12th ANNUAL WORKSHOP 2016

NVME™ OVER FABRICS

Presented by Phil Cayton
Intel® Corporation

April 6th, 2016
- NVM Express™* Organization
- Scaling NVMe in the datacenter
- Architecture / Implementation Overview
- Standardization and Enabling

*All names, trademarks, registered trademarks, service-marks, brands, and logos may be claimed as the property of their respective owners.
• 80+ companies strong and growing

• Workgroups:
  • Technical – Specifications
  • Driver – Linux™* Host and Target fabrics driver
  • Marketing – NVMe awareness

• Learn more at nvmexpress.org
- Effective scaling of PCIe\(^*\) attached NVMe\(^*\) is limited to 255/system
- Datacenter requires scaling out to 100’s or 1000’s of NVMe SSDs
- Traditional scaleout adds translations between Hosts and NVMe SSD
- NVMe over Fabrics extends NVMe and enables SSD Scale-Out & Low-Latency I/O needed by the datacenter at near local speeds

\(^*\)All names, trademarks, registered trademarks, service-marks, brands, and logos may be claimed as the property of their respective owners.
NVME OVER FABRICS
Industry standard definition of NVMe over Datacenter Network Fabrics

- Shares same base architecture and NVMe Host Software as PCIe
- Specification defines interface supporting multiple network fabrics

NVMe over Fabrics specification defines how NVMe is extended to support new network fabrics
The vast majority of NVMe architecture is leveraged as-is for Fabrics
- NVMe Multi-Queue Host Interface, Subsystem, Controllers, Namespaces, and Commands
- Allows for use of common NVMe Host software with very thin fabric dependent layers

Primary differences reside in the discovery and queuing mechanisms
Fabric Capsules are messages with “encapsulated” NVMe content

- Data Section may be sent either within the Capsule or via a fabric type dependent data transfer mechanism (RDMA_READ/ RDMA_WRITE)

Host to NVMe Controller

NVMe Controller to Host

Data transfer to/from host resident buffer
Discovery Log Entries

- Information needed to establish connections to NVM Subsystems
- Accommodates multiple transports and address types
1) Create a fabric-dependent transport connection

2) Send a Command Capsule with Fabric Connect Operation (AdminQ Connect establishes an “association” to an NVMe Controller)

3) Send <any> Authentication Fabric Commands

4) AdminQ or IOQ Ready for NVMe Commands
NVME OVER FABRICS
Capsule Exchange Example

Host

NVMe Host Software Driver

Send NVMe CMD Capsule 1

RCV NVMe RSP Capsule 7

Fabric Dependent Transport

Port

NVM Subsystem

NVMe Controller

RCV NVMe CMD Capsule 3

SND NVMe RSP Capsule 5

Fabric Dependent Transport

Port

NVMe CMD Capsule

Submission Queue Entry

In Capsule Data (if any)

Command data (if any)

Completion Queue Entry

NVMe RSP Capsule

OpenFabrics Alliance Workshop 2016
### Architecture
- Separation of NVMe Core Logic and Transport
- Common NVMe Core across all transports: OS and Block interface
- Common NVMe Fabric agnostic functions: Capsules, Properties, Connect, and Discovery
- Thin transports with limited NVMe awareness that can interoperate with multiple targets

### Implementation
- Separate Driver Components
- Components interface with Multi-Q Block layer
- Pluggable Transports

Retains efficiency of PCIe NVMe access over fabrics
### Architecture
- Virtual NVMe subsystem representation of PCIe SSD Subsystem
- Separate NVMe subsystem core components and transports
- Discovery Subsystem and controllers

### Implementation
- Creates logical NVMe Subsystems and Controllers that are presented to Hosts
- NVMe Namespaces are logically mapped to physical block devices
- **Linux Host and Target Kernel drivers operational**
  - Target configured with NVMe PCIe SSD
  - To ensure fabrics transport is fabric agnostic it has been tested on InfiniBand™*, iWARP, RoCE RDMA adapters
  - Host has been tested with multiple target implementations

- **Drivers are still being performance tuned**

---

*All names, trademarks, registered trademarks, service-marks, brands, and logos may be claimed as the property of their respective owners.*

---

*OpenFabrics Alliance Workshop 2016*
- NVMe over Fabrics Specification complete
  - In release candidate final review
  - Available publicly on nvmexpress.org: 05/16

- Host and Target drivers will available in upstream Linux kernel
  - Upstream Kernel under drivers/nvme/{host/target}
  - Available: shortly after specification released in ~4.8
  - Fabrics supported:
    - Local: PCIe
    - RDMA: Fabric agnostic kernel verbs transport
    - Fibre Channel (under development)
    - Other fabrics transports under evaluation / pathfinding (e.g., kFabric)

When the specification and Host and Target code is available
Please download, test with your fabric hardware and environments
No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document. Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade. 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. The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

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.

Copyright © 2016, Intel Corporation. All rights reserved. Intel, Pentium, Xeon, Xeon Phi, Core, VTune, Cilk, and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

*Other names and brands may be claimed as the property of others

<table>
<thead>
<tr>
<th>Optimization Notice</th>
</tr>
</thead>
<tbody>
<tr>
<td>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</td>
</tr>
</tbody>
</table>
THANK YOU
NVME OVER FABRICS

Target Components