optimizations
- Feature enhancements
  - Faster failure recover scenarios
  - Reservations support

Thanks

Test Methodology

- Host – Target Setup
  - Ubuntu 14.04.4 LTS Linux 4.7.0-rc2 kernel
  - Each host mapped with 2 subsystems each with 3 SSDs
  - Each subsystem mapped to 1x 25Gb/s NIC
- Benchmark tool
  - fio 2.6-20-g2caf
  - ioengine=libaio
  - Random IOPS: 4k, iodepth=16, numjobs=2
  - Sequential IO: 128k, iodepth=32, numjobs=1
  - Latency: random IO 4k, iodepth=1, numjobs=1