SPDK: Building Blocks For Scalable, High Performance Storage Applications

Benjamin Walker
Intel Corporation
Agenda

- What is the Storage Performance Development Kit (SPDK)?
- How did SPDK get started?
- What are the benefits of an NVM Express* (NVMe) polled mode driver?
- How does SPDK support protocols like NVMe over Fabrics?
- What are some of the future areas of development for SPDK?
- Summary and Next Steps
Agenda

- What is the Storage Performance Development Kit (SPDK)?
- How did SPDK get started?
- What are the benefits of an NVM Express* (NVMe) polled mode driver?
- How does SPDK support protocols like NVMe over Fabrics?
- What are some of the future areas of development for SPDK?
- Summary and Next Steps
The Problem: Software is becoming the bottleneck

Latency
- HDD: >2ms
- SATA NAND SSD: <100µs
- NVM Express® NAND SSD: <100µs
- Intel® Optane™ SSD: >400,000 IO/s

I/O Performance
- HDD: <500 IO/s
- SATA NAND SSD: >25,000 IO/s
- NVM Express® NAND SSD: <100µs
- Intel® Optane™ SSD: >400,000 IO/s

The Opportunity: Use Intel software ingredients to unlock the potential of new media
**Intel® Platform Storage Reference Architecture**
- Optimized for *Intel platform* characteristics
- Open source building blocks (BSD licensed)
- Available via github.com/spdk or spdk.io

**Scalable and Efficient Software Ingredients**
- User space, lockless, polled-mode components
- Up to millions of IOPS per core
- Designed for Intel Optane™ technology latencies
Storage Performance Development Kit (SPDK)

Client Software

Storage Protocols

- iSCSI Target
- NVMe-oF* Target
- SCSI

Storage Services

- Block Device Abstraction Layer (BDAL)
- BDAL Extension Modules (using BDAL API)

Hardware Drivers

- Intel® QuickData Technology Driver
- NVM Express* SSD Driver

All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice. The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps. Statements in this document that refer to Intel’s plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel’s results and plans is included in Intel’s SEC filings, including the annual report on Form 10-K.
Agenda

- What is the Storage Performance Development Kit (SPDK)?
- How did SPDK get started?
- What are the benefits of an NVM Express* (NVMe) polled mode driver?
- How does SPDK support protocols like NVMe over Fabrics?
- What are some of the future areas of development for SPDK?
- Summary and Next Steps
Data Plane Development Kit (DPDK)

Software solution for accelerating Packet Processing workloads

- Optimized for IA platforms
- Vibrant community support

- Free, Open Source, BSD License
- Website: dpdk.org

What does SPDK share with DPDK?

- Concepts
- Framework Code
- Community

What DPDK Primitives Does SPDK Use?

- DPDK Libraries
  - Buffer Management
  - Queue/Ring Functions
  - Thread Polling Model
  - Environment Abstraction Layer (EAL)
  - Linux Kernel space
Disclaimer: 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 http://www.intel.com/performance.

Data captured by Intel with DPDK l3fwd (Layer 3 forwarding) sample application. Packet generator: Ixia IxNetwork 8.03 EA.
Agenda

- What is the Storage Performance Development Kit (SPDK)?
- How did SPDK get started?
- What are the benefits of an NVM Express* (NVMe) polled mode driver?
- How does SPDK support protocols like NVMe over Fabrics?
- What are some of the future areas of development for SPDK?
- Summary and Next Steps
## NVM Express* Driver Key Characteristics

- Supports NVM Express* (NVMe) 1.2 spec-compliant devices
- Userspace Asynchronous Polled Mode operation
- Application owns I/O queue allocation and synchronization

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>End-to-end Data Protection</td>
<td>Integrity from host to drive with T10-DIF/DIX</td>
</tr>
<tr>
<td>Scatter-Gather Lists (SGL)</td>
<td>Eliminates buffer copies</td>
</tr>
<tr>
<td>Reservations</td>
<td>For dual port NVMe usage models</td>
</tr>
<tr>
<td>Namespace Management</td>
<td>Support multiple dynamic NVMe namespaces</td>
</tr>
<tr>
<td>Weighted Round Robin</td>
<td>Quality of Service for NVMe I/O queues</td>
</tr>
</tbody>
</table>
NVM Express* Driver Throughput Scalability

**I/O Performance on Single Intel® Xeon® core**

- Systems with multiple NVM Express* (NVMe) SSDs capable of millions of I/O per second
- Results in many cores of software overhead with kernel-based interrupt-driven driver model
- SPDK enables:
  - more CPU cycles for storage services
  - lower I/O latency

SPDK saturates 8 NVMe SSDs with a single CPU core!

System Configuration: 2x Intel® Xeon® E5-2695v4 (HT off), Intel® Speed Step enabled, Intel® Turbo Boost Technology enabled, 8x 8GB DDR4 2133 MT/s, 1 DIMM per channel, CentOS* Linux* 7.2, Linux kernel 4.7.0-rc1, 8x Intel® P3700 NVMe SSD (800GB), 4x per CPU socket, FW 8DV10102, 4KB Random Read I/O, Queue Depth: 32 per SSD. Performance measured by Intel using SPDK perf tool, 4KB Random Read I/O, Queue Depth: 128/SSD.
### NVM Express* Driver Software Overhead

SPDK reduces NVM Express* (NVMe) software overhead up to 10x!

<table>
<thead>
<tr>
<th>Kernel Source of Overhead</th>
<th>SPDK Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interrupts</td>
<td>Asynchronous Polled Mode</td>
</tr>
<tr>
<td>Synchronization</td>
<td>Lockless</td>
</tr>
<tr>
<td>System Calls</td>
<td>Userspace Hardware Access</td>
</tr>
<tr>
<td>DMA Mapping</td>
<td>Hugepages</td>
</tr>
<tr>
<td>Generic Block Layer</td>
<td>Specific for Flash Latencies</td>
</tr>
</tbody>
</table>

System Configuration: 2x Intel® Xeon® E5-2695v4 (HT off), Intel® Speed Step enabled, Intel® Turbo Boost Technology disabled, 8x 8GB DDR4 2133 MT/s, 1 DIMM per channel, CentOS® Linux® 7.2, Linux kernel 4.7.0-rc1, 1x Intel® P3700 NVMe SSD (800GB), 4x per CPU socket, FW 8DV10102, I/O workload 4KB random read, Queue Depth: 1 per SSD, Performance measured by Intel using SPDK overhead tool, Linux kernel data using Linux AIO
Agenda

- What is the Storage Performance Development Kit (SPDK)?
- How did SPDK get started?
- What are the benefits of an NVM Express* (NVMe) polled mode driver?
- How does SPDK support protocols like NVMe over Fabrics?
- What are some of the future areas of development for SPDK?
- Summary and Next Steps
iSCSI Performance

- iSCSI Target improvements stem from:
  - Non-blocking TCP sockets
  - Pinned iSCSI connections
  - SPDK storage access model

- TCP processing is limiting factor
  - 70%+ CPU cycles consumed in kernel network stack
  - Userspace polled mode TCP required for more improvement

SPDK improves efficiency almost 2x

System Configuration: 2S Intel® Xeon® E5-2699v3: 18C, 2.3GHz (HT off), Intel® Speed Step enabled, Intel® Turbo Boost Technology disabled, 8x4GB DDR4 2133 MT/s, 1 DIMM per channel, Ubuntu® Server 14.10, 3.16.0-30-generic kernel, Ethernet Controller XL710 for 40GbE, 8x Intel® P3700 NVM Express® SSD – 800GB (4 per CPU socket), FW 8DV10102
As measured by: fio – Direct=Yes, 4KB random read I/O, QueueDepth=32, Ramp Time=30s, Run Time=180s, Norandommap=1, I/O Engine = libaio, Numjobs=1
Why NVM Express* over Fabrics?

PCI Express* (PCIe*) Gen3

NVM Express* (NVMe)

Local NVMe Storage

Remote NVMe Storage

RDMA Network
NVM Express* over Fabrics Performance

<table>
<thead>
<tr>
<th>NVMe over Fabrics Target Features</th>
<th>Realized Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Utilizes NVM Express* (NVMe) Polled Mode Driver</td>
<td>Reduced overhead per NVMe I/O</td>
</tr>
<tr>
<td>RDMA Queue Pair Polling</td>
<td>No interrupt overhead</td>
</tr>
<tr>
<td>Connections pinned to CPU cores</td>
<td>No synchronization overhead</td>
</tr>
</tbody>
</table>

SPDK reduces NVMe over Fabrics software overhead up to 10x!

System Configuration: Target system: 2x Intel® Xeon® E5-2695v4 (HT off), Intel® Speed Step enabled, Intel® Turbo Boost Technology enabled, 8x 8GB DDR4 2133 MT/s, 1 DIMM per channel, 8x Intel® P3700 NVMe SSD (800GB), 4x per CPU socket, FW 8DV10102, Network: Mellanox® ConnectX-4 100Gb RDMA, direct connection between initiator and target; Initiator OS: CentOS Linux 7.2, Linux kernel 4.7.0-rc2, Target OS (SPDK): CentOS Linux 7.2, Linux kernel 3.10.0-327.el7.x86_64, Target OS (Linux kernel): CentOS Linux 7.2, Linux kernel 4.7.0-rc2. Performance as measured by: fio, 4KB Random Read I/O, 2 RDMA QP per remote SSD, Numjobs=4 per SSD, Queue Depth: 32/job
Block Device Abstraction Layer (BDAL)

- Block layer optimized for SPDK programming model
  - Lockless, event driven API
  - BDAL API for creating new BDAL drivers
  - Stackable

- Several BDAL modules available today
  - NVM Express* (NVMe) – SPDK NVMe polled mode driver
  - AIO – Linux libaio
  - malloc – Userspace ramdisk
BDAL Extension Modules – Example #1
Intel® Intelligent Storage Acceleration Library (Intel® ISA-L)

- Intel® Intelligent Storage Acceleration Library (Intel® ISA-L)
  - Optimized low-level functions targeting storage applications
  - Erasure coding, parity, CRC, compression, crypto, hashing
  - https://github.com/01org/isa-l

- Example:
  - User-provided deduplication extension module
BDAL Extension Modules – Example #2

nvml – Linux NVM Library

• Linux* NVM Library
  - Set of libraries to provide useful APIs for persistent memory server applications
  - Enables 3D XPoint™ memory

• Example:
  - User-provided write log
Agenda

- What is the Storage Performance Development Kit (SPDK)?
- How did SPDK get started?
- What are the benefits of an NVM Express* (NVMe) polled mode driver?
- How does SPDK support protocols like NVMe over Fabrics?
- What are some of the future areas of development for SPDK?
- Summary and Next Steps
Storage Performance Development Kit (SPDK)

Client Software
- NVMe-oF* Initiator

Storage Protocols
- iSCSI Target
- NVMe-oF Target
- SCSI
- NVMe

NVMe-oF*

Storage Services
- Block Device Abstraction Layer (BDAL)
- BDAL Extension Modules (using BDAL API)

Hardware Drivers
- Intel® QuickData Technology Driver
- NVM Express* (NVMe) SSD Driver

Released
User Provided
2H’16

All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice. The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps. Statements in this document that refer to Intel’s plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel’s results and plans is included in Intel’s SEC filings, including the annual report on Form 10-K.
NVM Express* over Fabrics Additions

• Initiator
  - Enable polled mode userspace access to remote NVM Express* (NVMe) devices
  - Same programming model as SPDK local NVMe access

• BDAL integration w/ NVMe over Fabrics target
  - Export SPDK block devices over NVMe over Fabrics
    ▪ Similar to iSCSI

• Continued performance tuning
  - Scaling to more NVMe devices, more RDMA throughput
Storage Performance Development Kit (SPDK)

Client Software
- NVMe-oF* Initiator

Storage Protocols
- iSCSI Target
- NVMe-oF Target
- SCSI
- NVMe

Storage Services
- Blobstore
- Block Device Abstraction Layer (BDAL)
- BDAL Extension Modules (using BDAL API)

Hardware Drivers
- Intel® QuickData Technology Driver
- NVM Express* (NVMe) SSD Driver

All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice. The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps. Statements in this document that refer to Intel’s plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel’s results and plans is included in Intel’s SEC filings, including the annual report on Form 10-K.
What about a filesystem?

- Most applications want some level of file semantics
  - Example: databases, key/value stores – small number of files, flat hierarchy, no permissions
- Kernel filesystems not usable in SPDK programming model
  - They are in the kernel
  - They are based on POSIX synchronous file semantics
- Need framework for SPDK file-like semantics – an SPDK “Blobstore”
  - Asynchronous, polled-mode, lockless, event driven (i.e., not POSIX)
  - Framework for building higher order services
    - Lightweight filesystem, extent allocator, etc.
Storage Performance Development Kit (SPDK)

Client Software
- NVMe-oF® Initiator

Storage Protocols
- iSCSI Target
- vhost-scsi Target
- NVMe-oF Target
- SCSI
- NVMe

Storage Services
- Blobstore
- Block Device Abstraction Layer (BDAL)
- BDAL Extension Modules (using BDAL API)

Hardware Drivers
- Intel® QuickData Technology Driver
- NVM Express® (NVMe) SSD Driver

All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice. The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps. Statements in this document that refer to Intel’s plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel’s results and plans is included in Intel’s SEC filings, including the annual report on Form 10-K.
SPDK vhost-scsi

- Serve SPDK storage to local virtual machines
  - NVM Express* ephemeral storage
  - SPDK-based BDAL storage
- Leverage existing infrastructure for
  - QEMU vhost-scsi
  - QEMU/DPDK vhost-net user
Agenda

- What is the Storage Performance Development Kit (SPDK)?
- How did SPDK get started?
- What are the benefits of an NVM Express* (NVMe) polled mode driver?
- How does SPDK support protocols like NVMe over Fabrics?
- What are some of the future areas of development for SPDK?
- Summary and Next Steps
Summary and Next Steps

- Fully realizing new media performance requires software optimizations.
- SPDK positioned to enable developers to realize this performance.
- SPDK available today via http://spdk.io.
- Help us build SPDK as an open source community!
Q&A
Legal Notices and Disclaimers

- 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 intel.com.

- Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

- 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 http://www.intel.com/performance.

- Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

- This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

- No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

- Statements in this document that refer to Intel’s plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel’s results and plans is included in Intel’s SEC filings, including the annual report on Form 10-K.

- All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice. The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

- 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.

- © 2016 Intel Corporation. Intel, the Intel logo, Xeon, Optane, QuickData, SpeedStep, Turbo Boost, ISA-L, 3D XPoint and others 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.
Backup