LEGAL NOTICE:

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 Group .
3855 SW 153rd Drive
Beaverton, OR 97003 USA
info@nvmexpress.org

NVM Express Technical Proposal for New Feature

Technical Proposal ID	**4018a – NVM Sets and Read Recovery Level**
Change Date	**7/23/2018**
Builds on Specification	**NVM Express 1.3**

Technical Proposal Author(s)

Name	Company
Chris Petersen	Facebook
David Black	Dell EMC
Lee Prewitt	Microsoft
Monish Shah	Google
Mark Carlson, Steve Wells	Toshiba
Peter Onufryk	Microsemi
Fred Knight	NetApp
Bill Martin	Samsung
Amber Huffman, Jonathan Hughes, Mike Allison	Intel
Christoph Hellwig	WDC
Mike Allison	SK Hynix
Edward Hsieh	SMI

This technical proposal defines foundational capabilities that are used as part of Predictable Latency Mode. These foundational capabilities are separated into their own Technical Proposal, as they may be leveraged by other Technical Proposals or on their own. The foundational concepts are NVM Sets, Read Recovery Levels, and Endurance Groups.

Revision History

Revision Date	Change Description
8/22/2017	Separated foundational content on NVM Sets, Read Recovery Level, and Asymmetric Namespace Access to a separate document. Added section on Endurance Group.
8/22/2017	Made Endurance Groups part of NVM Sets.

8/29/2017	Changes based on editorial feedback from Bill Martin, Mike Allison, and Chris Petersen. Removed Get Log Page use of NVM Set Identifier. Created a log specific identifier that can be used for Endurance Group Identifier, and also for NVM Set Identifier in the future. Added figure for Endurance Group.
9/4/2017	Removed two instances of duplicate text. Modified NVM Set definition to allow for isolation or access. Modified NVM Set Identifier Maximum to be for the NVM subsystem rather than per controller.
9/6/2017	Used ECN 003 versions of section 6.1 and Get Log Page list to be up to date. Many editorial changes from Fred's markup. A few opens from Curtis regarding making NVM Sets more generally usable.
9/10/2017	Editorial changes based on Bill Martin's mark-up. Made Endurance Group a top level concept. NVM Sets are for isolation only, and not access characteristics.
9/14/2017	Final minor changes from 9/14 Technical meeting.
9/24/2017	Interaction between Read Recovery Level and Limited Retry is implementation specific. Added Random 4KB QD=1 typical read time. Modified Read Recovery Level Config to place NVM Set Identifier in Dword 11 and Read Recovery Level in Dword 12 and return Dword 12 in Get Features.
9/27/2017	Fixed a 63 to 31 for max number of NVM Set entries. Minor editorial fixes.
9/29/2017	Fixed section 5.22 to 5.21 references.
10/18/2017	Editorial feedback from Curtis Ballard incorporated.
10/19/2017	Added "(Log Identifier 09h)" to the Endurance Group Information page.
11/2/2017	Editorial change, specifically modified "the NVM subsystem and all controllers shall" to "then all controllers in the NVM subsystem shall".
5/15/2018	Modified Figure 126 so that the NVM Set Identifier (NVMSETID) field is aligned on the same bytes as same field in Figure 114.
7/23/2018	Ratified

Description of Specification Changes

8.TBD2 Read Recovery Level (Optional)

The Read Recovery Level (RRL) is a configurable attribute that balances the completion time for read commands and the amount of error recovery applied to those read commands. The Read Recovery Level applies to an NVM Set with which it is associated. A namespace created within an NVM Set inherits the Read Recovery Level of that NVM Set. If NVM Sets are not supported, all namespaces in the NVM subsystem use an identical Read Recovery Level.

The controller indicates support for Read Recovery Levels in the Controller Attributes field in the Identify Controller data structure (refer to Figure 109). If Read Recovery Levels are supported, then the specific levels supported are indicated in the Read Recovery Levels Supported field in the Identify Controller data structure. There are 16 levels that may be supported. Level 0, if supported, provides the maximum amount of recovery. Level 4 is a mandatory level that provides a nominal amount of recovery and is the default level. Level 15 is a mandatory level that provides the minimum amount of recovery and is referred to as the 'Fast Fail' level. The levels are organized based on the amount of recovery supported, such that a higher numbered level provides less recovery than the preceding lower level.

Interactions between the Read Recovery Level and the Limited Retry (LR) field in IO commands are implementation specific.

The Read Recovery Level may be configured using Set Features for the Read Recovery Level Config Feature. The Read Recovery Level may be determined using Get Features for the Read Recovery Level Config Feature.

Figure 8.TBD2_TBDa: Read Recovery Level Overview

Level	O/M	Description
0	O	
1	O	
2	O	
3	O	
4	M	Default
5	O	
6	O	
7	O	
8	O	
9	O	
10	O	
11	O	
12	O	
13	O	
14	O	
15	M	Fast Fail

If Read Recovery Levels are supported, then the NVM subsystem and all controllers shall:

- Support at least Level 4 and Level 15;

- Indicate support for Read Recovery Levels in the Controller Attributes field in the Identify Controller data structure;
- Support the Read Recovery Levels Supported field in the Identify Controller data structure; and
- Support the Read Recovery Level Config Feature.

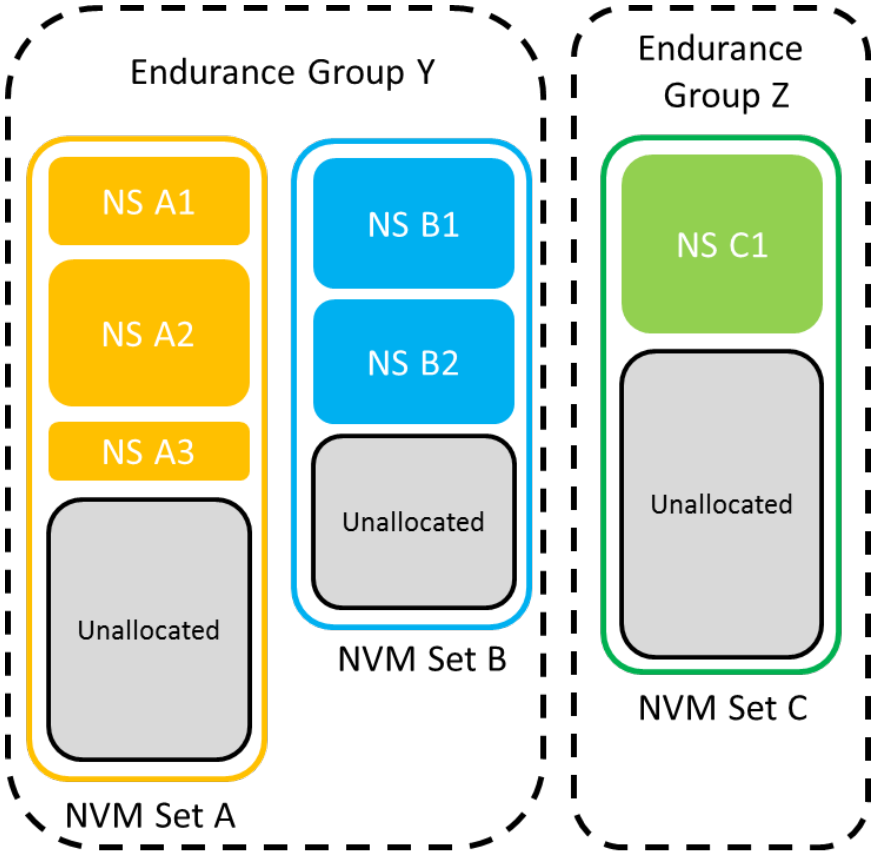
8.TBD3 Endurance Groups

Endurance may be managed within a single NVM Set (refer to section 4.TBD) or across a collection of NVM Sets. Each NVM Set is associated with an Endurance Group (refer to Fig5_15TBD1). If two or more NVM Sets have the same Endurance Group Identifier, then endurance is managed by the NVM subsystem across that collection of NVM Sets. If only one NVM Set is associated with a specific Endurance Group Identifier, then endurance is managed locally to that NVM Set.

The endurance information for an Endurance Group is specified in the Endurance Group Information log page (refer to section 5.14.1.9).

Figure Fig8.TBD3a shows Endurance Groups added to the example in Figure Fig4.TBDa. In this example, the endurance of NVM Set A and NVM Set B are managed together as part of Endurance Group Y, while the endurance of NVM Set C is managed only within NVM Set C as it is the only NVM Set that is part of Endurance Group Z.

Figure Fig8.TBD3a: NVM Sets and Associated Namespaces



If Endurance Groups are supported, then the NVM subsystem and all controllers shall:

- Indicate support for Endurance Groups in the Controller Attributes field in the Identify Controller data structure;

- Indicate the Endurance Group Identifier with which the namespace is associated in the Identify Namespace data structure; and
- Support the Endurance Group Information log page.

Modify section 6.1.1 (NSID and Namespace Usage) as shown below:
EDITORIAL NOTE: These changes build on NVMe 1.3 ECN 003.

If Namespace Management or NVM Sets (refer to section 4.TBD) are ~~is~~ supported (refer to the OACS field in Figure 109) then NSIDs shall be unique within the NVM subsystem (e.g., NSID of 3 shall refer to the same physical namespace regardless of the accessing controller). If Namespace Management and NVM Sets are ~~is~~ not supported then NSIDs:

a) for shared namespaces shall be unique; and
b) for private namespaces are not required to be unique.

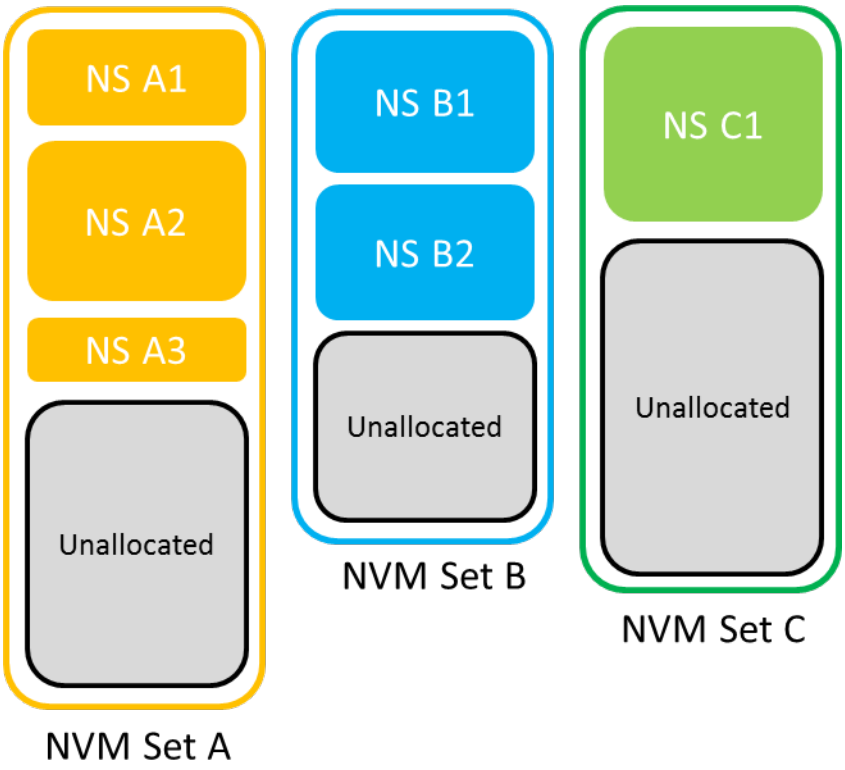
Add section 4.TBD (NVM Sets) prior to section 4.8 (Namespace List) as shown below:

4.TBD NVM Sets

An NVM Set is a collection of NVM that is separate (logically and potentially physically) from NVM in other NVM Sets. One or more namespaces may be created within an NVM Set and those namespaces inherit the attributes of the NVM Set. A namespace is wholly contained within a single NVM Set and shall not span more than one NVM Set.

Figure Fig4.TBDa shows an example of three NVM Sets. NVM Set A contains three namespaces (NS A1, NS A2, and NS A3). NVM Set B contains two namespaces (NS B1 and NS B2). NVM Set C contains one namespace (NS C1). Each NVM Set shown also contains 'Unallocated' regions that consist of NVM that is not yet allocated to a namespace.

Figure Fig4.TBDa: NVM Sets and Associated Namespaces



There is a subset of Admin commands that are NVM Set aware as described in Figure Fig4.TBDb.

Figure Fig4.TBDb: NVM Set Aware Admin Commands

Admin Command	Details
Identify	• The Identify Namespace data structure includes the associated NVM Set Identifier. • The NVM Set List data structure includes attributes for each NVM Set.
Namespace Management	• The create action includes the NVM Set Identifier as a host specified field.
Set Features	• The Read Recovery Level Feature specifies the associated NVM Set Identifier.

The host determines the NVM Sets present and their attributes using the Identify command with CNS value of 04h to retrieve the NVM Set List (refer to Fig5_15TBD0). For each NVM Set, the attributes include:

- an identifier associated with the NVM Set;
- the optimal size for writes to the NVM Set;
- the total capacity of the NVM Set; and
- the unallocated capacity for the NVM Set.

An NVM Set Identifier is a 16-bit value that specifies the NVM Set with which an action is associated. An NVM Set Identifier may be specified in NVM Set aware Admin commands (refer to Figure Fig4.TBD.b). An NVM Set Identifier value of 0h is reserved and is not a valid NVM Set Identifier. Unless otherwise specified, if the host specifies an NVM Set Identifier set to 0h for a command that requires an NVM Set Identifier, then that command shall fail with a status code of Invalid Field in Command.

Each NVM Set is associated with exactly one Endurance Group (refer to section 8.TBD3).

The NVM Set with which a namespace is associated is reported in the Identify Namespace data structure (refer to Figure 114). When a host creates a namespace using the Namespace Management command, the host specifies the NVM Set Identifier of the NVM Set that the namespace is to be created in. The namespace that is created inherits attributes from the NVM Set (e.g., the optimal write size to the NVM).

If NVM Sets are supported, then all controllers in the NVM subsystem shall:

- Indicate support for NVM Sets in the Controller Attributes field in the Identify Controller data structure;
- Support the NVM Set Identifier in all commands that use the NVM Set Identifier;
- Support the NVM Set List for the Identify command;
- Indicate the NVM Set Identifier with which the namespace is associated in the Identify Namespace data structure;
- Support Endurance Groups; and
- For each NVM Set, indicate the associated Endurance Group as an attribute.

Modify a portion of section 5.15 (Identify command) as shown below:

The Identify command uses the Data Pointer, ~~and~~ Command Dword 10, and Command Dword 11 fields. All other command specific fields are reserved.

Figure 107: Identify – Data Pointer

Bit	Description
127:00	**Data Pointer (DPTR):** This field specifies the start of the data buffer. Refer to Figure 11 for the definition of this field. If using PRPs, this field shall not be a pointer to a PRP List as the data buffer may not cross more than one page boundary.

Figure 108: Identify – Command Dword 10

Bit	Description
31:16	**Controller Identifier (CNTID):** This field specifies the controller identifier used as part of some Identify operations. If the field is not used as part of the Identify operation, then host software shall clear this field to 0h for backwards compatibility (0h is a valid controller identifier). Controllers that support Namespace Management shall support this field. This field is used for Identify operations with a CNS value of 12h or 13h. This field should be cleared to 0h for Identify operations with a CNS value of 00h, 01h, 02h, 10h, and 11h.
15:08	Reserved
07:00	**Controller or Namespace Structure (CNS):** This field specifies the information to be returned to the host. Refer to Figure 106.

Figure 108b: Identify – Command Dword 11

Bit	Description
31:16	Reserved
15:00	**NVM Set Identifier (NVMSETID):** This field specifies the identifier of the NVM Set. This field is used for Identify operations with a CNS value of 04h. This field should be cleared to 0h for Identify operations with other CNS values.

Modify a portion of Figure 109 (Identify – Identify Controller data structure) as shown below:

<table>
<tr><td>99:96</td><td>M</td><td>Controller Attributes (CTRATT): This field indicates attributes of the controller.

Bits ~~31:2~~ 31:5 are reserved.

Bit 4 (Endurance Groups): If set to ‘1’ then the controller supports Endurance Groups (refer to section 8.TBD3). If cleared to ‘0’ then the controller does not support Endurance Groups.

Bit 3 (Read Recovery Levels): If set to ‘1’ then the controller supports Read Recovery Levels (refer to section 8.TBD2). If cleared to ‘0’ then the controller does not support Read Recovery Levels.

Bit 2 (NVM Sets): If set to ‘1’ then the controller supports NVM Sets (refer to section 4.TBD). If cleared to ‘0’ then the controller does not support NVM Sets.

Bit 1 (Non-Operational Power State Permissive Mode): If set to ‘1’ then the controller supports host control of whether the controller may temporarily exceed the power of a non-operational power state for the purpose of executing controller initiated background operations in a non-operational power state (i.e., Non-Operational Power State Permissive Mode supported). If cleared to ‘0’ then the controller does not support host control of whether the controller may exceed the power of a non-operational state for the purpose of executing controller initiated background operations in a non-operational state (i.e., Non-Operational Power State Permissive Mode not supported). Refer to section 5.21.1.17.

Bit 0 if set to ‘1’ then the controller supports a 128-bit Host Identifier. Bit 0 if cleared to ‘0’ then the controller does not support a 128-bit Host Identifier.</td></tr>
<tr><td>101:100</td><td>O</td><td>Read Recovery Levels Supported (RRLS): If Read Recovery Levels (RRL) are supported, then this field shall be supported. If a bit is set to 1b, then the corresponding Read Recovery Level is supported. If a bit is cleared to 0b, then the corresponding Read Recovery Level is not supported.

<table>
<tr><th>Bit</th><th>Definition</th></tr>
<tr><td>0</td><td>Read Recovery Level 0</td></tr>
<tr><td>1</td><td>Read Recovery Level 1</td></tr>
<tr><td>2</td><td>Read Recovery Level 2</td></tr>
<tr><td>3</td><td>Read Recovery Level 3</td></tr>
<tr><td>4</td><td>Read Recovery Level 4 – Default[1]</td></tr>
<tr><td>5</td><td>Read Recovery Level 5</td></tr>
<tr><td>6</td><td>Read Recovery Level 6</td></tr>
<tr><td>7</td><td>Read Recovery Level 7</td></tr>
<tr><td>8</td><td>Read Recovery Level 8</td></tr>
<tr><td>9</td><td>Read Recovery Level 9</td></tr>
<tr><td>10</td><td>Read Recovery Level 10</td></tr>
<tr><td>11</td><td>Read Recovery Level 11</td></tr>
<tr><td>12</td><td>Read Recovery Level 12</td></tr>
<tr><td>13</td><td>Read Recovery Level 13</td></tr>
<tr><td>14</td><td>Read Recovery Level 14</td></tr>
<tr><td>15</td><td>Read Recovery Level 15 – Fast Fail[1]</td></tr>
<tr><td colspan="2">NOTE:
1. If Read Recovery Levels are supported, then this bit shall be set to 1b.</td></tr>
</table>
</td></tr>
<tr><td>~~111:100~~
111:102</td><td></td><td>Reserved</td></tr>
<tr><td>…</td><td>…</td><td>…</td></tr>
<tr><td>…</td><td>…</td><td>…</td></tr>
</table>

333:332	O	**NVM Set Identifier Maximum (NSETIDMAX):** This field defines the maximum value of a valid NVM Set Identifier for any controller in the NVM subsystem. The number of NVM Sets supported by the NVM subsystem is less than or equal to NSETIDMAX.
~~511:332~~ 511:334		Reserved

Modify Figure 114 (Identify – Identify Namespace data structure) as shown below:

63:48	O	**NVM Capacity (NVMCAP):** This field indicates the total size of the NVM allocated to this namespace. The value is in bytes. This field shall be supported if Namespace Management and Namespace Attachment commands are supported. Note: This field may not correspond to the logical block size multiplied by the Namespace Size field. Due to thin provisioning or other settings (e.g., endurance), this field may be larger or smaller than the Namespace Size reported.
~~103:64~~ 99:64		Reserved
101:100	O	**NVM Set Identifier (NVMSETID):** This field indicates the NVM Set with which this namespace is associated. If NVM Sets are not supported by the controller, then this field shall be cleared to 0h.
103:102	O	**Endurance Group Identifier (ENDGID):** This field indicates the Endurance Group with which this namespace is associated. If Endurance Groups are not supported by the controller, then this field shall be cleared to 0h.

Modify Figure 106 (Identify – Data Structure Returned) to add Identify NVM Set data structure as shown below:

Figure 106: Identify – Data Structure Returned

CNS Value	O/M	Definition
00h	M	The Identify Namespace data structure is returned to the host for the namespace specified in the Namespace Identifier (CDW1.NSID) field if it is an active NSID. If the specified namespace is not an active NSID, then the controller returns a zero filled data structure. If the controller supports Namespace Management and CDW1.NSID is set to FFFFFFFFh, the controller returns an Identify Namespace data structure that specifies capabilities that are common across namespaces for this controller. If the controller does not support Namespace Management and CDW1.NSID is set to FFFFFFFFh, the controller shall fail the command with a status code of Invalid Namespace or Format.
01h	M	The Identify Controller data structure is returned to the host for this controller.
02h	M	A list of 1024 namespace IDs is returned containing active NSIDs in increasing order that are greater than the value specified in the Namespace Identifier (CDW1.NSID) field of the command. The controller should abort the command with status code Invalid Namespace or Format if CDW1.NSID is set to FFFFFFFEh or FFFFFFFFh. Note that CDW1.NSID may be cleared to 0h to retrieve a Namespace List including the namespace starting with NSID of 1h. The data structure returned is a Namespace List (refer to section 4.8).
03h	M	A list of Namespace Identification Descriptor structures (refer to Figure 116) is returned to the host for the namespace specified in the Namespace Identifier (CDW1.NSID) field if it is an active NSID. The controller may return any number of variable length Namespace Identification Descriptor structures that fit into the 4096 byte Identify payload. All remaining bytes after the namespace identification descriptor structures should be cleared to 0h, and the host shall interpret a Namespace Identifier Descriptor Length (NIDL) value of 0h as the end of the list. If the hosts sees an unknown descriptor type it should continue parsing the structure. A controller shall not return multiple descriptors with the same Namespace Identification Descriptor Type (NIDT). A controller shall return at least one descriptor identifying the namespace.

04h	O	An NVM Set List (refer to Figure Fig5_15_TBD0) is returned to the host for up to 31 NVM Sets. The list contains entries for NVM Set identifiers greater than or equal to the value specified in the NVM Set Identifier (CDW11.NVMSETID) field.
~~04h – 0Fh~~ 05h – 0Fh		Reserved

Add the following material after Figure 116 (Identify – Namespace Identification Descriptor) as shown below:

Figure Fig5_15TBD0 defines an NVM Set List. The data structure is an ordered list by NVM Set Identifier, starting with the first NVM Set Identifier supported by the NVM subsystem that is equal to or greater than the NVM Set Identifier indicated in CDW11.NVMSETID. The NVM Set List describes the attributes for each NVM Set in the list based on the NVM Set Attributes Entry in Figure Fig5_15TBD1.

Figure Fig5_15TBD0: NVM Set List

Bytes	Description
0	**Number of Identifiers:** This field indicates the number of NVM Set Attributes Entries in the list. There are up to 31 entries in the list. A value of 0 indicates that there are no entries in the list.
127:1	Reserved
255:128	**Entry 0:** This field contains the first NVM Set Attributes Entry in the list, if present.
383:256	**Entry 1:** This field contains the second NVM Set Attributes Entry in the list, if present.
…	…
(N*128+255): (N*128+128)	**Entry N:** This field contains the N+1 NVM Set Attributes Entry in the list, if present.

Figure Fig5_15TBD1: NVM Set Attributes Entry

Bytes	Description
1:0	**NVM Set Identifier:** This field indicates the identifier of the NVM Set in the NVM subsystem that is described by this entry.
3:2	**Endurance Group Identifier:** This field indicates the Endurance Group for this NVM Set. Refer to section 8.TBD3.
7:4	Reserved
11:8	**Random 4KB Read Typical:** This field indicates the typical time to complete a 4KB random read at queue depth 1 in 100 nanosecond units.
15:12	**Optimal Write Size:** This field indicates the size in bytes for optimal write performance.
31:16	**Total NVM Set Capacity:** This field indicates the total NVM capacity in this NVM Set. The value is in bytes.
47:32	**Unallocated NVM Set Capacity:** This field indicates the unallocated NVM capacity in this NVM Set. The value is in bytes.
127:48	Reserved

Modify section 8.12 on Namespace Management as shown below:

8.12 Namespace Management (Optional)

The Namespace Management command is used to create a namespace or delete a namespace. The Namespace Attachment command is used to attach and detach controllers from a namespace. Namespace management is intended for use during manufacturing or by a system administrator.

When a namespace is detached from a controller or deleted it becomes an inactive namespace on that controller. Previously submitted but uncompleted or subsequently submitted commands to the affected namespace are handled by the controller as if they were issued to an inactive namespace.

The size of a namespace is based on the number of logical blocks requested in a create operation, the format of the namespace, and any characteristics (e.g., endurance). The controller determines the NVM capacity allocated for that namespace. Namespaces may be created with different usage characteristics (e.g., endurance) that utilize differing amounts of NVM capacity. Namespace characteristics and the mapping of these characteristics to NVM capacity usage are outside the scope of this specification.

The total and unallocated NVM capacity for the NVM subsystem is reported in the Identify Controller data structure.

For controllers that support NVM Sets, the total and unallocated NVM capacity for each NVM Set is reported as part of the NVM Set Attributes Entry (refer to Figure Fig5_15TBD1). For each namespace, the NVM Set in which the namespace is allocated is reported in the Identify Namespace data structure. The NVM Set to be used for a namespace is based on the value in the NVM Set Identifier field in a create operation. If the NVM Set Identifier field is cleared to 0h in a create operation, then the controller shall choose the NVM Set from which to allocate the namespace.

For each namespace, the NVM capacity used for that namespace is reported in the Identify Namespace data structure. The controller may allocate NVM capacity in units such that the requested size for a namespace may be rounded up to the next unit boundary. For example, if host software requests a namespace of 32 logical blocks with a logical block size of 4KB for a total size of 128KB and the allocation unit for the implementation is 1MB then the NVM capacity consumed may be rounded up to 1MB. The NVM capacity fields may not correspond to the logical block size multiplied by the total number of logical blocks.

To create a namespace, host software performs the following actions:

…

Modify Figure 126 (Namespace Management – Host Software Specified Fields) as shown below:

Figure 126: Namespace Management – Host Software Specified Fields

Bytes	Description	Host Specified
7:0	Namespace Size (NSZE)	Yes
15:8	Namespace Capacity (NCAP)	Yes
25:16	Reserved	
26	Formatted LBA Size (FLBAS)	Yes
28:27	Reserved	
29	End-to-end Data Protection Type Settings (DPS)	Yes
30	Namespace Multi-path I/O and Namespace Sharing Capabilities (NMIC)	Yes
99:31	Reserved	
101:100	NVM Set Identifier (NVMSETID)	Yes
~~383:31~~ 383:102	Reserved	

Modify Figure 87 (Get Log Page – Command Dword 11) in section 5.14 as shown below:

Figure 87: Get Log Page – Command Dword 11

Bit	Description
31:16	**Log Specific Identifier:** This field specifies an identifier that is required for a particular log page. The log pages that require a log specific identifier are indicated in the table below. **Log Page** / **Definition** Endurance Group Information / Endurance Group Identifier (refer to section 8.TBD3)
15:00	**Number of Dwords (NUMDU):** This field specifies the upper 16 bits of the number of Dwords to return.

Modify Figure 90 as shown below:
EDITORIAL NOTE: These changes build on NVMe 1.3 ECN 003.

Figure 1: Get Log Page – Log Page Identifiers

Log Identifier	O/M	Scope	Description	Reference Section
00h	Reserved			
01h	M	Controller	Error Information	5.14.1.1
02h	M	NVM subsystem [1]	SMART / Health Information	5.14.1.2
	O	Namespace [2]		
03h	M	NVM subsystem	Firmware Slot Information	5.14.1.3
04h	O	Controller	Changed Namespace List	5.14.1.4
05h	O	Controller	Commands Supported and Effects	5.14.1.5
06h	O	NVM subsystem	Device Self-test	5.14.1.6
07h	O	Controller	Telemetry Host-Initiated	5.14.1.7
08h	O	Controller	Telemetry Controller-Initiated	5.14.1.8
09h	O	NVM subsystem	Endurance Group Information	5.14.1.9
~~09h – 6Fh~~ 0Ah – 6Fh	Reserved			
70h	Discovery (refer to the NVMe over Fabrics specification)			
71h – 7Fh	Reserved for NVMe over Fabrics			
80h – BFh	I/O Command Set Specific			
C0h – FFh	Vendor specific			
KEY: O = Optional, M = Mandatory Namespace = The log page contains information about a specific namespace. Controller = The log page contains information about the controller that is processing the command. NVM subsystem = The log page contains information about the NVM subsystem.				
NOTES: 1. For namespace identifiers of 0h or FFFFFFFFh 2. For namespace identifiers other than 0h or FFFFFFFFh				

Add section 5.14.1.9 adding a new log page for Endurance Group Information as shown below:

5.14.1.9 Endurance Group Information (Log Identifier 09h)

This log page is used to provide endurance information based on the Endurance Group (refer to section 8.TBD3). An Endurance Group consists of zero or more NVM Sets. The information provided is over the life of the Endurance Group. The Endurance Group Identifier is specified in the Log Specific Identifier field in Command Dword 11 of the Get Log Page command. The log page is 512 bytes in size.

Figure 5_14_1_11Fig0: Get Log Page – Endurance Group Log

Bytes	Description
3:0	Reserved
4	**Available Spare Threshold:** The available spare is indicated as a normalized percentage (0 to 100%). The values 101-255 are reserved.
5	**Percentage Used:** Contains a vendor specific estimate of the percentage of life used for the NVM Set(s) that comprise the Endurance Group based on the actual usage and the manufacturer's prediction of NVM life. A value of 100 indicates that the estimated endurance of the NVM in the Endurance Group has been consumed, but may not indicate an NVM failure. The value is allowed to exceed 100. Percentages greater than 254 shall be represented as 255. This value shall be updated once per power-on hour when the controller is not in a sleep state. Refer to the JEDEC JESD218A standard for SSD device life and endurance measurement techniques.
31:6	Reserved
47:32	**Endurance Estimate:** This field is an estimate of the total number of data bytes that may be written to the NVM Set(s) that comprise the Endurance Group over the lifetime of the Endurance Group assuming a Write Amplification of 1. This value is reported in billions (i.e., a value of 1 corresponds to 1,000,000,000 bytes written) and is rounded up. A value of zero indicates that the controller does not report an Endurance Estimate.
63:48	**Data Units Read:** Contains the total number of data bytes that have been read from the NVM Set(s) that comprise the Endurance Group. This value does not include controller reads due to internal operations such as garbage collection. This value is reported in billions (i.e., a value of 1 corresponds to 1,000,000,000 bytes read) and is rounded up. A value of zero indicates that the controller does not report the number of Data Units Read.
79:64	**Data Units Written:** Contains the total number of data bytes that have been written to the NVM Set(s) that comprise the Endurance Group. This value does not include controller writes due to internal operations such as garbage collection. This value is reported in billions (i.e., a value of 1 corresponds to 1,000,000,000 bytes written) and is rounded up. A value of zero indicates that the controller does not report the number of Data Units Written.
95:80	**Media Units Written:** Contains the total number of data bytes that have been written to the NVM Set(s) that comprise the Endurance Group including both host and controller writes (e.g. garbage collection). This value is reported in billions (i.e., a value of 1 corresponds to 1,000,000,000 bytes written) and is rounded up. A value of zero indicates that controller does not report the number of Media Units Written.
511:96	Reserved

Figure 84: Get Features – Feature Identifiers

Description	Section Defining Format of Attributes Returned
Arbitration	Section 5.21.1.1
Power Management	Section 5.21.1.2
LBA Range Type	Section 5.21.1.3
Temperature Threshold	Section 5.21.1.4
Error Recovery	Section 5.21.1.5
Volatile Write Cache	Section 5.21.1.6
Number of Queues	Section 5.21.1.7
Interrupt Coalescing	Section 5.21.1.8
Interrupt Vector Configuration	Section 5.21.1.9
Write Atomicity	Section 5.21.1.10
Asynchronous Event Configuration	Section 5.21.1.11
Autonomous Power State Transition	Section 5.21.1.12
Host Memory Buffer	Section 5.21.1.13
Timestamp	Section 5.21.1.14
Keep Alive Timer	Section 5.21.1.15
Host Controlled Thermal Management	Section 5.21.1.16
Non-Operational Power State Config	Section 5.21.1.17
Read Recovery Level Config	Section 5.21.1.18
NVM Command Set Specific	
Software Progress Marker	Section ~~5.21.1.18~~ 5.21.1.19
Host Identifier	Section ~~5.21.1.19~~ 5.21.1.20
Reservation Notification Mask	Section ~~5.21.1.20~~ 5.21.1.21
Reservation Persistence	Section ~~5.21.1.21~~ 5.21.1.22

Modify Figure 134 (Set Features – Feature Identifiers) as shown below:

Figure 134: Set Features – Feature Identifiers

Feature Identifier	O/M[6]	Persistent Across Power Cycle and Reset[2]	Uses Memory Buffer for Attributes	Description
00h				Reserved
01h	M	No	No	Arbitration
02h	M	No	No	Power Management
03h	O	Yes	Yes	LBA Range Type
04h	M	No	No	Temperature Threshold
05h	M	No	No	Error Recovery
06h	O	No	No	Volatile Write Cache
07h	M	No	No	Number of Queues
08h	NOTE 5	No	No	Interrupt Coalescing
09h	NOTE 5	No	No	Interrupt Vector Configuration
0Ah	M	No	No	Write Atomicity Normal
0Bh	M	No	No	Asynchronous Event Configuration
0Ch	O	No	Yes	Autonomous Power State Transition
0Dh	O	No[3]	No[4]	Host Memory Buffer
0Eh	O	No	Yes	Timestamp
0Fh	O	No	No	Keep Alive Timer
10h	O	Yes	No	Host Controlled Thermal Management
11h	O	No	No	Non-Operational Power State Config
12h	O	Yes	No	Read Recovery Level Config
~~11h – 77h~~ 13h– 77h				Reserved
78h – 7Fh		Refer to the NVMe Management Interface Specification for definition.		
80h – BFh				Command Set Specific (Reserved)
C0h – FFh				Vendor Specific[1]

NOTES:
1. The behavior of a controller in response to an inactive namespace ID to a vendor specific Feature Identifier is vendor specific.
2. This column is only valid if the feature is not saveable (refer to 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.
3. The controller does not save settings for the Host Memory Buffer feature across power states and reset events, however, host software may restore the previous values. Refer to section 8.9.
4. The feature does not use a memory buffer for Set Features, but it does use a memory buffer for Get Features. Refer to section 8.9.
5. The feature is mandatory for NVMe over PCIe. This feature is not supported for NVMe over Fabrics.
6. O/M: O = Optional, M = Mandatory.

5.21.1.18 Read Recovery Level Config (Feature Identifier 12h)

This Feature is used to configure the Read Recovery Level (refer to section 8.TBD2). The attributes are indicated in Command Dword 11 and Command Dword 12. Modifying the Read Recovery Level has no effect on the data contained in any associated namespace.

If a Get Features command is submitted for this Feature, the attributes specified in Figure 5_21_1_18Fig1 are returned in Dword 0 of the completion queue entry for that command.

Figure 5_21_1_18Fig0: Read Recovery Level Config – Command Dword 11

Bit	Description
31:16	Reserved
15:00	**NVM Set Identifier (NVMSETID):** This field specifies the NVM Set to be modified. If NVM Sets are not supported, then this field is ignored and the command applies to all namespaces in the NVM subsystem.

Figure 5_21_1_18Fig1: Read Recovery Level Config – Command Dword 12

Bit	Description
31:04	Reserved
03:00	**Read Recovery Level (RRL):** This field sets the Read Recovery Level for the NVM Set specified.