NVMe™ Software Drivers: What’s New and What’s Supported?

Scott Lee – Windows; Sudhanshu Jain / Murali Rajgopal – vmware; Jim Harris – SPDK
Uma Parepalli, Session chair; Cameron Brett - Organizer

August 06, 2019

Sponsored by NVM Express™ organization, the owner of NVMe™, NVMe-oF™ and NVMe-MI™ standards
Speakers

Scott Lee
Sudhanshu (Suds) Jain
Murali Rajagopal
Jim Harris

Uma Parepalli, Session Chair
Cameron Brett, Organizer
NVMe Driver Ecosystem

Robust drivers available on all major platforms
Visit NVM Express Website [http://nvmexpress.org](http://nvmexpress.org) for Drivers related resources
UEFI NVMe Drivers – Very stable in 2019

• Highly stable UEFI NVMe drivers available on Intel and ARM platforms
• NVMe support available from preboot UEFI to booting all major Operating Systems.
Windows Inbox NVMe™ Driver

Scott Lee, Principle Software Engineer Lead, Microsoft
Agenda

• New Additions for Windows 10 version 1903, May 2019 Update (19H1)
• Windows NVMe™ Diagnostic
• New Additions for Next Windows version
• Futures
Windows 10 version 1903, May 2019 Update

- TP4018/4018a: NVM Set & Endurance Group
- Improved diagnostics of NVMe hardware issues
  - Controller Fatal Status (CFS)
- Device Self-Test
- Runtime D3 for NVMe™
- Host Controlled Thermal Management Feature
NVMe™ Diagnostic – Controller Fatal Status

- Checked when Async Event Notification (AEN), controller reset (e.g. IO timeout), invalid command ID in completion entry or command failure
- Storport event 534 in Microsoft-Windows-Storage-Storport/Operational channel
NVMe™ Diagnostic – SMART Log

- Regular monitoring and logging of NVMe™ SMART/Health Informational Log
  - NOTE: expect drive update SMART values asynchronously to the Get Log call
- Logs all contents of SMART/Health Informational Log
- Storport event 512 in Microsoft-Windows-Storage-Storport/Health channel
NVMe™ Diagnostic – AEN

• Driver will send Asynchronous Event Request as part of controller initialization
• Event logged when AEN indicates a warning or error event
  • Error Event - Critical warning bit set
  • Warning Event - Available spare below 2
  • Warning Event – Percentage used above 95
• Storport event 539 for error events in Microsoft-Windows-Storage-Storport/Health channel
• Storport event 543 for warning events in Microsoft-Windows-Storage-Storport/Health channel
NVMe™ Diagnostic – AEN (cont)

• Example AEN Error Event - Critical Failure
NVMe™ Diagnostic – AEN (cont)

- Example AEN Warning Event – Percentage Used Above Threshold
NVMe™ Diagnostic – IO Performance

• Classification of IO performance into pre-defined latency buckets
• Storport event 505 in Microsoft-Windows-Storage-Storport/Operational channel
NVMe™ Diagnostic – Command Tracing

• Support for tracing of NVMe command and response data

• Turn on by enabling Miniport and CommandTrace keywords for Microsoft-Windows-Storport ETW provider.

  • Method 1: Download Windows Performance Toolkit and run following commands in Command Prompt. Use Windows Performance Analyzer to view storport.etl.
    1. xperf -start STORPORT -on Microsoft-Windows-Storport:0x0000200000000080:4 -BufferSize 1024 -MinBuffers 4096 -MaxBuffers 4096
    2. <run test>
    3. xperf -stop STORPORT -d storport.etl

  • Method 2: Microsoft Message Analyzer.
    1. Add a new Live Trace and specify Microsoft-Windows-Storport as system provider. Configure the provider and select Miniport and CommandTrace keywords.
    2. Start the session and run your test. You should start to see some events.
    3. Stop the session to stop tracing.
NVMe™ Diagnostic – Command Tracing (cont)

- Example output from Windows Performance Analyzer
NVMe™ Diagnostic – Command Tracing (cont)

• Example output from Microsoft Message Analyzer
Next Windows Version

• Development for next Windows version in progress
• Non-Operational Power State Config Feature
• LED for NVMe™ Devices
  • ACPI-based: PCIe® SSD Status LED Management _DSM
  • PCI-based: Native PCIe Enclosure Management (NPEM)
Futures*

- Native NVMe™ Storage Stack
- Zoned Namespace (ZNS)
- Device Firmware Hang Detection
- Runtime Hardware Reset of NVMe Devices

* Not plan of record
vSphere NVMe™ Driver Support

Sudhanshu (Suds) Jain and Murali Rajagopal, VMware
# NVMe™ Focus @VMWare

## Driver
- Boot (UEFI)
- Firmware Update
- End-to-end protection
- Deallocate/TRIM/Unmap
- 4K
- SMART, Planned hot-remove

## Core Stack
- Reduced serialization
- Locality improvements
- vNVMe Adaption layer
- Multiple completion worlds support in NVMe

## Virtual Devices
- NVMe™ 1.0e spec
- Hot-plug support
- VM orchestration

## vSphere 6.5
- Performance enhancements
- Extended CLI
- Name space management
- Async event error handling
- Enhance diagnostic logs

## Future Direction
- PCIe Native Hot-plug
- LED Management
- NVMe Over Fabric
- Multiple fabric option
- Sanitize

## vSphere 6.7
- Optimized stack - Highly parallel execution for single path local NVMe devices
- Reach target of 90%+ performance of device spec

## Virtual Devices
- NVMe™ 1.0e spec
- Hot-plug support
- VM orchestration

## Performance improvements
- Async mode support
- unmap support

## Future Direction
- Next Generation Storage Stack with ultra-high IOPS
- End-to-end NVMe Stack
- NVMe Multi-pathing, ANA

## Virtual Devices
- NVMe™ 1.0e spec
- Hot-plug support
- VM orchestration

## Performance improvements
- Async mode support
- unmap support

## Future Direction
- Rev the specification
- Parallel execution @backend
- 4K Support
- Scatter-gather support
- Interrupt coalescing
**NVMe™ Performance Boost**

**Hardware:**
- Intel® Xeon® E5-2687W v3 @3.10GHz (10 cores + HT)
- 64 GB RAM
- NVM Express® 1M IOPS @ 4K Reads

**Software:**
- vSphere® 6.0U2 vs. Future prototype
- 1 VM, 8 VCPU, Windows® 2012, 4 VMDK eager-zeroed
- IOMeter:
  - 4K seq reads, 64 OIOs per worker, even distribution of workers to VMDK
(Future) NVMe™ Driver Architecture

ESXi Storage Stack

ESXi Next Generation Storage Stack

NVMe™ Transport Device Driver Framework

PCIe Transport Driver

RDMA Transport Driver (RoCEv1, RoCEv2, iWarp)

Fibre Channel Transport Driver

Stack Interface 1

Stack Interface 2

SCSI NVMe Translation

NVMe Core Functionality

CLI

NVMe-oF Transport Abstraction

Driver Interface

vmknvme
VMware’s NVMe™ Driver Ecosystem

- Available as part of base ESXi image from vSphere 6.0 onwards
  - Faster innovation with async release of VMware NVMe™ driver
- VMware Opensource its NVMe Driver to encourage ecosystem to innovate
  - [https://github.com/vmware/nvme](https://github.com/vmware/nvme)
- Broad VMware NVMe Driver Ecosystem
  - Close to 300 third party NVMe devices certified on VMware NVMe driver
- Beyond NVMe PCI Driver (Future)
  - Actively working with broad I/O controller and storage array partners to bring NVMe-oF solutions
Accelerating NVMe™ with SPDK

Jim Harris, Principal Software Engineer, Intel
Notices and disclaimers

• Intel technologies’ features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.

• Some results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

• Intel processors of the same SKU may vary in frequency or power as a result of natural variability in the production process.

• Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

• Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice Revision #20110804.

• The benchmark results may need to be revised as additional testing is conducted. The results depend on the specific platform configurations and workloads utilized in the testing, and may not be applicable to any particular user's components, computer system or workloads. The results are not necessarily representative of other benchmarks and other benchmark results may show greater or lesser impact from mitigations.

• Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks.

• Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

• Intel technologies’ features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at www.intel.com.

• The cost reduction scenarios described are intended to enable you to get a better understanding of how the purchase of a given Intel based product, combined with a number of situation-specific variables, might affect future costs and savings. Circumstances will vary and there may be unaccounted-for costs related to the use and deployment of a given product. Nothing in this document should be interpreted as either a promise of or contract for a given level of costs or cost reduction.

• No computer system can be absolutely secure.

• © 2019 Intel Corporation. Intel, the Intel logo, Xeon and Xeon logos are trademarks of Intel Corporation in the U.S. and/or other countries.

• *Other names and brands may be claimed as the property of others.
**User Space Storage Software Stack**

- Extreme performance (>10M IO/s on one thread)
- Block device abstraction and device drivers
- Network and virtualization protocols
- Resets, timeouts, I/O splitting, volume management

**Widely Adopted**

- Powering major storage systems in production today

**C Libraries and Applications**

- Open Source (GitHub, BSD License)
- Active Community (~50 contributors each quarter)
## SPDK Architecture

### Block Storage Protocols
- **Networking**: NVMe-oF (RDMA, TCP), iSCSI
- **Virtualization**: vhost-scsi, vhost-bk, vhost-nvme

### File Storage Services
- **Filesystems**: BlobFS

### Block Storage Services
- **Partitioning**: Logical Volumes, GPT
- **Caching**: OCF
- **Host FTL**: Open Channel
- **Pooling**: RAID-0
- **Transforms**: Crypto, Compression

### Block Storage Providers
- NVMe, Ceph RBD, Linux AIO, virtio, iSCSI, pmemblk, malloc, null

### Drivers
- NVMe (PCIe, RDMA, TCP), virtio (scsi, blk), Intel Quickdata

### Integration
- **Orchestration**: Cinder
- **Database**: RocksDB
- **Scale-out Storage**: Ceph

### Tools
- **Benchmarking**: fio
- **Management**: nvme-cli, spdk-cli
SPDK and Kernel

SPDK has better performance and efficiency compared to interrupt-driven kernel mode approaches

BUT...

SPDK is not a general-purpose solution

- covers some use cases very well – others not at all (or at least not well)

Polled mode design and userspace implementation drove much of the SPDK design
NVMe™ Transport Abstraction

Enables different implementations for different transports

- construct/destruct controller
- set/get register value
- create/delete I/O queue pair
- submit request
- process completions
NVMe-oF™ Target

Spec-compliant, fully functional NVMe-oF™ target

▪ No modifications on client/compute node

Supports broad range of storage services – including:

▪ Sharing SSD across multiple clients (Logical Volumes)
▪ At-rest data encryption with crypto offload
▪ SSD pooling/striping
Supported Features

Explicit Queue Pair Allocation
Metadata and Data Protection
Controller Memory Buffer
Timeout Handling
SGL
Asynchronous Attach
AER
NVMe-oF™ Persistent Reservations
NVMe™ /TCP

NVMe™ TP ratified November 2018

SPDK added TCP transport for

- NVMe driver
- NVMe-oF™ target

Supports alternative TCP stack implementations
Host Block FTL

Host FTL enabling smart data placement

- Based on OC2.0 specification

Block FTL support added to bdev nvme module

Long term goal: Zoned Namespace API

- With ZNS/OC adapters
NVMe™ Performance: Avoid MMIO

- Past: Simple completion queue doorbell batching
  - Ring doorbell after processing first 3 completions

- Recent: Leverage polling
  - Delay ringing submission queue doorbell until end of poll call

- Future: Advanced completion queue batching
  - Track number of free cq slots
  - Only ring doorbell when slots are needed
SPDK NVMe™ Driver Performance

https://spdk.io/news/2019/05/06/nvme/

System Configuration: 2S Intel(R) Xeon(R) Platinum 8280L (use single thread for testing), 192GB DDR4 Memory, 6x Memory Channels per socket, Fedora 29, Linux kernel 5.0.0-rc+, BIOS: HT enabled, p-states enabled, turbo enabled, SPDK 19.04+, SPDK nvme-perf tool used for benchmarking, numjobs=1, direct=1, 21x Intel P4610 1.6T SSD or 20x Intel P4800X 375GB SSD.
Questions?