



**LEGAL NOTICE:**

© **Copyright 2008 to 2022 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 2022 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.

The NVM Express® design mark is a registered trademark of NVM Express, Inc.

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

## NVM Express Technical Proposal for New Feature

<b>Technical Proposal ID</b>	<b>TP 4095a Namespace Capability Reporting</b>
<b>Change Date</b>	<b>2022-04-28</b>
<b>Builds on Specification</b>	<b>NVM Express Base Specification 2.0a</b> <b>NVM Express NVM Command Set 1.0a</b> <b>NVM Express Key Value Command Set 1.0a</b> <b>NVM Express Zone Namespace Command Set 1.1a</b>
<b>References Specification</b>	<b>ECN101</b>

Technical Proposal Author(s)

<b>Name</b>	<b>Company</b>
Mike Allison, Ambika Krishnamoorthy, Jonathan Hughes, Kapil Karkra	Intel
Mike Allison, Judy Brock	Samsung

This proposal intends to:

- Define the common namespace capabilities.
- Define a mechanism to support LBA formats that do not report the same namespace attributes across all LBA formats and define a mechanism to report the namespace attributes of LBA formats without having to create the namespace.

### Revision History

<b>Revision Date</b>	<b>Change Description</b>
2021-01-21	Initial rough version
2021-02-24	Aligned to latest NVMe 2.0 specifications.
2021-02-25	Made all new CNS require the CSI field. Removed comments where WG approved text.
2021-02-26	Made the new CNS values optional. Modified the CNS Specific Identifier to be patterned after the Log Specific Field in the Get Log Page command in that the description of the identifier used is defined by the definition section of a CNS value.
2021-03-11	Accepted all changes and resolved comment. Ready for Phase 2 exit.
2021-03-25	Aligned to latest member review documents. Editorial changes due to Phase 3 review in Technical WG meeting.
2021-04-01	Corrected the data structure names. Minor edited during WG discussion.
2021-04-08	Changes the “may” to a “shall” on the format of the returned namespace data structures per WG request.
2021-04-22	Accepted all changes and converted all references/cross-references to text for the 30 day member review.
2021-04-27	Corrected the summing of the NLBAF field and the NULBAF field to account for one being 0-based and one being 1-based as reported by Edward Hseih.
2021-04-29	Modified figure X to refer to fields as the diagram to reference the value represented by the field and not just the field value.
2021-05-27	Correct pluralization and removed redundant words.

2021-06-03	Addressed comments from Gerry Houlder (Seagate). Approved in NVMe Technical WG
2021-06-03	Aligned to the released versions of NVMe 2.0 family of specifications. Removed large chunks of unchanged text. Removed ZNS Command Spec section 4.1.2 as change already applied. Corrected the order of section 5.17.2.4 in the NVMe Base Spec.
2021-06-21	Integrated into the NVMe Command Set Specification, revision 1.0; the NVMe Key Value Command Set specification, revision 1.0; the NVMe Zoned Namespace Command Set Specification, revision 1.1; and the NVMe Base Specification, revision 2.0.
2021-06-23	Removed all comments, accepted all changes, and converted references/cross-references to text.
2021-07-27	Aligned with NVMe 2.0 ECN 001 new definition of Format Index and updated the TP to align with the following specifications: <ul style="list-style-type: none"> <li>• NVM Express Base Specification 2.0a</li> <li>• NVM Command Set Specification 1.0a</li> <li>• Zoned Namespace Command Set Specification 1.1a</li> <li>• Key Value Command Set Specification 1.0a.</li> </ul> Judy Brock provided editorial updates.
2021-08-05	Updates the Capability Fields usage descriptions to clarify “yes” versus “no”. Updated the definition CNS Specific Identifier to align with the other specific field definitions.
2021-08-13	Removed all comments, accepted all changes, and converted references/cross-references to text.
2021-10-12	Updated the definition of FLBAS to reflect that bit 6:5 should be ignored based on the number of supported LBA Formats and not NLBAF. Updated the definition of the LBAFEE field to refer to section 5.TBD for the definition of the number of supported LBA Formats.
2021-10-14	Minor editorial changes during NVMe Technical Work Group
2021-10-18	Replaced “LBA Format Index” and “KV Format Index” with “Format Index” as the definition of “Format Index” in each specification defines the difference. Fixed the Identify command with CNS 09h and 0Ah to indicate the capability fields returned data is based on the specified Format Index.
2021-10-21	Corrected spelling of “Format Index”. Updated the MAR and MOR fields to the released text in the Zoned Namespace Command Set Specification 1.1a.
2021-11-22	Addressed Gerry Houlder’s and Judy Brock’s comments to the member review.
2021-11-23	Used the word “total” in reference to the number of LBA formats supported since a sum is required.
2021-12-03	Removed comments that were reviewed. Removed the issue with LBAFEE field that is about TP4068 and not part of this TP’s new capability and that issue will be put into a different TP. Changed “Capability Fields” to “Host-Selectable Attribute”.
2021-12-22	NMIC Report column changed to Yes. Modified the CNS 08h to scope all namespaces and not be tied to LBA Formats when NSID is set to FFFFFFFh. Modified the dates for 2022. Updated section #;s for the navigation pane.
2021-12-23	Fixed the list of CNS values where the “common” definition was updated in the description of changes.
2021-12-2	Updated figure Y and the text that introduces figure Y.
2021-12-28	Changed the name of the column to Reported. Made the NMIC field a Yes for Reported. Editorial fixes.
2022-1-5	Added a note to the Reported columns.
2022-1-6	Removed comments and accepted all changed for member review.
2022-1-13	Added clarification to the Zoned Namespace Command Set specification the Command Set Index for both the NVM Command Set (i.e., 00h) and the Zoned Namespace Command Set (i.e., 02h) are utilized by the Zoned Namespace Command Set.
2022-1-14	Editorial changes to the Notes in figure 47 of the Zoned Namespace Command Set Specification. Updated the Navigation pane to show each level of headings to specific changes.

2022-1-20	Added descriptions on accessing the Identify Namespace data structures in each of the I/O Command Set specification. Added the definitions for the Identify Controller data structures and Identify Namespace data structures in each I/O Command Set specification to clearly state the exact list of data structures. Change the word “capabilities” to “attributes” to align to field name and usage.
2022-1-24	Removed the word “common” from the Zoned Namespace Command Set Specification for CNS 00h in Figure 47 (this aligns to the change made in
2022-1-31	Updated CNS 04h, 18h, and 19h to add the reference to the identifiers defining sections that was removed.
2022-2-11	Corrected trademarks. Moved the change for CNS 00h in Figure 96 in the NVM Command Set Specification to ECN109.
2022-2-16	Moved the text on the definition of a supported LBA Format that has a LBA Data Size field cleared to 0h to section 5.TBD. Updated the LBA Data Size field to refer to section 5.TBD for when the value is cleared to 0h.
2022-3-3	Added “section” for a reference. Deleted comments and accepted tracking changes.
2022-04-27	Integrated
2022-04-28	Updated some “NVMe Base Specification” that changed to “NVM Express Base Specification.”

## Description for NVM Express NVM Command Set Specification 1.0a Changes Document

### Feature Enhancement:

- Added CNS value 09h to the Identify command to return an NVM Command Set Identify Namespace data structure associated with the LBA Format specified by the Format Index. The NVM Command Set Identify Namespace data structure is updated to identify the fields that report valid values (refer to the Reported column).
- Added CNS value 0Ah to the Identify command to return an Identify I/O Command Set specific Namespace data structure for the NVM Command Set associated with the LBA Format specified by the Format Index. The Identify I/O Command Set specific Namespace data structure for the NVM Command Set is updated to identify the fields that report valid values (refer to the Reported column).
- New requirement / incompatible change:**
  - The Identify command for CNS value 00h removed the term “common” and clearly specified the requirements on all fields when the specified NSID is set to FFFFFFFFh
  - The requirement to abort an Identify command with CNS values 00h, 05h, and 08h, when the specified NSID is FFFFFFFFh and Namespace Management capability is not supported, was changed to optional (previously, aborting the command was mandatory).
  - The LBA Format list structure is extended to have both the existing group of LBA Formats that have the same capabilities for creating and formatting a namespace as well as a group of LBA Formats that each may have a unique set of capabilities for creating and formatting a namespace.
- References:
  - TP4095a Namespace Capability Reporting

## Description for NVM Express Key Value Command Set Specification 1.0a Changes Document

### Feature Enhancement:

- Added CNS value 0Ah to the Identify command to return an I/O Command Set specific Identify Namespace data structure for the Key Value Command Set associated with the specified Format Index and Command Set Identifier. The I/O Command Set specific Identify Namespace data structure for the Key Value Command is updated to identify the fields that report valid values (refer to the Reported column).

- References:
  - TP4095a

## Description for NVM Express Zoned Namespaces Command Set Specification 1.0a Changes Document

### Feature Enhancement:

- Added CNS value **Ah** to the Identify command to return an I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set associated with the specified Format Index and Command Set Identifier. The I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set Command is updated to identify the fields that report valid values (refer to the Reported column).
- **New requirement / incompatible change:**
  - The Identify command for CNS values 05h removed the term “common” and clearly specified the requirements on all fields when the specified NSID is set to FFFFFFFFh
  - The requirement to abort an Identify command with CNS values 00h, 05h, and 08h, when the specified NSID is FFFFFFFFh and Namespace Management capability is not supported, was changed to optional (previously, aborting the command was mandatory).
- References:
  - TP4095a Namespace Capability Reporting

## Description for NVM Express Base Specification 2.0a Changes Document

### Feature Enhancement:

- Added CNS value **09h** to the Identify command to return an Identify Namespace data structure associated with the specified Format Index
- Added CNS value **0Ah** to the Identify command to return an I/O Command Set specific Identify Namespace data structure associated with the specified Format Index and Command Set Identifier.
- **New requirement / incompatible change:**
  - The Identify command for CNS values 00h, 05h and 08h removed the term “common” and clearly specified the requirements on all fields when the specified NSID is set to FFFFFFFFh.
  - The requirement to abort an Identify command with CNS values 00h, 05h, and 08h, when the specified NSID is FFFFFFFFh and Namespace Management capability is not supported, was changed to optional (previously, aborting the command was mandatory).
- References:
  - TP 4095a Namespace Capability Reporting

### Markup Conventions:

Black:	Unchanged (however, hot links are removed)
<del>Red Strikethrough:</del>	Deleted
Blue:	New
<b>Blue Highlighted:</b>	TBD values, anchors, and links to be inserted in new text.
<Green Bracketed>:	Notes to editor

**Modify portions of NVM Command Set Specification 1.0a as shown below:**

***Modify section 1.4.2 as shown below:***

## **1 Introduction**

...

### **1.4 Definitions**

...

#### **1.4.2 Definitions in the NVM Express Base Specification specified in the NVM Command set**

...

##### **1.4.2.TBD Identify Controller data structures**

All controller data structures that are able to be retrieved via the Identify command for the NVM Command Set:

- the Identify Controller data structure (refer to the NVM Express Base Specification and section 4.1.5.2); and
- the I/O Command Set specific Identify Controller data structure for the NVM Command Set (refer to section 4.1.5.4).

##### **1.4.2.TBD1 Identify Namespace data structures**

All namespace data structures that are able to be retrieved via the Identify command for the NVM Command Set:

- the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification);
- the Identify Namespace data structure (refer to section 4.1.5.1); and
- the I/O Command Set specific Identify Namespace data structure for the NVM Command Set (refer to section 4.1.5.3).

...

***Modify section 4.1.2 as shown below:***

## **4 Admin Commands for the NVM Command Set**

...

### **4.1 Admin Command behavior for the NVM Command Set**

...

#### **4.1.2 Format NVM command**

The Format NVM command operates as defined in the NVM Express Base Specification. The Format Index indicates a valid LBA User Data Format from the LBA Format field in the Identify Namespace data structure (refer to section 5.TBD). Other NVM Command Set specific fields are defined in Figure 78.

*Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2022 NVM Express, Inc.*

...

**Modify a portion of figure 86 in section 4.1.3.4 as shown below:**

#### 4.1.3 Get Features and Set Features Commands

...

##### 4.1.3.4 Host Behavior Support (Feature Identifier 16h)

The Host Behavior Support feature operates as defined in the NVMe Express Base Specification. In addition to the requirements in the NVMe Express Base Specification, this specification provides NVM Command Set specific definitions.

**Figure 86: Host Behavior Support – Data Structure**

Bytes	Description
02	<p><b>LBA Format Extension Enable (LBAFEE):</b> This field allows the host to specify support for the extended LBA formats (refer to the EBLAS field in the Identify Controller data structure in the NVMe Express Base Specification). If this field is set to 1h and the ELBAS field is set to '1', then the controller:</p> <ol style="list-style-type: none"><li>shall report a maximum number that is less than or equal to 64 for:<ol style="list-style-type: none"><li>the total number of LBA formats supported (refer to section 5.TBD <del>the NLBAF field in the Identify Namespace data structure in Figure 97</del>); and</li><li>the number of namespace granularity descriptors (refer to Figure 104);</li></ol></li><li>and</li><li>is enabled to create, format, and perform I/O commands on namespaces formatted with (refer to section 5.2.1):<ol style="list-style-type: none"><li>16b Guard Protection Information with the STS field set to a non-zero value;</li><li>32b Guard Protection Information; and</li><li>64b Guard Protection Information,</li></ol></li></ol> <p>where the extended LBA formats (refer to Figure 101) define the actual protection information formats supported.</p> <p>If this field is cleared to 0h, then the controller:</p> <ol style="list-style-type: none"><li>shall report a maximum that is less than or equal to 16 for:<ol style="list-style-type: none"><li>the total number of LBA formats supported (refer to section 5.TBD); and</li><li>the number of namespace granularity descriptors;</li></ol></li><li>and</li><li>shall not create, format, and perform I/O commands on namespaces formatted with (refer to section 5.2.1):<ol style="list-style-type: none"><li>16b Guard Protection Information with the STS field set to a non-zero value;</li><li>32b Guard Protection Information; and</li><li>64b Guard Protection Information,</li></ol></li></ol> <p>and commands requesting these restrictions shall be aborted with a status code of Invalid Namespace or Format.</p> <p>All values other than 0h and 1h are reserved.</p>

...

**Modify figure 96 in section 4.1.5 as shown below:**

#### 4.1.5 Identify Command

...

**Figure 96: Identify – CNS Values**

CNS Value	O/M <sup>1</sup>	Definition	NSID <sup>2</sup>	CNTID <sup>3</sup>	CSI <sup>4</sup>	Reference Section
<b>Active Namespace Management</b>						
...						
06h	M	Identify I/O Command Set specific Controller data structure for the controller processing the command. <sup>6</sup>	Y	N	Y	4.1.5.4
09h	O	Identify Namespace data structure for the specified Format Index containing the namespace capabilities for the NVM Command Set. <sup>6</sup>	N	N	Y	4.1.5.TBD
0Ah	O	I/O Command Set specific Identify Namespace data structure for the specified Format Index for the I/O Command Set specified in the CSI field. <sup>6</sup>	N	N	Y	4.1.5.TBD1
11h	O	Identify Namespace data structure for the specified allocated NSID.	Y	N	N	4.1.5.5
...						
NOTES: 1. O/M definition: O = Optional, M = Mandatory. 2. The NSID field is used: Y = Yes, N = No. 3. The CDW10.CNTID field is used: Y = Yes, N = No. 4. The CDW11.CSI field is used: Y = Yes, N = No. 5. Mandatory for controllers that support the Namespace Management capability (refer to the NVM Express Base Specification). 6. Selection of a UUID may be supported. Refer to the Universally Unique Identifiers (UUIDs) for Vendor Specific Information section in the NVM Express Base Specification.						

**Modify portions of section 4.1.5.1 as shown below:**

##### 4.1.5.1 NVM Command Set Identify Namespace data structure (CNS 00h)

If the Namespace Identifier (NSID) field specifies an active NSID, then the NVM Command Set Identify Namespace data structure (refer to Figure 97) is returned to the host for that specified namespace. If that specified namespace is an inactive NSID, then the controller returns a zero filled data structure. If the specified namespace is not associated with an I/O Command Set that supports this data structure, then the controller shall abort the command with the status code of Invalid I/O Command Set.

~~If the controller supports the Namespace Management capability (refer to the Namespace Management section in the NVM Express Base Specification) and the NSID field is set to FFFFFFFFh, then the controller returns an Identify Namespace data structure that specifies NVM Command Set capabilities that are common across namespaces for the controller. If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller shall abort the command with a status code of Invalid Namespace or Format.~~

The Reported column in Figure 97 specifies fields in the NVM Command Set Identify Namespace data structure that define namespace capabilities used by a host to format or create a namespace. If the NSID



field is set to FFFFFFFFh, then the controller shall return an NVM Command Set Identify Namespace data structure that contains:

- fields in Figure 97 set to a value that is the same for all namespaces using any of the LBA formats associated with the Number of LBA Formats field (refer to section 5.TBD) that the Reported column indicates “Yes”; and
- fields in Figure 97 cleared to 0h that the Reported column indicates “No”.

If the controller supports the Namespace Management capability (refer to the Namespace Management section in the NVM Express Base Specification) and the NSID field is set to FFFFFFFFh, then the controller shall return an NVM Command Set Identify Namespace data structure. If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller may abort the command with a status code of Invalid Namespace or Format.

**<Note to readers: This is about adding the right most column**

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
07:00	M	<b>Namespace Size (NSZE):</b> This field indicates the total size of the namespace in logical blocks. A namespace of size $n$ consists of LBA 0 through $(n - 1)$ . The number of logical blocks is based on the formatted LBA size.	No
15:08	M	<b>Namespace Capacity (NCAP):</b> This field indicates the maximum number of logical blocks that may be allocated in the namespace at any point in time. The number of logical blocks is based on the formatted LBA size. Spare LBAs are not reported as part of this field.  Refer to section 2.1.1 for details on the usage of this field.	No
23:16	M	<b>Namespace Utilization (NUSE):</b> This field indicates the current number of logical blocks allocated in the namespace. This field is smaller than or equal to the Namespace Capacity. The number of logical blocks is based on the formatted LBA size.  Refer to section 2.1.1 for details on the usage of this field.	No

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
24	M	<p><b>Namespace Features (NSFEAT):</b> This field defines features of the namespace.</p> <p>Bits 7:5 are reserved.</p> <p>Bit 4 (<b>OPTPERF</b>) if set to '1' indicates that the fields NPWG, NPWA, NPDG, NPDA, and NOWS are defined for this namespace and should be used by the host for I/O optimization (refer to the NVM Set List section in the NVMe Base Specification). If cleared to '0', then the controller does not support the fields NPWG, NPWA, NPDG, NPDA, and NOWS for this namespace.</p> <p>Bit 3 (<b>UIDREUSE</b>) This bit is as defined in the UIDREUSE bit in the I/O Command Set Independent Identify Namespace data structure (refer to the I/O Command Set Independent Identify Namespace data structure section in the NVMe Express Base Specification).</p> <p>Bit 2 (<b>DAE</b>) if set to '1' indicates that the controller supports the Deallocated or Unwritten Logical Block error for this namespace. If cleared to '0', then the controller does not support the Deallocated or Unwritten Logical Block error for this namespace. Refer to section 3.2.3.2.1.</p> <p>Bit 1 (<b>NSABP</b>) if set to '1' indicates that the fields NAWUN, NAWUPF, and NACWU are defined for this namespace and should be used by the host for this namespace instead of the AWUN, AWUPF, and ACWU fields in the Identify Controller data structure. If cleared to '0', then the controller does not support the fields NAWUN, NAWUPF, and NACWU for this namespace. In this case, the host should use the AWUN, AWUPF, and ACWU fields defined in the Identify Controller data structure in the NVMe Express Base Specification. Refer to section 2.1.4.</p> <p>Bit 0 (<b>THINP</b>) if set to '1' indicates that the namespace supports thin provisioning. If cleared to '0' indicates that thin provisioning is not supported Refer to section 2.1.1 for details on the usage of this bit.</p>	No

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
25	M	<p><b>Number of LBA Formats (NLBAF):</b> This field defines the number of supported LBA data size and metadata size combinations supported by the namespaces <a href="#">that share the same set of host-selectable attributes</a>. LBA formats shall be allocated in order (starting with 0) and packed sequentially. This is a 0's based value.</p> <p>Refer to section <a href="#">5.TBD</a> for the structure of the LBA formats, the association to the NULBAF field, and the maximum values of this field.</p> <p><del>The maximum number of LBA formats that may be indicated as supported is:</del></p> <ul style="list-style-type: none"> <li><del>16 if the LBA Format Extension Enable (LBAFEE) field is cleared to 0h in the Host Behavior Support feature (refer to the Host Behavior Support section in the NVMe Base Specification); or</del></li> <li><del>64 if the LBAFEE field is set to 1h in the Host Behavior Support feature (refer to the Host Behavior Support section in the NVMe Base Specification).</del></li> </ul> <p>The supported LBA formats are indicated in bytes 128 to 383 in this data structure. The LBA Format fields with an index <a href="#">greater than</a> <del>beyond</del> the value <a href="#">defined by section 5.TBD</a> <del>set in this field</del> are invalid and not supported. <del>LBA Format that valid, but not currently available indicated by the LBA Data Size for that LBA Format &lt;sentence moved to section 5.TBD&gt;.</del></p> <p>The metadata may be either transferred as part of the LBA (creating an extended LBA which is a larger LBA size that is exposed to the application) or may be transferred as a separate contiguous buffer of data. The metadata shall not be split between the LBA and a separate metadata buffer.</p> <p>It is recommended that software and controllers transition to an LBA size that is 4 KiB or larger for ECC efficiency at the controller. If providing metadata, it is recommended that at least 8 bytes are provided per logical block to enable use with end-to-end data protection, refer to section 5.8.3.</p>	Yes
26	M	<p><b>&lt;Editor's note: This field has been updated to reflect the changes defined by ECN101. ECN101 text modifid bits 6:5 that conflicted with TP4095. No other section in this document is updated by ECN101.&gt;</b></p> <p><b>Formatted LBA Size (FLBAS):</b> This field indicates the LBA data size &amp; metadata size combination that the namespace has been formatted with (refer to section 4.1.2).</p> <p>Bits 7 is reserved.</p> <p>Bits 6:5 indicate the most-significant 2 bits of the Format Index of the supported LBA Format indicated in this data structure that was used to format the namespace. If the <a href="#">total number of LBA formats supported (refer to section 5.TBD)</a> <del>NLBAF field</del> is less than or equal to 16, then the host should ignore these bits.</p> <p>Bit 4 if set to '1' indicates that the metadata is transferred at the end of the data LBA, creating an extended data LBA. Bit 4 if cleared to '0' indicates that all of the metadata for a command is transferred as a separate contiguous buffer of data. Bit 4 is not applicable when there is no metadata.</p> <p>Bits 3:0 indicates the least-significant 4 bits of the Format Index <del>one</del> of the supported LBA Format indicated in this data structure that was used to format the namespace.</p>	No

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>														
27	M	<p><b>Metadata Capabilities (MC):</b> This field indicates the capabilities for metadata.</p> <p>Bits 7:2 are reserved.</p> <p>Bit 1 if set to '1' indicates the namespace supports the metadata being transferred as part of a separate buffer that is specified in the Metadata Pointer. Bit 1 if cleared to '0' indicates that the namespace does not support the metadata being transferred as part of a separate buffer.</p> <p>Bit 0 if set to '1' indicates that the namespace supports the metadata being transferred as part of an extended data LBA. Bit 0 if cleared to '0' indicates that the namespace does not support the metadata being transferred as part of an extended data LBA.</p>	Yes														
28	M	<p><b>End-to-end Data Protection Capabilities (DPC):</b> This field indicates the capabilities for the end-to-end data protection feature. Multiple bits may be set in this field. Refer to section 5.1.</p> <table><tr><th>Bits</th><th>Description</th></tr><tr><td>7:5</td><td>Reserved</td></tr><tr><td>4</td><td><b>Protection Information In Last Bytes (PIILB):</b> If set to '1' indicates that the namespace supports protection information transferred as the last bytes of metadata. If cleared to '0' indicates that the namespace does not support protection information transferred as the last bytes of metadata.</td></tr><tr><td>3</td><td><b>Protection Information In First Bytes (PIIFB):</b> If set to '1' indicates that the namespace supports protection information transferred as the first bytes of metadata. If cleared to '0' indicates that the namespace does not support protection information transferred as the first bytes of metadata. For versions later than version 1.4 of this specification, this bit shall be cleared to '0'.</td></tr><tr><td>2</td><td><b>Protection Information Type 3 Supported (PIT3S):</b> If set to '1' indicates that the namespace supports Protection Information Type 3. If cleared to '0' indicates that the namespace does not support Protection Information Type 3.</td></tr><tr><td>1</td><td><b>Protection Information Type 2 Supported (PIT2S):</b> If set to '1' indicates that the namespace supports Protection Information Type 2. If cleared to '0' indicates that the namespace does not support Protection Information Type 2.</td></tr><tr><td>0</td><td><b>Protection Information Type 1 Supported (PIT1S):</b> If set to '1' indicates that the namespace supports Protection Information Type 1. If cleared to '0' indicates that the namespace does not support Protection Information Type 1.</td></tr></table>	Bits	Description	7:5	Reserved	4	<b>Protection Information In Last Bytes (PIILB):</b> If set to '1' indicates that the namespace supports protection information transferred as the last bytes of metadata. If cleared to '0' indicates that the namespace does not support protection information transferred as the last bytes of metadata.	3	<b>Protection Information In First Bytes (PIIFB):</b> If set to '1' indicates that the namespace supports protection information transferred as the first bytes of metadata. If cleared to '0' indicates that the namespace does not support protection information transferred as the first bytes of metadata. For versions later than version 1.4 of this specification, this bit shall be cleared to '0'.	2	<b>Protection Information Type 3 Supported (PIT3S):</b> If set to '1' indicates that the namespace supports Protection Information Type 3. If cleared to '0' indicates that the namespace does not support Protection Information Type 3.	1	<b>Protection Information Type 2 Supported (PIT2S):</b> If set to '1' indicates that the namespace supports Protection Information Type 2. If cleared to '0' indicates that the namespace does not support Protection Information Type 2.	0	<b>Protection Information Type 1 Supported (PIT1S):</b> If set to '1' indicates that the namespace supports Protection Information Type 1. If cleared to '0' indicates that the namespace does not support Protection Information Type 1.	Yes
Bits	Description																
7:5	Reserved																
4	<b>Protection Information In Last Bytes (PIILB):</b> If set to '1' indicates that the namespace supports protection information transferred as the last bytes of metadata. If cleared to '0' indicates that the namespace does not support protection information transferred as the last bytes of metadata.																
3	<b>Protection Information In First Bytes (PIIFB):</b> If set to '1' indicates that the namespace supports protection information transferred as the first bytes of metadata. If cleared to '0' indicates that the namespace does not support protection information transferred as the first bytes of metadata. For versions later than version 1.4 of this specification, this bit shall be cleared to '0'.																
2	<b>Protection Information Type 3 Supported (PIT3S):</b> If set to '1' indicates that the namespace supports Protection Information Type 3. If cleared to '0' indicates that the namespace does not support Protection Information Type 3.																
1	<b>Protection Information Type 2 Supported (PIT2S):</b> If set to '1' indicates that the namespace supports Protection Information Type 2. If cleared to '0' indicates that the namespace does not support Protection Information Type 2.																
0	<b>Protection Information Type 1 Supported (PIT1S):</b> If set to '1' indicates that the namespace supports Protection Information Type 1. If cleared to '0' indicates that the namespace does not support Protection Information Type 1.																

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>																				
29	M	<b>End-to-end Data Protection Type Settings (DPS):</b> This field indicates the protection information Type settings for the end-to-end data protection feature. Refer to section 5.1.	No																				
		<table><tr><th>Bits</th><th>Description</th></tr><tr><td>7:4</td><td>Reserved</td></tr><tr><td>3</td><td><b>Protection Information Position (PIP):</b> This bit indicates that the protection information, if enabled, is transferred as the first bytes of metadata. Bit 3 if cleared to '0' indicates that the protection information, if enabled, is transferred as the last bytes of metadata. For implementations compliant to version 1.0 or later of the NVM Command Set Specification, this bit shall be cleared to '0'.</td></tr><tr><td>2:0</td><td><b>Protection Information Type (PIT):</b> This field indicates whether protection information is enabled and the type of protection information enabled. The values for this field have the following meanings:<table><tr><th>Value</th><th>Definition</th></tr><tr><td>000b</td><td>Protection information is not enabled</td></tr><tr><td>001b</td><td>Protection information is enabled, Type 1</td></tr><tr><td>010b</td><td>Protection information is enabled, Type 2</td></tr><tr><td>011b</td><td>Protection information is enabled, Type 3</td></tr><tr><td>100b to 111b</td><td>Reserved</td></tr></table></td></tr></table>		Bits	Description	7:4	Reserved	3	<b>Protection Information Position (PIP):</b> This bit indicates that the protection information, if enabled, is transferred as the first bytes of metadata. Bit 3 if cleared to '0' indicates that the protection information, if enabled, is transferred as the last bytes of metadata. For implementations compliant to version 1.0 or later of the NVM Command Set Specification, this bit shall be cleared to '0'.	2:0	<b>Protection Information Type (PIT):</b> This field indicates whether protection information is enabled and the type of protection information enabled. The values for this field have the following meanings: <table><tr><th>Value</th><th>Definition</th></tr><tr><td>000b</td><td>Protection information is not enabled</td></tr><tr><td>001b</td><td>Protection information is enabled, Type 1</td></tr><tr><td>010b</td><td>Protection information is enabled, Type 2</td></tr><tr><td>011b</td><td>Protection information is enabled, Type 3</td></tr><tr><td>100b to 111b</td><td>Reserved</td></tr></table>	Value	Definition	000b	Protection information is not enabled	001b	Protection information is enabled, Type 1	010b	Protection information is enabled, Type 2	011b	Protection information is enabled, Type 3	100b to 111b	Reserved
		Bits		Description																			
		7:4		Reserved																			
		3		<b>Protection Information Position (PIP):</b> This bit indicates that the protection information, if enabled, is transferred as the first bytes of metadata. Bit 3 if cleared to '0' indicates that the protection information, if enabled, is transferred as the last bytes of metadata. For implementations compliant to version 1.0 or later of the NVM Command Set Specification, this bit shall be cleared to '0'.																			
2:0	<b>Protection Information Type (PIT):</b> This field indicates whether protection information is enabled and the type of protection information enabled. The values for this field have the following meanings: <table><tr><th>Value</th><th>Definition</th></tr><tr><td>000b</td><td>Protection information is not enabled</td></tr><tr><td>001b</td><td>Protection information is enabled, Type 1</td></tr><tr><td>010b</td><td>Protection information is enabled, Type 2</td></tr><tr><td>011b</td><td>Protection information is enabled, Type 3</td></tr><tr><td>100b to 111b</td><td>Reserved</td></tr></table>	Value	Definition	000b	Protection information is not enabled	001b	Protection information is enabled, Type 1	010b	Protection information is enabled, Type 2	011b	Protection information is enabled, Type 3	100b to 111b	Reserved										
Value	Definition																						
000b	Protection information is not enabled																						
001b	Protection information is enabled, Type 1																						
010b	Protection information is enabled, Type 2																						
011b	Protection information is enabled, Type 3																						
100b to 111b	Reserved																						
30	O	<b>Namespace Multi-path I/O and Namespace Sharing Capabilities (NMIC):</b> This field is as defined in the I/O Command Set Independent Identify Namespace data structure (refer to the I/O Command Set Independent Identify Namespace data structure section in the NVMe Express Base Specification).	Yes																				
31	O	<b>Reservation Capabilities (RESCAP):</b> This field is as defined in the I/O Command Set Independent Identify Namespace data structure (refer to the I/O Command Set Independent Identify Namespace data structure section in the NVMe Express Base Specification).	No																				
32	O	<b>Format Progress Indicator (FPI):</b> This field is as defined in the I/O Command Set Independent Identify Namespace data structure (refer to the I/O Command Set Independent Identify Namespace data structure section in the NVMe Express Base Specification).	No																				

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>										
33	O	<p><b>Deallocate Logical Block Features (DLFEAT):</b> This field indicates information about features that affect deallocating logical blocks for this namespace.</p> <p>Bits 7:5 are reserved.</p> <p>Bit 4 if set to '1' indicates that the Guard field for deallocated logical blocks that contain protection information is set to the CRC for the value read from the deallocated logical block and its metadata (excluding protection information). If cleared to '0' indicates that each byte in the Guard field for the deallocated logical blocks that contain protection information is set to FFh.</p> <p>Bit 3 if set to '1' indicates that the controller supports the Deallocate bit in the Write Zeroes command for this namespace. If cleared to '0' indicates that the controller does not support the Deallocate bit in the Write Zeroes command for this namespace. This bit shall be set to the same value for all namespaces in the NVM subsystem.</p> <p>Bits 2:0 indicate deallocated logical block read behavior. For a logical block that is deallocated, this field indicates the values read from that deallocated logical block and its metadata (excluding protection information). The values for this field have the following meanings:</p> <table><tr><th>Value</th><th>Definition</th></tr><tr><td>000b</td><td>The read behavior is not reported</td></tr><tr><td>001b</td><td>A deallocated logical block returns all bytes cleared to 0h</td></tr><tr><td>010b</td><td>A deallocated logical block returns all bytes set to FFh</td></tr><tr><td>011b to 111b</td><td>Reserved</td></tr></table>	Value	Definition	000b	The read behavior is not reported	001b	A deallocated logical block returns all bytes cleared to 0h	010b	A deallocated logical block returns all bytes set to FFh	011b to 111b	Reserved	No
Value	Definition												
000b	The read behavior is not reported												
001b	A deallocated logical block returns all bytes cleared to 0h												
010b	A deallocated logical block returns all bytes set to FFh												
011b to 111b	Reserved												
35:34	O	<p><b>Namespace Atomic Write Unit Normal (NAWUN):</b> This field indicates the namespace specific size of the write operation guaranteed to be written atomically to the NVM during normal operation. If the NSABP bit is cleared to '0', then this field is reserved.</p> <p>A value of 0h indicates that the size for this namespace is the same size as that reported in the AWUN field of the Identify Controller data structure. All other values specify a size in terms of logical blocks using the same encoding as the AWUN field. Refer to section 2.1.4.</p>	No										
37:36	O	<p><b>Namespace Atomic Write Unit Power Fail (NAWUPF):</b> This field indicates the namespace specific size of the write operation guaranteed to be written atomically to the NVM during a power fail or error condition. If the NSABP bit is cleared to '0', then this field is reserved.</p> <p>A value of 0h indicates that the size for this namespace is the same size as that reported in the AWUPF field of the Identify Controller data structure. All other values specify a size in terms of logical blocks using the same encoding as the AWUPF field. Refer to section 2.1.4.</p>	No										
39:38	O	<p><b>Namespace Atomic Compare &amp; Write Unit (NACWU):</b> This field indicates the namespace specific size of the write operation guaranteed to be written atomically to the NVM for a Compare and Write fused command. If the NSABP bit is cleared to '0', then this field is reserved.</p> <p>A value of 0h indicates that the size for this namespace is the same size as that reported in the ACWU field of the Identify Controller data structure. All other values specify a size in terms of logical blocks using the same encoding as the ACWU field. Refer to section 2.1.4.</p>	No										

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
41:40	O	<p><b>Namespace Atomic Boundary Size Normal (NABSN):</b> This field indicates the atomic boundary size for this namespace for the NAWUN value. This field is specified in logical blocks. Writes to this namespace that cross atomic boundaries are not guaranteed to be atomic to the NVM with respect to other read or write commands.</p> <p>A value of 0h indicates that there are no atomic boundaries for normal write operations. All other values specify a size in terms of logical blocks using the same encoding as the AWUN field. Refer to section 2.1.4.</p> <p>Refer to section 5.8.2 for how this field is utilized.</p>	No
43:42	O	<p><b>Namespace Atomic Boundary Offset (NABO):</b> This field indicates the LBA on this namespace where the first atomic boundary starts.</p> <p>If the NABSN and NABSPF fields are cleared to 0h, then the NABO field shall be cleared to 0h. NABO shall be less than or equal to NABSN and NABSPF. Refer to section 2.1.4.</p> <p>Refer to section 5.8.2 for how this field is utilized.</p>	No
45:44	O	<p><b>Namespace Atomic Boundary Size Power Fail (NABSPF):</b> This field indicates the atomic boundary size for this namespace specific to the Namespace Atomic Write Unit Power Fail value. This field is specified in logical blocks. Writes to this namespace that cross atomic boundaries are not guaranteed to be atomic with respect to other read or write commands and there is no guarantee of data returned on subsequent reads of the associated logical blocks.</p> <p>A value of 0h indicates that there are no atomic boundaries for power fail or error conditions. All other values specify a size in terms of logical blocks using the same encoding as the AWUPF field. Refer to section 2.1.4.</p>	No
47:46	O	<p><b>Namespace Optimal I/O Boundary (NOIOB):</b> This field indicates the optimal I/O boundary for this namespace. This field is specified in logical blocks. The host should construct Read and Write commands that do not cross the I/O boundary to achieve optimal performance. A value of 0h indicates that no optimal I/O boundary is reported.</p> <p>Refer to section 5.8.2 for how this field is utilized to improve performance and endurance.</p>	No
63:48	O	<p><b>NVM Capacity (NVMCAP):</b> This field indicates the total size of the NVM allocated to this namespace. The value is in bytes. This field shall be supported if the Namespace Management capability (refer to section 5.3) is supported.</p> <p>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 product of the logical block size and the Namespace Size reported.</p> <p>If the controller supports Asymmetric Namespace Access Reporting (refer to the CMIC field), and the relationship between the controller and the namespace is in the ANA Inaccessible state (refer to the ANA Inaccessible state section in the NVM Express Base Specification) or the ANA Persistent Loss state (refer to the ANA Persistent Loss state section in the NVM Express Base Specification), then this field shall be cleared to 0h.</p>	No

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
65:64	O	<p><b>Namespace Preferred Write Granularity (NPWG):</b> This field indicates the smallest recommended write granularity in logical blocks for this namespace. This is a 0's based value. If the OPTPERF bit is cleared to '0', then this field is reserved.</p> <p>The size indicated should be less than or equal to Maximum Data Transfer Size (MDTS) that is specified in units of minimum memory page size. The value of this field may change if the namespace is reformatted. The size should be a multiple of Namespace Preferred Write Alignment (NPWA).</p> <p>Refer to section 5.8.2 for how this field is utilized to improve performance and endurance.</p>	No
67:66	O	<p><b>Namespace Preferred Write Alignment (NPWA):</b> This field indicates the recommended write alignment in logical blocks for this namespace. This is a 0's based value. If the OPTPERF bit is cleared to '0', then this field is reserved.</p> <p>The value of this field may change if the namespace is reformatted.</p> <p>Refer to section 5.8.2 for how this field is utilized to improve performance and endurance.</p>	No
69:68	O	<p><b>Namespace Preferred Deallocate Granularity (NPDG):</b> This field indicates the recommended granularity in logical blocks for the Dataset Management command with the Attribute – Deallocate bit set to '1' in Dword 11. This is a 0's based value. If the OPTPERF bit is cleared to '0', then this field is reserved.</p> <p>The value of this field may change if the namespace is reformatted. The size should be a multiple of Namespace Preferred Deallocate Alignment (NPDA).</p> <p>Refer to section 5.8.2 for how this field is utilized to improve performance and endurance.</p>	No
71:70	O	<p><b>Namespace Preferred Deallocate Alignment (NPDA):</b> This field indicates the recommended alignment in logical blocks for the Dataset Management command with the Attribute – Deallocate bit set to '1' in Dword 11. This is a 0's based value. If the OPTPERF bit is cleared to '0', then this field is reserved.</p> <p>The value of this field may change if the namespace is reformatted.</p> <p>Refer to section 5.8.2 for how this field is utilized to improve performance and endurance.</p>	No
73:72	O	<p><b>Namespace Optimal Write Size (NOWS):</b> This field indicates the size in logical blocks for optimal write performance for this namespace. This is a 0's based value. If the OPTPERF bit is cleared to '0', then this field is reserved.</p> <p>The size indicated should be less than or equal to Maximum Data Transfer Size (MDTS) that is specified in units of minimum memory page size. The value of this field may change if the namespace is reformatted. The value of this field should be a multiple of Namespace Preferred Write Granularity (NPWG).</p> <p>If the namespace is associated with an NVM set, NOWS defined for this namespace shall be set to the Optimal Write Size field setting defined in NVM Set Attributes Entry (refer to the Namespace Identification Descriptor in the NVM Express Base Specification) for the NVM Set with which this namespace is associated. If NOWS is not supported, the Optimal Write Size field in NVM Sets Attributes Entry (refer to the Namespace Identification Descriptor in the NVMe Base Specification) for the NVM Set with which this namespace is associated should be used by the host for I/O optimization.</p> <p>Refer to section 5.8.2 for how this field is utilized to improve performance and endurance.</p>	No



**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
75:74	O	<b>Maximum Single Source Range Length (MSSRL):</b> This field indicates the maximum number of logical blocks that may be specified in each valid Source Range field (refer to 3) of a Copy command.  If the controller supports the Copy command, then this field shall be set to a non-zero value.	No
79:76	O	<b>Maximum Copy Length (MCL):</b> This field indicates the maximum number of logical blocks that may be specified in a Copy command (i.e., the sum of the number of logical blocks specified in all Source Range entries).  If the controller supports the Copy command, then this field shall be set to a non-zero value.	No
80	O	<b>Maximum Source Range Count (MSRC):</b> This field indicates the maximum number of Source Range entries that may be used to specify source data in a Copy command. This is a 0's based value.	No
82	M	<b>Number of Unique Attribute LBA Formats (NULBAF):</b> This field defines the number of user data size and metadata size combinations supported by the namespace that may not share the same host-selectable attributes. These LBA formats shall be allocated in order (starting at the first index after the LBA formats defined by the NLBAF field)) and packed sequentially (refer to section 5.TBD).  Refer to section 5.TBD for the structure of the LBA formats, the association to the NLBAF field, and the maximum value of this field.	Yes
...	...	...	...
95:92	O	<b>ANA Group Identifier (ANAGRPID):</b>  This field is as defined in the I/O Command Set Independent Identify Namespace data structure (refer to the I/O Command Set Independent Identify Namespace data structure section in the NVM Express Base Specification).	No
98:96		Reserved	
99	O	<b>Namespace Attributes (NSATTR):</b>  This field is as defined in the I/O Command Set Independent Identify Namespace data structure (refer to the I/O Command Set Independent Identify Namespace data structure section in the NVM Express Base Specification).	No
101:100	O	<b>NVM Set Identifier (NVMSETID):</b> This field is as defined in the I/O Command Set Independent Identify Namespace data structure (refer to the I/O Command Set Independent Identify Namespace data structure section in the NVM Express Base Specification).	No
103:102	O	<b>Endurance Group Identifier (ENDGID):</b> This field is as defined in the I/O Command Set Independent Identify Namespace data structure (refer to the I/O Command Set Independent Identify Namespace data structure section in the NVM Express Base Specification).	No

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
119:104	O	<p><b>Namespace Globally Unique Identifier (NGUID):</b> This field contains a 128-bit value that is globally unique and assigned to the namespace when the namespace is created. This field remains fixed throughout the life of the namespace and is preserved across namespace and controller operations (e.g., Controller Level Reset, namespace format, etc.).</p> <p>This field uses the EUI-64 based 16-byte designator format. Bytes 114:112 contain the 24-bit Organizationally Unique Identifier (OUI) value assigned by the IEEE Registration Authority. Bytes 119:115 contain an extension identifier assigned by the corresponding organization. Bytes 111:104 contain the vendor specific extension identifier assigned by the corresponding organization. Refer to the IEEE EUI-64 guidelines for more information. This field is big endian (refer to the Namespace Globally Unique Identifier section in the NVMe Express Base Specification).</p> <p>The controller shall specify a globally unique namespace identifier in this field, the EUI64 field, or a Namespace UUID in the Namespace Identification Descriptor (refer to the Namespace Identification Descriptor figure in the NVMe Express Base Specification) when the namespace is created. If the controller is not able to provide a globally unique identifier in this field, then this field shall be cleared to 0h. Refer to the Unique Identifier section in the NVMe Express Base Specification.</p>	No
127:120	O	<p><b>IEEE Extended Unique Identifier (EUI64):</b> This field contains a 64-bit IEEE Extended Unique Identifier (EUI-64) that is globally unique and assigned to the namespace when the namespace is created. This field remains fixed throughout the life of the namespace and is preserved across namespace and controller operations (e.g., Controller Level Reset, namespace format, etc.).</p> <p>The EUI-64 is a concatenation of a 24-bit or 36-bit Organizationally Unique Identifier (OUI or OUI-36) value assigned by the IEEE Registration Authority and an extension identifier assigned by the corresponding organization. Refer to the IEEE EUI-64 guidelines for more information. This field is big endian (refer to the IEEE Extended Unique Identifier section in the NVMe Express Base Specification).</p> <p>The controller shall specify a globally unique namespace identifier in this field, the NGUID field, or a Namespace UUID in the Namespace Identification Descriptor (refer to the Namespace Identification Descriptor figure in the NVMe Express Base Specification) when the namespace is created. If the controller is not able to provide a globally unique 64-bit identifier in this field, then this field shall be cleared to 0h. Refer to the Unique Identifier section in the NVMe Express Base Specification.</p>	No
<b>LBA Formats (refer to section 5.TBD)</b>			
131:128	M	<p><b>LBA Format 0 Support (LBAF0):</b> This field indicates the LBA format 0 that is supported by the controller. The LBA format field is defined in Figure 98.</p> <p>Additional information may be provided in the ELBAF0 field (refer to Figure 100).</p>	Yes
135:132	O	<p><b>LBA Format 1 Support (LBAF1):</b> This field indicates the LBA format 1 that is supported by the controller. The LBA format field is defined in Figure 98.</p> <p>Additional information may be provided in the ELBAF1 field (refer to Figure 100).</p>	Yes
...			
383:380	O	<p><b>LBA Format 63 Support (LBAF63):</b> This field indicates the LBA format 63 that is supported by the controller. The LBA format field is defined in Figure 98.</p> <p>Additional information may be provided in the ELBAF63 field (refer to Figure 100).</p>	Yes
383:192		Reserved	

**Figure 97: Identify – Identify Namespace Data Structure, NVM Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
4095:384	O	Vendor Specific	No
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.			

...

**Figure 98: LBA Format Data Structure, NVM Command Set Specific**

Bits	Description
...	
23:16	<b>LBA Data Size (LBADS):</b> This field indicates the LBA data size supported. The value is reported in terms of a power of two ( $2^n$ ). A non-zero value smaller than 9 (i.e., 512 bytes) is not supported. If the value reported is 0h, then the LBA format is not supported / used or is not currently available (refer to section 5.TBD).
...	

...

### **Modify portions of section 4.1.5.3 as shown below:**

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

Figure 100 defines the I/O Command Set specific Identify Namespace data structure for the NVM Command Set.

The Reported column in Figure 100 specifies fields in the I/O Command Set specific Identify Namespace data structure for the NVM Command Set that define namespace capabilities used by a host to format or create a namespace. If the NSID field is set to FFFFFFFFh, then the controller shall return an I/O Command Set specific Identify Namespace data structure for the NVM Command Set that contains:

- fields in Figure 100 set to a value that is the same for all namespaces using any of the LBA formats associated with the Number of LBA Formats field (refer to section 5.TBD) that the Reported column indicates “Yes”; and
- fields in Figure 100 cleared to 0h that the Reported column indicates “No”.

If the controller supports the Namespace Management capability (refer to the Namespace Management section in the NVM Express Base Specification) and the NSID field is set to FFFFFFFFh, then the controller shall return an I/O Command Set specific Identify Namespace data structure for the NVM Command Set. If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller may abort the command with a status code of Invalid Namespace or Format.

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

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>								
7:0	O	<p><b>Logical Block Storage Tag Mask (LBSTM):</b> Identifies the mask for the Storage Tag field for the protection information (refer to section 5.1). The size of this field is defined by the STS field. If the size of this field is less than 64 bits, the mask is contained in the least significant bits of this field.</p> <p>If end-to-end protection is not enabled in the namespace, then this field is ignored.</p> <p>If:</p> <ul style="list-style-type: none"><li>• end-to-end protection is enabled;</li><li>• 16b Guard Protection Information format is used; and</li><li>• The 16BPISTM bit is set to '1',</li></ul> <p>then all bits in the mask shall be set to '1'.</p>	No								
8	O	<p><b>Protection Information Capabilities (PIC):</b> This field indicates the capabilities for the protection information formats.</p> <table><tr><th>Bits</th><th>Description</th></tr><tr><td>7:2</td><td>Reserved</td></tr><tr><td>1</td><td><p><b>16b Guard Protection Information Storage Tag Mask (16BPISTM):</b> If set to '1', then the LBSTM field shall have all bits set to '1' for the 16b Guard Protection Information. If cleared to '0' then the Logical Block Storage Tag Mask field is allowed to have any bits set to '1' for the 16b Guard Protection Information.</p></td></tr><tr><td>0</td><td><p><b>16b Guard Protection Information Storage Tag Support (16BPISTS):</b> If set to '1', then the end-to-end protection 16b Guard Protection Information format (refer to section 5.2.1.1) supports a non-zero value in the STS field. If cleared to '0', then the end-to-end protection 16b Guard Protection Information format support requires that the STS field be cleared to 0h (i.e., the Storage Tag field is not supported).</p><p>If the 32b Guard Protection Information or 64b Guard Protection Information is supported in any LBA format (refer to Figure 97 and Figure 100), then this bit shall be set to '1'.</p></td></tr></table>	Bits	Description	7:2	Reserved	1	<p><b>16b Guard Protection Information Storage Tag Mask (16BPISTM):</b> If set to '1', then the LBSTM field shall have all bits set to '1' for the 16b Guard Protection Information. If cleared to '0' then the Logical Block Storage Tag Mask field is allowed to have any bits set to '1' for the 16b Guard Protection Information.</p>	0	<p><b>16b Guard Protection Information Storage Tag Support (16BPISTS):</b> If set to '1', then the end-to-end protection 16b Guard Protection Information format (refer to section 5.2.1.1) supports a non-zero value in the STS field. If cleared to '0', then the end-to-end protection 16b Guard Protection Information format support requires that the STS field be cleared to 0h (i.e., the Storage Tag field is not supported).</p> <p>If the 32b Guard Protection Information or 64b Guard Protection Information is supported in any LBA format (refer to Figure 97 and Figure 100), then this bit shall be set to '1'.</p>	Yes
Bits	Description										
7:2	Reserved										
1	<p><b>16b Guard Protection Information Storage Tag Mask (16BPISTM):</b> If set to '1', then the LBSTM field shall have all bits set to '1' for the 16b Guard Protection Information. If cleared to '0' then the Logical Block Storage Tag Mask field is allowed to have any bits set to '1' for the 16b Guard Protection Information.</p>										
0	<p><b>16b Guard Protection Information Storage Tag Support (16BPISTS):</b> If set to '1', then the end-to-end protection 16b Guard Protection Information format (refer to section 5.2.1.1) supports a non-zero value in the STS field. If cleared to '0', then the end-to-end protection 16b Guard Protection Information format support requires that the STS field be cleared to 0h (i.e., the Storage Tag field is not supported).</p> <p>If the 32b Guard Protection Information or 64b Guard Protection Information is supported in any LBA format (refer to Figure 97 and Figure 100), then this bit shall be set to '1'.</p>										
11:9		Reserved									
Extended LBA Format (refer to section 5.TBD)											
15:12	O	<b>Extended LBA Format 0 Support (ELBAF0):</b> 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								
19:16	O	<b>Extended LBA Format 1 Support (ELBAF1):</b> This field indicates additional LBA Format 1 information related to the LBA Format 1 Support (LBAF1) field in the Identify Namespace data structure. The Extended LBA format field is defined in Figure 101..	Yes								
...	...	...									
267:264	O	<b>Extended LBA Format 63 Support (ELBAF63):</b> 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								
4095:268	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.

## **Add subsections to section 4.1.5 as shown below:**

### **4.1.5.TBD NVM Command Set Identify Namespace Data Structure (CNS 09h, CSI 00h)**

An NVM Command Set Identify Namespace data structure (refer to Figure 97) is returned to the host for the Format Index specified by the CNS Specific Identifier field as defined in Figure CNS\_TBDH. The returned NVM Command Set Identify Namespace data structure specifies fields that define capabilities used by a host to format or create a namespace. If the specified Format Index is valid (refer to section 5.TBD), then the controller shall return an NVM Command Set Identify Namespace data structure that contains:

- fields in Figure 97 set to a value that is the same for all namespaces using the specified Format Index that the Reported column indicates “Yes”; and
- fields in Figure 97 cleared to 0h that the Reported column indicates “No”.

**Figure CNS\_TBDH: Command Dword 11 – CNS Specific Identifier**

Bits	Description
15:0	<b>Format Index:</b> This field specifies the Format Index identifying the LBA Format for which capabilities are to be returned. Refer to section 5.TBD.

## **Add section 4.1.5.TBD1 as shown below:**

### **4.1.5.TBD1 Identify I/O Command Set specific Namespace data structure (CNS 0Ah, CSI 00h)**

An I/O Command Set specific Identify Namespace data structure for the NVM Command Set (refer to Figure 100) is returned to the host for the Format Index specified by the CNS Specific Identifier field as defined in Figure CNS\_TBDH. The returned I/O Command Set specific Identify Namespace data structure for the NVM Command Set specifies fields that define capabilities used by a host to format or create a namespace. If the specified Format Index is valid (refer to section 5.TBD), then the controller shall return an I/O Command Set specific Identify Namespace data structure for the NVM Command Set that contains:

- fields in Figure 100 set to a value that is the same for all namespaces using the specified Format Index that the Reported column indicates “Yes”; and
- fields in Figure 100 cleared to 0h that the Reported column indicates “No”.

### **4.1.5.TBD2 Command Set Index Usage for the NVM Command Set**

The following sections provide an example on how a host uses the CSI value of 00h for accessing Identify Namespace data structures for a namespace associated with the NVM Command Set.

#### **4.1.5.TBD2.1 Determining the Identify Command Information Associated with a Namespace**

For a host to determine the Identify Namespace Data Structures (refer to section 1.4.2.TBD1) for a namespace associated with the NVM Command Set, the host is required to issue the following Identify commands in any order:

- a) An Identify command with:
  - a. the CNS field set to 08h; and
  - b. the NSID field set to the NSID of the namespace,to access the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification);
- b) An Identify command with:
  - a. the CNS field set to 00h; and
  - b. the NSID field set to the NSID of the namespace,to access the Identify Namespace data structure (refer to section 4.1.5.1);

c) An Identify command with:

- a. the CNS field set to 05h;
- b. the CSI field set to 00h; and
- c. the NSID field set to the NSID of the zoned namespace,

to access the I/O Command Set specific Identify Namespace data structure for the NVM Command Set (refer to section 4.1.5.3).

#### 4.1.5.TBD2.2 Determining the Identify Command Information Associated with a Format Index

For a host to determine the Identify Namespace Data Structures associated with a specific Format Index (i.e., determining information about a namespace associated with the NVM Command Set prior to creating that namespace), the host is required to issue the following Identify commands in any order:

a) An Identify command with:

- a. the CNS field set to 08h; and
- b. the NSID field set to FFFFFFFFh,

to access the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification);

b) An Identify command with:

- a. the CNS field set to 09h;
- b. the CSI field set to 00h;
- c. the NSID field set to 0h; and
- d. the CNS Specific Identifier field set to the Format Index,

to access the Identify Namespace data structure (refer to section 4.1.5.TBD);

c) An Identify command with:

- a. the CNS set to 0Ah;
- b. the CSI set to 00h;
- c. the NSID field set to 0h; and
- d. the CNS Specific Identifier field set to the Format Index,

to access the I/O Command Set specific Identify Namespace data structure for the NVM Command Set (refer to section 4.1.5.TBD1).

...

**Modify figure 103 in section 4.1.5.6 as shown below:**

#### 4.1.5.6 Namespace Granularity List (CNS 16h)

...

Figure 103: Namespace Granularity List

Bytes	Description						
03:00	<b>Namespace Granularity Attributes:</b> This field indicates attributes of the Namespace Granularity List.  <del>Bits 31:1 are reserved.</del>  <del>Bit 0 (<b>Granularity Descriptor Mapping</b>): If set to '1', then each valid namespace granularity descriptor applies to the LBA format having the same index and the Number of Descriptors field shall be equal to the Number of LBA Formats field in the Identify Namespace data structure (refer to Figure 97). If cleared to '0', then NG Descriptor 0 shall apply to all LBA formats and the Number of Descriptors field shall be cleared to 0h.</del>						
	<table><tr><th>Bits</th><th>Description</th></tr><tr><td>31:1</td><td>Reserved</td></tr><tr><td>0</td><td><b>Granularity Descriptor Mapping:</b> If set to '1', then each valid namespace granularity descriptor applies to the LBA format having the same <b>Format Index</b> and the Number of Descriptors field shall be equal to <b>the sum of the values represented by the Number of LBA Formats field and the Number of Unique Attribute LBA Formats field</b> in the Identify Namespace data structure (refer to Figure 97 <b>and section 5.TBD</b>). If cleared to '0', then NG Descriptor 0 shall apply to all LBA formats and the Number of Descriptors field shall be cleared to 0h.</td></tr></table>	Bits	Description	31:1	Reserved	0	<b>Granularity Descriptor Mapping:</b> If set to '1', then each valid namespace granularity descriptor applies to the LBA format having the same <b>Format Index</b> and the Number of Descriptors field shall be equal to <b>the sum of the values represented by the Number of LBA Formats field and the Number of Unique Attribute LBA Formats field</b> in the Identify Namespace data structure (refer to Figure 97 <b>and section 5.TBD</b> ). If cleared to '0', then NG Descriptor 0 shall apply to all LBA formats and the Number of Descriptors field shall be cleared to 0h.
	Bits	Description					
31:1	Reserved						
0	<b>Granularity Descriptor Mapping:</b> If set to '1', then each valid namespace granularity descriptor applies to the LBA format having the same <b>Format Index</b> and the Number of Descriptors field shall be equal to <b>the sum of the values represented by the Number of LBA Formats field and the Number of Unique Attribute LBA Formats field</b> in the Identify Namespace data structure (refer to Figure 97 <b>and section 5.TBD</b> ). If cleared to '0', then NG Descriptor 0 shall apply to all LBA formats and the Number of Descriptors field shall be cleared to 0h.						
...							

Add section 5.TBD as shown below:

## 5 Extended Capabilities

...

### 5.TBD LBA Format List Structure

To create or format a namespace, a host specifies an LBA Format (i.e., the Format Index) referencing the LBA Format list in the NVM Command Set Identify Namespace data structure. The LBA Format list has a structure as is illustrated in **Figure X**. The NLBAF field, a 0-based number (i.e. at least one is required to be defined), identifies the number of LBA Formats that have the same capabilities used to format and create a namespace (i.e., the green LBA Formats). The Identify command provides the ability to access these same capabilities in a namespace data structure by specifying an NSID of FFFFFFFFh.

The NULBAF field, a 1-based number (i.e., none may be defined), identifies the number of LBA Formats that have unique attributes (i.e., that may not be the same capabilities as other LBA Formats) used to format and create a namespace (i.e., the orange LBA Formats). A host should use the Identify command with a CNS value of **09h** to access the capabilities of a specific LBA Format for the NVM Command Set Identify Namespace data structure. A host should use the Identify command with a CNS value of **0Ah** to access the capabilities of a specific LBA Format for an I/O Command Set specific Identify Namespace data structure for the NVM Command Set.

**Figure Y** shows the association of CNS values of the Identify command that provides the ability to access these same capabilities to specific LBA Format entries referenced by the NLBAF field and NULBF field.

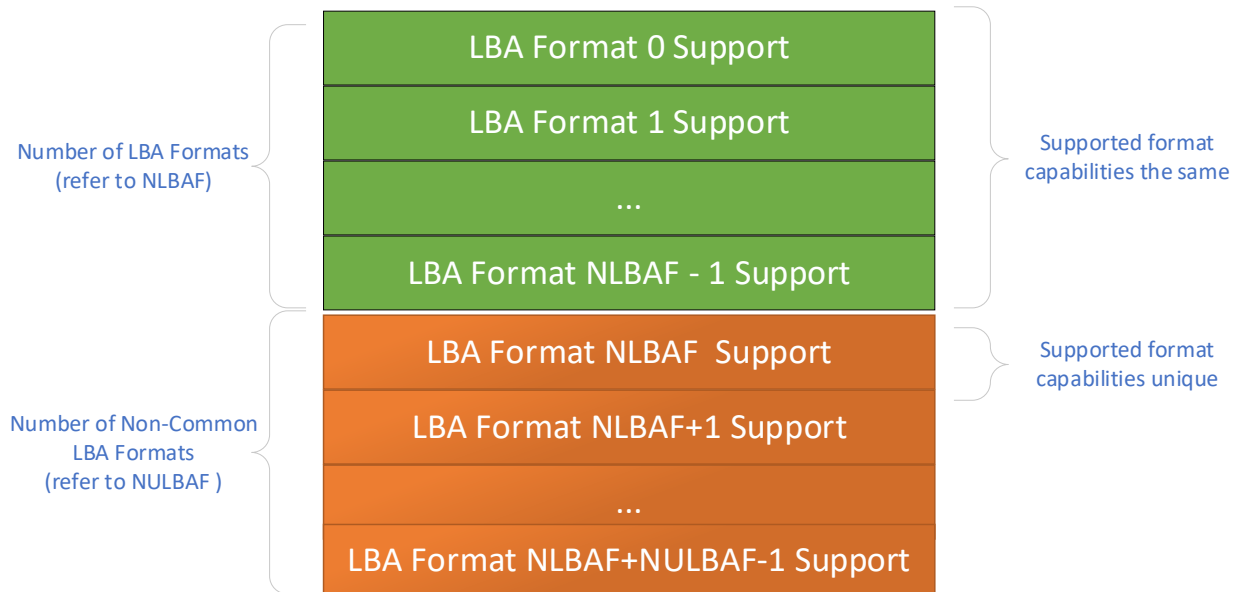
The maximum number of LBA formats allowed to be supported is:



- 16 if the LBA Format Extension Enable (LBAFEE) field is cleared to 0h in the Host Behavior Support feature (refer to the Host Behavior Support section in the NVM Express Base Specification); or
- 64 if the LBAFEE field is set to 1h in the Host Behavior Support feature (refer to the Host Behavior Support section in the NVM Express Base Specification).

The total number of LBA formats supported is the sum of the values represented by the NLBAF field and the NULBAF field. A Format Index is valid if the value is less than the sum of the values represented by the NLBAF field and the NULBAF field. An LBA Format that is supported, but not currently available is indicated by clearing the LBA Data Size field to 0h for that LBA Format.

**Figure X: LBA Format List Structure**



**Figure Y: LBA Format List Entries Applicability to Identify Command CNS Value**

CNS Value	Returned data is associated with LBA Format entries referenced by only the NLBAF field	Returned data is associated with LBA Format entries referenced by both the NLBAF field and NULBAF field
00h	Yes	No
05h	Yes	No
08h	Yes	No
09h	No	Yes
0Ah	No	Yes



# Modify portions of Key Value Command Set Specification 1.0a as shown below:

*Modify section 1.4.2 as shown below:*

## 1 Introduction

...

### 1.4 Definitions

...

#### 1.4.2 Definitions in the NVM Express Base Specification specified in the Key Value Command set

...

##### 1.4.2.TBD Identify Controller data structures

All controller data structures that are able to be retrieved via the Identify command for the Key Value Command Set:

- the Identify Controller data structure (refer to the NVM Express Base Specification); and
- the I/O Command Set specific Identify Controller data structure for the Key Value Command Set (refer to section 4.1.5.2).

##### 1.4.2.TBD1 Identify Namespace data structures

All namespace data structures that are able to be retrieved via the Identify command for the Key Value Command Set:

- the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification); and
- the I/O Command Set specific Identify Namespace data structure for the Key Value Command Set (refer to section 4.1.5.1).

...

*Modify section 4.1.5 as shown below:*

## 4 Admin Commands for the Key Value Command Set

...

### 4.1 Admin Command behavior for the Key Value Command Set

...

#### 4.1.5 Identify Command

<Editor Note: This section level is not correct in the released specification and an ECN will be used to correct the issue note this TP>

This specification implements the Identify Command and associated Identify data structures defined in the NVM Express Base specification. Additionally, the Key Value Command Set specifies the data structures defined in this section.

Each I/O Command Set is assigned a specific Command Set Identifier (CSI) value by the NVM Express Base Specification. The Key Value Command Set is assigned a CSI value of 01h.

**Figure 35: Identify – CNS Values**

CNS Value	O/M <sup>1</sup>	Definition	NSID <sup>2</sup>	CNTID <sup>3</sup>	CSI <sup>4</sup>	Reference Section
<b>Active Namespace Management</b>						
05h	M <sup>5</sup>	Identify I/O Command Set specific Namespace data structure for the specified NSID for the I/O Command Set specified in the CSI field.	Y	N	Y	4.1.5.1
06h	M	Identify I/O Command Set Specific Controller data structure for the controller processing the command.	Y	N	Y	4.1.5.2
0Ah	O	I/O Command Set specific Identify Namespace data structure for the specified Format Index for the I/O Command Set specified in the CSI field. <sup>6</sup>	N	N	Y	4.1.5.TBD
NOTES: 1. O/M definition: O = Optional, M = Mandatory. 2. The NSID field is used: Y = Yes, N = No. 3. The CDW10.CNTID field is used: Y = Yes, N = No. 4. The CDW11.CSI field is used: Y = Yes, N = No. 5. Mandatory for controllers that support the Namespace Management capability (refer to the NVM Express Base Specification). 6. Selection of a UUID may be supported. Refer to the Universally Unique Identifiers (UUIDs) for Vendor Specific Information section in the NVM Express Base Specification.						

**Modify section 4.1.5.1 as shown below:**

#### 4.1.5.1 I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 01h)

The I/O Command Set specific Identify Namespace data structure (i.e., CNS 05h) for the Key Value Command Set is defined in Figure 36.

**Figure 36: Identify – Identify Namespace Data Structure, Key Value Type Specific**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
07:00	M	<b>Namespace Size (NSZE):</b> This field indicates the total size of the namespace in bytes. This is the space to store KV keys and KV values. This field is undefined prior to the namespace being formatted.	No
15:08		Reserved	
23:16	M	<b>Namespace Utilization (NUSE):</b> This field indicates the current number of bytes allocated in the namespace. This is the space to store KV keys and KV values. This field is smaller than or equal to the Namespace Size.  A key value pair is allocated when it is written with a Store command. A key value pair is deallocated using the Delete command.  If the controller supports Asymmetric Namespace Access Reporting (refer to the CMIC field), and the relationship between the controller and the namespace is in the ANA Inaccessible state (refer to the NVM Express Base Specification) or the ANA Persistent Loss state (refer to the NVM Express Base Specification), then this field shall be cleared to 0h.	No

**Figure 36: Identify – Identify Namespace Data Structure, Key Value Type Specific**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
24	M	<b>Namespace Features (NSFEAT):</b> This field defines features of the namespace. Bits 7:4 are reserved. Bit 3 if set to '1' indicates that the non-zero NGUID and non-zero EUI64 fields for this namespace are never reused by the controller. If cleared to '0', then the NGUID and EUI64 values may be reused by the controller for a new namespace created after this namespace is deleted. This bit shall be cleared to '0' if both NGUID and EUI64 fields are cleared to 0h. Refer to the NVMe Express Base Specification. Bits 2:0 are reserved.	No
25	M	<b>Number of KV Formats (NKVF):</b> This field defines the number of KV format descriptors supported by the namespace. KV formats shall be allocated in order (starting with 0) and packed sequentially. This is a 0's based value. The maximum number of KV formats that may be indicated as supported is 16. The supported KV formats are indicated in bytes 72 to 327 in this data structure. The KV Format fields with an index beyond the value set in this field are invalid and not supported. KV Formats that are valid, but not currently available may be indicated by setting the KV Key Max and KV Value Max both to 0000h for that KV Format.	Yes
26	O	<b>Namespace Multi-path I/O and Namespace Sharing Capabilities (NMIC):</b> Refer to the NMIC field in the Identify Namespace data structure in the NVMe Express Base Specification.	No
27	O	<b>Reservation Capabilities (RESCAP):</b> Refer to NVMe Express Base Specification.	No
28	O	<b>Format Progress Indicator (FPI):</b> Refer to NVMe Express Base Specification.	No
31:29		Reserved	
35:32	O	<b>Namespace Optimal Value Granularity (NOVG):</b> This field indicates the optimal value granularity for this namespace. This field is specified in bytes. The host should construct Store commands that store multiples of NOVG bytes to achieve optimal performance. A value of 0h indicates that no optimal value granularity is reported.	No
39:36	O	<b>ANA Group Identifier (ANAGRPID):</b> Refer to NVMe Express Base Specification.	No
42:40		Reserved	
43	O	<b>Namespace Attributes (NSATTR):</b> Refer to NVMe Express Base Specification.	No
45:44	O	<b>NVM Set Identifier (NVMSETID):</b> Refer to NVMe Express Base Specification.	No
47:46	O	<b>Endurance Group Identifier (ENDGID):</b> Refer to NVMe Express Base Specification.	No
63:48	O	<b>Namespace Globally Unique Identifier (NGUID):</b> Refer to NVMe Express Base Specification.	No
71:64	O	<b>IEEE Extended Unique Identifier (EUI64):</b> Refer to NVMe Express Base Specification.	No
<b>KV Formats</b>			
87:72	M	<b>KV Format 0 Support (KVF0):</b> This field indicates the KV format 0 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
103:88	O	<b>KV Format 1 Support (KVF1):</b> This field indicates the KV format 1 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
<del>119:104</del>	<del>O</del>	<del><b>KV Format 2 Support (KVF2):</b> This field indicates the KV format 2 that is supported by the controller. The KV format field is defined in Figure 37.</del>	<del>Yes</del>
<del>135:120</del>	<del>O</del>	<del><b>KV Format 3 Support (KVF3):</b> This field indicates the KV format 3 that is supported by the controller. The KV format field is defined in Figure 37.</del>	<del>Yes</del>

**Figure 36: Identify – Identify Namespace Data Structure, Key Value Type Specific**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
151:136	O	KV Format 4 Support (KVF4): This field indicates the KV format 4 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
167:152	O	KV Format 5 Support (KVF5): This field indicates the KV format 5 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
183:168	O	KV Format 6 Support (KVF6): This field indicates the KV format 6 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
199:184	O	KV Format 7 Support (KVF7): This field indicates the KV format 7 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
215:200	O	KV Format 8 Support (KVF8): This field indicates the KV format 8 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
231:216	O	KV Format 9 Support (KVF9): This field indicates the KV format 9 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
247:232	O	KV Format 10 Support (KVF10): This field indicates the KV format 10 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
263:248	O	KV Format 11 Support (KVF11): This field indicates the KV format 11 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
279:264	O	KV Format 12 Support (KVF12): This field indicates the KV format 12 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
295:280	O	KV Format 13 Support (KVF13): This field indicates the KV format 13 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
311:296	O	KV Format 14 Support (KVF14): This field indicates the KV format 14 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
...	...	...	
327:312	O	KV Format 15 Support (KVF15): This field indicates the KV format 15 that is supported by the controller. The KV format field is defined in Figure 37.	Yes
3839:328		Reserved	
4095:3840	O	Vendor Specific	No
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.			

**Add new subsections to section 4.1.5.1 as shown below:**

#### 4.1.5.TBD I/O Command Set specific Identify Namespace data structure (CNS 0Ah, CSI 01h)

An I/O Command Set specific Identify Namespace data structure for the Key Value Command Set (refer to Figure 9) is returned to the host for the Format Index specified by the CNS Specific Identifier field as defined in Figure CNS\_TBDH. The returned I/O Command Set specific Identify Namespace data structure for the Key Value Command Set specifies fields that define capabilities used by a host to format or create a namespace. If the Format Index specified is valid, then the controller shall return an I/O Command Set specific Identify Namespace data structure for the Key Value Command Set that contains:

- fields in Figure 9 set to a value that is the same for all namespaces using the specified Format Index that the Reported column indicates “Yes”; and
- fields in Figure 9 cleared to 0h that the Reported column indicates “No”.

**Figure CNS\_TBDH: Command Dword 11 – CNS Specific Identifier**

Bits	Description
15:0	<b>Format Index:</b> This field specifies the Format Index (refer to Figure 36) identifying the KV Format capabilities that are to be returned.

#### 4.1.5.TBD1 Command Set Index Usage for the Key Value Command Set

The following sections provide an example on how a host uses the CSI value of 01h for accessing Identify Namespace data structures for a namespace associated with the Key Value Command Set.

##### 4.1.5.TBD1.1 Determining the Identify Command Information Associated with a Namespace

For a host to determine the Identify Namespace Data Structures (refer to section 1.4.2.TBD1) for a namespace associated with the Key Value Command Set, the host is required to issue the following Identify commands in any order:

- a) An Identify command with:
  - a. the CNS field set to 08h; and
  - b. the NSID field set to the NSID of the namespace,to access the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification);
- b) An Identify command with:
  - a. the CNS field set to 05h;
  - b. the CSI field set to 01h; and
  - c. the NSID field set to the NSID of the namespace,to access the I/O Command Set specific Identify Namespace data structure for the Key Value Command Set (refer to section 4.1.5.1).

##### 4.1.5.TBD1.2 Determining the Identify Command Information Associated with a Format Index

For a host to determine the Identify Namespace Data Structures associated with a specific Format Index (i.e., determining information about a namespace associated with the Key Value Command Set prior to creating that namespace), the host is required to issue the following Identify commands in any order:

- a) An Identify command with:
  - a. the CNS field set to 08h; and
  - b. the NSID field set to FFFFFFFFh,to access the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification);
- b) An Identify command with:
  - a. the CNS set to 0Ah;
  - b. the CSI set to 01h;
  - c. the NSID field set to 0h; and
  - d. the CNS Specific Identifier field set to the Format Index,to access the I/O Command Set specific Identify Namespace data structure for the Key Value Command Set (refer to 4.1.5.TBD).

## Modify portions of Zoned Namespace Command Set Specification 1.1a as shown below:

*Modify section 1.4.2 as shown below:*

### 1 Introduction

...

#### 1.4 Definitions

...

##### 1.4.2 Terms in the NVM Express Base Specification specified in the Zoned Namespace Command set

...

##### 1.4.2.TBD Identify Controller data structures

All controller data structures that are able to be retrieved via the Identify command for the Zoned Namespace Command Set:

- the Identify Controller data structure (refer to the NVM Express Base Specification);
- the I/O Command Set specific Identify Controller data structure for the NVM Command Set (refer to the NVM Command Set Specification); and
- the I/O Command Set specific Identify Controller data structure for the Zoned Namespace Command Set (refer to section 4.1.5.2).

##### 1.4.2.TBD1 Identify Namespace data structures

All namespace data structures that are able to be retrieved via the Identify command for the Zoned Namespace Set (refer to section 4.1.5.TBD1 for specific field settings for the Identify command):

- the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification);
- Identify Namespace data structure (refer to NVM Command Set Specification);
- the I/O Command Set specific Identify Namespace data structure for the NVM Command Set (refer to section NVM Command Set Specification); and
- the I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set (refer to section 4.1.5.1).

...

*Modify section 4.1.5 as shown below:*

### 4 Admin Commands for the Zoned Namespace Command Set

...

#### 4.1 Admin Command behavior for the Zoned Namespace Command Set

...

## 4.1.5 Identify Command

...

**Figure 47: CNS Values**

CNS Value	O/M <sup>1</sup>	Definition	NSID <sup>2</sup>	CNTID <sup>3</sup>	CSI <sup>4</sup>	Reference Section
<b>Active Namespace Management</b>						
00h	M	Identify Namespace data structure for the specified NSID or the <del>common</del> namespace capabilities for the NVM Command Set. <sup>5</sup>	Y	N	N	NVM Command Set Specification
01h	M	Identify Controller Data Structure, I/O Command Set Independent. <sup>5</sup>	N	N	N	NVM <del>Express</del> Base Specification
05h	M	I/O Command Set specific Identify Namespace data structure for the specified NSID for the I/O Command Set specified in the CSI field. <sup>5</sup>	Y	N	<del>Y</del> 00h	<del>CSI 00h:</del> NVM Command Set Specification
					02h	<del>CSI 02h:</del> 4.1.5.1
06h	M	Identify I/O Command Set Specific Controller data structure for the controller processing the command. <sup>5</sup>	Y	N	<del>Y</del> 00h	NVM Command Set Specification
					02h	4.1.5.2
09h	O	Identify Namespace data structure for the specified Format Index containing the namespace capabilities for the NVM Command Set. <sup>5</sup>	N	N	00h	NVM Command Set Specification
0Ah	O	I/O Command Set specific Identify Namespace data structure for the specified Format Index for the I/O Command Set specified in the CSI field. <sup>5</sup>	N	N	00h	NVM Command Set Specification
					02h	4.1.5.TBD
16h	O	A Namespace Granularity List is returned to the host for up to sixteen Namespace Granularity Entries.	N	N	N <sup>6</sup>	NVM Command Set Specification

NOTES:

- O/M definition: O = Optional, M = Mandatory.
- The NSID field is used: Y = Yes, N = No.
- The CDW10.CNTID field is used: Y = Yes, N = No.
- The CDW11.CSI field value or N if the field is not used: ~~Y = Yes, N = No~~. Since this specification is an extension of the NVM Command Set (refer to section 1.1) a host is required to use the CSI value of 00h and the CSI value of 02h to obtain all of the refer to section 1.4.2.TBD, section 1.4.2.TBD1, and section 4.1.5.TBD1.
- Selection of a UUID may be supported. Refer to the UUIDs for Vendor Specific Information section of the NVM Command Set Specification.
- ~~This Identify data structure applies to namespace that are associated with command sets that specify logical blocks (i.e., Command Set Identifier 0h or 02h).~~

### Modify section 4.1.5.1 as shown below:

#### 4.1.5.1 Identify I/O Command Set Specific Namespace Data Structure for the Zoned Namespace Command Set (CNS 05h, CSI 02h)

Figure 48 defines the Identify I/O Command Set specific Namespace data structure for the Zoned Namespace Command Set.

Figure 48 specifies fields in the I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set that define capabilities used by a host to format or create a namespace. If the



NSID field is set to FFFFFFFFh, then the controller shall return an I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set that contains:

- fields in Figure 48 set to a value that is the same for all namespaces using any of the LBA formats associated with the Number of LBA Formats field (refer to the LBA Format List Structure section in the NVM Command Set Specification) that the Reported column indicates “Yes”; and
- fields in Figure 48 cleared to 0h that the Reported column indicates “No”.

If the controller supports the Namespace Management capability (refer to the Namespace Management section in the NVM Express Base Specification) and the NSID field is set to FFFFFFFFh, then the controller shall return an I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set. If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller may abort the command with a status code of Invalid Namespace or Format.

**Figure 48: Identify I/O Command Set Specific Namespace Data Structure for the Zoned Namespace Command Set**

Command Set

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>								
01:00	O	<b>Zone Operation Characteristics (ZOC):</b> This field indicates the zone operation characteristics of the zoned namespace. <table><tr><th>Bits</th><th>Description</th></tr><tr><td>15:2</td><td>Reserved</td></tr><tr><td>1</td><td><b>Zone Active Excursions:</b> If set to '1', then a controller may transition a zone in the ZSIO:Implicitly Opened state, the ZSEO:Explicitly Opened state, or the ZSC:Closed state to the ZSF:Full state due to a vendor specific excursion event. If cleared to '0', then a controller shall not transition a zone due to a vendor specific excursion event. Refer to section 5.6.</td></tr><tr><td>0</td><td><b>Variable Zone Capacity:</b> if set to '1', then the capacity for a zone may change without a change to the format of the zoned namespace. If cleared to '0', then the capacity for a zone does not change without a change to the format of the zoned namespace. Refer to Figure 37.</td></tr></table>	Bits	Description	15:2	Reserved	1	<b>Zone Active Excursions:</b> If set to '1', then a controller may transition a zone in the ZSIO:Implicitly Opened state, the ZSEO:Explicitly Opened state, or the ZSC:Closed state to the ZSF:Full state due to a vendor specific excursion event. If cleared to '0', then a controller shall not transition a zone due to a vendor specific excursion event. Refer to section 5.6.	0	<b>Variable Zone Capacity:</b> if set to '1', then the capacity for a zone may change without a change to the format of the zoned namespace. If cleared to '0', then the capacity for a zone does not change without a change to the format of the zoned namespace. Refer to Figure 37.	No
Bits	Description										
15:2	Reserved										
1	<b>Zone Active Excursions:</b> If set to '1', then a controller may transition a zone in the ZSIO:Implicitly Opened state, the ZSEO:Explicitly Opened state, or the ZSC:Closed state to the ZSF:Full state due to a vendor specific excursion event. If cleared to '0', then a controller shall not transition a zone due to a vendor specific excursion event. Refer to section 5.6.										
0	<b>Variable Zone Capacity:</b> if set to '1', then the capacity for a zone may change without a change to the format of the zoned namespace. If cleared to '0', then the capacity for a zone does not change without a change to the format of the zoned namespace. Refer to Figure 37.										
03:02	O	<b>Optional Zoned Command Support (OZCS):</b> This field defines optional features of the zoned namespace. <table><tr><th>Bits</th><th>Description</th></tr><tr><td>15:1</td><td>Reserved</td></tr><tr><td>0</td><td><b>Read Across Zone Boundaries:</b> If set to '1', then any User Data Read Access Command is allowed to perform read operations that specify an LBA range containing logical blocks in more than one zone.  If cleared to '0', then any command that performs a read operation that specifies an LBA range containing logical blocks in more than one zone is aborted as defined in section 2.1.1.2.1.2.</td></tr></table>	Bits	Description	15:1	Reserved	0	<b>Read Across Zone Boundaries:</b> If set to '1', then any User Data Read Access Command is allowed to perform read operations that specify an LBA range containing logical blocks in more than one zone.  If cleared to '0', then any command that performs a read operation that specifies an LBA range containing logical blocks in more than one zone is aborted as defined in section 2.1.1.2.1.2.	No		
Bits	Description										
15:1	Reserved										
0	<b>Read Across Zone Boundaries:</b> If set to '1', then any User Data Read Access Command is allowed to perform read operations that specify an LBA range containing logical blocks in more than one zone.  If cleared to '0', then any command that performs a read operation that specifies an LBA range containing logical blocks in more than one zone is aborted as defined in section 2.1.1.2.1.2.										
07:04	M	<b>Maximum Active Resources (MAR):</b> This field defines the maximum number of concurrently active zones in the zoned namespace. A value of FFFFFFFFh indicates that there is no limit. This is a 0's based value.	No								
11:08	M	<b>Maximum Open Resources (MOR):</b> This field defines the maximum number of concurrently open zones in the zoned namespace. This field shall be less than or equal to the Maximum Active Resources field. A value of FFFFFFFFh indicates that there is no limit. This is a 0's based value.	No								
15:12	O	<b>Reset Recommended Limit (RRL):</b> If the zone attribute Reset Zone Recommended Time Limit field is set to 00b, then this field indicates the number of seconds before the NVM subsystem may perform a vendor specific action on a zone after the Reset Zone Recommended zone attribute is set to '1' for that zone. If this field is cleared to 0h, then no Reset Recommended Limit is reported. Refer to section 5.4.	No								



**Figure 48: Identify I/O Command Set Specific Namespace Data Structure for the Zoned Namespace Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
19:16	O	<b>Finish Recommended Limit (FRL):</b> If the zone attribute Finish Zone Recommended Time Limit field is set to 00b, then this field indicates the number of seconds before the NVM subsystem may perform the vendor specific action on a zone after the Finish Zone Recommended zone attribute is set to '1' for that zone. If this field is cleared to 0h, then no Finish Recommended Limit is reported. Refer to section 5.5.	No
23:20	O	<b>Reset Recommended Limit 1 (RRL 1):</b> If the zone attribute Reset Zone Recommended Time Limit field is set to 01b, then this field indicates the number of seconds before the NVM subsystem may perform a vendor specific action on a zone after the Reset Zone Recommended zone attribute is set to '1' for that zone. If this field is cleared to 0h, then no Reset Recommended Limit is reported. Refer to section 5.4.	No
27:24	O	<b>Reset Recommended Limit 2 (RRL 2):</b> If the zone attribute Reset Zone Recommended Time Limit field is set to 10b, then this field indicates the number of seconds before the NVM subsystem may perform a vendor specific action on a zone after the Reset Zone Recommended zone attribute is set to '1' for that zone. If this field is cleared to 0h, then no Reset Recommended Limit is reported. Refer to section 5.4.	No
31:28	O	<b>Reset Recommended Limit 3 (RRL 3):</b> If the zone attribute Reset Zone Recommended Time Limit field is set to 11b, then this field indicates the number of seconds before the NVM subsystem may perform a vendor specific action on a zone after the Reset Zone Recommended zone attribute is set to '1' for that zone. If this field is cleared to 0h, then no Reset Recommended Limit is reported. Refer to section 5.4.	No
35:32	O	<b>Finish Recommended Limit 1 (FRL1):</b> If the zone attribute Finish Zone Recommended Time Limit field is set to 01b, then this field indicates the number of seconds before the NVM subsystem may perform the vendor specific action on a zone after the Finish Zone Recommended zone attribute is set to '1' for that zone. If this field is cleared to 0h, then no Finish Recommended Limit is reported. Refer to section 5.5.	No
39:36	O	<b>Finish Recommended Limit 2 (FRL2):</b> If the zone attribute Finish Zone Recommended Time Limit field is set to 10b, then this field indicates the number of seconds before the NVM subsystem may perform the vendor specific action on a zone after the Finish Zone Recommended zone attribute is set to '1' for that zone. If this field is cleared to 0h, then no Finish Recommended Limit is reported. Refer to section 5.5.	No
43:40	O	<b>Finish Recommended Limit 3 (FRL3):</b> If the zone attribute Finish Zone Recommended Time Limit field is set to 11b, then this field indicates the number of seconds before the NVM subsystem may perform the vendor specific action on a zone after the Finish Zone Recommended zone attribute is set to '1' for that zone. If this field is cleared to 0h, then no Finish Recommended Limit is reported. Refer to section 5.5.	No
2815:44		Reserved	
<b>LBA Format Extensions</b> (refer to the LBA Format List Structure section in the NVM Command Set Specification)			
2831:2816	M	<b>LBA Format 0 Extension (LBAFE0):</b> This field indicates the LBA format Extension 0 that is supported by the controller. The Zone format field is defined in Figure 49.	Yes
2847:2832	O	<b>LBA Format 1 Extension (LBAFE1):</b> This field indicates the LBA format Extension 1 that is supported by the controller. The LBA Format Extension field is defined in Figure 49.	Yes
...			
3839:3824	O	<b>LBA Format 63 Extension (LBAFE63):</b> This field indicates the LBA format Extension 63 that is supported by the controller. The LBA Format Extension field is defined in Figure 49.	Yes

**Figure 48: Identify I/O Command Set Specific Namespace Data Structure for the Zoned Namespace Command Set**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
4095:3840	O	Vendor Specific	No
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.			

**Add new subsections to section 4.1.5 as shown below:**

#### 4.1.5.TBD Identify I/O Command Set specific Namespace data structure (CNS 0Ah, CSI 02h)

An I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set (refer to Figure 48) is returned to the host for the Format Index specified by the CNS Specific Identifier field as defined in Figure CNS\_TBDH. The returned I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set specifies fields that define capabilities used by a host to format or create a namespace. If the Format Index specified is valid, then the controller shall return an I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set that contains:

- fields in Figure CNS\_TBDH set to a value that is the same for all namespaces using the specified Format Index that the Reported column indicates “Yes”; and
- fields in Figure CNS\_TBDH cleared to 0h that the Reported column indicates “No”.

**Figure CNS\_TBDH: Command Dword 11 – CNS Specific Identifier**

Bits	Description
15:0	<b>Format Index:</b> This field specifies the Format Index identifying the LBA Format capabilities that are to be returned. Refer to the NVM Command Set Specification.

#### 4.1.5.TBD1 Command Set Index Usage for the Zoned Namespace Command Set

The Zoned Namespace Command Set is an extension to the NVM Command Set as specified in section 1.1. Therefore, the CNS values defined by the NVM Command Set (i.e, 00h) and Zoned Namespace Command Set (i.e., 02h) are used by the host for the Zoned Namespace Command Set.

The following sections provide an example on how a host uses the CSI value 00h and CSI value 02h for accessing Identify Namespace data structures (refer to section 1.4.2.TBD1) for a namespace associated with the Zoned Namespace Command Set.

##### 4.1.5.TBD1.1 Determining the Identify Command Information Associated with a Namespace

For a host to determine the Identify Namespace Data Structures (refer to section 1.4.2.TBD1) for a namespace associated with the Zoned Namespace Command Set, the host is required to issue the following Identify commands in any order:

- An Identify command with:
  - the CNS field set to 08h; and
  - the NSID field set to the NSID of the namespace,
 to access the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification);
- An Identify command with:
  - the CNS field set to 00h; and
  - the NSID field set to the NSID of the namespace,

to access the Identify Namespace data structure (refer to the NVM Command Set Specification);  
c) An Identify command with:

- a. the CNS field set to 05h;
- b. the CSI field set to 00h; and
- c. the NSID field set to the NSID of the namespace,

to access the I/O Command Set specific Identify Namespace data structure for the NVM Command Set (refer to the NVM Command Set Specification); and

d) An Identify command with:

- a. the CNS field set to 05h;
- b. the CSI field set to 02h; and
- c. the NSID field set to the NSID of the namespace,

to access the I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set (refer to section 4.1.5.1).

#### 4.1.5.TBD1.2 Determining the Identify Command Information Associated with a Format Index

For a host to determine the Identify Namespace Data Structures associated with a specific Format Index (i.e., determining information about a namespace associated with the Zoned Namespace Command Set prior to creating that namespace), the host is required to issue the following Identify commands in any order:

a) An Identify command with:

- a. the CNS field set to 08h; and
- b. the NSID field set to FFFFFFFFh,

to access the I/O Command Set Independent Identify Namespace data structure (refer to the NVM Express Base Specification);

b) An Identify command with:

- a. the CNS field set to 09h;
- b. the CSI field set to 00h;
- c. the NSID field set to 0h; and
- d. the CNS Specific Identifier field set to the Format Index,

to access the Identify Namespace data structure (refer to the NVM Command Set Specification);

c) An Identify command with:

- a. the CNS field set to 0Ah;
- b. the CSI field set to 00h;
- c. the NSID field set to 0h; and
- d. the CNS Specific Identifier field set to the Format Index,

to access the I/O Command Set specific Identify Namespace data structure for the NVM Command Set (refer to the NVM Command Set Specification); and

d) An Identify command with:

- a. the CNS field set to 0Ah;
- b. the CSI field set to 02h;
- c. the NSID field set to 0h; and
- d. the CNS Specific Identifier field set to the Format Index,

to access the I/O Command Set specific Identify Namespace data structure for the Zoned Namespace Command Set (refer to section 4.1.5.TBD).

# Modify portions of NVM Express Base Specification 2.0a:

Modify section 5.17 as shown below:

## 5 Admin Command Set

...

### 5.17 Identify command

#### 5.17.1 Identify command overview

...

Figure 271: Identify – Command Dword 11

Bits	Description								
31:24	<b>Command Set Identifier (CSI):</b> This field is CNS value specific. This field specifies the I/O Command Set to be used by the command for CNS values that require a Command Set Identifier. Refer to Figure 273 for Identify command CNS values that use this field. This field shall be cleared to 0h for Identify operations with CNS values that do not use this field.  Values for this field are defined by Figure 274.								
23:16	Reserved								
15:00	<b>CNS Specific Identifier:</b> <del>This field specifies an identifier that is required for a particular CNS value.</del> This field is dependent on the specified CNS value and if not defined by that CNS value, then this field is reserved. <del>The CNS values that require a CNS specific identifier are indicated in the table below.</del> <table border="1"> <thead> <tr> <th>CNS-Value</th><th>Definition</th></tr> </thead> <tbody> <tr> <td>NVM-Set-List (04h)</td><td>NVM-Set-Identifier (NVMSETID) (refer to section 3.2.2)</td></tr> <tr> <td>Domain-List (18h)</td><td>Domain-Identifier (DOMID) (refer to section 3.2.4.3)</td></tr> <tr> <td>Endurance-Group-List (19h)</td><td>Endurance-Group-Identifier (ENDGID) (refer to section 3.2.3)</td></tr> </tbody> </table> << Note – the information in the table in this field are being moved to the definition of the CNS value. >>	CNS-Value	Definition	NVM-Set-List (04h)	NVM-Set-Identifier (NVMSETID) (refer to section 3.2.2)	Domain-List (18h)	Domain-Identifier (DOMID) (refer to section 3.2.4.3)	Endurance-Group-List (19h)	Endurance-Group-Identifier (ENDGID) (refer to section 3.2.3)
CNS-Value	Definition								
NVM-Set-List (04h)	NVM-Set-Identifier (NVMSETID) (refer to section 3.2.2)								
Domain-List (18h)	Domain-Identifier (DOMID) (refer to section 3.2.4.3)								
Endurance-Group-List (19h)	Endurance-Group-Identifier (ENDGID) (refer to section 3.2.3)								

...

Figure 273: Identify – CNS Values

CNS Value	O/M <sup>1</sup>	Definition	NSID <sup>2</sup>	CNTID <sup>3</sup>	CSI <sup>4</sup>	Reference Section
<b>Active Namespace Management</b>						
00h	M	Identify Namespace data structure for the specified NSID or the <del>common</del> namespace capabilities for the NVM Command Set. <sup>7</sup>	Y	N	N <sup>8</sup>	NVM Command Set Specification
...						
07h	M	Active Namespace ID list associated with the specified I/O Command Set.	Y	N	Y	5.17.2.7
09h	O	Identify Namespace data structure for the specified Format Index containing the namespace capabilities for the NVM Command Set. <sup>7</sup>	N	N	Y	NVM Command Set Specification
0Ah	O	I/O Command Set specific Identify Namespace data structure for the specified Format Index containing the namespace capabilities for the I/O Command Set specified in the CSI field. <sup>7</sup>	N	N	Y	I/O Command Set Specification

**Figure 273: Identify – CNS Values**

CNS Value	O/M <sup>1</sup>	Definition	NSID <sup>2</sup>	CNTID <sup>3</sup>	CSI <sup>4</sup>	Reference Section
0E08h to 0Fh		Reserved				
<b>Controller and Namespace Management</b>						
...						
18h to 1Fh		Reserved				
<b>Future Definition</b>						
20h to FFh		Reserved				
NOTES: 1. O/M definition: O = Optional, M = Mandatory. 2. The NSID field is used: Y = Yes, N = No. 3. The CDW10.CNTID field is used: Y = Yes, N = No. 4. The CDW11.CSI field is used: Y = Yes, N = No. 5. Mandatory for controllers that support the Namespace Management capability (refer to section 8.11). 6. Mandatory for controllers that support Virtualization Enhancements (refer to section 8.26). 7. Selection of a UUID may be supported (refer to section 8.25). 8. This Identify data structure applies to namespaces that are associated with command sets that specify logical blocks (i.e., Command Set Identifier 0h or 2h). 9. Mandatory for controllers that support Variable Capacity Management (refer to section 8.3.3). 10. Mandatory for controllers that support Capacity Management (refer to section 8.3) in an NVM subsystem that supports multiple domains (refer to section 3.2.4). 11. Only applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.						

**Modify section 5.17.2.4 as shown below:**

#### 5.17.2.4 NVM Set List (CNS 04h)

Figure 278 defines an NVM Set List. The data structure is an ordered list of NVM Set Attribute Entry data structures, sorted 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 specified by the CNS Specific Identifier field as defined in Figure CNS\_04H indicated in CDW11.NVMSETID and are accessible by the controller processing the command. The NVM Set List describes the attributes for each NVM Set in the list based on the NVM Set Attributes Entry in Figure 278.

**Figure CNS\_04H: Command Dword 11 – CNS Specific Identifier**

Bits	Description
15:0	NVM Set Identifier (NVMSETID): This field specifies the NVM Set Identifier (refer to section 3.2.2) of the first NVM Set of the ordered list of NVM Set Attribute Entry data structures to be returned.

The NVM Set List shall not contain an entry cleared to 0h.

...

**Modify section 5.17.2.5 as shown below:**

#### 5.17.2.5 I/O Command Set specific Identify Namespace data structure (CNS 05h)

An I/O Command Set specific Identify Namespace data structure (refer to the applicable I/O Command Set specification) is returned to the host for the namespace specified in the Namespace Identifier (NSID) field if the NSID is active. If the specified namespace is an inactive NSID, then the controller returns a zero filled data structure.

The specific Identify Namespace data structure that is returned by this command is specified by the Command Set Identifier (CSI) field (refer to Figure 274). If the I/O Command Set associated with the namespace identified by the NSID field does not support the Identify Namespace data structure specified by the CSI field, the controller shall abort the command with a status code of Invalid Field in Command.

~~If the controller supports the Namespace Management capability (refer to section 8.11), the I/O Command Set requested in the CSI field has been enabled with the I/O Command Set Profile feature (refer to section 5.27.1.21), and the NSID field is set to FFFFFFFFh, then the controller returns an I/O Command Set specific Identify Namespaces data structure that specifies capabilities that are common across namespaces for the I/O Command Set specified in the CSI field (refer to Figure 274). If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller shall abort the command with a status code of Invalid Namespace or Format.~~

If the controller supports the Namespace Management capability (refer to section 8.11) and the NSID field is set to FFFFFFFFh, then the controller shall return an I/O Command Set specific Identify Namespace data structure for the I/O Command Set specified in the CSI field. If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller may abort the command with a status code of Invalid Namespace or Format.

### **Modify section 5.17.2.8 as shown below:**

#### **5.17.2.8 I/O Command Set Independent Identify Namespace data structure (CNS 08h)**

If the Namespace Identifier (NSID) field specifies an active NSID, then the I/O Command Set Independent Identify Namespace data structure (refer to Figure 280) is returned to the host for that specified namespace. If that specified namespace is an inactive NSID, then the controller returns a zero filled data structure.

~~If the controller supports the Namespace Management capability (refer to section 8.11) and the NSID field is set to FFFFFFFFh, then the controller returns an I/O Command Set Independent Identify Namespace data structure that specifies capabilities that are common for the controller. If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller shall abort the command with a status code of Invalid Namespace or Format.~~

The Reported column in Figure 280 specifies fields in the I/O Command Set Independent Identify Namespace data structure that define namespace capabilities used by a host to format or create a namespace. If the NSID field is set to FFFFFFFFh, then the controller shall return an I/O Command Set Independent Identify Namespace data structure that contains:

- fields in Figure 280 set to the same value for all namespaces that the Reported column indicates “Yes”; and
- fields in Figure 280 cleared to 0h that the Reported column indicates “No”.

If the controller supports the Namespace Management capability (refer to section 8.11) and the NSID field is set to FFFFFFFFh, then the controller shall return an I/O Command Set Independent Identify Namespace data structure. If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller may abort the command with a status code of Invalid Namespace or Format.

**Figure 280: Identify – I/O Command Set Independent Identify Namespace Data Structure**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
00	M	<p><b>Common Namespace Features (NSFEAT):</b> This field defines features of the namespace.</p> <p>Bits 7:5 are reserved.</p> <p>Bit 4 <b>Rotational Media (RMEDIA)</b> if set to '1' indicates that the namespace stores data on rotational media (refer to section 8.20). If cleared to '0', indicates that the namespace does not store data on rotational media.</p> <p>Bit 3 (<b>UIDREUSE</b>) if set to '1' indicates that the value in the NGUID field for this namespace, if non-zero, is never reused by the controller and that the value in the EUI64 field for this namespace, if non-zero, is never reused by the controller. If cleared to '0', then the NGUID value may be reused and the EUI64 value may be reused by the controller for a new namespace created after this namespace is deleted. This bit shall be cleared to '0' if both NGUID and EUI64 fields are cleared to 0h. Refer to section 4.5.1.</p> <p>Bit 2:0 are reserved.</p>	No
01	O	<p><b>Namespace Multi-path I/O and Namespace Sharing Capabilities (NMIC):</b> This field specifies multi-path I/O and namespace sharing capabilities of the namespace.</p> <p>Bits 7:1 are reserved.</p> <p>Bit 0: If set to '1', then the namespace may be attached to two or more controllers in the NVM subsystem concurrently (i.e., may be a shared namespace). If cleared to '0', then the namespace is a private namespace and is able to be attached to only one controller at a time.</p>	Yes



**Figure 280: Identify – I/O Command Set Independent Identify Namespace Data Structure**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>
02	O	<p><b>Reservation Capabilities (RESCAP):</b> This field indicates the reservation capabilities of the namespace. A value of 0h in this field indicates that reservations are not supported by this namespace. Refer to section 8.19 for more details.</p> <p>Bit 7 if set to '1' indicates that Ignore Existing Key is used as defined in revision 1.3 or later of this specification. Bit 7 if cleared to '0' indicates that Ignore Existing Key is used as defined in revision 1.2.1 or earlier of this specification. This bit shall be set to '1' if the controller supports revision 1.3 or later as indicated in the Version register.</p> <p>Bit 6 if set to '1' indicates that the namespace supports the Exclusive Access – All Registrants reservation type. If this bit is cleared to '0', then the namespace does not support the Exclusive Access – All Registrants reservation type.</p> <p>Bit 5 if set to '1' indicates that the namespace supports the Write Exclusive – All Registrants reservation type. If this bit is cleared to '0', then the namespace does not support the Write Exclusive – All Registrants reservation type.</p> <p>Bit 4 if set to '1' indicates that the namespace supports the Exclusive Access – Registrants Only reservation type. If this bit is cleared to '0', then the namespace does not support the Exclusive Access – Registrants Only reservation type.</p> <p>Bit 3 if set to '1' indicates that the namespace supports the Write Exclusive – Registrants Only reservation type. If this bit is cleared to '0', then the namespace does not support the Write Exclusive – Registrants Only reservation type.</p> <p>Bit 2 if set to '1' indicates that the namespace supports the Exclusive Access reservation type. If this bit is cleared to '0', then the namespace does not support the Exclusive Access reservation type.</p> <p>Bit 1 if set to '1' indicates that the namespace supports the Write Exclusive reservation type. If this bit is cleared to '0', then the namespace does not support the Write Exclusive reservation type.</p> <p>Bit 0 if set to '1' indicates that the namespace supports the Persist Through Power Loss capability. If this bit is cleared to '0', then the namespace does not support the Persist Through Power Loss Capability.</p>	No
03	O	<p><b>Format Progress Indicator (FPI):</b> If a format operation is in progress, this field indicates the percentage of the namespace that remains to be formatted.</p> <p>Bit 7 if set to '1' indicates that the namespace supports the Format Progress Indicator defined by bits 6:0 in this field. If this bit is cleared to '0', then the namespace does not support the Format Progress Indicator and bits 6:0 in this field shall be cleared to 0h.</p> <p>Bits 6:0 indicate the percentage of the Format NVM command that remains to be completed (e.g., a value of 25 indicates that 75% of the Format NVM command has been completed and 25% remains to be completed). If bit 7 is set to '1', then a value of 0h indicates that the namespace is formatted with the format specified by the FLBAS and DPS fields in this data structure and there is no Format NVM command in progress.</p>	No
07:04	O	<p><b>ANA Group Identifier (ANAGRPID):</b> For NSID other than FFFFFFFFh, this field indicates the ANA Group Identifier of the ANA group (refer to section 8.1.2) of which the namespace is a member. Each namespace that is attached to a controller that supports Asymmetric Namespace Access Reporting (refer to the CMIC field) shall report a valid ANAGRPID. If the controller does not support Asymmetric Namespace Access Reporting, then this field shall be cleared to 0h.</p> <p>If the value in this field changes and Asymmetric Namespace Access Change Notices are supported and enabled, then the controller shall issue an Asymmetric Namespace Access Change Notice.</p>	No



**Figure 280: Identify – I/O Command Set Independent Identify Namespace Data Structure**

Bytes	O/M <sup>1</sup>	Description	Reported <sup>2</sup>						
08	O	<b>Namespace Attributes (NSATTR):</b> This field specifies attributes of the namespace.  Bits 7:1 are reserved.  Bit 0: If set to '1', then the namespace is currently write protected due to any condition (e.g., namespace write protection set for the namespace, media errors) and all write access to the namespace shall fail. If cleared to '0', then the namespace is not currently write protected.	No						
09		Reserved							
11:10	O	<b>NVM Set Identifier (NVMSETID):</b> For NSID other than FFFFFFFFh, 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.	No						
13:12	O	<b>Endurance Group Identifier (ENDGID):</b> For NSID other than FFFFFFFFh, 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.	No						
14	M	<b>Namespace Status (NSTAT):</b> This field indicates the status of the namespace with the specified NSID. <table><tr><th>Bits</th><th>Description</th></tr><tr><td>7:1</td><td>Reserved</td></tr><tr><td>0</td><td><b>Namespace Ready (NRDY):</b> A value of '1' indicates that the namespace is ready (refer to section 3.5.3). A value of '0' indicates that the namespace is not ready.</td></tr></table>	Bits	Description	7:1	Reserved	0	<b>Namespace Ready (NRDY):</b> A value of '1' indicates that the namespace is ready (refer to section 3.5.3). A value of '0' indicates that the namespace is not ready.	No
Bits	Description								
7:1	Reserved								
0	<b>Namespace Ready (NRDY):</b> A value of '1' indicates that the namespace is ready (refer to section 3.5.3). A value of '0' indicates that the namespace is not ready.								
4095:15		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.

**Modify section 5.17.2.17 as shown below:**

#### 5.17.2.17 Domain List (CNS 18h)

Figure 286 defines a Domain List. The data structure is an ordered list by Domain Identifier, starting with the first Domain Identifier that is greater than or equal to ~~or greater than~~ the Domain Identifier specified by the CNS Specific Identifier field as defined in **Figure CNS\_18H** ~~CDW41.NVMSETID~~ and is accessible by the controller processing the command. The Domain List describes the attributes for each Domain in the list based on the Domain Attributes Entry in Figure 287.

**Figure CNS\_18H: Command Dword 11 – CNS Specific Identifier**

Bits	Description
15:0	<b>Domain Identifier (DOMID):</b> This field specifies the Domain Identifier (refer to section 3.2.4.3) of the first Domain of the ordered list of Domain Attribute Entry data structures to be returned.

...

**Modify section 5.17.2.18 as shown below:**

#### 5.17.2.18 Endurance Group List (19h)

An Endurance Group List (refer to Figure 288) of up to 2,047 Endurance Group Identifiers in increasing order is returned containing an Endurance Group Identifier greater than or equal to the ~~value specified in the~~ Endurance Group Identifier ~~(CDW41.ENDGID)~~ specified by the CNS Specific Identifier field as defined

in Figure CNS\_19H. The list contains Endurance Group Identifiers of Endurance Groups that are accessible by the controller processing the command. If the value specified in the Endurance Group Identifier is greater than ENDGIDMAX, then the controller shall complete the command with a status of Successful Completion and return an Endurance Group List containing no Endurance Group Identifiers.

**Figure CNS\_19H: Command Dword 11 – CNS Specific Identifier**

Bits	Description
15:0	<b>Endurance Group Identifier (ENGID):</b> This field specifies the Endurance Group Identifier (refer to section 3.2.3) of the first Endurance Group of the ordered list of Endurance Group Identifiers to be returned.

...