LEGAL NOTICE:

© Copyright 2008 to 2023 NVM Express, Inc. ALL RIGHTS RESERVED.

This 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 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: **"© 2008 to 2023 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

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

NVM Express® Technical Proposal

Technical Proposal ID	**TP4165 Tracking LBA Allocation with Granularity**
Revision Date	**2023.09.07**
Builds on Specification(s)	**NVM Express® Base Specification 2.0c** **NVM Express® NVM Command Set Specification 1.0c**
References	**TP4095a Namespace Capability Reporting**

Technical Proposal Author(s)

Name	**Company**
Mike Allison, Judy Brock, Bill Martin, Dan Helmick, Hanjae Lee, Vipin Agrawal, Jiwon Chang, Mark Gaertner	Samsung
Lee Prewitt, Jake Oshins, Scott Lee	Microsoft
Amber Huffman, Chris Sabol, David Dunn, Nicolae Mogoreanu, Yuliya Tarnikova	Google
Paul Suhler	Kioxia

Technical Proposal Overview

This proposal adds the ability for a controller to report allocated LBAs managed at a granularity.

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Revision History

Revision Date	Change Description
2023.04.05	Initial draft
2023.04.06	Changes due to initial review in Technical WG: • Removed power of 2 on Get LBA Status command allocation granularity • Moved fix of staring LBA on subsequent LBA Status Descriptor Entries to Bug 96 to go into an ECN. • Removed Action Types Supported filed as moving this to another TP. • Fixed figure #s and references.
2023.04.11	Editorial changes: • Updated the location of the LBAAG field in section 4.2.1. Comments from Paul Suhler • Updated the definition of LBAAG field to include alignment is the same as the granularity. • Updated section 4.2.1 to state that the LBAs within the same granularity unit have the same Allocated LBA status. • Updated section 5.8.1 to indicate reporting support for Action Types 10h and 11h. Note that TPAR4168 is going to allow reporting all Action Types supported.
2023.04.13	Addressed the items defined by Technical WG review. • Added a minimum recommendation to the LBAAG field. • Moved text around the Get LBA Status command to make it clear that the command data words are the same definition for each Action Type value. • Changed the Status field in the LBA Status Descriptor Entry to be 3 bits • Added a specific requirement that the Get LBA Status command is supported if the capability it supported. • Simplified the text for the additional support required if Action Types 10h and 11h are supported. Incorporated editorial feedback from Dan Hubbard.
2023.04.18	Clearly identified the support requirements of the Action Types.
2023.04.20	Added the reserved coded values to Figure 112.
20234.04.21	Editorial changes to specify “shall” and “may” for the reporting of LBAs deallocated as opposed to factual statements.
2023.05.02	• Removed old comments. • Updated the definition of the returned logical block allocation to emphasize if all LBAs within the alignment and granularity are deallocated, then the Get LBA Status command shall not return any of those LBAs. • Removed large sections of text that are not modified by this TP. • Replaced “LBA” with “logical block” where appropriate.
2023.05.04	• Corrected plurality.

2023.05.11	• Separated out the different Action Types into their own sections for the Get LBA Status command.
	• Moved the definition of the LBA Status Descriptor List to the command completion section.
	• Updated the requirements for Action Type 02h return data to be relative the tracked data instead of per LBA with the LBA range of the Get LBA Status command.
	• For Action Type 02h clarified the reporting LBAs when the LBAs within a tracked granularity:
	  ○ are all deallocated,
	  ○ are all allocated, and
	  ○ are a mixture of allocated and deallocated.

2023.05.18	• Moved LBA Status Descriptor List to its own section.
2023.06.21	• Editorial feedback from Judy Brock • Editorial feedback from Scott Lee • With TP4168 being withdrawn, add a support bit for the Action Type value 02h.

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Revision Date	Change Description
2023.06.22	• Added a column to the Status field in the LBA Status Descriptor Entry to indicate which calues are allowed to be returned by which Action Type field values.
2023.09.06	• Integrated
2023.09.07	• Minor integration cleanup

Description for Changes Document for NVM Express NVM Command Set Specification 1.0c

New Features/Feature Enhancements/Required Changes:

- Add the ability for a controller to report the allocated LBAs in a namespace
    - Description of change.
        - The Get LBA Status command supports a new Action Type to allow a host to request reporting of allocated LBAs of a namespace. In addition, a controller identifies which Action Types are supported by the Get LBA Status command.
    - **New requirement and incompatible change**
        - A controller is allowed to support the Get LBA Status command and not support the Action Type values of 10h and 11h for reporting Tracked LBAs and Untracked LBAs.
    - References
        - Technical Proposal TP4165

Markup Conventions:

Black:	Unchanged (however, hot links are removed)
~~Red Strikethrough~~:	Deleted
Blue:	New
Blue Highlighted:	TBD values, anchors, and links to be inserted in new text.
~~Purple Strikethrough~~:	Existing text moved to a different location.
Purple:	Inserted existing text moved from a different location.
Orange:	Text from a reference specification (e.g., another ratified TP)
<Green Bracketed>:	Notes to editor

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Description of Specification Changes for the NVM Express NVM Command Set Specification 1.0c

Modify a portion of section 4 as shown below:

…

4 Admin Commands for the NVM Command Set

…

4.1 Admin Command behavior for the NVM Command Set

…

4.1.5 Identify Command

…

4.1.5.3 I/O Command Set Specific Identify Namespace Data Structure (CNS 05h)

…

Figure 100: NVM Command Set I/O Command Set Specific Identify Namespace Data Structure (CSI 00h)

Bytes	O/M [1]	Description	Reported [2]
…			
Extended LBA Format (refer to section 5.5)			
15:12	O	**Extended LBA Format 0 Support (ELBAF0):** This field indicates additional LBA Format 0 information related to the LBA Format 0 Support (LBAF0) field in the Identify Namespace data structure. The Extended LBA format field is defined in Figure 101.	Yes
…			
267:264	O	**Extended LBA Format 63 Support (ELBAF63):** This field indicates additional LBA Format 63 information related to the LBA Format 63 Support (LBAF63) field in the Identify Namespace data structure. The Extended LBA format field is defined in Figure 101.	Yes
…			
295:292	O	**Tracked LBA Allocation Granularity (TLBAAG):** This field indicates the alignment and granularity, in logical blocks, for the reporting of allocated LBAs for the namespace by the Get LBA Status command (refer to section 4.2.1). If this field is cleared to the value of 0h, then the alignment and granularity are not reported. It is recommended that value of this field multiplied by the logical block data size for this namespace be greater than or equal to 4 KiB.	Yes
4095: 296~~268~~	O	Reserved	
Notes: 1. O/M definition: O = Optional, M = Mandatory. 2. Identifies fields that report information for the Identify command when querying the capabilities of LBA formats.			

…

4.1.5.4 I/O Command Set Specific Identify Controller Data Structure (CNS 06h, CSI 00h)

…

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Figure 102: I/O Command Set Specific Identity Controller Data Structure for the NVM Command Set

<table>
<tr><th>Bytes</th><th>O/M [1]</th><th>Description</th></tr>
<tr><td colspan="3">…</td></tr>
<tr><td>15:08</td><td>O</td><td>Dataset Management Size Limit (DMSL): If bit 2 in the Optional NVM Command Support (ONCS) field is set to ‘1’ then:
a) a non-zero value in this field indicates the recommended maximum total number of logical blocks for a Dataset Management command (refer to section 3.2.3).
b) a value of 0h in this field indicates that no recommended maximum total number of logical blocks for a Dataset Management command is reported.
If bit 2 in the ONCS field is cleared to ‘0’, then:
a) a non-zero value in this field indicates that the controller supports the Dataset Management command with the maximum total number of logical blocks limit indicated by this field (refer to section 3.2.3); and
b) a value of 0h in this field indicates that the controller does not support the Dataset Management command.</td></tr>
<tr><td>17:16</td><td>M</td><td>Reserved</td></tr>
<tr><td>19:18</td><td>M</td><td>Admin Optional Command Support (AOCS): This field indicates the optional Admin commands and features supported by the controller.
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>15:01</td><td>Reserved</td></tr>
<tr><td>00</td><td>Reporting Allocated LBA Supported (RALBAS): If this bit is set to ‘1’, then the controller supports the Get LBA Status capability with the Action Type value of 02h (refer to section 4.2.1). If this bit is cleared to ‘0’, then the controller does not support the Get LBA Status capability with the Action Type value of 02h.</td></tr>
</table></td></tr>
<tr><td>4095:20</td><td>M</td><td>Reserved</td></tr>
<tr><td colspan="3">Notes:
1. O/M definition: O = Optional, M = Mandatory.</td></tr>
</table>

…

4.2 I/O Command Set Specific Admin commands

…

4.2.1 Get LBA Status command

The Get LBA Status command requests information about LBAs ~~Potentially Unrecoverable LBAs~~ (refer to section 5.8.1). If the Get LBA Status command completes successfully, then the LBA Status Descriptor List, defined in Figure 111, is returned in the data buffer for that command.

The Get LBA Status command uses the Data Pointer, Command Dword 10, Command Dword 11, Command Dword 12, and Command Dword 13 fields. All other command specific fields are reserved.

The Maximum Number of Dwords (MNDW) field contains the maximum number of dwords to return. Upon successful command completion, the actual amount of data returned by the controller is indicated by the Number of LBA Status Descriptors (NLSD) field in the LBA Status Descriptor List.

~~A controller identifies Potentially Unrecoverable LBAs using the following two report types:~~

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2023 NVM Express, Inc.

a) **Tracked LBAs:** a list of Potentially Unrecoverable LBAs associated with physical storage. These may be discovered through a background scan where the controller examines the media in the background or discovered through other means. The Tracked LBA list is able to be returned without significant delay; or

b) **Untracked LBAs:** a list of Potentially Unrecoverable LBAs generated by a scan originated by a Get LBA Status command with the ATYPE field set to 10h. The controller scans internal data structures related to the specified range of LBAs to determine which LBAs are Potentially Unrecoverable LBAs. The controller may use this scan to determine which LBAs in which namespaces are affected by a component (e.g., die or channel) failure. Significant delays may be incurred during the processing of a Get LBA Status command with the ATYPE field set to 10h. After discovery of Untracked LBAs, they may or may not be added to the list of Tracked LBAs.

In response to a Get LBA Status command, the controller returns LBA Status Descriptors that describe LBAs written by a Write Uncorrectable command in addition to any other LBAs that may return an Unrecovered Read Error status discovered through other mechanisms. The list of Tracked LBAs and the list of Untracked LBAs may be included in LBA Status Descriptor Entries that describe LBAs written by a Write Uncorrectable command. If an LBA Status Descriptor Entry describes only LBAs written by a Write Uncorrectable command, then bits 1:0 in the Status field should be set to 11b in that entry.

Figure 107: Get LBA Status – Data Pointer

Bits	Description
127:00	**Data Pointer (DPTR):** This field specifies the start of the data buffer. Refer to the Common Command Format figure in the NVM Express Base Specification for the definition of this field.

Figure 108: Get LBA Status – Command Dword 10 and Command Dword 11

Bits	Description
63:00	**Starting LBA (SLBA):** This field indicates the 64-bit address of the first logical block addressed by this command. Command Dword 10 contains bits 31:00; Command Dword 11 contains bits 63:32.

Figure 109: Get LBA Status – Command Dword 12

Bits	Description
31:00	**Maximum Number of Dwords (MNDW):** This field specifies the maximum number of dwords to return. This is a 0's based value.

Figure 110: Get LBA Status – Command Dword 13

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>31:24</td><td>Action Type (ATYPE): This field specifies the mechanism the controller uses in determining the LBA Status Descriptors to return as defined in Figure 112.
<table>
<tr><th>Value</th><th>M/O/P[1]</th><th>Description</th><th>Reference Section</th></tr>
<tr><td>02h</td><td>O</td><td>Return tracked allocated LBAs in the specified range</td><td>4.2.1.TBD.1</td></tr>
<tr><td>10h</td><td>O</td><td>Perform a scan and return Untracked LBAs and Tracked LBAs in the specified range</td><td rowspan="2">4.2.1.TBD.2</td></tr>
<tr><td>11h</td><td>O</td><td>Return Tracked LBAs in the specified range</td></tr>
<tr><td>All others</td><td></td><td>Reserved</td><td></td></tr>
<tr><td colspan="4">Notes:
1. O = Optional, M = Mandatory, P = Prohibited</td></tr>
</table></td></tr>
<tr><td>23:16</td><td>Reserved</td></tr>
</table>

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Figure 110: Get LBA Status – Command Dword 13

Bits	Description
15:00	**Range Length (RL):** This field specifies the length of the range of contiguous LBAs, beginning at Starting LBA (SLBA), that the action specified in the Action Type (ATYPE) field shall be performed on. A value of 0h in this field specifies the length of a range beginning at Starting LBA and ending at Namespace Size (NSZE) minus 1 (refer to Figure 97).

4.2.1.TBD Get LBA Status Action Type Mechanisms

4.2.1.TBD.1 Return Allocated LBAs (Action Type 02h)

The Get LBA Status command specifying an Action Type of 02h requests information about allocated LBAs. The controller shall return information about allocated LBAs in the range specified in the Get LBA Status command for the namespace specified in the NSID field. A controller is allowed to track the Allocated LBAs on the alignment and granularity specified by the Tracked LBA Allocation Granularity (TLBAAG) field in the I/O Command Set Specific Identify Namespace data structure for the NVM Command Set (refer to Figure 100).

If all the logical blocks within the same alignment and granularity unit are deallocated, then all LBAs in that granularity and alignment unit that are in the LBA range specified in the Get LBA Status command shall not be reported as being allocated in the returned information for the Get LBA Status command.

If all of the logical blocks within the same alignment and granularity unit are allocated, then all LBAs in that granularity and alignment unit that are in the LBA range specified in the Get LBA Status command shall be reported as being allocated in the returned information for the Get LBA Status command as being allocated.

If within the same alignment and granularity unit:

- one or more logical blocks are not deallocated; and
- one or more logical blocks are deallocated,

then all LBAs in that granularity and alignment unit that are in the LBA range specified in the Get LBA Status command shall be reported as being allocated in the returned information for the Get LBA Status command as being allocated.

4.2.1.TBD.2 Potentially Unrecoverable LBAs (Action Type 10h and 11h)

A Get LBA Status command specifying an Action Type of 10h or 11h requests information about Potentially Unrecoverable LBAs.

A controller identifies Potentially Unrecoverable LBAs using the following two report types:

a) **Tracked LBAs:** a list of Potentially Unrecoverable LBAs associated with physical storage. These may be discovered through a background scan where the controller examines the media in the background or discovered through other means. The Tracked LBA list is able to be returned without significant delay; or

b) **Untracked LBAs:** a list of Potentially Unrecoverable LBAs generated by a scan originated by a Get LBA Status command with the ATYPE field set to 10h. The controller scans internal data structures related to the specified range of LBAs to determine which LBAs are Potentially Unrecoverable LBAs. The controller may use this scan to determine which LBAs in which namespaces are affected by a component (e.g., die or channel) failure. Significant delays may be incurred during the processing of a Get LBA Status command with the ATYPE field set to 10h. After discovery of Untracked LBAs, they may or may not be added to the list of Tracked LBAs.

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

If the value in the Action Type (ATYPE) field is set to 10h, then:

a) the controller shall generate a list of Untracked LBAs as described in this section;
b) the controller shall return Untracked LBAs and Tracked LBAs in the range specified in the Get LBA Status command for the namespace specified in the Namespace Identifier (CDW1.NSID);
c) the controller shall remove all LBAs in the range specified in the Get LBA Status command, which prior to processing the Get LBA Status command were successfully re-written, from relevant internal data structures (e.g., internal Tracked LBA list);
d) the controller shall ensure that any such successfully re-written logical blocks are not reported in any LBA Status Descriptor Entries returned by the Get LBA Status command unless, after having been removed from relevant internal data structures and prior to processing the Get LBA Status command, those LBAs were newly detected as being Potentially Unrecoverable LBAs; and
e) the list of Untracked LBAs returned by the Get LBA Status command may be discarded by the controller or added to the Tracked LBA list once the command has completed.

If the value in the Action Type (ATYPE) field is set to 11h, then the controller shall:

a) return Tracked LBAs in the range specified in the Get LBA Status command for the namespace specified in the Namespace Identifier (CDW1.NSID) field;
b) remove all LBAs in the range specified in the Get LBA Status command, which prior to processing the Get LBA Status command were successfully re-written, from relevant internal data structures (e.g., internal Tracked LBA list);
c) ensure that any such successfully re-written logical blocks are not reported in any LBA Status Descriptor Entries returned by the Get LBA Status command unless, after having been removed from relevant internal data structures and prior to processing the Get LBA Status command, those LBAs were newly detected as being Potentially Unrecoverable LBAs; and
d) not perform a foreground scan to generate and return Untracked LBAs.

In response to a Get LBA Status command, the controller returns LBA Status Descriptors that describe LBAs written by a Write Uncorrectable command in addition to any other LBAs that may return an Unrecovered Read Error status discovered through other mechanisms. The list of Tracked LBAs and the list of Untracked LBAs may be included in LBA Status Descriptor Entries that describe LBAs written by a Write Uncorrectable command. If an LBA Status Descriptor Entry describes only LBAs written by a Write Uncorrectable command, then bits 1:0 in the Status field should be set to 11b in that entry.

4.2.1.TBD1 LBA Status Descriptor List

Figure 111 defines the LBA Status Descriptor List returned in the data buffer.

Figure 111: LBA Status Descriptor List

Bytes	Description
03:00	**Number of LBA Status Descriptors (NLSD):** This field indicates the number of LBA Status Descriptor Entries returned by the controller in this data structure. A value of 0h in this field indicates that no LBA Status Descriptor Entries are returned.

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Figure 111: LBA Status Descriptor List

Bytes	Description
04	**Completion Condition (CMPC):** This field indicates the condition that caused completion of the Get LBA Status command. **Code** / **Definition** 0h: No indication of the completion condition. 1h: **INCOMPLETE:** The command completed as a result of transferring the number of Dwords specified in the MNDW field and: • for any ATYPE~~for ATYPE set to 10h, or ATYPE set to 11h,~~ additional LBA Status Descriptor Entries are available to transfer that are associated with the specified LBA range; or • ~~F~~for ATYPE set to 10h, the scan did not complete. 2h: **COMPLETE:** The command completed as a result of completing the action specified in the Action Type field over the number of logical blocks specified in the Range Length field and there are no additional LBA Status Descriptor Entries available to transfer that are associated with the specified range. All others: Reserved
07:05	Reserved
23:08	**LBA Status Descriptor Entry 0:** This field contains the first LBA Status Descriptor Entry in the list, if present.
39:24	**LBA Status Descriptor Entry 1:** This field contains the second LBA Status Descriptor Entry in the list, if present.
…	**…**
(N*16+23): (N*16+8)	**LBA Status Descriptor Entry N:** This field contains the N+1 LBA Status Descriptor Entry in the list, if present.

Figure 112: LBA Status Descriptor Entry

Bytes	Description
07:00	**Descriptor Starting LBA (DSLBA):** This field indicates the 64-bit address of the first logical block of the LBA range for which this LBA Status Descriptor Entry reports LBA status.
11:08	**Number of Logical Blocks (NLB):** This field indicates the number of contiguous logical blocks reported in this LBA Status Descriptor Entry. The controller should perform the action specified in the Action Type field in such a way that the value in this field reports the largest number of contiguous logical blocks possible (i.e., multiple consecutive LBA Status Descriptor Entries should not report contiguous LBAs that span those entries, but rather, LBA Status Descriptor Entries should be consolidated into the fewest number of LBA Status Descriptor Entries possible). This is a 0's based value.
12	Reserved

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Figure 112: LBA Status Descriptor Entry

<table>
<tr><th>Bytes</th><th>Description</th></tr>
<tr><td>13</td><td>Status: This field contains information about this LBA range.
<table>
<tr><th>Bits</th><th>Definition</th></tr>
<tr><td>7:
3~~2~~</td><td>Reserved</td></tr>
<tr><td>2~~1~~:0</td><td>This field ~~These bits~~ indicates information about the logical blocks indicated in this LBA Status Descriptor Entry for the Action Type field values supported.
<table>
<tr><th>Value</th><th>Definition</th><th>Used with Action Type field Values</th></tr>
<tr><td>000b</td><td>Each logical block may:<ul><li>report Unrecovered Read Error status as a result of media errors;</li><li>be a logical block for which the most recent write to the logical block was a Write Uncorrectable command; or</li><li>be read successfully.</li></ul></td><td>10h and 11h</td></tr>
<tr><td>001b</td><td>Each logical block may:<ul><li>report Unrecovered Read Error status as a result of media errors; or</li><li>be a logical block for which the most recent write to the logical block was a Write Uncorrectable command.</li></ul></td><td>10h and 11h</td></tr>
<tr><td>010b</td><td>~~Reserved~~
One or more of the reported logical blocks are allocated.</td><td>02h</td></tr>
<tr><td>011b</td><td>Each logical block is a~~:~~<ul><li>logical blo~~s~~ck for which the most recent write to the logical block was a Write Uncorrectable command.</li></ul><Remove the bullet and just make 1 sentence></td><td>10h and 11h</td></tr>
<tr><td>100b to 111b</td><td>Reserved</td><td></td></tr>
</table></td></tr>
</table></td></tr>
<tr><td>15:14</td><td>Reserved</td></tr>
</table>

The Descriptor Starting LBA (DSLBA) field in the first LBA Status Descriptor Entry returned in the LBA Status Descriptor List shall contain the lowest numbered LBA that is greater than or equal to the value specified in the Starting LBA field in the Get LBA Status command.

For subsequent LBA Status Descriptor Entries, the contents of the Descriptor Starting LBA field shall contain the value of the lowest numbered LBA meeting the requirements for the specified Action Type value that is greater than the sum of the values in:

a) the Descriptor Starting LBA field in the previous LBA Status Descriptor Entry; and
b) the Number of Logical Blocks field in the previous LBA Status Descriptor Entry.

4.2.1.1 Command Completion

When the command is completed, the controller posts a completion queue entry to the Admin Completion Queue indicating the status for the command.

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Modify a portion of section 5 as shown below:

5 Extended Capabilities

…

5.8 Command Set Specific Capability

5.8.1 Get LBA Status

The Get LBA Status capability enables the host to obtain information about LBAs:

- Identifying LBAs that may be allocated in a namespace provides the host with the ability to minimize the user data accessed when copying that namespace for use cases such as namespace migration and snapshots. The Get LBA Status capability provides the host the ability to identify logical blocks that may be allocated in a namespace.
- Potentially Unrecoverable LBAs are LBAs that, when read, may result in the command that caused the media to be read being aborted with a status code of Unrecovered Read Error. The Get LBA Status capability provides the host with the ability to identify Potentially Unrecoverable LBAs. The logical block data is able to be recovered from another location and re-written.

If the Get LBA Status capability is supported, then the controller shall support the Get LBA Status command.

~~To support the Get LBA Status capability, the NVM subsystem shall:~~

If Action Types 10h and 11h are supported (refer to the Get LBA Status Supported (GLSS) bit in the Optional Admin Command Support (OACS) field in the Identify Controller data structure), then the controller shall:

- ~~indicate support for the Get LBA Status capability in the Optional Admin Command Support (OACS) field in the Identify Controller data structure;~~
- ~~indicate~~ support ~~for~~ LBA Status Information Alert Notices ~~in~~ (refer to the Optional Asynchronous Events Supported (OAES) field in the Identify Controller data structure);
- support the LBA Status Information log page;
- ~~indicate~~ support ~~for~~ the Log Page Offset and extended Number of Dwords (i.e., 32 bits rather than 12 bits) in the Get Log Page command (refer to the Attributes field of the Identify Controller data structure); and
- support the LBA Status Information Attributes Feature~~;~~
- ~~support the Get LBA Status command; and~~
- ~~support the LBA Status Information Alert Notices event~~.

Prior to using ~~the~~ a Get LBA Status ~~capability~~ command with the Action Type (ATYPE) field values 10h or 11h:

- The host should use the Get Features and Set Features commands with the LBA Status Information Attributes Feature (refer to section 4.1.3.3) to retrieve and optionally configure the LBA Status Information Report Interval; and
- If the host wishes to receive LBA Status Information Alert asynchronous events, the host should enable LBA Status Information Alert Notices (refer to Figure 88).

…

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Description of Specification Changes for the NVM Express Base Specification 2.0c

Modify a portion of section 5 as shown below:

5 Admin Command Set

…

5.17 Identify command

…

5.17.2 Identify Data Structures

5.17.2.1 Identify Controller Data Structure (CNS 01h)

…

Figure 275: Identify – Identify Controller Data Structure, I/O Command Set Independent

<table>
<tr><th>Bytes</th><th>I/O[1]</th><th>Admin[1]</th><th>Disc[1]</th><th>Description</th></tr>
<tr><td colspan="5">…</td></tr>
<tr><td colspan="5">Admin Command Set Attributes & Optional Controller Capabilities</td></tr>
<tr><td>257:256</td><td>M</td><td>M</td><td>R</td><td>Optional Admin Command Support (OACS): This field indicates the optional Admin commands and features supported by the controller. Refer to section 3.2.1.

<table>
<tr><th>Bits</th><th>Definition</th></tr>
<tr><td>15:11</td><td>Reserved</td></tr>
<tr><td>10</td><td>Command and Feature Lockdown Supported (CFLS): I~~f~~ this bit is set to ‘1’, then the controller supports the Command and Feature Lockdown capability (refer to section 8.4). If this bit is cleared to ‘0’, then the controller does not support the Command and Feature Lockdown capability. The ~~This~~ value of this bit shall be the same for all controllers in the NVM subsystem.</td></tr>
<tr><td>9</td><td>Get LBA Status Supported (GLSS): I~~f~~ this bit is set to ‘1’, then the controller supports the Get LBA Status capability with the Action Type values of 10h and 11h (refer to the NVM Command Set Specification). If this bit is cleared to ‘0’, then the controller does not support the Get LBA Status capability with the Action Type values of 10h and 11h.</td></tr>
<tr><td>8</td><td>Doorbell Buffer Config Supported (DBCS): I~~f~~ this bit is set to '1', then the controller supports the Doorbell Buffer Config command. If this bit is cleared to '0', then the controller does not support the Doorbell Buffer Config command.</td></tr>
<tr><td>7</td><td>Virtualization Management Supported (VMS): I~~f~~ this bit is set to ‘1’, then the controller supports the Virtualization Management command. If this bit is cleared to ‘0’, then the controller does not support the Virtualization Management command.</td></tr>
<tr><td>6</td><td>NVMe-MI Send Receive Supported (NSRS): I~~f~~ this bit is set to ‘1’, then the controller supports the NVMe-MI Send and NVMe-MI Receive commands. If this bit is cleared to ‘0’, then the controller does not support the NVMe-MI Send and NVMe-MI Receive commands.</td></tr>
<tr><td>5</td><td>Directives Supported (DIRS): I~~f~~ this bit is set to ‘1’, then the controller supports Directives. I~~f~~ this bit is cleared to ‘0’, then the controller does not support Directives. A controller that supports Directives shall support the Directive Send and Directive Receive commands. Refer to section 8.7.</td></tr>
</table>
</td></tr>
</table>

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Figure 275: Identify – Identify Controller Data Structure, I/O Command Set Independent

Bytes	I/O[1]	Admin[1]	Disc[1]	Description
				4 **Device Self-test Supported (DSTS):** If this bit is set to '1', then the controller supports the Device Self-test command. If this bit is cleared to '0', then the controller does not support the Device Self-test command. 3 **Namespace Management Supported (NMS):** If this bit is set to '1', then the controller supports the Namespace Management capability (refer to section 8.11). If this bit is cleared to '0', then the controller does not support the Namespace Management capability. 2 **Firmware Download Supported (FWDS):** If this bit is set to '1', then the controller supports the Firmware Commit and Firmware Image Download commands. If this bit is cleared to '0', then the controller does not support the Firmware Commit and Firmware Image Download commands. 1 **Format NVM Supported (FNVMS):** If this bit is set to '1', then the controller supports the Format NVM command. If this bit is cleared to '0', then the controller does not support the Format NVM command. 0 **Security Send Receicve Supported (SSRS):** If this bit is set to '1', then the controller supports the Security Send and Security Receive commands. If this bit is cleared to '0', then the controller does not support the Security Send and Security Receive commands. ~~Bits 15:11 are reserved.~~ ~~Bit 10 if set to '1', then the controller supports the Command and Feature Lockdown capability (refer to section 8.4). If cleared to '0', then the controller does not support the Command and Feature Lockdown capability. This value shall be the same for all controllers in the NVM subsystem.~~ ~~Bit 9 if set to '1', then the controller supports the Get LBA Status capability (refer to the NVM Command Set Specification). If cleared to '0', then the controller does not support the Get LBA Status capability.~~ ~~Bit 8 if set to '1', then the controller supports the Doorbell Buffer Config command. If cleared to '0', then the controller does not support the Doorbell Buffer Config command.~~ ~~Bit 7 if set to '1', then the controller supports the Virtualization Management command. If cleared to '0', then the controller does not support the Virtualization Management command.~~ ~~Bit 6 if set to '1', then the controller supports the NVMe-MI Send and NVMe-MI Receive commands. If cleared to '0', then the controller does not support the NVMe-MI Send and NVMe-MI Receive commands.~~ ~~Bit 5 if set to '1', then the controller supports Directives. If cleared to '0', then the controller does not support Directives. A controller that supports Directives shall support the Directive Send and Directive Receive commands. Refer to section 8.7.~~ ~~Bit 4 if set to '1', then the controller supports the Device Self-test command. If cleared to '0', then the controller does not support the Device Self-test command.~~ ~~Bit 3 if set to '1', then the controller supports the Namespace Management capability (refer to section 8.11). If cleared to '0', then the controller does not support the Namespace Management capability.~~ ~~Bit 2 if set to '1', then the controller supports the Firmware Commit and Firmware Image Download commands. If cleared to '0', then the controller does not support the Firmware Commit and Firmware Image Download commands.~~ ~~Bit 1 if set to '1', then the controller supports the Format NVM command. If cleared to '0', then the controller does not support the Format NVM command.~~

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Figure 275: Identify – Identify Controller Data Structure, I/O Command Set Independent

Bytes	I/O[1]	Admin[1]	Disc[1]	Description
				~~Bit 0 if set to '1', then the controller supports the Security Send and Security Receive commands. If cleared to '0', then the controller does not support the Security Send and Security Receive commands.~~
…				

…

Modify a portion of section 8 as shown below:

8 Extended Capabilities

…

8.4 Command and Feature Lockdown

…

If the Command and Feature Lockdown capability is supported (i.e., the CFLS bit ~~10~~ in the OACS field in Figure 275 is set to '1'), then the controller shall support the Lockdown command and the Command and Feature Lockdown log.

…

8.7 Directives

…

Support for Directives is optional and is indicated by the Directives Supported (DIRS) bit in the Optional Admin Command Support (OACS) field in the Identify Controller data structure (refer to Figure 275).

If a controller supports Directives, then the controller shall:

- Indicate support for Directives in the Optional Admin Command Support (OACS) field by setting the DIRS bit to '1' in the Identify Controller data structure;

…

8.11 Namespace Management

The Namespace Management capability consists of the Namespace Management command (refer to section 5.23) and the Namespace Attachment command (refer to section 5.22). 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. The Namespace Management capability is intended for use during manufacturing or by a system administrator.

If the Namespace Management capability is supported, then the controller:

a) shall support the Namespace Management command and the Namespace Attachment command;
b) shall set the NMS bit ~~3~~ to '1' in the OACS field (refer to Figure 275);
c) should support the Namespace Attribute Changed asynchronous event (refer to Figure 147 and section 5.27.1.8); and
d) may support Namespace Granularity (refer to the NVM Command Set Specification).

…

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

8.26 Virtualization Enhancements

…

To support the Virtualization Enhancements capability, the NVM subsystem shall support the following:

- One or more primary controllers, each of which supports:
  - One or more secondary controllers;
  - A pool of unassigned Flexible Resources that supports allocation to a primary controller and dynamic assignment to its associated secondary controllers;
  - Two or more Private Resource queue pairs;
  - Indicate support for the Virtualization Management command by setting the VMS bit to '1' in the Optional Admin Command Support (OACS) field in the Identify Controller data structure;
  - The Virtualization Management command;
  - The Primary Controller Capabilities Structure defined in Figure 281 (Identify command with CNS value of 14h);
  - The Secondary Controller List defined in Figure 282 (Identify command with CNS value of 15h); and
  - The Namespace Management capability (refer to section 8.11);
- One or more secondary controllers; and
- Flexible Resources, each of which supports all of the following:
  - Assignment and removal by exactly one primary controller; and
  - Assignment to no more than one controller at a time.

…

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.