



**LEGAL NOTICE:**

© Copyright 2007 - 2018 NVM Express, Inc. ALL RIGHTS RESERVED.

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

**NVM Express Technical Proposal for New Feature**

<b>Technical Proposal ID</b>	<b>4007 – Persistent Event Log</b>
<b>Change Date</b>	<b>7/31/2018</b>
<b>Builds on Specification</b>	<b>NVM Express 1.3</b>

## Technical Proposal Author(s)

Name	Company
Nanci Olson, Curtis Ballard, Olga Otte	Hewlett Packard Enterprise
David Black, Austin Bolen, Gerry Fredette	Dell EMC
Jim Hatfield	Seagate
Michael Allison	SKHynix
Paul Suhler	Micron
Michael Allison	Intel

This technical proposal defines a mechanism for a persistent event log in NVMe subsystems. The log is intended to persistently capture significant events for use by software/system vendors that are not the NVMe subsystem manufacturer such as operating systems, management software, storage system vendors, etc. The information may overlap with information that is reported in the telemetry log but Persistent Event log information is reported using a defined format which is able to be used directly by an application client unlike the Telemetry log which is reported in a vendor specific format only able to be parsed by vendor specific tools.

This TP includes the Persistent Event Log model and a minimal set of events. Additional events will be defined in one or more follow-on technical proposals. The first follow-on TP defining additional events is TP 4042, Further Persistent Event Log Events. See the TP 4042 document for further details of events defined in that TP.

**Editorial Conventions:** Black text indicates existing text or entirely new content if designated as new content with **dark green comments** at start of the new content. New content mixed with existing content is shown in **blue text**. Existing text to be removed is shown in ~~red strikethrough text~~. Text to be moved is shown in **brown text**. Comments intended only for discussion or explanation and not for inclusion in the specification are shown in **dark green text**. Text that has changed since the most recent presentation is shown in Blue/Grey underline.

**Revision History**

Revision Date	Change Description
9/24/2017	First version
6 Oct. 2017	Copy model used elsewhere for reading, lock on first command then read
24 Oct. 2017	Simplified rules for retaining log data during reads Specified that lost events result in 00h bytes at tail of log Simplified PCIe specific values in header to just point to Identify Controller Inserted discussion topic, severity field in events? Inserted a SMART/Health log discussion topic at end of document
30 Oct. 2017	Created explicit error codes (cmd seq error) for important cases <ul style="list-style-type: none"> <li>Request new context and one exists (other user already reading)</li> <li>Try to read more data and context is gone (controller lost context)</li> </ul> Created possible subsystem SMART/Health log for discussion
1 Nov. 2017	Changed log to subsystem wide with following effects: <ul style="list-style-type: none"> <li>SMART log captures duplicated for each controller</li> <li>Can't sort by timestamp, leave up to subsystem to report in general time order</li> <li>Each event needs identifier for associated controller</li> <li>Timestamp is subsystem events if multiple controllers exist is picked by subsystem from one controller</li> </ul> Created a proposed format for a new 'Timestamp change' event
20 Dec. 2017	Accepted earlier edits Changed power on hours in timestamp changed event to milliseconds Added power cycle counter to event header Revised byte numbering on all events to match updated header
24 Jan. 2018	Updated following Jan 2018 meeting/decisions <ul style="list-style-type: none"> <li>Removed requirements not present in other NVMe log pages such as validating of requested offset</li> <li>Chose to not place requirements on what events need to be in page</li> <li>Removed Power Cycle Count from event header, it remains in the reset event</li> <li>Added a Power Cycle Count field to the log page header</li> <li>Clarified that support for Timestamp feature is not required, timestamps in this log just use the same format</li> <li>Used fewer bytes for power cycle count and power on time counters</li> </ul>
31 Jan 2018	<ul style="list-style-type: none"> <li>Changed suggested size from 5MB to 2.5MB/physical controller</li> <li>Added allowance that controller may filter events if they are occurring too rapidly</li> <li>Created guidance to replace old events when log is full</li> <li>Clarified events that effect multiple controllers only need logged once by a vendor select one of the controllers</li> <li>Added requirement that new events are still logged during reporting</li> <li>Moved firmware was activated and Format was in progress flags to controller specific space in reset event.</li> </ul>
7 February 2018	<ul style="list-style-type: none"> <li>Updated NVM Express TP intro to 2018 date</li> <li>Changed event vendor specific offset to vendor specific data length</li> <li>Removed event header from individual event structures</li> <li>Added guidance for suppressing events if the same event repeats too quickly</li> </ul>
14 February 2018	<ul style="list-style-type: none"> <li>Added Rebuild Assist event to future events list</li> <li>Made SMART/Health snapshot optional for fabrics</li> <li>Changed guidance on log size to remove specific values</li> <li>Revised Vendor Event wording to clarify what was vendor specific and what was defined structure</li> </ul>
22 February 2018	Clean version without change tracking for phase 2 exit vote

19 March 2018	Edits from phase 2 exit review <ul style="list-style-type: none"> <li>Two tables had inconsistent column formatting, corrected</li> <li>'alteration' allowed by sanitize operation clarified to not clear log</li> <li>SMART/Health event qualifier for primary controllers when using virtualization reformatted to match other similar text.</li> </ul>
11 April 2018	Incorporating changes requested during Technical WG deep dive discussion
18 April 2018	Added bitmap to report supported Persistent Event Log events Revised format of event tables to match core specification preferred format
25 April 2018	Changed supported events bitmap alignment to end of log header so it will always be the last field in the header Accepted all proposed changes and named with latest naming scheme Phase 3 exit vote will be requested on this version
26 April 2018	Fixed a date that hadn't been updated Fixed a missing close parenthesis Technical WG approved sending this version for 30 day member review
18 June 2018	Minor edits from 30 day member review <ul style="list-style-type: none"> <li>Two incorrect table references were corrected</li> <li>A generic "event data" phrase was replaced with more precise text</li> <li>Several capitalization errors were corrected</li> <li>Three incorrect table byte count calculations were corrected</li> </ul>
21 June 2018	Final version for integration accepting all changes, no further edits from 18 June version
25 June 2018	Editorial changes requested in late review comments
31 July 2018	Updated to match text as incorporated into 1.NEXTc draft core specification All TBD values replaced with values matching first draft incorporation
20 August 2018	Ratified

## Description of Specification Changes

Comment: Add reporting of support for this log page

Figure 223: Identify – Identify Controller Data Structure

Bytes	O/M	Description
Unchanged rows removed for readability		
Admin Command Set Attributes & Optional Controller Capabilities		
261	M	<p><b>Log Page Attributes (LPA):</b> This field indicates optional attributes for log pages that are accessed via the Get Log Page command.</p> <p>Bits 7:54 are reserved.</p> <p>Bit 4 if set to '1' then the controller supports the Persistent Event log. If cleared to '0' then the controller does not support the Persistent Event log.</p> <p>Bit 3 if set to '1' then the controller supports the Telemetry Host-Initiated and Telemetry Controller-Initiated log pages and sending Telemetry Log Notices. If cleared to '0' then the controller does not support the Telemetry Host-Initiated and Telemetry Controller-Initiated log pages and Telemetry Log Notice events.</p> <p>Bit 2 if set to '1' then the controller supports extended data for Get Log Page (including extended Number of Dwords and Log Page Offset fields). Bit 2 if cleared to '0' then the controller does not support extended data for Get Log Page.</p> <p>Bit 1 if set to '1' then the controller supports the Commands Supported and Effects log page. Bit 1 if cleared to '0' then the controller does not support the Commands Supported and Effects log page.</p> <p>Bit 0 if set to '1' then the controller supports the SMART / Health information log page on a per namespace basis. If cleared to '0' then the controller does not support the SMART / Health information log page on a per namespace basis.</p>
Unchanged rows removed for readability		

Bytes	O/M	Description
355:352	O	<b>Persistent Event Log Size (PELS):</b> This field indicates the maximum reportable size for the Persistent Event Log (Refer to section 5.14.1.13) in 64KB units. If the Persistent Event Log is not supported, then this field is reserved.
511: <del>352</del> 356		Reserved
Unchanged rows removed for readability, no further changes in this table		
NVM Command Set Attributes		
512	M	<b>Submission Queue Entry Size (SQES):</b> This field defines . . . . .

## 5.14 Get Log Page command

The Get Log Page command returns a data buffer containing the log page requested.

Unchanged text removed for readability

### 5.14.1 Log Specific Information

Figure and Figure 94 define the Log pages that may be retrieved with the Get Log Page command.

**Figure 183: 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 <sup>1</sup>	SMART / Health Information	5.14.1.2
	O	Namespace <sup>2</sup>		
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
0Ah	O	NVM subsystem	Predictable latency Per NVM Set	5.14.1.10
0Bh	O	NVM subsystem	Predictable Latency Event Aggregate	5.14.1.11
0Ch	O	Controller	Asymmetric Namespace Access	5.14.1.12
0Dh	O	NVM subsystem	Persistent Event Log	5.14.1.13
<del>0Eh</del> 0Eh – 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				

**Comment: All of the following content is new and shown in black text**

#### 5.14.1.13 Persistent Event Log (Log Identifier 0Dh)

The Persistent Event Log page contains information about significant events not specific to a particular command. The information in this log page shall be retained across power cycles and resets. This log consists of a header describing the log and zero or more Persistent Events (refer to section 5.14.1.13.1).

This log page is global to the NVM subsystem.

A sanitize operation may alter this log page (e.g., remove or modify events to prevent derivation of user data from log page information, refer to section 8.15). A sanitize operation should not clear the log page.

Persistent Event Log events specified in this section should be reported in an order such that more recent events are generally reported earlier in the log data than older events. The method by which the NVM subsystem determines the order in which events occurred is vendor specific.

The number of events supported is vendor specific. The supported maximum size for the Persistent Event Log is indicated in the PELS field of Identify Controller. The number of events supported and the supported maximum size should be large enough that the number of events or the size of the Persistent Event Log data does not reach the maximum supported size over the usable life of the NVM subsystem.

The controller shall log all supported events at each event occurrence unless the controller determines that the same event is occurring at a frequency that exceeds a vendor specific threshold for the frequency of event creation. If the same event is occurring at a frequency that exceeds a vendor specific threshold then the vendor may suppress further entries for the same event. A controller may indicate if events have been suppressed in vendor specific event data.

It is vendor specific which events are deleted (e.g., important events may be retained and events that are newer than an important event that was retained may be deleted) to make room for future events if:

- a) the size of the Persistent Event Log data reaches the maximum supported size;
- b) the number of events reaches the largest reportable number of events; or
- c) an event category reaches the largest reportable number of events for that category (e.g., information regarding 1,000 occurrences of changes to the timestamp is stored in internal data structures and extracted for reporting as Timestamp Change events in the Persistent Event Log and more than 1,000 Timestamp Change events have occurred).

Events that affect multiple controllers (e.g., an NVM subsystem reset) should be logged once by a controller selected by the vendor and not logged by any other controllers.

The Action bit in the Log Specific Field (refer to Figure 202) specifies if:

- a) A persistent event log page reporting context is created at the start of processing this Get Log Page command and log page data, if any, is read from the log page data associated with that log page reporting context;
- b) Log page data is read from the log page data associated with a preexisting log page reporting context; or
- c) The persistent event log page reporting context, if any, is released.

The persistent event log page reporting context is vendor specific information that the controller creates for determining what information will be included in the Persistent Event Log Page data (e.g., the persistent event log page reporting context may be the persistent event log page data or may contain a set of pointers to the events to report).

The controller should retain the persistent event log page reporting context:

- a) Until the controller processes a Get Log Page command requesting the Persistent Event Log Page with the Action field set to 02h (i.e., Release Context), an NVM subsystem reset, or a controller reset; or
- b) For a vendor specific time long enough to allow retrieval of the persistent event log page data.

Persistent Event Log events that occur while a persistent event log page reporting context exists shall not be reported in the existing reporting context but shall be logged.

The host is expected to issue a Get Log Page command with the Action field set to 02h to release the persistent event log page reporting context after reading the persistent event log page data.

**Figure 202: Command Dword 10 - Log Specific Field**

Bit	Description										
11:10	Reserved										
09:08	<b>Action:</b> This field specifies the action the controller shall take during processing this Get Log Page command										
	<table><tr><th>Value</th><th>Definition</th></tr><tr><td>00b</td><td><b>Read Log Data:</b> Return persistent event log page data starting at the address indicated by the LPOU field and the LPOL field in the Get Log Page command. If the controller does not have a persistent event log page reporting context then the controller shall fail the command with a status code of Command Sequence Error.</td></tr><tr><td>01b</td><td><b>Establish Context and Read Log Data:</b> The controller shall:<ul style="list-style-type: none"><li>a) determine the length of the persistent event log page data;</li><li>b) determine the set of events to report in the persistent event log page data; and</li><li>c) establish a persistent event log reporting context to store information describing the persistent event log data to be reported and track persistent event log page data accesses.</li></ul><p>After establishing a persistent event log reporting context the controller shall return persistent log page data starting at the address indicated by the LPOU field and the LPOL field in the Get Log Page command.</p><p>A number of dwords to return set to zero in the Get Log Page command indicates that no log data shall be returned.</p><p>If a persistent event log reporting context already exists, then the controller shall fail the command with a status code of Command Sequence Error.</p></td></tr><tr><td>10b</td><td><b>Release Context:</b> The controller shall release the persistent event log reporting context, if any. It is not an error if the controller does not have a persistent log page reporting context.</td></tr><tr><td>11b</td><td>Reserved</td></tr></table>	Value	Definition	00b	<b>Read Log Data:</b> Return persistent event log page data starting at the address indicated by the LPOU field and the LPOL field in the Get Log Page command. If the controller does not have a persistent event log page reporting context then the controller shall fail the command with a status code of Command Sequence Error.	01b	<b>Establish Context and Read Log Data:</b> The controller shall: <ul style="list-style-type: none"><li>a) determine the length of the persistent event log page data;</li><li>b) determine the set of events to report in the persistent event log page data; and</li><li>c) establish a persistent event log reporting context to store information describing the persistent event log data to be reported and track persistent event log page data accesses.</li></ul> <p>After establishing a persistent event log reporting context the controller shall return persistent log page data starting at the address indicated by the LPOU field and the LPOL field in the Get Log Page command.</p> <p>A number of dwords to return set to zero in the Get Log Page command indicates that no log data shall be returned.</p> <p>If a persistent event log reporting context already exists, then the controller shall fail the command with a status code of Command Sequence Error.</p>	10b	<b>Release Context:</b> The controller shall release the persistent event log reporting context, if any. It is not an error if the controller does not have a persistent log page reporting context.	11b	Reserved
	Value	Definition									
	00b	<b>Read Log Data:</b> Return persistent event log page data starting at the address indicated by the LPOU field and the LPOL field in the Get Log Page command. If the controller does not have a persistent event log page reporting context then the controller shall fail the command with a status code of Command Sequence Error.									
	01b	<b>Establish Context and Read Log Data:</b> The controller shall: <ul style="list-style-type: none"><li>a) determine the length of the persistent event log page data;</li><li>b) determine the set of events to report in the persistent event log page data; and</li><li>c) establish a persistent event log reporting context to store information describing the persistent event log data to be reported and track persistent event log page data accesses.</li></ul> <p>After establishing a persistent event log reporting context the controller shall return persistent log page data starting at the address indicated by the LPOU field and the LPOL field in the Get Log Page command.</p> <p>A number of dwords to return set to zero in the Get Log Page command indicates that no log data shall be returned.</p> <p>If a persistent event log reporting context already exists, then the controller shall fail the command with a status code of Command Sequence Error.</p>									
	10b	<b>Release Context:</b> The controller shall release the persistent event log reporting context, if any. It is not an error if the controller does not have a persistent log page reporting context.									
11b	Reserved										

The log page returned is defined in **Figure 203**.

**Figure 203: Get Log Page - Persistent Event Log (Log Identifier 0Dh)**

Bytes	Description
<b>Persistent Event Log Header</b>	
00	<b>Log Identifier:</b> This field shall be set to 0Dh.
03:01	Reserved
07:04	<b>Total Number of Events (TNEV):</b> Contains the number of event entries in the log.
15:08	<b>Total Log Length (TLL):</b> Contains the total number of bytes of persistent event log page data available, including the header.
16	<b>Log Revision:</b> Contains a number indicating the revision of the Get Log Page data structure that this log page data complies with. Shall be set to 01h.
17	Reserved
19:18	<b>Log Header Length:</b> This field contains the length in bytes of the log header information that follows. The total length of the log header in bytes is the value in this field plus 20.

Bytes	Description																											
27:20	<b>Timestamp:</b> Shall return a timestamp using the format Timestamp – Data Structure for Get Features defined in figure 270 containing the timestamp for the time when the most recent Get Log Page command requesting the Persistent Event Log and with the Continue bit set to '0' was processed.																											
43:28	<b>Power on Hours (POH):</b> This field indicates the number of power-on hours at the time the Persistent Event log was retrieved. This may not include time that the controller was powered and in a non-operational state.																											
51:44	<b>Power Cycle Count:</b> contains the number of power cycles for this controller																											
53:52	<b>PCI Vendor ID (VID):</b> This is the same value as reported in the Identify Controller Data Structure PCI Vendor ID field (i.e., bytes 01:00).																											
55:54	<b>PCI Subsystem Vendor ID (SSVID):</b> This is the same value as reported in the Identify Controller Data Structure PCI Subsystem Vendor ID field (i.e., bytes 03:02).																											
75:56	<b>Serial Number (SN):</b> This field contains the same value as reported in the Serial Number field of the Identify Controller Data Structure, bytes 23:04.																											
115:76	<b>Model Number (MN):</b> This field contains the same value as reported in the Model Number field of the Identify Controller Data Structure, bytes 63:24.																											
371:116	<b>NVM Subsystem NVMe Qualified Name (SUBNQN):</b> This field contains the same value as reported in the NVM Subsystem NVMe Qualified Name field of the Identify Controller Data Structure, bytes 1023:768. If the NVM Subsystem NVMe Qualified Name field of the Identify Controller Data Structure is not supported, then all bytes of this field shall be set to 00h.																											
479:372	Reserved																											
511:480	<p><b>Supported Events Bitmap:</b> This field contains a bitmap indicating support for the persistent event log events. Each bit in the bitmap corresponds to the value for the event type (refer to Figure 205) (e.g., bit 222 decimal, DEh, corresponds to event type value DEh, Vendor Specific Event), A bit set to '1' indicates that the corresponding event is supported. A bit cleared to '0' indicates that the corresponding event is not supported.</p> <table><tr><th>Bit</th><th>Description</th><th>Reference</th></tr><tr><td>255:223</td><td>Reserved</td><td></td></tr><tr><td>222</td><td><b>Vendor Specific Event Supported</b></td><td>5.14.1.13.1.5</td></tr><tr><td>221:005</td><td>Reserved</td><td></td></tr><tr><td>004</td><td><b>Power-on or Reset Event Supported</b></td><td>5.14.1.13.1.4</td></tr><tr><td>003</td><td><b>Timestamp Change Event Supported</b></td><td>5.14.1.13.1.3</td></tr><tr><td>002</td><td><b>Firmware Commit Event Supported</b></td><td>5.14.1.13.1.2</td></tr><tr><td>001</td><td><b>SMART / Health Log Snapshot Event Supported</b></td><td>5.14.1.13.1.1</td></tr><tr><td>000</td><td>Reserved</td><td></td></tr></table>	Bit	Description	Reference	255:223	Reserved		222	<b>Vendor Specific Event Supported</b>	5.14.1.13.1.5	221:005	Reserved		004	<b>Power-on or Reset Event Supported</b>	5.14.1.13.1.4	003	<b>Timestamp Change Event Supported</b>	5.14.1.13.1.3	002	<b>Firmware Commit Event Supported</b>	5.14.1.13.1.2	001	<b>SMART / Health Log Snapshot Event Supported</b>	5.14.1.13.1.1	000	Reserved	
Bit	Description	Reference																										
255:223	Reserved																											
222	<b>Vendor Specific Event Supported</b>	5.14.1.13.1.5																										
221:005	Reserved																											
004	<b>Power-on or Reset Event Supported</b>	5.14.1.13.1.4																										
003	<b>Timestamp Change Event Supported</b>	5.14.1.13.1.3																										
002	<b>Firmware Commit Event Supported</b>	5.14.1.13.1.2																										
001	<b>SMART / Health Log Snapshot Event Supported</b>	5.14.1.13.1.1																										
000	Reserved																											
<b>Persistent Event Log Events</b>																												
(M-1)+512:512	<b>Persistent Event 0:</b> This field contains the first persistent event log entry (refer to Figure 204) where M is the length of this persistent event.																											
...	...																											
(TLL-1):(TLL-K)	<b>Persistent Event N:</b> This field contains the last persistent event log entry (refer to Figure 204) where K is the length of this persistent event and TLL is the size specified in the Total Log Length field.																											

The format of the Persistent Events in the Persistent Event log is shown in Figure 204.

**Figure 204: Persistent Event Format**

Bytes	Description
<b>Persistent Event Log Event Header</b>	
00	<b>Event Type:</b> This field indicates the event type for this entry. Refer to section 5.14.1.13.1 for the definition of the event types.
01	<b>Event Type Revision:</b> This field contains a number indicating the revision of the event data structure for the event indicated by the Event Type field that this event data complies with.
02	<b>Event Header Length (EHL):</b> This field contains the length in bytes of the event header information that follows. The total length of the event header in bytes is the value in this field plus 3. The host should use the value in this field to calculate the offset to the start of the Vendor Specific Information field.
03	Reserved
05:04	<b>Controller Identifier:</b> This field contains the NVM subsystem unique controller identifier for the controller that created this event. If the event is controller specific, then the event data is associated with this controller. If the event is not controller specific, then this is the controller that the NVM subsystem selected for creating the event.
13:06	<b>Event Timestamp:</b> This field contains a timestamp using the format Timestamp – Data Structure for Get Features defined in Figure 270 containing the timestamp for the time when this event occurred.
19:14	Reserved
21:20	<b>Vendor Specific Information Length (VSIL):</b> This field indicates the length in bytes of the Vendor Specific Information. If no Vendor Specific Information is present, then this field shall be set to 0000h. The length of the Vendor Specific Information is included in the Event Length field (bytes 23:22). Information associated with this event that is not able to be described in the event data structure fields may be reported in Vendor Specific Information fields in this event.
23:22	<b>Event Length (EL):</b> This field indicates the length in bytes of the vendor specific information, if any, and the persistent event log event data that follows. The total length of the event in bytes is the value in this field plus the value in the Event Header Length field plus 3.
<b>Vendor Specific Information, if any</b>	
EHL+2+VSIL:EHL+3	<b>Vendor Specific Information:</b> This field contains the vendor specific information, if any. This field is omitted if there is no vendor specific information (i.e., if VSIL is cleared to 0000h).
<b>Persistent Event Log Event Data</b>	
EHL+EL+2 :EHL+3+VSIL	<b>Event Data:</b> This field contains persistent event log events (refer to section 5.14.1.13.1).

#### 5.14.1.13.1 Persistent Event Log Events

The values that may be reported in the Event Type field (refer to section 5.14.1.13) of the event header for events in the Persistent Event log are defined in Figure 205.

**Figure 205: Persistent Event Log Event Types**

Type	O/M <sup>1</sup>	Event	Reference Section
00h		Reserved	
01h	NOTE 2	SMART / Health Log Snapshot	5.14.1.13.1.1
02h	M	Firmware Commit	5.14.1.13.1.2
03h	M	Timestamp Change	5.14.1.13.1.3
04h	M	Power-on or Reset	5.14.1.13.1.4
05h to DDh		Reserved	
DEh	O	Vendor Specific Event	5.14.1.13.1.5
DFh to FFh		Reserved	
NOTES: 1. O/M definition: O = Optional, M = Mandatory 2. Mandatory for NVMe over PCIe, Optional for NVMe over Fabrics			

#### 5.14.1.13.1.1 SMART / Health Log Snapshot Event (Event Type 01h)

NVM subsystems that support the Persistent Event Log shall create a SMART / Health Log Snapshot Event:

- a) If virtualization management is not implemented, then for every controller in the NVM subsystem; or
- b) If virtualization management is implemented, then for every primary controller,

at least once every 24 power on hours at a time determined by the controller.

The SMART / Health Log Snapshot Event shall set the Persistent Event Log Event Header:

- a) Event Type field to 01h; and
- b) Event Type Revision field to 01h.

The SMART / Health Log Snapshot Event data is specified in Figure 206.

**Figure 206: SMART / Health Log Snapshot Event Data Format (Event Type 01h)**

Bytes	Description
511:0	<b>Event Data:</b> Contains a snapshot of the SMART/Health Information Log data specified in Figure 186

#### 5.14.1.13.1.2 Firmware Commit Event (Event Type 02h)

A firmware commit event shall be recorded in the Persistent Event Log when a Firmware Commit command is completed. The Firmware Commit Event shall set the Persistent Event Log Event Format Header:

- a) Event Type field to 02h; and
- b) Event Type Revision field to 01h.

The Firmware Commit Event data is specified in Figure 207.

**Figure 207: Firmware Commit Event Data Format (Event Type 02h)**

Bytes	Description
07:00	<b>Old Firmware Revision:</b> Contains the firmware revision of the active firmware before this firmware commit event.
15:08	<b>New Firmware Revision:</b> Contains the firmware revision for the firmware that was requested to become active.
16	<b>Firmware Commit Action:</b> Contains the value from the Commit Action field in the Firmware Commit command.
17	<b>Firmware Slot:</b> Contains the value from the Firmware Slot field in the Firmware Commit command.
18	<b>Status Code Type for Firmware Commit Command:</b> Contains the status code type from the completion queue entry for the Firmware Commit command.
19	<b>Status Returned for Firmware Commit Command:</b> Contains the status code from the completion queue entry for the Firmware Commit command.
21:20	<b>Vendor Assigned Firmware Commit Result Code:</b> Contains a vendor specific value that provides more information about the result of the firmware commit. A value of '0000h' indicates that no vendor assigned firmware commit result code is provided.

#### 5.14.1.13.1.3 Timestamp Change Event (Event Type 03h)

The Timestamp Change Event (refer to Figure 208) contains the current timestamp, reported in the event header, and the timestamp from the time at which the timestamp was changed (i.e., the old timestamp).

The Timestamp Change Event shall set the Persistent Event Log Event Format Header:

- a) Event Type field to 03h; and
- b) Event Type Revision field to 01h.

The Timestamp Change Event data is specified in Figure 208.

**Figure 208: Timestamp Change Event Format (Event Type 03h)**

Bytes	Description
07:00	<b>Previous Timestamp:</b> Contains a timestamp using the format Timestamp – Data Structure for Get Features as defined in Figure 270 containing the timestamp for the time immediately before the timestamp was changed (i.e., the old timestamp).
15:08	<b>Milliseconds Since Reset:</b> Contains the time since the last Controller Level Reset reported in milliseconds.

#### 5.14.1.13.1.4 Power-on or Reset Event (Event Type 04h)

A Power-on or Reset event shall be recorded in the Persistent Event Log when an NVM subsystem reset (e.g., due to a power-on) or a controller level reset is completed. The Power-on or Reset Event reports information about resets due to power-on or other events that cause resets (refer to section 7.3) followed by descriptors reporting information about the controller at the time the reset occurred, including timestamp values for all controllers for use in synchronization of timestamp values across controllers.

The controller shall set the Persistent Event Log Event Format Header:

- a) Event Type field to 04h; and
- b) Event Type Revision field to 01h.

The Power-on or Reset Event data is specified in Figure 209.

**Figure 209: Power-on or Reset Event (Event Type 04h)**

Bytes	Description
7:0	<b>Firmware Revision:</b> Contains the firmware revision that will be in effect when CC.EN transitions from '0' to '1'
EL-VSIL-1:8	<b>Reset Information List:</b> Contains a list of one or more Controller Reset Information descriptors (refer to Figure 210). If virtualization management is not implemented then the list shall contain a Controller Reset Information descriptor for every controller in the NVM subsystem. If virtualization management is implemented, then the list shall contain a Controller Reset Information descriptor for every primary controller. The Controller Reset Information descriptor is shown in Figure 210.

**Figure 210: Controller Reset Information descriptor**

Bytes	Description								
1:0	<b>Controller ID:</b> Contains the Controller ID for the controller with the timestamp in the Controller Timestamp field.								
2	<b>Firmware Activation:</b> Contains a code indicating if this event triggered a firmware activation. <table border="1"> <tr> <td>00h</td><td>Indicates that this event did not trigger a firmware activation on this controller.</td></tr> <tr> <td>01h</td><td>Indicates that new firmware was activated on this controller due to this power on or reset</td></tr> <tr> <td>02h</td><td>Indicates that an attempt to activate new firmware on this controller due to this power-on or reset failed</td></tr> <tr> <td>03h to FFh</td><td>Reserved</td></tr> </table>	00h	Indicates that this event did not trigger a firmware activation on this controller.	01h	Indicates that new firmware was activated on this controller due to this power on or reset	02h	Indicates that an attempt to activate new firmware on this controller due to this power-on or reset failed	03h to FFh	Reserved
00h	Indicates that this event did not trigger a firmware activation on this controller.								
01h	Indicates that new firmware was activated on this controller due to this power on or reset								
02h	Indicates that an attempt to activate new firmware on this controller due to this power-on or reset failed								
03h to FFh	Reserved								
3	<b>Operation in Progress:</b> Bits 7:1 are reserved.  Bit 0: A value of '1' indicates that a Format NVM command was in progress for a namespace attached to this controller when this reset event occurred. A value of '0' indicates that no Format NVM commands were in progress for any namespace attached to this controller when this reset event occurred.								
15:4	Reserved								
19:16	<b>Controller Power Cycle:</b> Contains the power cycle count for the controller indicated in the Controller ID field.								
27:20	<b>Power on milliseconds:</b> Contains the power on hours in milliseconds since being manufactured. This may not include time that the controller was powered and in a non-operational power state.  The resolution of this field is vendor specific (e.g., an NVM subsystem that only counts power on time in hours will only report values corresponding to whole hours).								

Bytes	Description
35:28	<b>Controller Timestamp:</b> Contains a timestamp using the format Timestamp – Data Structure for Get Features as defined in Figure 270 containing the timestamp for the controller specified in the Controller ID field at the time when this event occurred.

#### 5.14.1.13.1.5 Vendor Specific Event (Event Type DEh)

The Vendor Specific Event (refer to Figure 211) contains a set of Vendor Specific Event Descriptors that describe an event that the vendor has determined is a significant event which should be reported to a host in the persistent event log and that is not described by any of the other persistent event log events.

The Vendor Specific Event Descriptors follow the format shown in Figure 212 and contain vendor specific data of the type indicated in the Vendor Specific Event Data Type field of the Vendor Specific Event Descriptor.

The controller shall set the Vendor Specific Event Format Header:

- a) Event Type field to DEh; and
- b) Event Type Revision field to 01h.

The Vendor Specific Event data is specified in Figure 211.

**Figure 211: Vendor Specific Event Format (Event Type DEh)**

Bytes	Description
M-1:0	<b>Vendor Specific Event Descriptor 0:</b> Contains the first vendor specific event descriptor (refer to Figure 212). Where M is the length of this vendor specific event descriptor.
...	
EL-VSIL-1: EL-VSIL-K	<b>Vendor Specific Event Descriptor N:</b> Contains the last vendor specific event descriptor (refer to Figure 212). Where K is the length of this vendor specific event descriptor.

The format of the Vendor Specific Event Descriptor is shown in Figure 212.

**Figure 212: Vendor Specific Event Descriptor Format**

Bytes	Description
01:00	<b>Vendor Specific Event Code:</b> Contains a vendor specific code that uniquely identifies the type of event that is described in the data that follows. All vendor specific events of the same type should report the same Vendor Specific Event Code field value.
02	<b>Vendor Specific Event Data Type:</b> Contains a code indicating the type of data reported in the Vendor Specific Event Data field (refer to Figure 213).
03	Reserved
05:04	<b>Vendor Specific Event Data Length:</b> Contains the length in bytes of the Vendor Specific Event Data field.
M+5:06	<b>Vendor Specific Event Data:</b> Contains vendor specific data that is associated with this event and is of the type specified in the Vendor Specific Event Data Type field. Where M is the length of the vendor specific event data.

The data types that are able to be reported in a Vendor Specific Event Descriptor are shown in Figure 213.

**Figure 213: Vendor Specific Event Data Type Codes**

Code	Description
00	Reserved
01	<b>Event Name:</b> The Vendor Specific Event Data field contains a null terminated ASCII string with a vendor specific name for the value in the Vendor Specific Event Code field. The value reported in this field shall be the same for every vendor specific event containing a vendor specific event code that is the same as the value in the Vendor Specific Event Code field in this event. If this type is reported in a Vendor Specific Event Descriptor, then it shall be the first vendor specific event descriptor in that event.
02	<b>ASCII String:</b> The Vendor Specific Event Data field contains a null terminated ASCII string.
03	<b>Binary:</b> The Vendor Specific Event Data field contains binary data. The byte ordering in the binary data is determined by the NVM subsystem vendor.
04	<b>Signed Integer:</b> The Vendor Specific Event Data field contains a 64-bit signed integer in two's complement form.