DISTRIBUTED ENDPOINT MANAGEMENT
AN NVME-OF™ SCALE-OUT MANAGEMENT SOLUTION

Phil Cayton
Intel Corporation
March, 2019
AGENDA

- NVMe over Fabrics (NVMe-oF) Overview
- Current State of NVMe-oF Management and Administration
- Distributed Endpoint Management (DEM) Project
- Brief Demonstration
- Development Opportunities and Wrap-up
NVMe: industry standard interface and storage protocol for PCIe SSDs

- High-performance, low-latency PCI SSD interface
- Eliminates unnecessary protocol translations (i.e., SCSI)
- Defines partitioning PCIe SSDs into one or more subsystems

NVMe-oF: extends NVMe efficiency over Fabrics

- Builds on base NVMe architecture with thin encapsulation of base NVMe across a fabric
- Enables low-latency and high IOPS access to remote NVMe storage
- Defines end-to-end mechanisms to transfer NVMe commands and data structures

The NVMe-oF spec is not Fabric specific; separate Transport Bindings are defined for each Transport.
**NVME OVER FABRICS**

**NVMe-oF Overview**

**Targets**
- Create logical NVMe subsystems and Controllers that are presented to Hosts
- Logically map NVMe Namespaces to physical NVMe block devices
- Export NVMe subsystems virtualizing NVM Namespaces
- May be provisioned to allow individual Hosts access to specific resources

**Hosts**
- Discover provisioned NVMe-oF resources from Targets
- Connect to provisioned resources
Linux In-kernel Implementation only supports local management

Each individual Target is either manually or statically configured
- Configure each Fabric with Address, Port, …
- Define each NVMe-oF Subsystem
- Assign NVMe resources to each NVMe-oF Subsystem
- Set up Individual Host Access rights to each NVMe-oF Subsystem

Each individual Host either accesses NVMe-oF resources through:
- Static predefined configurations
- Manual resource discovery process (as defined by NVMe-oF specification)
  - Connects to each individual Target
  - Requests resources on that Target that they may access

Limits usability flexibility, scale of dynamic installations of NVMe-oF
DISTRIBUTED ENDPOINT MANAGEMENT (DEM) PROJECT
Distributed Endpoint Management (DEM): An Open-Source Project

Enable efficient, dynamic configuration and provisioning of NVMe-oF Resources

Started out as a strawman driving specification changes into NVMe, NVMe-oF, and NVMe-MI

Management suite enabling:
- Remote configuration of NVMe-oF resources through RESTful interface
- Centralized enumeration of provisioned resources
- Single source for notification of changes to resources

Adopted by UNH-IOL with additional scripts written for Interoperability Test Suite
Distributed Endpoint Management (DEM): An Open-Source Project

Enable efficient, dynamic configuration and provisioning of NVMe-oF Resources

Started out as a strawman driving specification changes into NVMe, NVMe-oF, and NVMe-MI

Management suite supporting:

NVMe-oF Transports
- RDMA - Validated on IB/iWARP/RoCE
- TCP

Configuration via
- In-band (i.e., using NVMe-oF protocol)
- Out-of-Band (i.e., using RESTful interface via JSON)
Project Components

Discovery controller + Management (DEM)

Endpoint Manager (DEM-EM) – customized for Target implementation
- In-Band Mode
- Out-of-Band Mode

Optional Host Tools
- Auto Connect (DEM-AC)
- Discovery Log Page Monitor (DEM-DM)

RESTful Management Interfaces
- Web Pages (HTML / CSS files)
- Command Line Interface (DEM-CLI)
Discovery controller + Management (DEM)

- Primary component for remote configuration and provisioning
- Plug-in module architecture for NVMe-oF supported Fabrics

- Configures remote NVMe resources via In-Band or Out-of-Band interfaces
- Collects & distributes tailored Discovery Log Pages to Hosts
- Receives notification of changes to NVMe-oF resources
- Reports changes to NVMe-oF resources to affected registered Hosts
- Enables additional access restrictions
Endpoint Manager (DEM-EM)

- Agent running on Target enabling remote configuration
- Reuses DEM plug-in module architecture for NVMe-oF supported Fabrics

- Plug-in configuration model enabling implementation-specific management of NVMe-oF resources
- In-Band configuration based on proposed changes to NVMe-MI specification
- Out-of-Band RESTful configuration based on proposed changes to RF/SF
- Used for Targets not managed by other means
Optional Host Tools

- Reuses DEM plug-in module architecture for NVMe-oF supported Fabrics

Auto Connect (DEM-AC)

- Establishes persistent connection
- Collects Discovery Log Pages
- Automatically connects to its provisioned resources

Discovery Log Page Monitor (DEM-DM)

- Establishes persistent connection
- Reports Log Change Events and displays updated Discovery Log Pages
RESTful Management Interfaces

DEM Command Line Interface (DEM-CLI)
- Local Interface to the DEM via console command line

Web interface
- Interface to the DEM via web interface
- Project contains complete set of HTML, CSS, and JS files
BRIEF DEMONSTRATION
Screen shows interfaces Hosts use to access Log Pages from the DEM
Configure remote Target to determine existing configuration
First step, add the Target
Set the Target as Locally Managed. This will allow DEM to view a Target’s configuration.
Next, view the Target to add a Port ID and Subsystem to establish a NVMe-oF Controller
Next, add a Port ID
Provide the existing information to connect to the Target. Info must be known a priory for connection to the existing configuration.
Next, add an arbitrary Subsystem for the creation of a NVMe-oF Controller
Provide the Subsystem NQN and set to Allow Any Host access
Now that there is a Controller to connect to, DEM can query the Target for the set of preconfigured Log Pages.
This screen shows a preconfigured Log Page that does not match how the DEM has been configured as is evident from the Unattached heading.
Next, the Target will be reconfigured to allow the DEM to configure the Target.
The Target is configured for Out-of-Band Management through the DEM-EM residing on the Target. Admin must know how Target EM is configured/started.
The DEM has now retrieved the Interfaces and NVMe Namespace Devices exported by the Target.
Once the Interfaces have been retrieved, setting of the Port ID will be different from the prior screen.
Instead of providing the Transport Type, Address and Family, the Interface must be selected from the list.
Once a configuration is created, it can be pushed to the Target to replace any existing configuration.
Given this operation is destructive, it needs to be confirmed before being executed.
View the Target Log Pages to confirm the new configuration.
View the Target Log Pages to confirm the new configuration
To complete the NVMe-oF configuration, the Subsystem needs NS Devices via NSIDs.
To link the NS Device, an NS ID is required as well as selecting a NS Device from the available list.
Finally, a complete Target configuration
Discovery controller + Management (dem)
- Redundancy and Failover
- Generate Log Pages for Targets without a Discovery Controller
- Target usage monitoring
- Convert DEM to RedFish / SwordFish schema
- Convert DEM in-band configuration to current NVMe-MI Specification proposal

Endpoint Manager (dem-em)
- Target usage monitoring
- Extend for other Targets

Support:
- More NVMe-oF Transports (e.g., FC)
- Asymmetric Namespace Access Groups
- Namespace Subtypes/Partitions
- Transport Required (TREQ)
- Multipath
Currently Available:

- As a Dual GPL / BSD licensed project
THANK YOU

Phil Cayton

Intel Corporation