LEGAL NOTICE:

This NVM Express revision 1.3 technical proposal is proprietary to the NVM Express, Inc. (also referred to as "Company") and/or its successors and assigns.

NOTICE TO USERS WHO ARE NVM EXPRESS, INC. MEMBERS: Members of NVM Express, Inc. have the right to use and implement this NVM Express revision 1.3 technical proposal subject, however, to the Member's continued compliance with the Company's Intellectual Property Policy and Bylaws and the Member's Participation Agreement.

NOTICE TO NON-MEMBERS OF NVM EXPRESS, INC.: If you are not a Member of NVM Express, Inc. and you have obtained a copy of this document, you only have a right to review this document or make reference to or cite this document. Any such references or citations to this document must acknowledge NVM Express, Inc. copyright ownership of this document. The proper copyright citation or reference is as follows: **"© 2007 - 2018 NVM Express, Inc. ALL RIGHTS RESERVED."** When making any such citations or references to this document you are not permitted to revise, alter, modify, make any derivatives of, or otherwise amend the referenced portion of this document in any way without the prior express written permission of NVM Express, Inc. Nothing contained in this document shall be deemed as granting you any kind of license to implement or use this document or the specification described therein, or any of its contents, either expressly or impliedly, or to any intellectual property owned or controlled by NVM Express, Inc., including, without limitation, any trademarks of NVM Express, Inc.

LEGAL DISCLAIMER:

THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN "**AS IS**" BASIS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, NVM EXPRESS, INC. (ALONG WITH THE CONTRIBUTORS TO THIS DOCUMENT) HEREBY DISCLAIM ALL REPRESENTATIONS, WARRANTIES AND/OR COVENANTS, EITHER EXPRESS OR IMPLIED, STATUTORY OR AT COMMON LAW, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, VALIDITY, AND/OR NONINFRINGEMENT.

All product names, trademarks, registered trademarks, and/or servicemarks may be claimed as the property of their respective owners.

NVM Express Workgroup
c/o VTM, Inc.
3855 SW 153rd Drive
Beaverton, OR 97003 USA
info@nvmexpress.org

NVM Express Technical Proposal for New Feature

Technical Proposal ID	**8002 – Resource Enumeration / State Change Announcements**
Change Date	**10/1/2018**
Builds on Specification	**NVM Express 1.3c and NVM Express over Fabrics 1.0a**

Technical Proposal Author(s)

Name	Company
Phil Cayton	Intel
Jay Sternberg	Intel
Dave Minturn	Intel
David Black	Dell EMC
Frederick Knight	Net App
Rob Davis	Mellanox
Idan Burstein	Mellanox

This technical proposal defines enhancements to provide NVMe-oF with the same level of resource enumeration as is available with local PCIe-based NVMe devices. It defines an NVMe-oF level mechanism for notifying provisioned hosts about remote resource state changes (i.e., added, removed, modified), specifically Discovery Log changes.

These changes allow an optional explicit persistent connection to Discovery controllers, and defines a mechanism to indicate changes occurring on Subsystems and Namespace controllers via:

- Asynchronous Notification for optional events; and
- Transport Binding specifications may need to be updated for methods of:
    - Host determination of active Discovery controller(s): and
    - Discovery controller binding.

Revision History

Revision Date	Change Description
01/08/18	Initial proposal as drafted by Phil Cayton and Jay Sternberg
01/22/18	Integrate review input from Dave Minturn
02/05/18	Integrate review input from David Black and Rob Davis
03/05/18	Integrate review input from NVMe Technical Workgroup
03/29/18	Integrate review input from Frederick Knight
04/19/18	Integrate comments from NVMe WG
04/26/18	Submit for Phase 2 Technical ballot
06/11/18	Integrate 'page turner' review comments from Fredrick Knight, Rob Davis, Dave Minturn, Jay Sternberg, Phil Cayton
06/20/18	Integrate 'page turner' review comments from David Black, Phil Cayton, Jay Sternberg, Dave Minturn
06/28/18	Integrate 'page turner' review comments from David Black.
07/20/18	Integrate comments from NVMe WG
08/01/18	Integrate comments from Hannes Reinecke
09/25/18	Integration comments by the technical editor
10/1/2018	Ratified

[Description of Specification Changes]

[Changes to NVM Express over Fabrics 1.0 Specification]

[Modify section 1.5.6 "Discovery" as shown below:]

1.5.6 Discovery

NVMe over Fabrics defines a discovery mechanism that a host may use to determine the NVM subsystems the host may access. A Discovery controller supports minimal functionality for providing Discovery Logs. A Discovery controller may support notification of Discovery Log changes using Asynchronous Events. ~~and only implements the required features that allow the Discovery Log Page to be retrieved.~~ A Discovery controller does not implement I/O Queues or expose namespaces. A Discovery Service is an NVM subsystem that exposes only Discovery controllers.

[Modify Figure 19 "Connect Command – Submission Queue Entry" in section 3.3 "Connect Command and Response" to fix specification errata as shown below:]

Figure 19: Connect Command – Submission Queue Entry

Byte	Description
…	…
51:48	**Keep Alive Timeout (KATO):** This field has the same definition as the Keep Alive Timeout defined in section ~~5.15.1.14~~ 5.12 of the ~~the~~ NVMe Base specification. The controller shall set the Keep Alive Timeout Feature to this value.

[Modify section 5 "Discovery Service" as shown below:]

5 Discovery Service

NVMe over Fabrics defines a discovery mechanism that a host uses to determine the NVM subsystems that expose namespaces that the host may access. The Discovery Service provides a host with the following capabilities:

- The ability to discover a list of NVM subsystems with namespaces that are accessible to the host~~.~~;
- The ability to discover multiple paths to an NVM subsystem~~.~~; ~~and~~
- The ability to discover controllers that are statically configured~~.~~;
- The optional ability to establish explicit persistent connections to the Discovery controller; and
- The optional ability to receive Asynchronous Event Notifications from the Discovery controller.

A Discovery Service is an NVM subsystem that supports only Discovery controllers. A Discovery controller supports minimal functionality and only implements ~~the~~ ~~required~~ features related to Discovery Log Pages ~~that allow the Discovery Log Page to be retrieved~~ and does not implement I/O Queues or expose namespaces. The functionality supported by the Discovery controller is defined in section 5.4.

The host uses the well-known Discovery Service NQN (nqn.2014-08.org.nvmexpress.discovery) in the Connect command to a Discovery Service. The method that a host uses to obtain the NVMe Transport information necessary to connect to the well-known Discovery Service is implementation specific.

The Discovery Log Page provided by a Discovery controller contains one or more entries. Each entry specifies information necessary for the host to connect to an NVM subsystem. An entry may be associated with an NVM subsystem that exposes namespaces or a referral to another Discovery Service. There are no ordering requirements for log page entries within the Discovery Log Page.

Discovery controller(s) may provide different log page contents depending on the Host NQN provided (e.g., different NVM subsystems may be accessible to different hosts). The set of Discovery Log entries should

~~return~~ include all applicable addresses on the same fabric as the Discovery Service and may include addresses on other fabrics.

Discovery controllers that support explicit persistent connections shall support both Asynchronous Event Request and Keep Alive commands (refer to NVMe Base specification sections 5.2 and 5.12 respectively). A host requests an explicit persistent connection to a Discovery controller and Asynchronous Event Notifications from the Discovery controller on that persistent connection by specifying a non-zero Keep Alive Timer value in the Connect command. If the Connect command specifies a non-zero Keep Alive Timer value and the Discovery controller does not support Asynchronous Events, then the Discovery controller shall return a status value of Connect Invalid Parameters (refer to Figure 22) for the Connect command. Discovery controllers shall indicate support for Discovery Log Change Notifications in the Identify Controller Data Structure (refer to Figure 33).

~~The Keep Alive command is reserved for Discovery controllers. A transport~~ Discovery controllers that do not support explicit persistent connections shall not support Keep Alive commands and may use ~~specify~~ a fixed Discovery controller activity timeout value (e.g., 2 minutes). If no commands are received by such a Discovery controller within that time period, the controller may perform the actions for Keep Alive Timer expiration defined in section 7.1.2.

[Modify section 5.1 "Discovery Controller Initialization" as shown below:]

5.1 Discovery Controller Initialization

The initialization process for ~~the~~ Discovery controllers is described in Figure 36.~~below:~~

Figure 36: Discovery Controller Initialization process flow

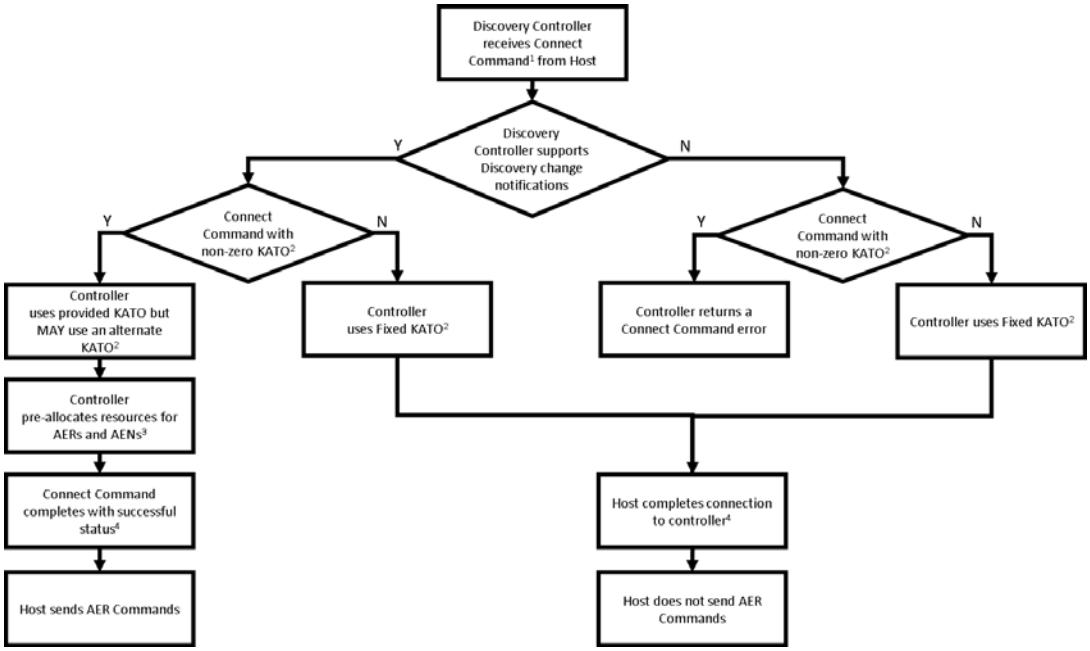

Figure 36 references:

1. Refer to section 3.3;
2. Refer to NVM Express Base specification Asynchronous Event Request command in section 5.2;
3. Refer to NVM Express Base specification Keep Alive command in section 5.12; and
4. Refer to the following steps in this section.

After the Connect Command completes with a status of Successful Completion, the host performs the following steps:

1. NVMe ~~in-band~~ authentication is performed if required (refer to section 6.2);
2. The host determines the controller's capabilities by reading the Controller Capabilities property;
3. The host configures the controller's settings by writing the Controller Configuration property, including setting CC.EN to '1' to enable command processing;
4. The host waits for the controller to indicate it is ready to process commands. The controller is ready to process commands when CSTS.RDY is set to '1' in the Controller Status property; and
5. The host determines the features and capabilities of the controller by issuing the Identify command, specifying the Controller data structure.

After initializing the Discovery controller, the host reads the Discovery Log Page. Refer to section 5.3.

[Modify figure 31 "Discovery Controller – Admin Commands" in section 5.2 "Discovery Controller Properties and Command Support" to add the additional optional commands this TP introduces as shown below:]

Figure 31: Discovery Controller – Admin Commands

Opcode by Field			Combined Opcode[2]	O/M[1]	Namespace Identifier Used[3]	Command
(07)	(06:02)	(01:00)				
Generic Command	Function	Data Transfer[4]				
0b	000 00b	10b	02h	M	n/a	Get Log Page
0b	000 01b	10b	06h	M	n/a	Identify
0b	000 10b	01b	09h	NOTE 5	n/a	Set Features
0b	000 10b	10b	0Ah	NOTE 5	n/a	Get Features
0b	000 11b	00b	0Ch	NOTE 5	n/a	Asynchronous Event Request
0b	001 10b	00b	18h	NOTE 5	n/a	Keep Alive

NOTES:
1. O/M definition: O = Optional, M = Mandatory.
2. Opcodes not listed are reserved.
3. The Namespace Identifier field (CDW1.NSID) is reserved for Discovery controllers.
4. 00b = no data transfer; 01b = host to controller; 10b = controller to host; 11b = bidirectional
5. For Discovery controllers that do not support explicit persistent connections, the command is reserved. For Discovery controllers that support explicit persistent connections, the command is mandatory.

[Modify figure 33 "Discovery Controller – Identify Controller Data Structure" in section 5.2 "Discovery Controller Properties and Command Support" to add OAES field as shown below:]

Bytes	O/M	Description
~~260:849~~1:84		Reserved
95:92	M	**Optional Asynchronous Events Supported (OAES):** This field indicates the optional asynchronous events supported by the controller. A controller shall not send optional asynchronous events before they are enabled by host software. Bit 31 is set to '1' if the controller supports sending Discovery Log Change Notifications. If cleared to '0', then the controller does not support the Discovery Log Change Notification events. Bits 30:0 are reserved.
260:96		Reserved

[Add section 5.4 "Discovery Controller* Features and Command Support*" as shown below:]

5.4 Discovery Controller Features and Command Support

These features indicate the attributes of a Discovery controller (refer to Figure 37). This is optional information not required for proper behavior of the system (refer to NVM Express Base specification – Figure 105: Set Features – Feature Identifiers).

Figure 37: Set Features Identifier

Feature Identifier	O/M[2]	Persistent Across Power Cycle and Reset[1]	Uses Memory Buffer for Attributes	Description
00h to 0Ah				Reserved
0Bh	O	No	No	Asynchronous Event Configuration
0Ch to 0Eh				Reserved
0Fh	O	No	No	Keep Alive Timer
10h to BFh				Reserved
C0h to FFh				Vendor Specific[3]

NOTES:
1. This column is only valid if the feature is not saveable (refer to NVM Express Base specification – Feature Values in section 7.8). If the feature is saveable, then this column is not used and any feature may be configured to be saved across power cycles and reset.
2. O/M definition: O = Optional, M = Mandatory.
3. The behavior of a controller in response to a vendor specific Feature Identifier is vendor specific.

5.4.1 Asynchronous Event Configuration (Feature Identifier 0Bh), (Optional)

Discovery controllers that support Asynchronous Event Notifications shall implement the Get Features and Set Features commands. A Discovery controller shall enable Asynchronous Discovery Log Event Notifications, if a non-zero KATO value is received in the Connect command sent to that controller.

Figure 38 defines Discovery controller Asynchronous Event Notifications.

Figure 38: Asynchronous Event Configuration – Command Dword 11

Bit	Description
31	**Discovery Log Page Change Notification:** This bit indicates that the Discovery controller reports Discovery Log Page Change Notifications. If set to '1', the Discovery controller shall send a notification if Discovery Log Page changes occur.
30:00	Reserved

[Add section 5.5 "Discovery Controller* Asynchronous Event Information – Requests and Notifications*" as shown below:]

5.5 Discovery Controller Asynchronous Event Information – Requests and Notifications

If Discovery controllers detect events about which a host has requested notification, then the Discovery controller shall send an Asynchronous Event with the:

- Asynchronous Event Type field set to Notice (i.e., 2h);
- Log Page Identifier field set to Discovery (i.e., 70h); and
- Asynchronous Event Information field set as defined in Figure 39.

Figure 39: Asynchronous Event Information – Notice

Value	Description
F0h	**Discovery Log Page Change:** A change has occurred to one or more of the Discovery Log Pages. The host should submit a Get Log Page command to receive updated Discovery Log Pages.
F1h to FFh	Reserved for future NVMe-oF Asynchronous Event Notifications

5.5.1 Discovery Log Page Change Asynchronous Event Notification (Event Information F0h)

When a Discovery controller updates Discovery Log Page(s), the Discovery controller shall send a Discovery Log Page Change Asynchronous Event notification to each host that has requested asynchronous event notifications of this type.

[Increment subsequent figure numbers in the NVMe over Fabrics 1.0 specification to reflect added Figures 36-39 above.]

[Changes to NVM Express Base specification]

[Modify two figures below to add a range for NVMe over Fabrics]

[Modify Figure 49 "Asynchronous Event Information – Notice" in section "5.2.1 Command Completion" to add a range for NVMe over Fabrics as shown below:]

Value	Description
~~3h – FFh~~03h to EFh	Reserved
F0h to FFh	Refer to NVMe over Fabrics specification

[Modify Figure 143 "Asynchronous Event Configuration – Command DWORD 11" in section "5.21.1.11 Asynchronous Event Configuration (Feature Identifier 0Bh)" to add a range for NVMe over Fabrics as shown below:]

Bit	Description
~~31:11~~31:28	Refer to NVMe over Fabrics specification
27:11	Reserved