The Evolution and Future of NVMe™

J Metz, Ph.D
R&D Engineer, Advanced Storage
Cisco Systems, Inc.
@drjmetz

What This Presentation Is… and Isn't

This presentation is:

A cursory overview of the status of the development of NVMe™, NVMe Management Interface, and NVMe over Fabrics™

This presentation is not:

Comprehensive

Carved in stone

Some of these topics are subject to change!

Evolution of NVMe™

2011
- NVM Express Specification 1.0 published by industry leaders on March 1

2012
- NVM Express Specification 1.1 released on October 11

2014
- NVM Express Specification 1.2 released on November 3
- NVM Express Work Group was incorporated at NVM Express, Inc., the consortium responsible for the development of the NVM Express specification
- Work on the NVM Express over Fabrics (NVMe-oF™) Specification kicked-off

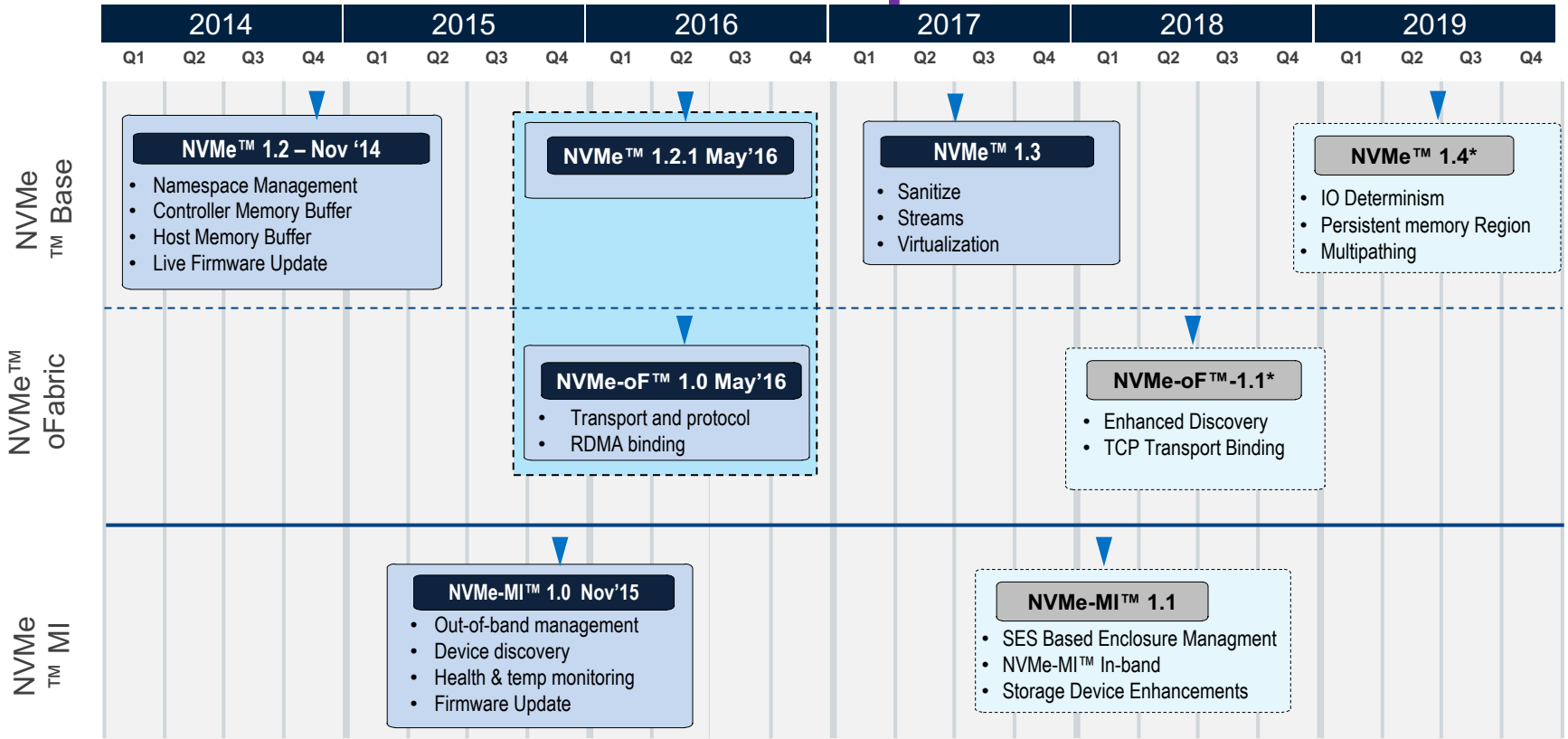
2015
- NVM Express Management Interface (NVMe-MI™) Specification officially released. Provides out-of-band management for NVMe™ components and systems and a common baseline management feature set across all NVMe™ devices and systems.

2016
- NVM Express over Fabrics (NVMe-oF™) Specification published; extending NVMe™ onto fabrics such as Ethernet, Fibre Channel and InfiniBand®, providing access to individual NVMe™ devices and storage systems.

2017
- NVM Express Specification 1.3 published. Addresses the needs of mobile devices, with their need for low power consumption and other technical features, making it the only storage interface available for all platforms from mobile devices through data center storage systems.

So...what's next?

NVMe™ Feature Roadmap



The Future of NVMe™

NVMe™ 1.4
- IO Determinism
- Persistent Controller Mem Buffer and Event Log
- Multipathing

NVMe-MI™ 1.1
- SCSI Enclosure Services (SES)
- NVMe-MI™ In-band
- Native Enclosure Management

NVMe-oF™ 1.1
- Enhanced Discovery
- TCP Transport Binding

6

NVMe Roadmap - Continuous Improvements

Ever-Advancing Performance and Features

NVMe™ 1.4*

- I/O Determinism
- Persistent memory Region
- Multipathing

- Data latency
  - Improvement: I/O Determinism (IOD)
- High Performance Non-Volatile data needs
  - Improvement: Persistent Memory Region
- Ease of Data sharing
  - Improvements: Multi-Pathing access

Management Needs

NVMe-MI™ 1.1

- SES Based Enclosure Management
- NVMe-MI™ In-band
- Storage Device Enhancements

- Standardized Management for ease of adoption
  - Industry standard tools and compliance
- Improvements and updates to managing the subsystems and end devices
  - Event logging
  - Incorporating robust industry adopted enclosure management
  - Diverse connections to end devices (SSDs)
  - Additional In-band mechanisms

SDC 18

Enterprise Networking Needs

NVMe-oF™-1.1*
- Enhanced Discovery
- TCP Transport Binding

- Robustness in networking topologies
  - Congestion Management
- New and interesting transport capabilities
  - TCP bindings for NVMe-oF™
- Improvements in automation
  - Discovery
- Security Enhancements
  - In-band authentication

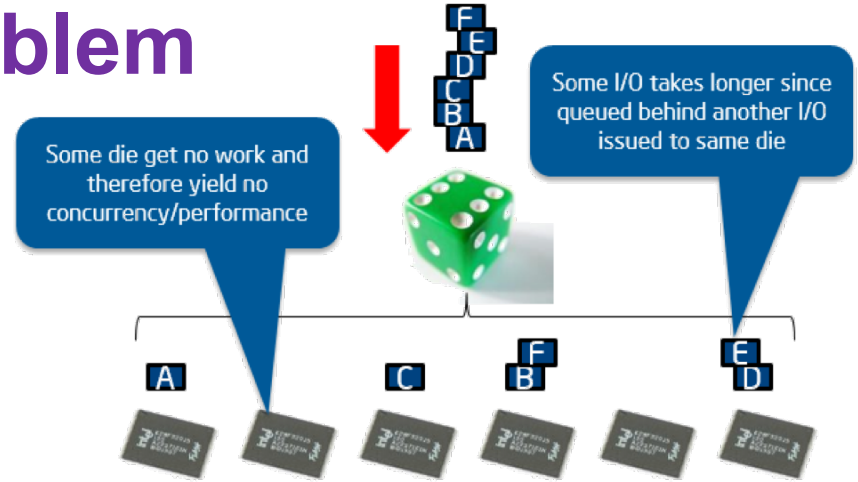
NVMe™ 1.4

Project Completion: 2019

SDC 18

The I/O Determinism Problem

- Customers are demanding max (tail) latency controls (e.g., 99.99%) with NAND Flash today
  - New low latency Storage Class Memory makes this much more demanding
- To meet current and future needs, need host awareness of “units of parallelism”
  - Avoid I/O collisions inside SSD
- Also need to minimize and control SSD background operations
  - Can’t get 99.99% tail latency control without this



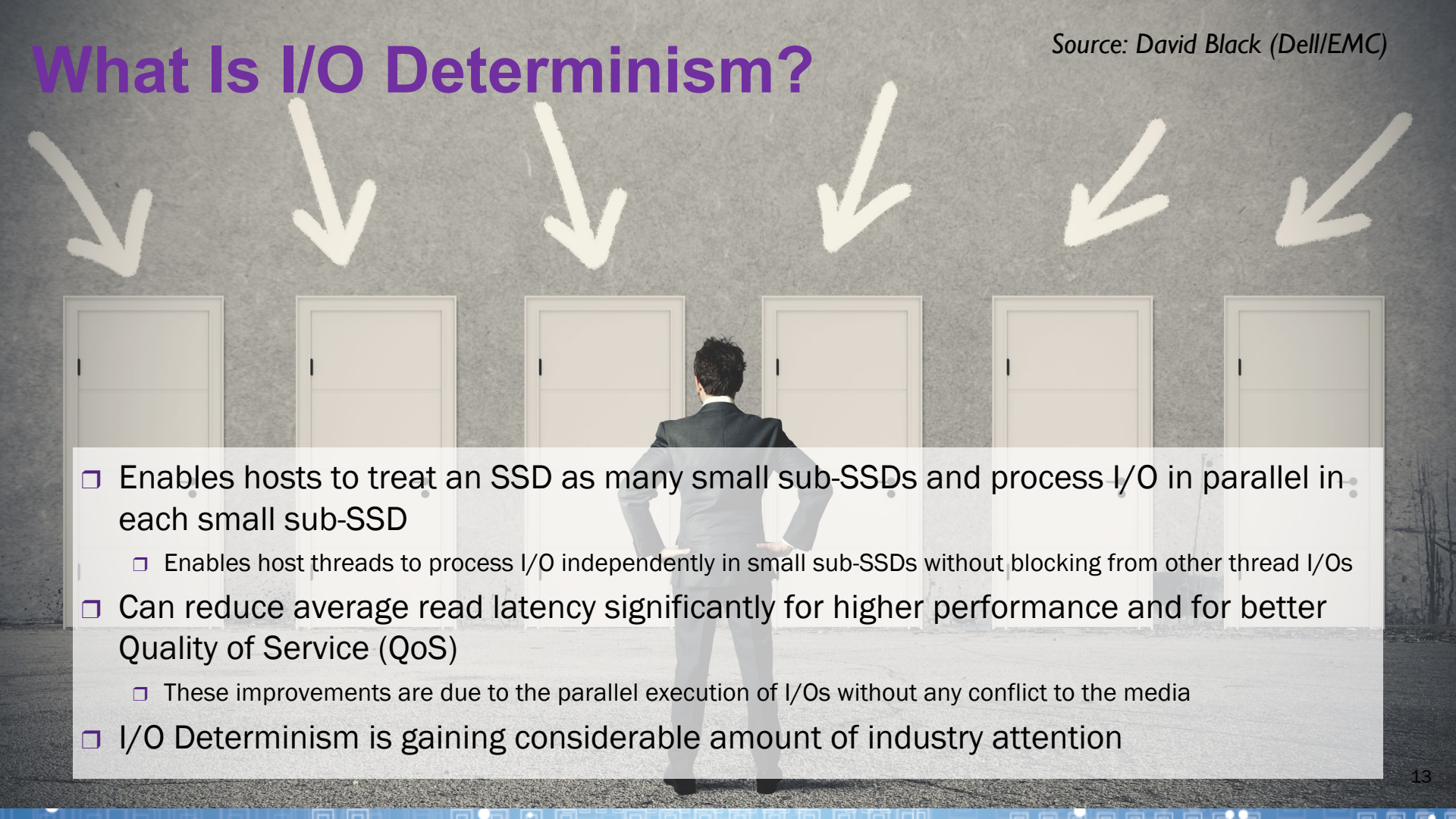
Yahtzee Effect: Statistical Clumping

When rolling 6 dice with 6 faces, on average only 4 of the 6 values will come up

(Odds of all 6 values appearing: ~1.5%)

What Is I/O Determinism?

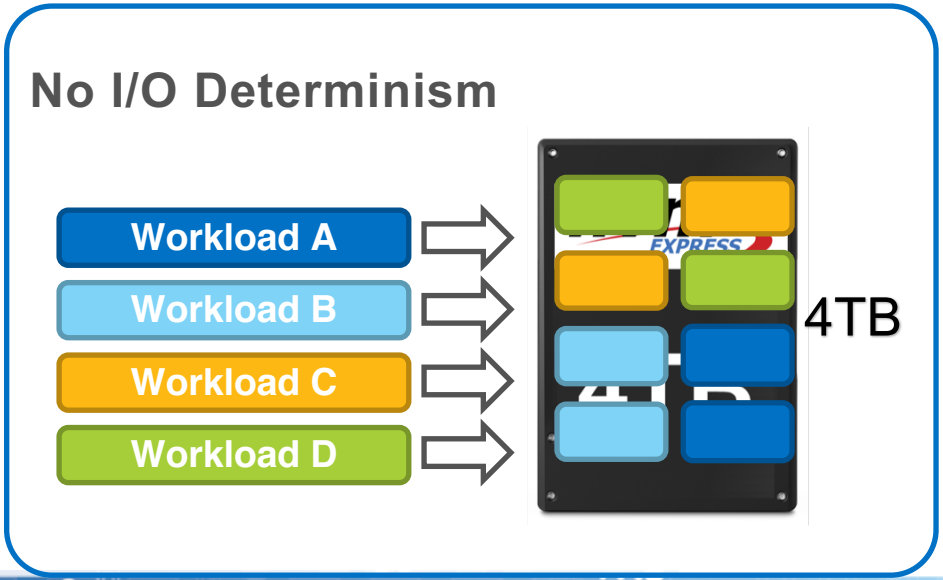
Source: David Black (Dell/EMC)

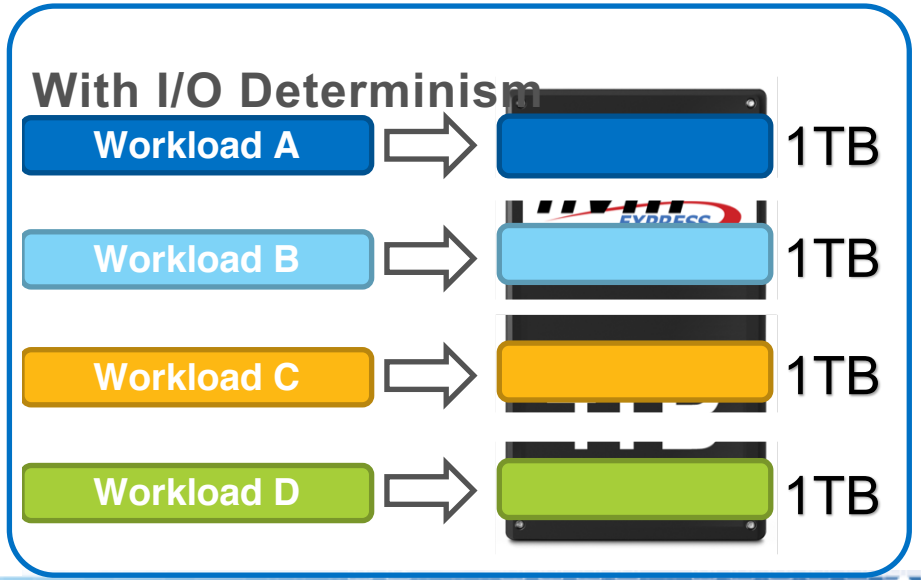


- Enables hosts to treat an SSD as many small sub-SSDs and process I/O in parallel in each small sub-SSD
  - Enables host threads to process I/O independently in small sub-SSDs without blocking from other thread I/Os
- Can reduce average read latency significantly for higher performance and for better Quality of Service (QoS)
  - These improvements are due to the parallel execution of I/Os without any conflict to the media
- I/O Determinism is gaining considerable amount of industry attention

What Is I/O Determinism? (cont.)

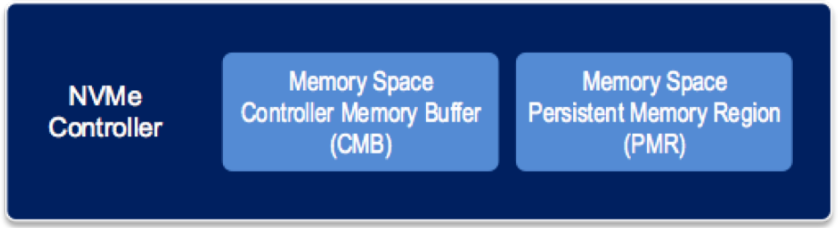
- Service isolation region
- Increase Read I/OPs and reduce max latency
- Provides strict QoS profile
- Significantly improves P99 and P9999 for a well-behaved host





Persistent Memory Region (PMR)

- Controller Memory Buffer (CMB)
  - Introduced in NVMe™ 1.2
  - PCI memory space exposed to host
  - May be used to store commands and command data
  - Contents do not persist across power cycles and resets

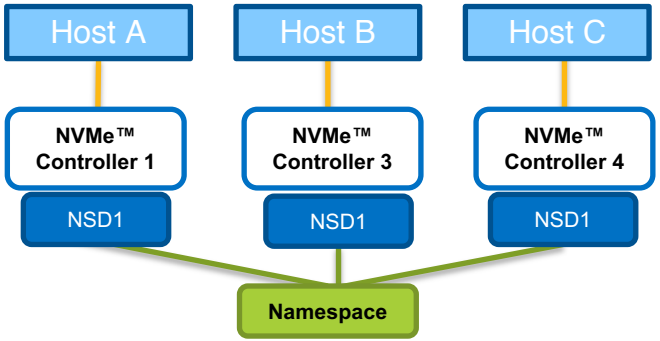


- Persistent Memory Region (PMR)
  - PCI memory space exposed to host
  - May be used to store command data
  - Content persist across power cycles and resets

NVMe™ Multipathing and Namespace Sharing

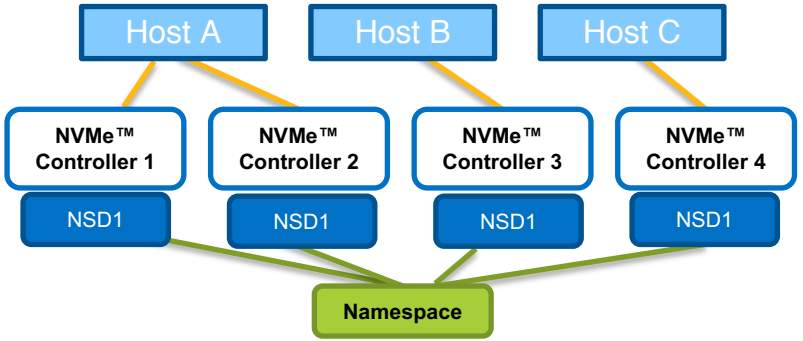
Technical Term: Asymmetric Namespace Access (ANA)

NVMe™ Multipathing I/O refers to two or more completely independent PCI Express paths between a single host and a namespace



NVMe™ Multipathing

Namespace sharing enables two or more hosts to access a common shared namespace using different NVM Express controllers

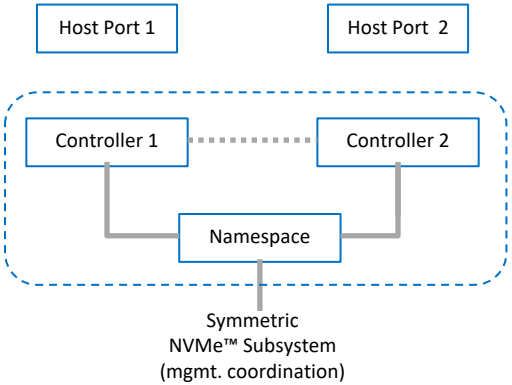


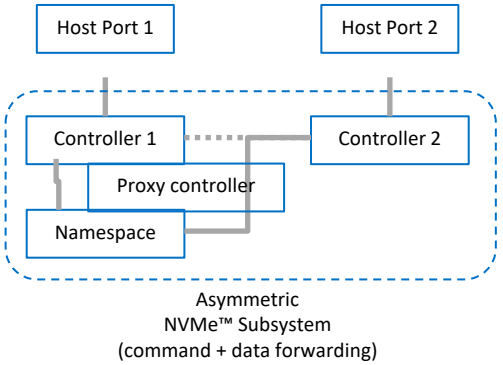
Namespace Sharing

Both multi-path I/O and namespace sharing require that the NVM subsystem contain two or more controllers

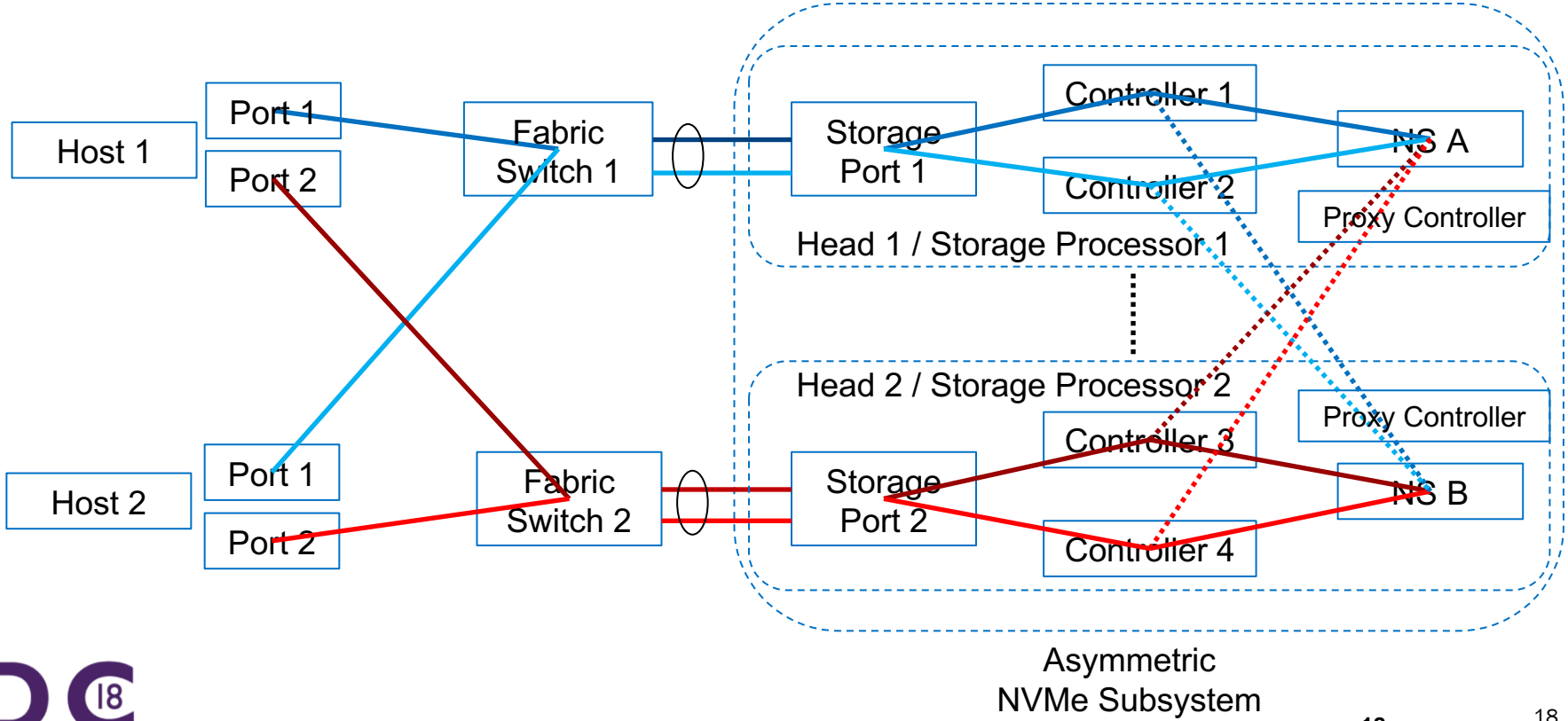
Multipathing-Related Capabilities

- Symmetric Multipathing already exists in the NVMe™ specification
  - Symmetric access: The same access characteristics on all paths
    - The host doesn't care which path is used – because they are all the same
  - Identify controller data (CMIC field)
  - Identify namespace data (NMIC field)
- Fabrics provide additional opportunities for multiple asymmetric paths
  - Asymmetric access: Different access characteristics on different paths
    - The host cares which path is used – because they are NOT all the same
  - New capabilities to be added to the NVMe™ specification





Example: Fully-Redundant Fabric and Array



NVMe Management Interface (NVMe-MI™) 1.1

Project Completion: 2018

NVMe-MI™ 1.1 Key Work Items

NVMe-MI™ 1.1

- SES Based Enclosure Management
- NVMe-MI™ In-band
- Storage Device Enhancements

- SCSI Enclosure Services (SES) Based Enclosure Management
  - Draft completed, working through final technical items
  - SCSI translation (completed)
- Support for In-Band NVMe-MI™
  - Draft complete and in workgroup review
- NVMe™ Storage Device Enhancement – In work

Enclosure Management

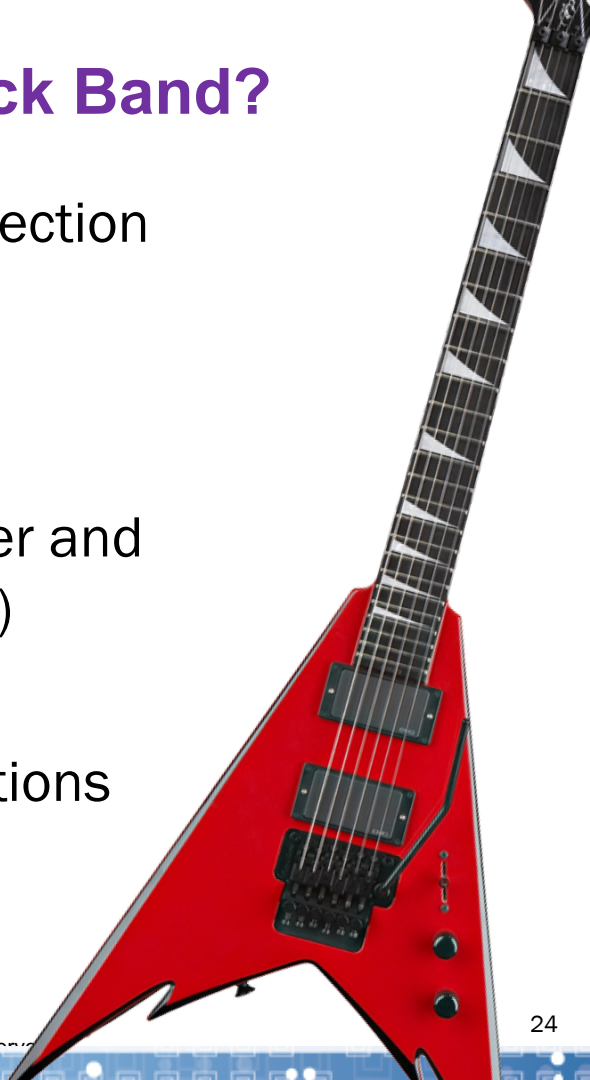
- Native PCIe Enclosure Management (NPEM)
  - Transport specific basic enclosure manager
  - Submitted to the PCI-SIG Protocol Workgrou on behalf of the NVMe™ Management Inter Workgroup
  - Approved by PCI-SIG on August 10, 2017
- SES Based Enclosure Management
  - Technical proposal being developed in NVM workgroup
  - Comprehensive enclosure management

SES-Based Enclosure Management

- SCSI Enclosure Services (SES) is a standard developed by T10 for management of enclosures using the SCSI architecture
- While the NVMe and SCSI architectures differ, the elements of an enclosure and the capabilities required to manage these elements are the same
  - Example enclosure elements: power supplies, fans, display or indicators, locks, temperature sensors, current sensors, voltage sensors, and ports
- NVMe-MI leverages SES for enclosure management
  - SES manages the elements of an enclosure using control and status diagnostic pages transferred using SCSI commands (SCSI SEND DIAGNOSTIC & SCSI RECEIVE DIAGNOSTIC RESULTS)
  - NVMe-MI uses these same control and status diagnostic pages, but transfers them using the SES Send and SES Receive commands

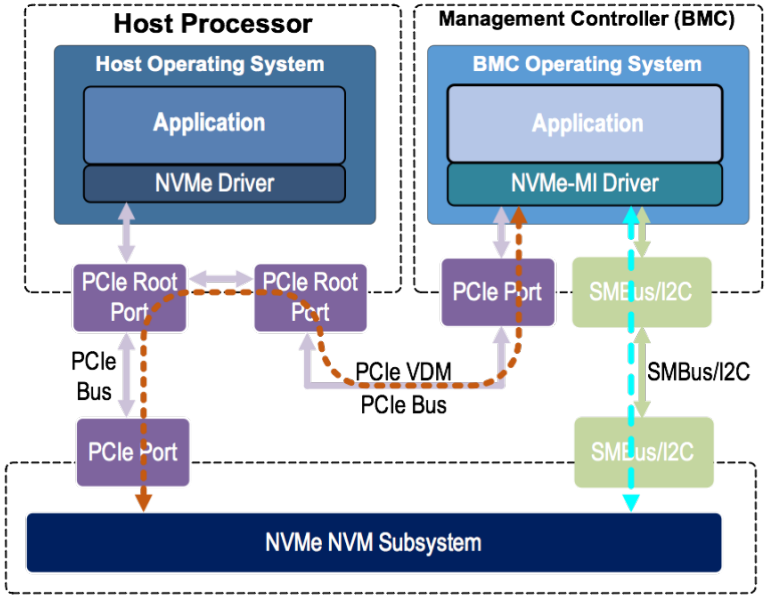
Management – In-Band? Out-of-Band? Rock Band?

- Out-of-Band Management – Independent connection separate from the main IO path and operation system (Usually SMBus/I2C physical interface)
- In-Band Management – Utilizes the NVMe driver and the main data path interface (Usually PCIe Bus)
- Provides “Rockin” NVMe-MI management solutions and flexibility of implementations

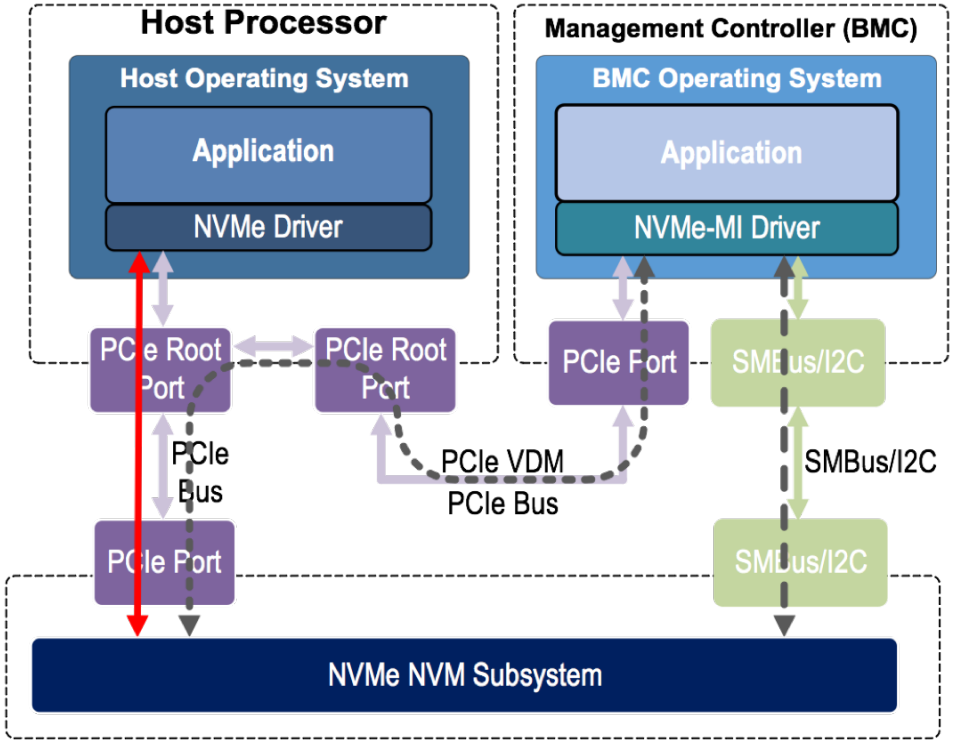


NVMe-MI™ Out-of-Band Management

- Out-of-Band Management – Management that operates with hardware and components that are independent of the operation system control
- NVMe™ Out-of-Band Management Interfaces
  - SMBus/I2C
  - PCIe Vendor Defined Messages (VDM)
  - IPMI FRU Data (VPD) accessed over SMBus/I2C



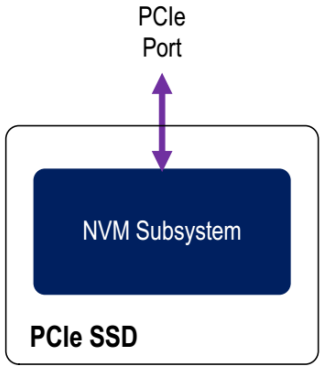
In-Band Management and NVMe-MI™



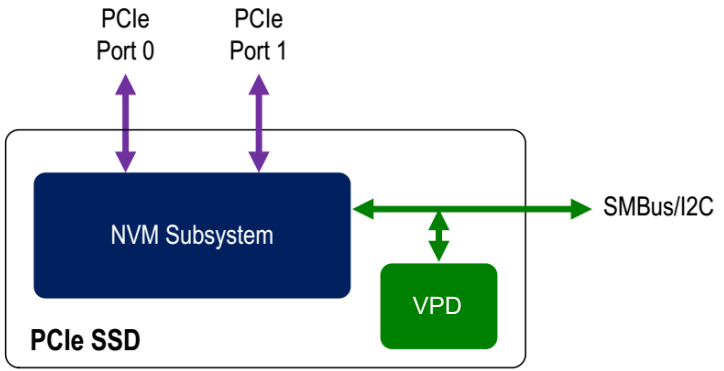
- In-band mechanism allows application to tunnel NVMe-MI™ commands through NVMe™ driver
  - Two new NVMe™ Admin commands
    - NVMe-MI™ Send
    - NVMe-MI™ Receive
- Benefits
  - Provides management capabilities not available in-band via NVMe™ commands
    - Efficient NVM subsystem health status reporting
    - Ability to manage NVMe™ at a FRU level
    - Vital Product Data (VPD) access
    - Enclosure management

NVMe-MI 1.0a Storage Device Management

- NVM Storage Device – One NVM Subsystem with one or more ports and an optional SMBus/I2C interface

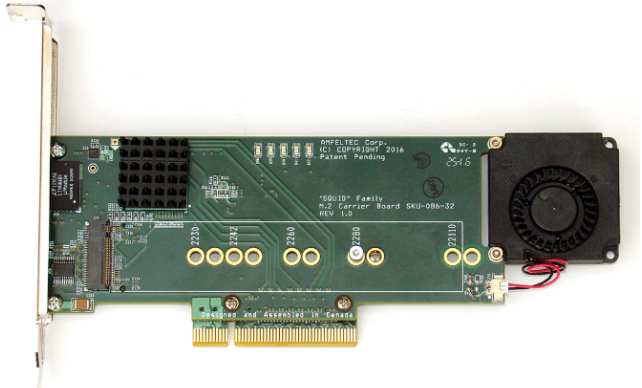


Single Ported PCIe SSD



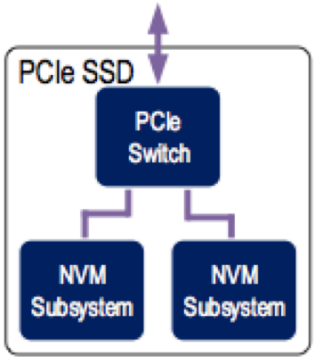
Dual Ported PCIe SSD with SMBus/I2C

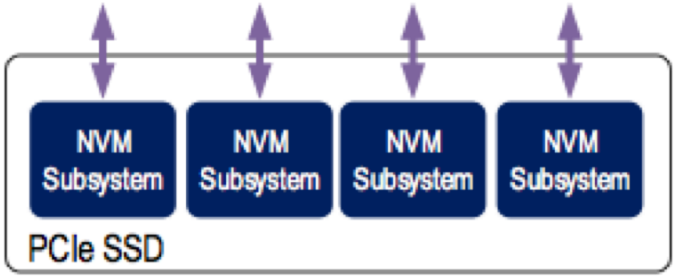
NVMe Storage Device with Multiple NVM Subsystems



M.2 Carrier Board from Amfeltec

ANA Carrier Board from Facebook





NVMe™ over Fabrics 1.1

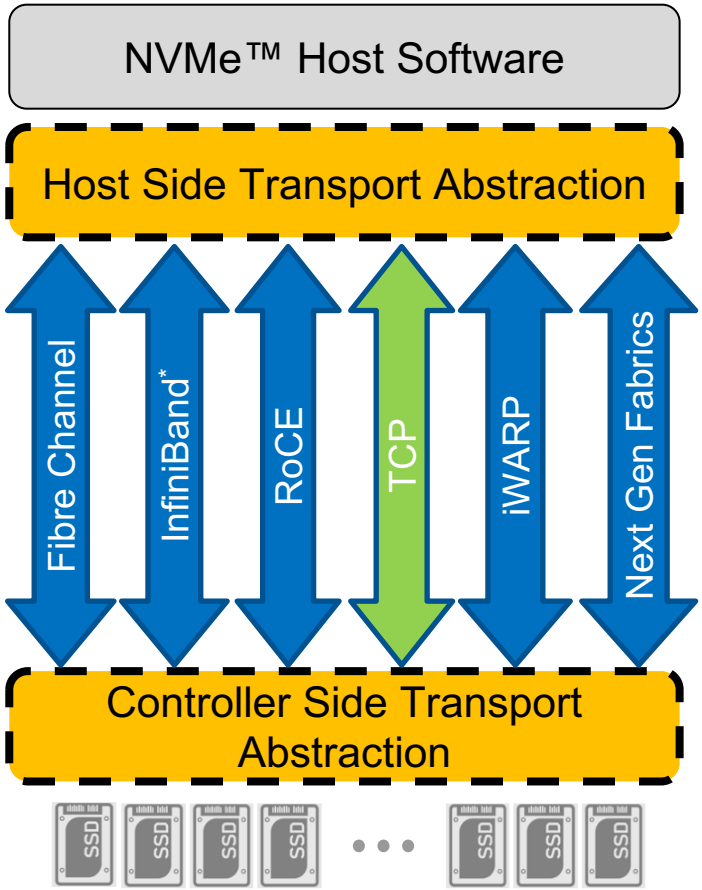
Project Completion: 2018

Quick and Dirty Topics

- NVMe-TCP
- Discovery

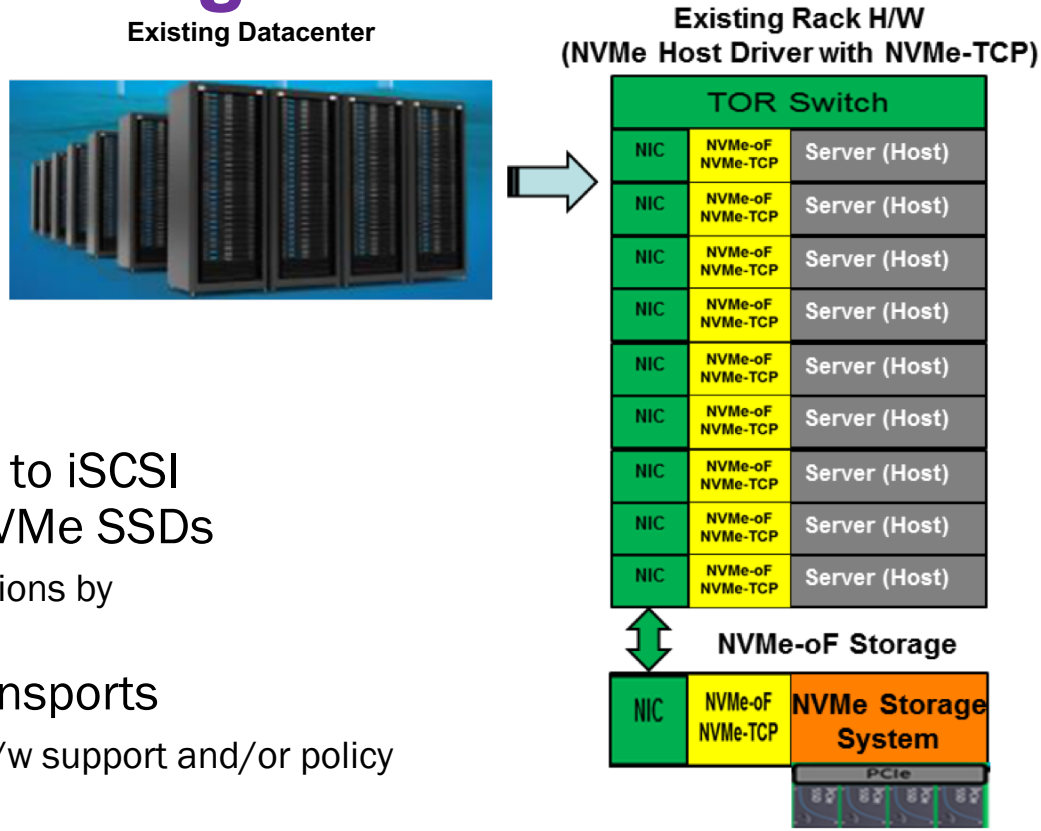
NVMe-TCP

- NVMe™ block storage protocol over standard TCP/IP transport
- Enables disaggregation of NVMe™ SSDs *without compromising latency and without requiring changes to networking infrastructure*
- Independently scale storage & compute to maximize resource utilization and optimize for specific workload requirements
- Maintains NVMe™ model: sub-systems, controllers namespaces, admin queues, data queues



NVMe-TCP Data Path Usage

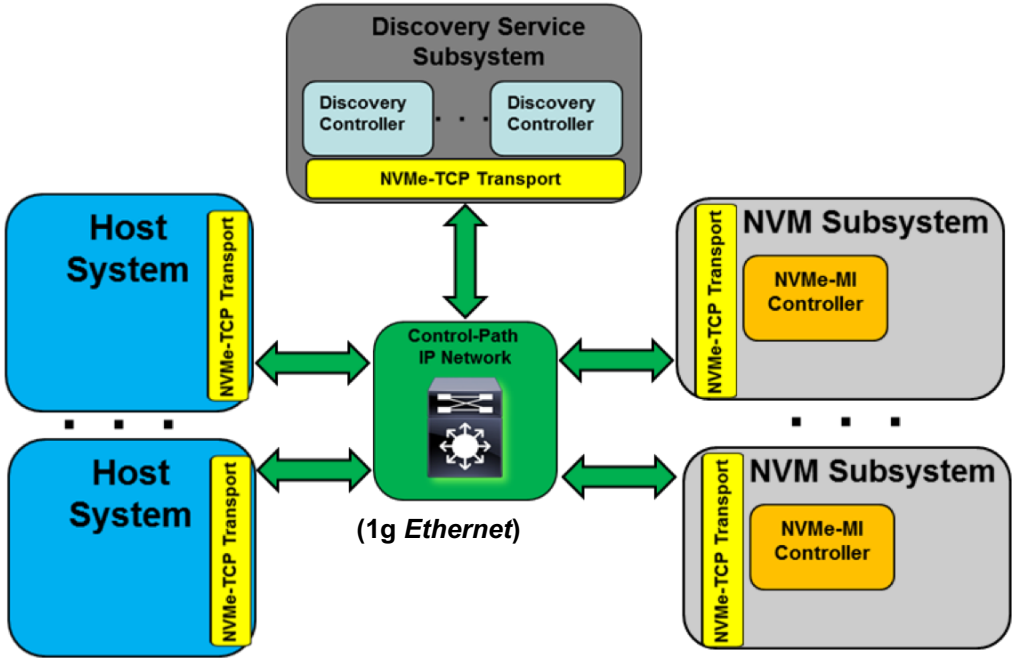
- Enables NVMe-oF I/O operations in existing IP Datacenter environments
  - Software-only NVMe Host Driver with NVMe-TCP transport
- Provides an NVMe-oF alternative to iSCSI for Storage Systems with PCIe NVMe SSDs
  - More efficient End-to-End NVMe Operations by eliminating SCSI to NVMe translations
- Co-exists with other NVMe-oF transports
  - Transport selection may be based on h/w support and/or policy



Source: Dave Minturn (Intel)

NVMe-TCP Control Path Usage

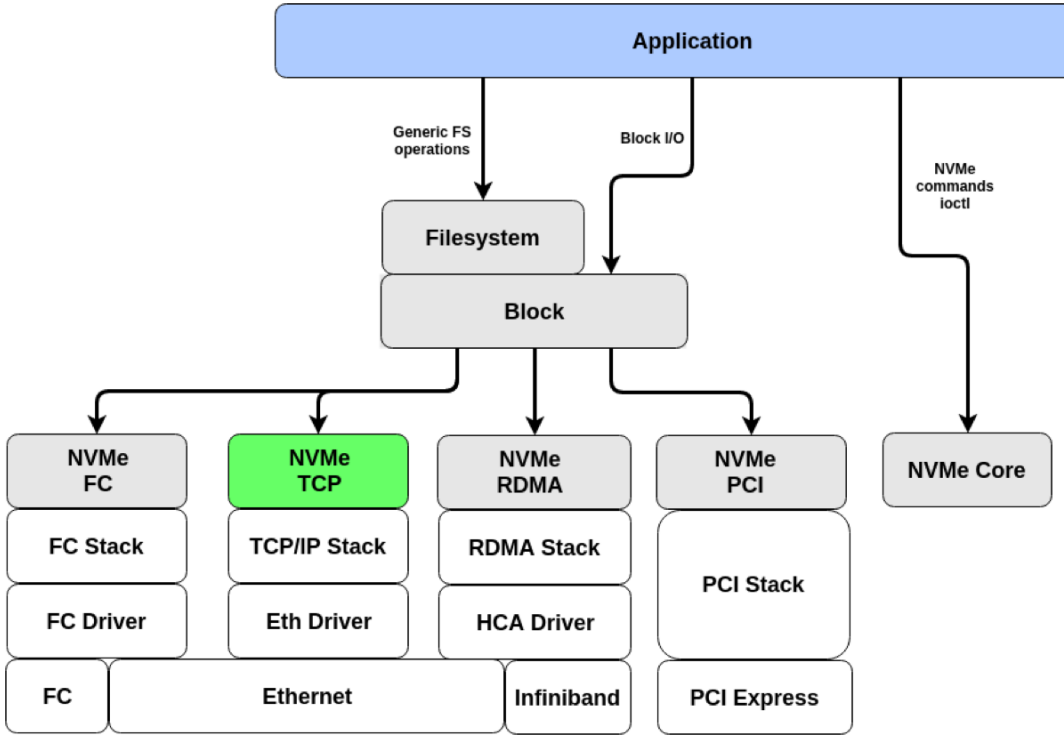
- Enables use of NVMe-oF on Control-Path Networks (example: 1g Ethernet)
- Discovery Service Usage
  - Discovery controllers residing on a common control network that is separate from data-path networks
- NVMe-MI Usage
  - NVMe-MI endpoints on control processors (BMC, ..) with simple IP network stacks
  - NVMe-MI on separate control network



Source: Dave Minturn (Intel)

NVMe-TCP Standardization

- Expect NVMe over TCP standard to be ratified in 1H 2018
  - The NVMe-oF 1.1 TCP layer ballot passed in April 2017
  - NVMe Workgroup adding TCP to spec alongside RDMA
  - Lead by Lightbits, Facebook, Intel, and other industry leaders



Source: Kam Eshghi (Lightbits Labs)

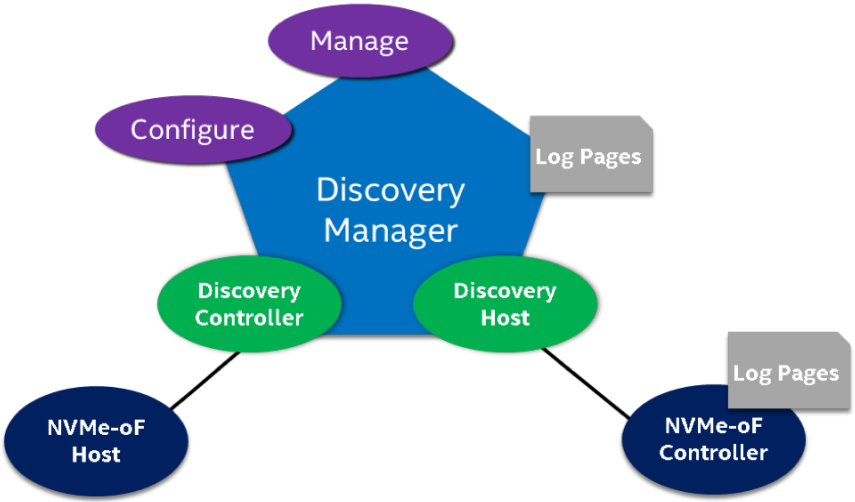
Current State of NVMe-oF Discovery and Management (Linux)

- NVMe-oF architecture and drivers enable high performance access to statically-defined and statically-configured remote NVMe resources.
- The current implementation of NVMe-oF Host and Remote Controller Linux driver stacks today rely on manual configuration and provisioning to remote NVMe controllers
  - Hosts are manually configured with the remote controllers to which they connect
  - Remote Controllers are either manually or statically configured with how local NVMe devices are represented on a fabric to specific Hosts based on provided NQN
- The only *discovery* defined in the current specification is how the remote NVMe controller informs Hosts of the representation of the subsystems they have access to on the fabric being used

Issues with NVMe-oF Discovery and Management

- The current NVMe-oF specification and Linux implementation lacks
  - Dynamic resource discovery and enumeration of remote resources
  - Clear definition for methods of how to discover the proper discovery controller defining remote storage resource provisioning
- To support large-scale deployment of NVMe-oF, more is needed
  - Specification enhancement for efficient, dynamic resource management
  - Fabric-transport specific mechanisms to determine where to get provisioning information from
  - Linux kernel driver stack changes as the specification evolves
  - Management tools to enable NVMe-oF management and scale-out

Enhanced Discovery



Simple solution within the confines of the current specification

1. Centralized Discovery Manager (CDM) – a generic controller – configured to know about every NVMe-oF Controller
2. NVMe-oF Controllers configured to allow the CDM access to all configured NVMe-oF subsystems for discovery purposes
3. Hosts configured to connect to and query the CDM to determine for NVMe-oF subsystems provisioned to that Host
4. Hosts use the discovery log pages provided by the CDM to connect directly to remote NVMe-oF controllers

Possible Specification Changes

- Asynchronous notification of updates to Hosts specific Discovery Log Pages – being worked as an approved NVMe-oF TPAR.
- Asynchronous notification of fabric availability and status
- Host->remote-controller utilization statistics (MI specification)
- Remote configuration of NVMe Subsystems presented on the fabric (MI specification)
- Log pages informing the CDM of local NVMe devices that may be remotely provisioned to fabrics

NVMe™ over Fabrics 1.1

Project Completion: 2018

- NVMe has over 50 ongoing projects in the technical working groups
- There are over 130 participating companies working on these projects
- Key improvements to the NVMe base spec, Fabrics, and Management will help facilitate more robust, resilient, and powerful tools for data centers.
- Special Thanks:
  - David Allen (Seagate)
  - Brandon Hoff (Broadcom)
  - David Black (Dell/EMC)
  - Fred Knight (NetApp)
  - Kam Eshghi (Lightbits Labs)
  - Phil Cayton (Intel)
  - Dave Minturn (Intel)
  - Peter Onufryk (Microsemi)

Thank you!