NVM Express*
Past, Present, and Future

Amber Huffman
Fellow, Intel Corporation
President, NVM Express, Inc.

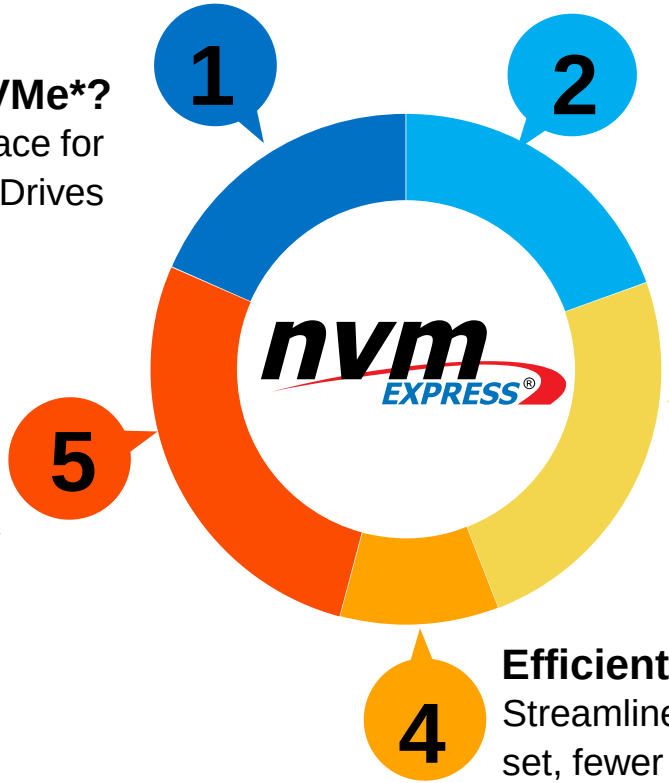
August 9, 2016

Outline

- NVMe* Explained and The Past 5 Years
- NVMe Management Interface
- NVMe over Fabrics
- What is Coming in NVMe Revision 1.3

NVM Express* Explained

What is NVMe*?
THE industry standard interface for PCI Express* Solid State Drives



Architected for SSDs
Eliminates hard drive legacy that SATA and SAS are based upon.

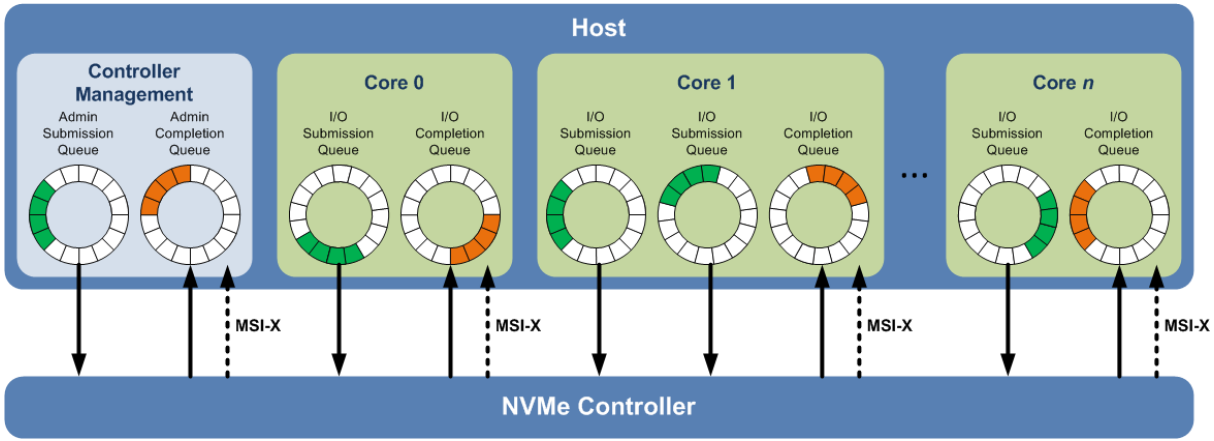
Works Out of The Box
Drivers for all major OSes

Efficient and Scalable
Streamlined protocol and command set, fewer clock cycles per IO

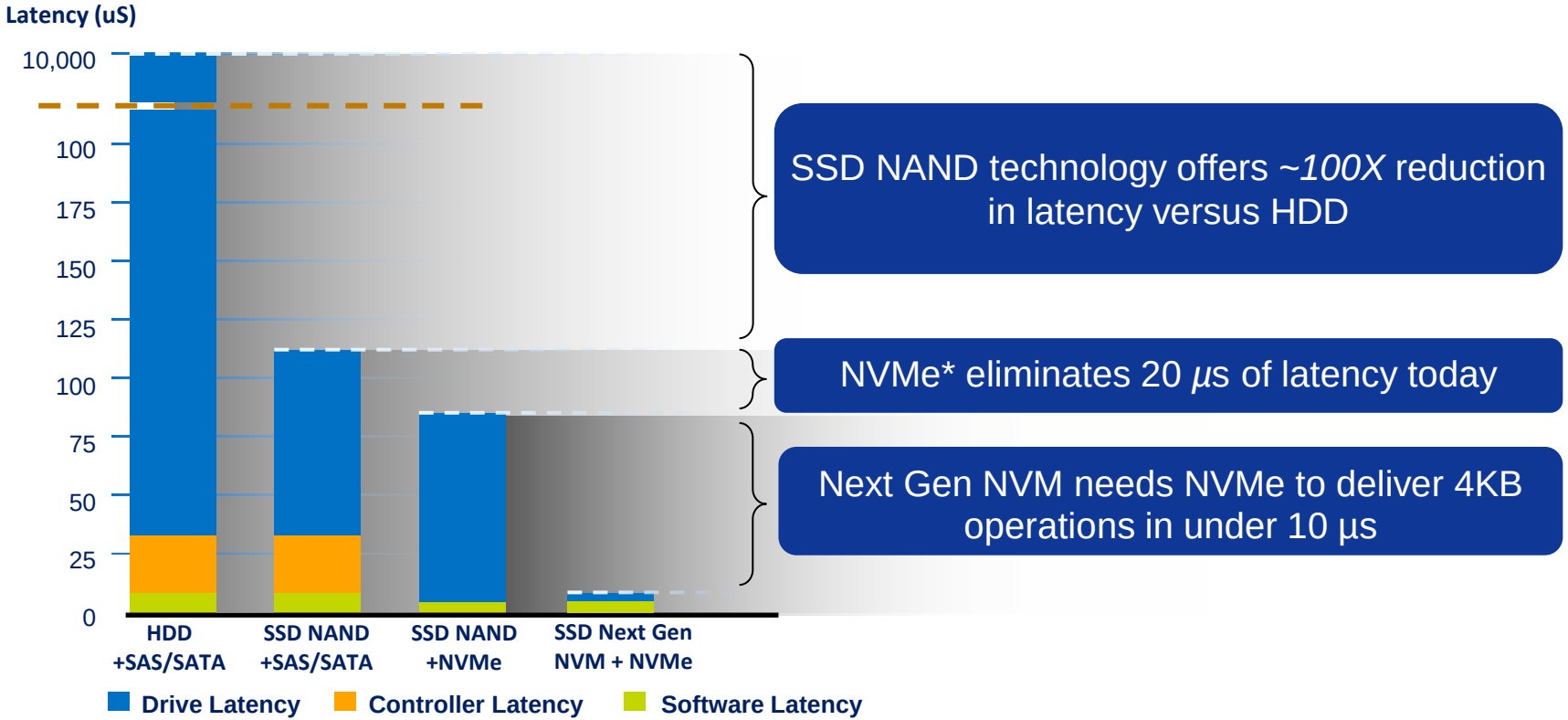
Optimized for Next Gen NVM
New storage stack with low latency to take full advantage

Technical Basics of NVMe*

- Supports deep queues (64K commands per queue, up to 64K queues)
- Supports MSI-X and interrupt steering
- Streamlined and simple command set (13 required commands)
- Optional features to address target segment
  - **Data Center**: Reservations, etc. **Client**: Power features, etc.
- Designed to scale for next generation NVM, agnostic to NVM type used



Delivering Benefit of Next Gen NVM

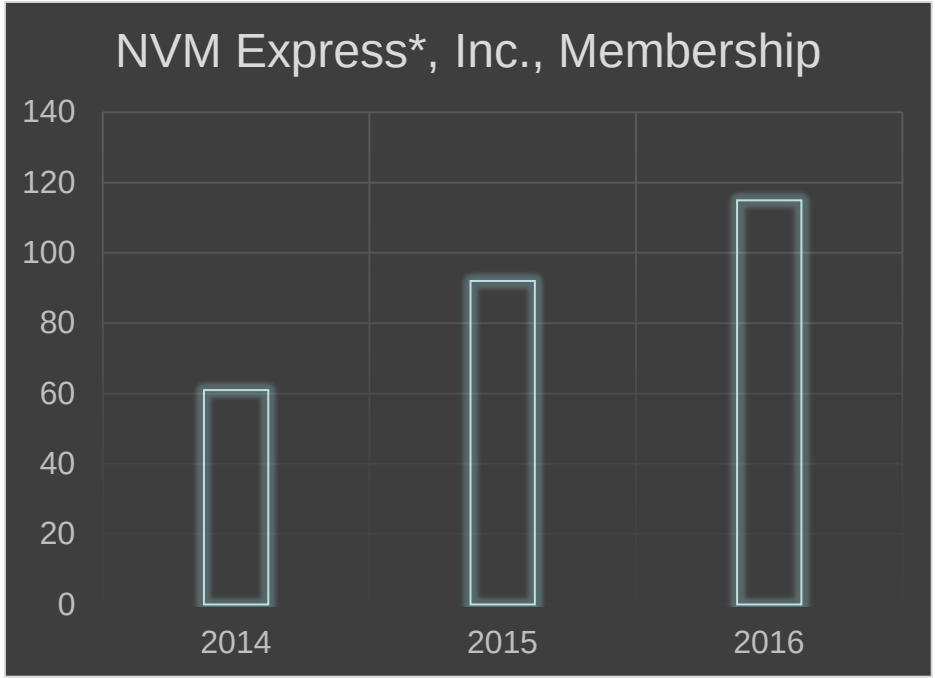


*Other names and brands may be claimed as the property of others.

Source: Storage Technologies Group, Intel. Comparisons between memory technologies based on in-market product specifications and internal Intel specifications.

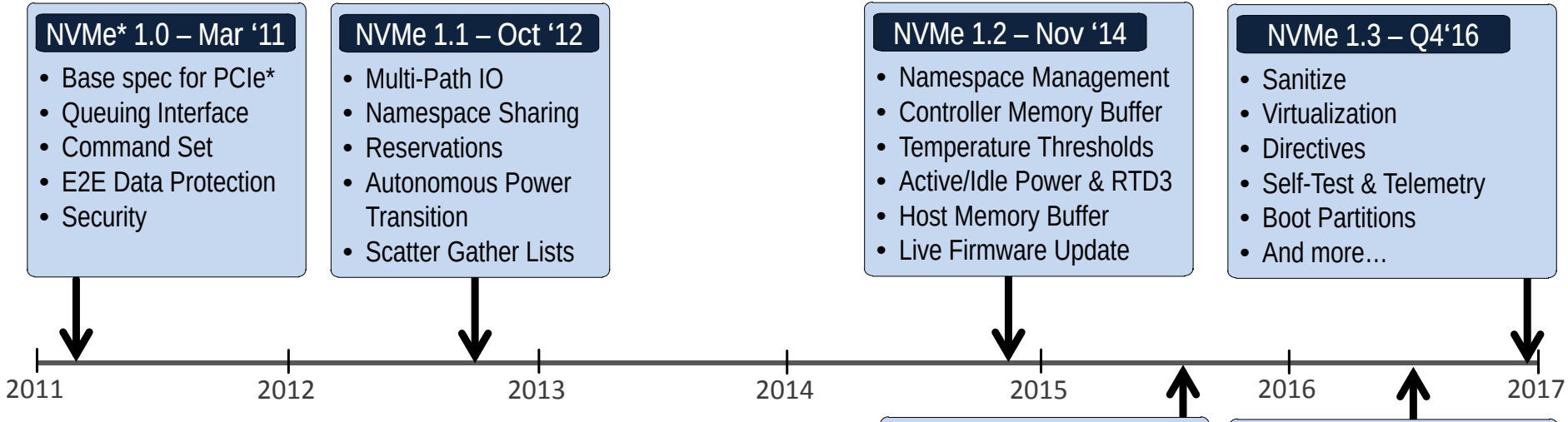
Workgroup Continues to Grow

- The Workgroup has gone from 61 to 115 members in under 2 years
- Direction and governance led by a 12 company Board of Directors



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

NVMe* Development Timeline



NVMe is the place for innovation in SSDs.

Outline

- NVMe* Explained and The Past 5 Years
- NVMe Management Interface
- NVMe over Fabrics
- What is Coming in NVMe Revision 1.3

NVMe* Management Interface

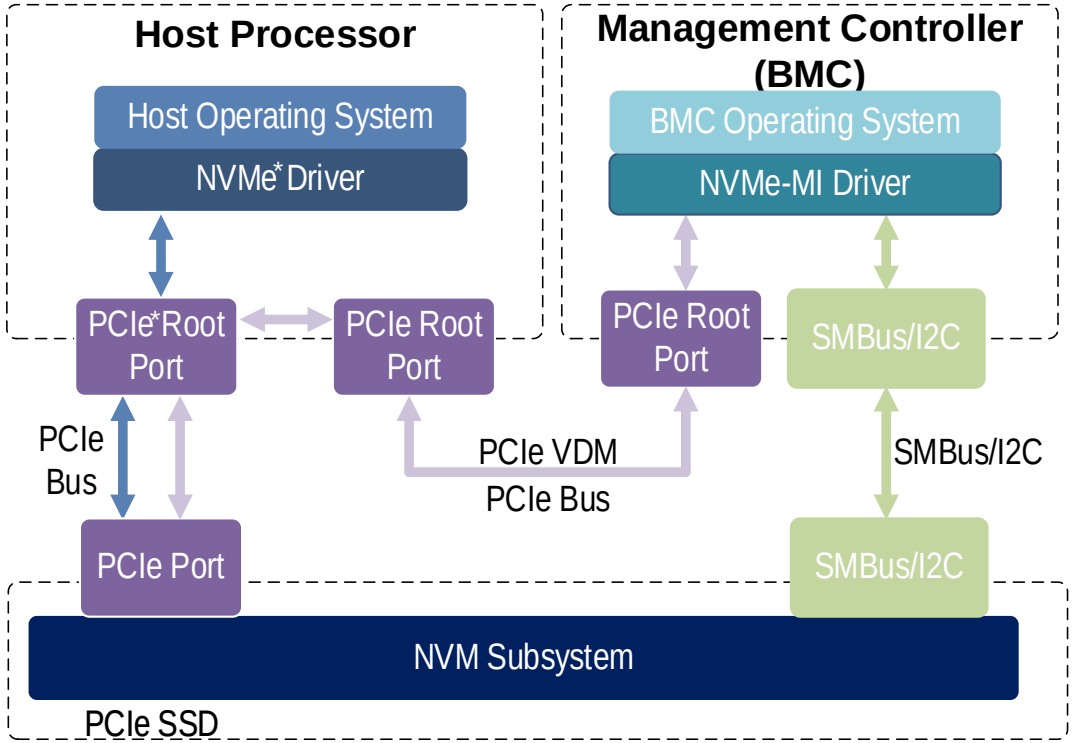
- The Management Interface standardizes out-of-band management for NVMe* devices
  - This is independent of the physical transport (e.g., PCIe* or SMBus)

- The benefit of a standard management interface includes:
  - Reducing cost and broadening adoption
  - Consistent feature set
  - Industry ecosystem (e.g., compliance tests, development tools)

NVMe Management Interface 1.0 published in November 2015

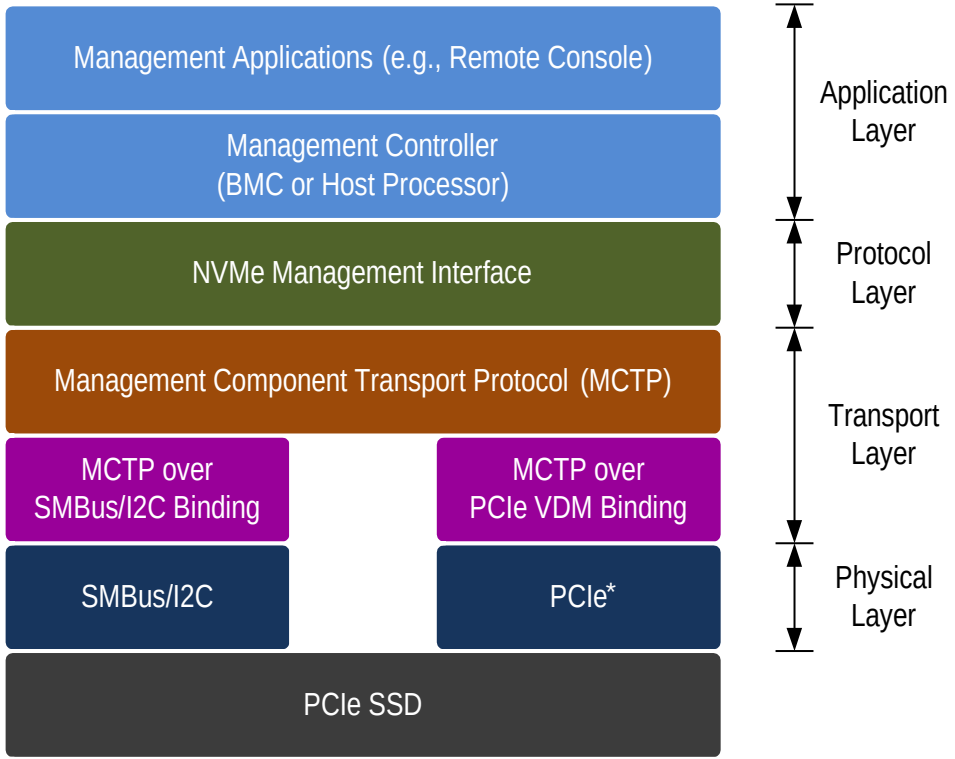
In-Band vs Out-of-Band

- Normal host traffic handled “in-band” via the host NVMe* driver
- Management is done “out-of-band” using PCIe* VDMs or SMBus

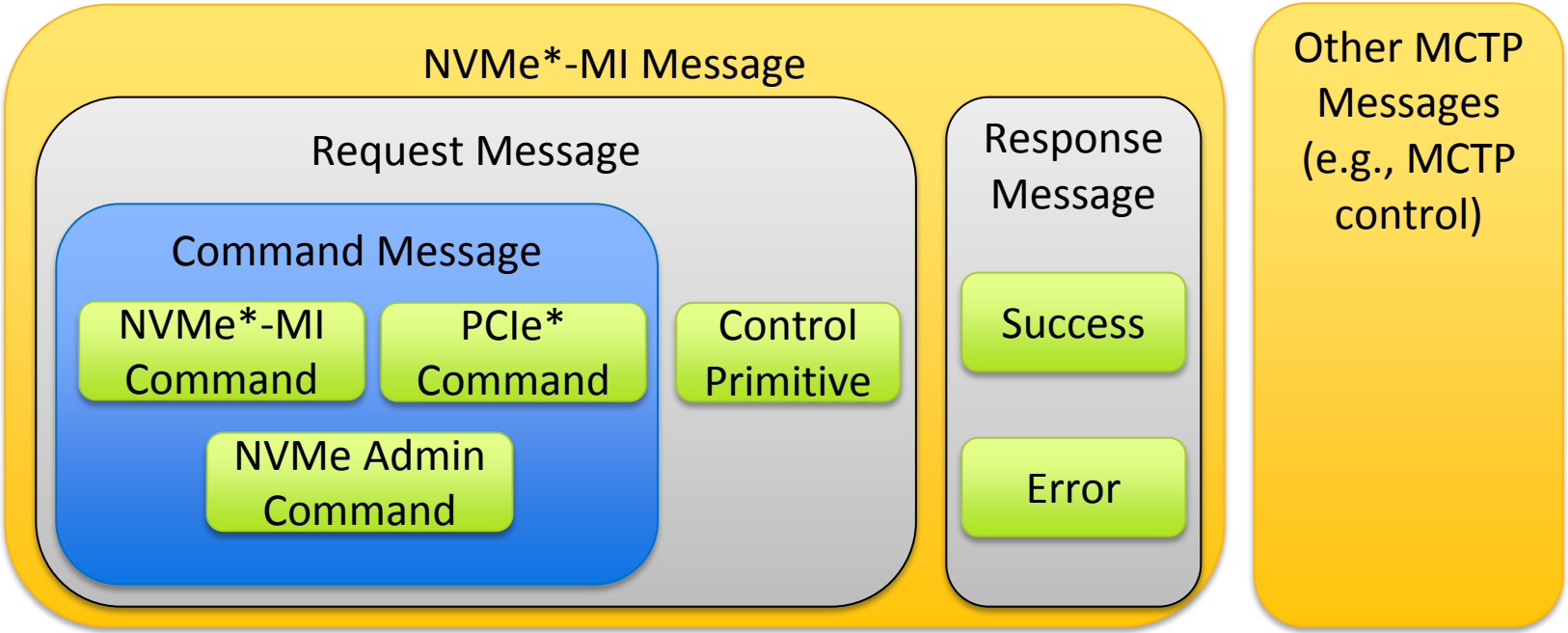


Protocol Layering

- MCTP defines the transport layer
  - Refer to http://dmtf.org/ for more info on MCTP
- NVMe* defines:
  - MCTP messages for BMC to NVMe out-of-band communication
  - Additional flow control and exception handling on top of MCTP



Types of MCTP Messages



Management Interface defines a few new commands while also leveraging existing NVMe Admin commands.

Management Interface Summary

- NVMe* Management Interface 1.0 focuses on drive level functionality (e.g., firmware update out-of-band)
- NVMe Management Interface 1.1 has kicked off and will focus on enclosure management
  - E.g., LEDs for an enclosure, etc

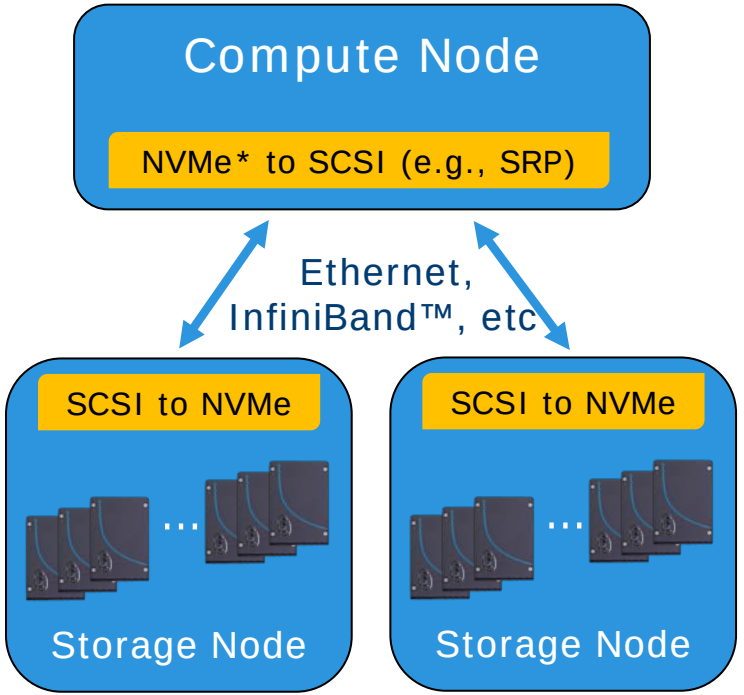
Support NVMe Management Interface 1.0 in your Datacenter devices. Get involved in NVMe Management Interface 1.1 definition.

Outline

- NVMe* Explained and The Past 5 Years
- NVMe Management Interface
- NVMe over Fabrics
- What is Coming in NVMe Revision 1.3

The Challenge

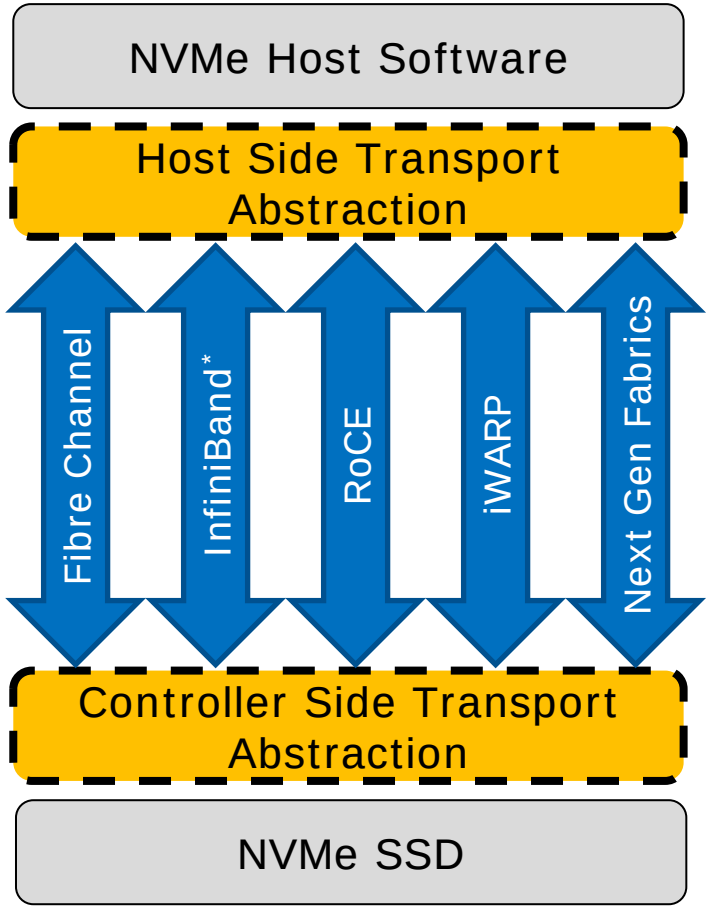
- Many Datacenters go from a compute node to a storage node across a fabric
- The back-end of many deployments is PCIe*-based NVM Express* SSDs
  - With next gen NVM SSDs, an end-to-end 4KB read may be achieved in < 10 µs
- But… Existing SCSI-based protocols (iSCSI, iSER, SRP) add ~ 100+ µs of latency



Challenge: Experience the benefit of NVMe SSDs throughout the Data Center.

NVMe* over Fabrics

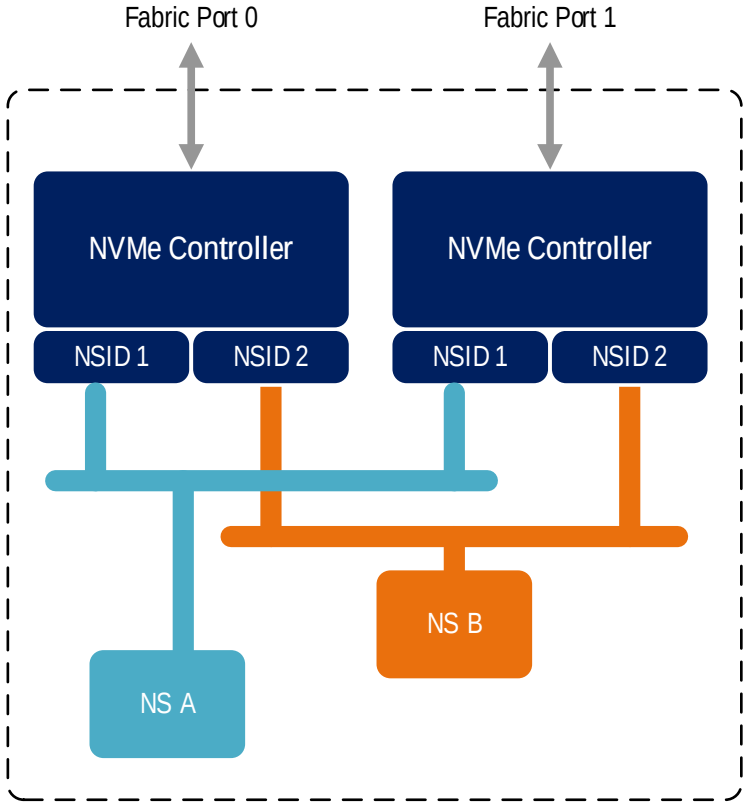
- Use NVMe* end-to-end to get the simplicity, efficiency and low latency benefits
- NVMe over Fabrics is a thin encapsulation of the base NVMe protocol across a fabric
  - No translation to another protocol (e.g., SCSI)
- NVMe over Fabrics 1.0 includes RDMA binding enabling Ethernet and InfiniBand™
  - INCITS T11 defining Fibre Channel binding



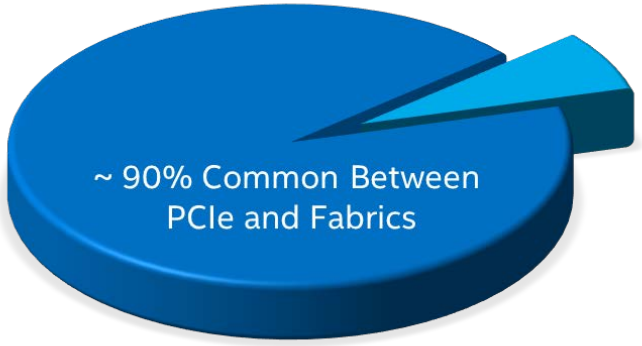
*Other names and brands may be claimed as the property of others.

Architectural Foundation

- Leveraging NVMe* building blocks
  - NVMe controllers
  - Multi-path and multi-host
  - Namespaces
  - Admin Queue and I/O Queues
  - NVMe command set
  - Scatter/Gather Lists



Differences



Differences	PCI Express* (PCIe*)	Fabrics
Identifier	Bus/Device/ Function	NVMe* Qualified Name (NQN)
Discovery	Bus Enumeration	Discovery and Connect commands
Queuing	Memory-based	Message-based
Data Transfers	PRPs or SGLs	SGLs only, added Key

The primary differences are in enumeration and the queuing mechanism.

NVMe* over Fabrics Released

- The NVMe* over Fabrics specification has been released – download at:
  http://nvmexpress.org/specifications
- A Linux host and target driver has been developed by the NVMe WG and released – download at:
  http://nvmexpress.org/resources/nvme-over-fabrics-drivers/
- Further details: Webinar by Mike Shapiro, VP Software @ EMC:
  https://www.brighttalk.com/webcast/12367/181249

Expect NVMe over Fabrics initial products later this year!
And learn MUCH more in the second part of this forum.

Outline

- NVMe* Explained and The Past 5 Years
- NVMe Management Interface
- NVMe over Fabrics
- What is Coming in NVMe Revision 1.3

NVMe* Revision 1.3

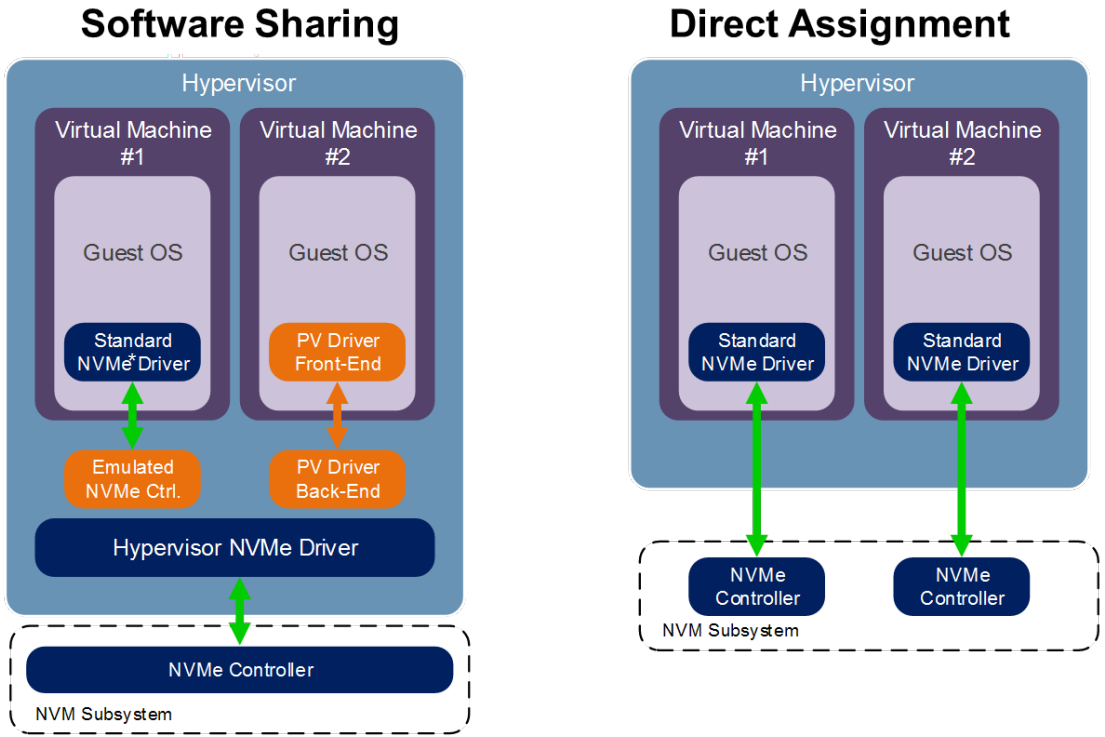
- Revision 1.3 is targeted for end of year
- The “anchor” features are Sanitize, Virtualization, and Directives
- And there is much more …

And here is a high level preview of Virtualization and Directives …

NVMe* 1.3 – Q4‘16

- Sanitize
- Virtualization
- Directives
- Thin Provisioning / Elastic Capacity
- Host Controlled Thermal Management
- Device Self-Test
- Timestamp
- Boot Partitions
- Error Log Enhancements
- Telemetry
- And more …

Options to Share an SSD

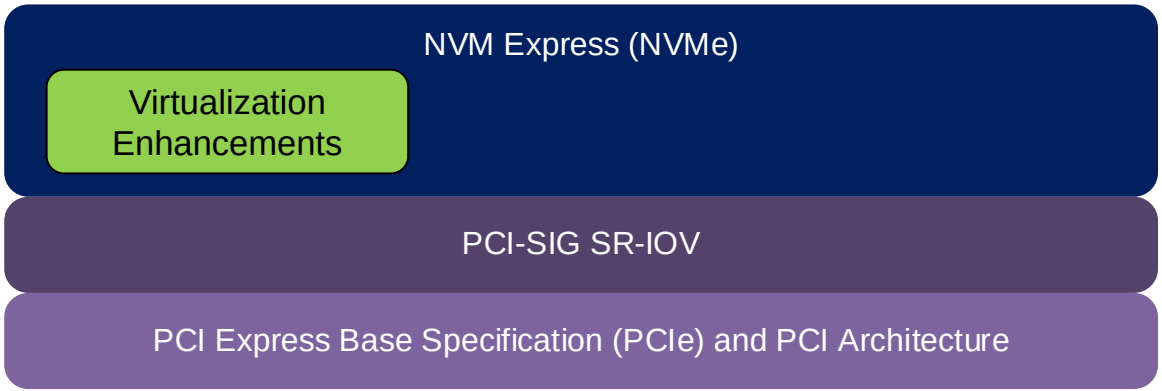


Challenge: Going through hypervisor software adds 20+ µs of latency.

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

Virtualization Enhancements

- Device specific functionality is outside the scope of SR-IOV
  - SR-IOV defines PCIe* related behavior only
- Virtualization Enhancements define:
  - How to allocate interrupt and queue resources to a Virtual Function (VF)
  - How to measure and allocate performance (bandwidth) to a VF
  - Physical Function (PF) Interface: Allowing a standardized NVMe* PF driver to setup VFs

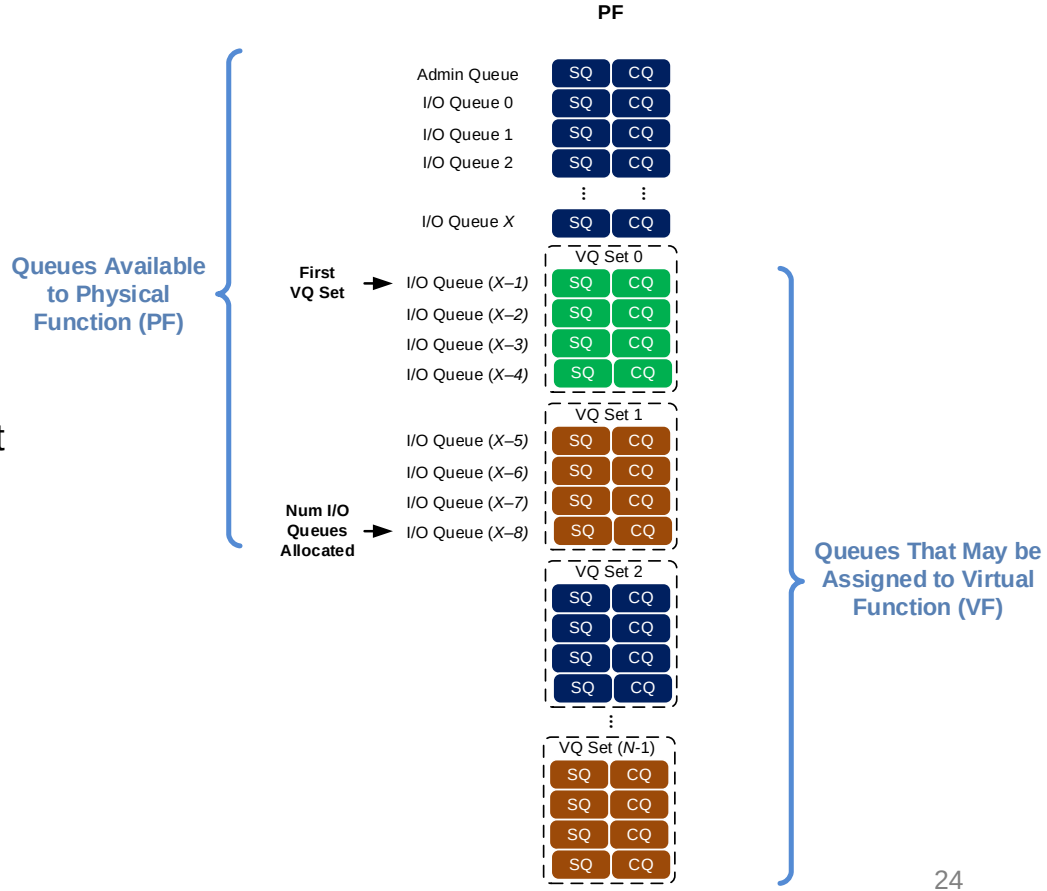


Allocating Resources

- Resources may be moved between the PF and VF(s)

VQ Set – A set of (four) Submission Queue (SQ) and Completion Queue (CQ) pairs that may be assigned to a VF

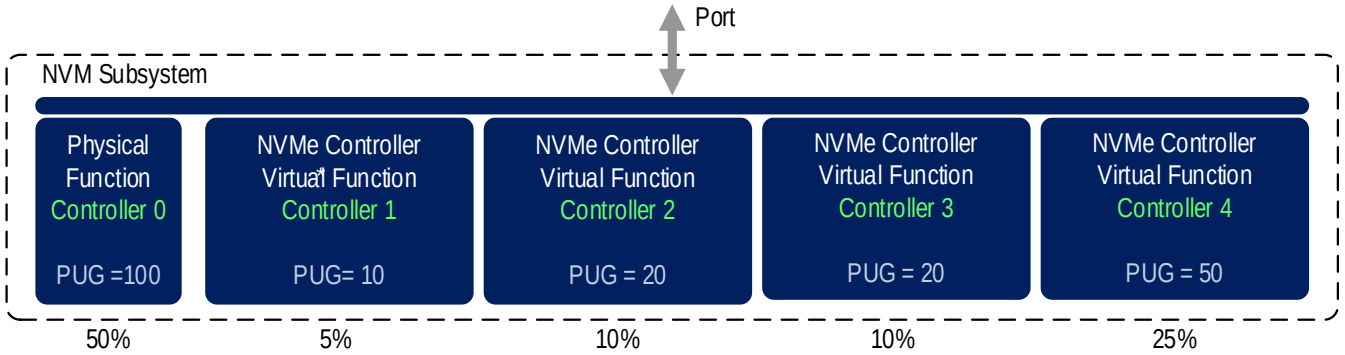
VI Set – A set of (four) MSI-X interrupt resources that may be assigned to a VF



Performance Partitioning

- A key challenge is the “noisy neighbor” – need to ensure that one VF does not consume all of the bandwidth or IOPs
- Performance Unit Grant (PUG) – performance grant relative to other controllers in the NVM subsystem

$$\text{Controller x Relative Performance} = \frac{PUG_x}{\sum_{i=0}^{N-1} PUG_i}$$



Host/Device Information Exchange

- Host to device interaction is needed for the best use of NVM
- Info exchange may happen at different times and on different “targets”

Time	Target	Example
Before Access	Logical Block Address	Dataset Management: LBA Range Context Attributes
	Physical Media or Partition	
	Access Relationship	Dataset Management: Integral Dataset for Read/Write
During Access	Logical Block Address	Read/Write: Dataset Management hints
	Physical Media or Partition	
	Access Relationship	Stream Identifier
After Access	Logical Block Address	Dataset Management: Deallocate (Trim)
	Physical Media or Partition	Host Initiated Garbage Collection
	Access Relationship	Host Initiated Garbage Collection

Directives Overview

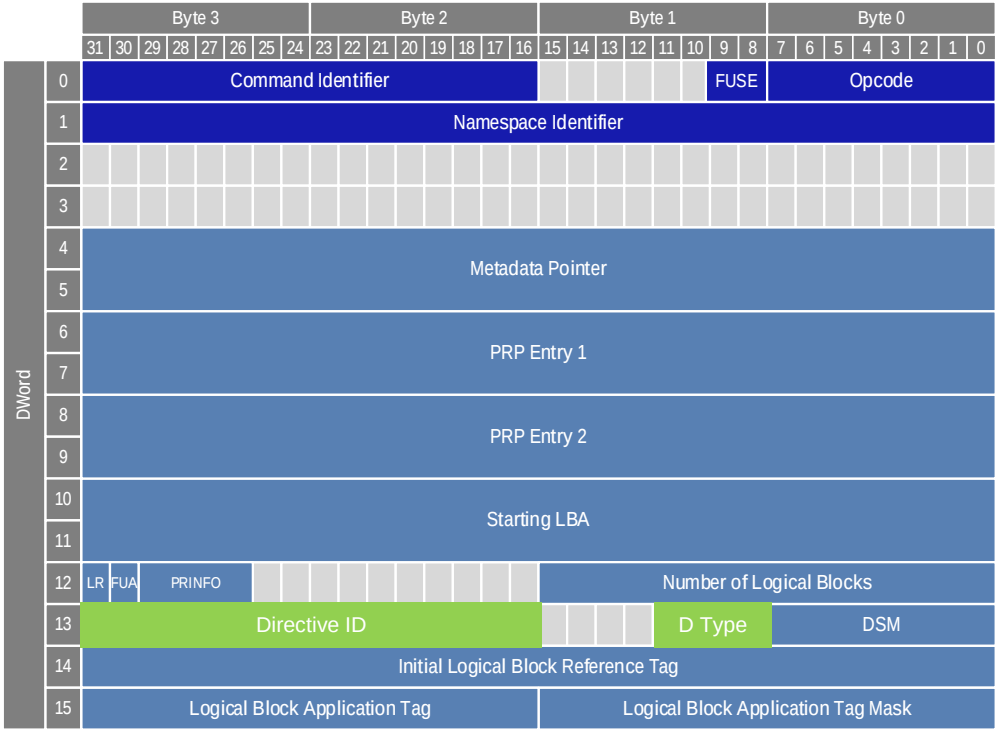
- Define **Directives** as a first class element of NVMe* for host/device info exchange
- Up to 256 Directive Types including Streams and Advanced Background Control
- Base commands: Directive Receive and Directive Send
- Each Directive defines operations valid with Directive Receive/Send
- A Directive Identifier is supported on a per command basis
  - Enables a read or write to be tagged with information (as in the case of Streams)

Figure 5.7_3: Directive Receive – Command Dword 11

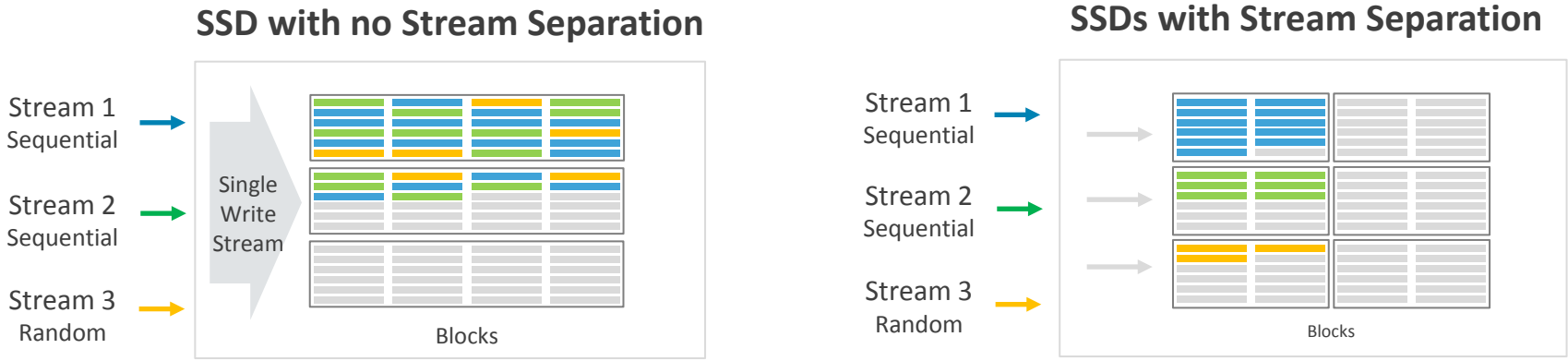
Bit	Description
31:16	**Directive Identifier (DIDENT):** This field specifies the Directive Identifier. The interpretation of this field is Directive Type dependent. Refer to section 9.
15:08	**Directive Type (DTYPE):** This field specifies the Directive Type. Refer to Figure 9_1 for the list of Directive Types.
07:00	**Directive Operation (DOPER):** This field specifies the Directive Operation to perform. The interpretation of this field is Directive Type dependent. Refer to section 9.

Enabling Future Enhancements

- Streams uses 16-bits in Write commands to identify stream
- NVMe* commands have little available space …
- Make re-useable Directive ID / Directive Type field
- ID can be used for Streams today and future ideas tomorrow



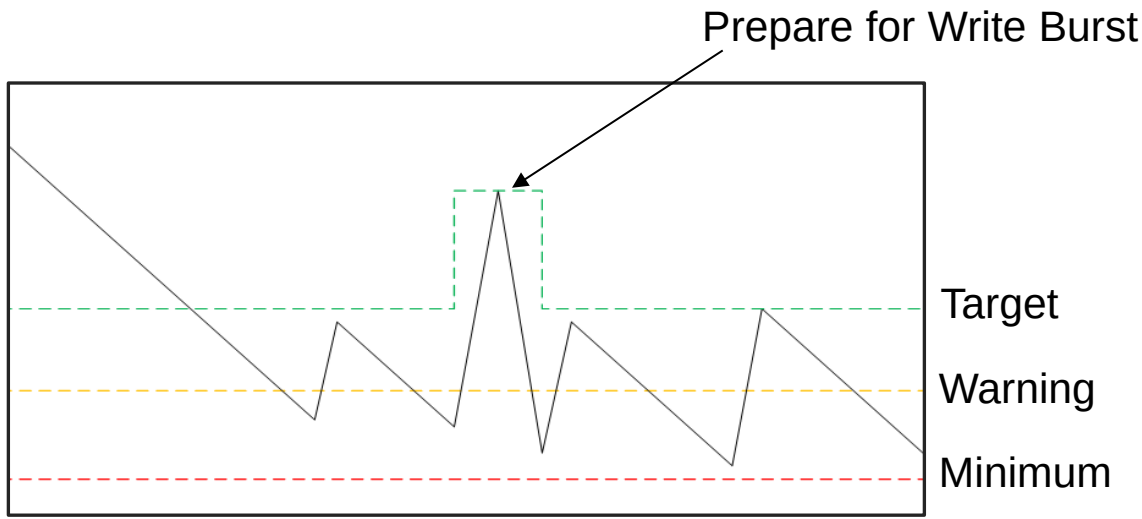
Streams Concept



Streams can reduce write amplification for host managed workloads.

Advanced Background Control

- Used by the device to indicate ahead of time when garbage collection required
- Can also be used by the host to dynamically configure write allocation resources



Learn More Today

Forum A-11: NVMe* over Fabrics Panel – Which Transport is Best?

- Introduction to NVMe over Fabrics and Linux Software Stack
- NVMe over RDMA with Ethernet
- NVMe over RDMA with InfiniBand™
- NVMe over Fibre Channel

Forum A-12: NVMe Drivers – Current Status and Future Expectations

- Updates from Microsoft, VMware, Intel, and Seagate

Forum A-12: Lessons Learned Deploying NVMe in Real Systems

- Perspectives from Dell, E8 Storage, Intel, NetApp, and Facebook

The NVMe Workgroup has delivered continual innovation the last five years. Learn more details from industry leaders today.

Legal 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 **www.intel.com.**

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

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 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.

Q&A

Thank You!