NVM Express™ Technical Proposal

Technical Proposal ID	TP 4110
Revision Date	2022-01-11
Builds on Specification(s)	NVM Express Base Specification 2.0a
References	None

Technical Proposal Author(s)

Name	Company
Curtis Ballard, Matt Goepfert	HPE

Technical Proposal Overview

The HOSTID value selected by a host has different rules for when it is set and if/when it can be changed for fabrics and for PCIe attached devices. The fabrics rules are too restrictive for some common scenarios that want to use HOSTIDs. Software using HOSTIDs has to know whether the attached storage device is a fabrics device or a PCIe device, complicating software development.

This TP proposes modifying the HOSTID rules to create behaviors that are as consistent as possible between fabrics and PCIe.

Text that is existing text which is proposed to be moved is in purple font.

Decisions made in Technical WG discussion of 'Decision Point' slides at the 12 August, 2021 Technical WG (also in meeting notes)

- Allow fabrics connections with the Host Identifier value set to 0h
- Allow Host Identifier to change only if set to 0h or if changing it to 0h
- Define method to allow PCIe devices to prohibit reservations when Host Identifier is set to 0h and prohibit fabrics supporting reservations when Host Identifier is set to 0h
- Additional decision not in slides – look into allowing streams that use Host Identifiers being prohibited by PCIe and Fabrics devices if the Host Identifier value is set to 0h

Decision made in Technical WG discussion at the 23 September, 2021 Technical WG (also in meeting notes)

- Host Identifier shall not be a saveable feature

Revision Date	Change Description
2021.04.26	Initial draft
2021.05.04	Minor updates: • Added discussion topic for whether only change Host Identifier when 0 is too large of a change for PCIe • Added discussion item for allow fabrics devices to fail reservation commands when Host Identifier is set to 0h.
2021.06.14	• Removed proposed PCIe restriction for only changing HOSTID from 0h as valid use case exists for non-zero HOSTID changes and current spec allows that
2021.07.13	Included IO Command Specific Error codes table Move Reserve commands may be aborted if host identifier not set text to align with Reservation Conflict text and reworked to match error code
2021.07.21	Incorporated comments from Mike Allison, Samsung Clarified HOSTID value cleared to 0h handling on connect command to only apply to I/O queues
2021.07.22	Minor live edits during FMDS call
2021.07.23	Added a reference and a comment during 2021.7.22 Technical WG call
2021.08.17	Aligned text with decisions from August 12 Technical WG – see details in "Overview" above Added 'host identifier required to be set' indicators for reservations in Identify Controller and for Streams in the Steams Capabilities
2021.08.26	Changed definition of new reservation and host interaction requirements bit to describe behavior that is new for both fabrics and PCIe Made all references to host identifier "value of 0h" consistent Revised host identifier feature describing use of reservations with Host Identifier value of 0h to be PCIe only Changed the allowed Set Features to allow changing 0 to non-zero and saved non-zero back to zero but no other changes. Modified fabrics connect command to add "current" to the Host Identifier value that is set in a connect.
2021.09.09	Editorial corrections from Mike Allison Added Discover Log Page entry bit for fabrics connect with HOSTID=0h Added cleared to '0' cases to reservations and streams require non-zero HOSTID requirements Proposed alternate reservations require non-zero HOSTID bit wording
2021.09.13	New text in Reservations sub-clause describing that a host with a Host Identifier value of 0h may encounter Reservation Conflict errors even when that host is prohibited from using reservations Reworked Host Identifier and Reservations Interactions reporting to add a "Not Reported" case with optional explicit model reporting capability for both Fabrics and PCIe devices
2021.09.15	Revised reservations require non-zero host identifier indicator based on conclusion of reflector discussion
2021.09.17	Specified that change of Host Identifier from/to value of 0h is a No-op

2021.09.22	Reworded Host Identifier processing in Set Features based on FMDS input
	Clarified text for commands send from host with Host Identifier of 0h conflicting with reservations will fail with Reservation Conflict.
2021.09.27	Changed from Host Identifier should not be saveable to shall not be saveable as instructed in technical WG meeting
	New text in PCIe following default value specifications indicating that any old saved value is reset to default
	Repeated statement about resources staying with a HOSTID if the HOSTID is changed
2021.09.28	Editorial improvements from Fred Knight
	Lots of unchanged text no longer needed for context was removed
2021.10.11	Editorial changes requested in 10 October 2021 Technical taskgroup
	Changes from 9/27 and 9/28 that were reviewed have been accepted
2021.10.13	Changes from 10/12 FMDS meeting, • editorial refinements to statements about settings established before Host Identifier is set remain with the 0h Host Identifier • more explicit Host Identifier saved based on earlier specification shall be reset to default
2021.10.14	Technical WG 10/14 minor editorial changes
	Cleaned up resolved comments for clean phase 2 exit copy
2021.11.05	30 day member review version, no text changes
2021.12.08	Changes requested from 30 day member review • Highlighted references likely to change in incorporation • Capitalization corrections • Moved NVM Subsystem Stream Capability bit descriptions into a table and made a few editorial corrections of 'bit x' to 'this bit' • Corrected several table references
2021.12.09	Changed text referencing the "Identify Controller, I/O Command Set Independent Data Structure" to match the rest of the specification text, "Identify Controller data structure"
	Removed change tracking
2022.01.10	Integrated
2022.01.11	Updated TBD value for Host Identifier Not Initialize status code.

Description for Changes Document for:

TP 4110 Align PCIe and Fabrics HOSTID management

New Features/Feature Enhancements/Required Changes:

- Align PCIe and Fabrics HOSTID management - mandatory

  - Made changes to both PCIe and Fabrics rules for setting and changing the Host Identifier so that both support the same model and same commands.

  - For both PCIe and Fabrics:
    - the Host Identifier value is allowed to be 0h when a controller is enabled and then set to a non-zero value with a Set Features command (new for Fabrics, existing behavior for PCIe)
    - once set to a non-zero value the Host Identifier value is not able to be changed with a Set Features command (new for PCIe, existing behavior for Fabrics)
    - the Host Identifier should be set to a non-zero value before using streams or reservations

  - New optional behavior:
    - the NVM subsystem may require the Host Identifier be set to a non-zero value before using streams or reservations
    - New capability to report whether non-Zero Host Identifier value is required for reservations
    - New capability to report whether non-zero Host Identifier value is required for Streams

  - **New requirement / incompatible change**

  - PCIe
    - Host Identifier defaults to a value of 0h (this behavior was expected but unstated)
    - Host Identifier is not allowed to be changed if it is set to a non-zero value

  - Fabrics
    - Host Identifier may be set to a value of 0h or set to a non-zero value in a Connect command to create an Admin Queue
    - Host Identifier may be set to a value of 0h or set to the Host Identifier value associated with the Admin Queue in a Connect command to create an I/O Queue
    - If the Host Identifier was set to a value of 0h in a connect command, then it may be changed to a non-zero value
    - previous specification versions required a non-zero Host Identifier established at connection and never changed

  - References
    - Technical Proposal TP4110
    - NVMe Base Specification 2.0a Section 3
    - NVMe Base Specification 2.0a Section 5
    - NVMe Base Specification 2.0a Section 6
    - NVMe Base Specification 2.0a Section 8

Markup Conventions:

Black: Unchanged (however, hot links are removed)

~~Red Strikethrough~~:   Deleted

Blue: New

<mark>Blue Highlighted</mark>:   TBD values, anchors, and links to be inserted in new text.

<Green Bracketed>:   Notes to editor

Description of Specification Changes

3   NVM Express Architecture

3.1   NVM Controller Architecture

A controller is the interface between a host and an NVM subsystem.

3.1.1   Controller Model

. . .

In a dynamic controller model, the controller is allocated by the NVM subsystem on demand with no state (e.g., Feature settings) preserved from prior associations. In a static controller model, the host may request a particular controller based on the Controller ID where state (e.g., Feature settings) is preserved from prior associations.

~~While an association exists between a host and a controller, only that host may establish connections with I/O Queues of that controller by presenting the same Host NQN,~~ <mark>~~Host Identifier~~</mark>~~, NVM Subsystem NQN and Controller ID in subsequent Connect command(s) using the same NVM subsystem port, NVMe Transport type, and NVMe Transport address.~~

While an association exists between a host and a controller, only that host may establish connections with I/O Queues of that controller. To establish a new connection with I/O Queues of that controller, the host sends subsequent Connect commands using the same NVM subsystem port, NVMe Transport type, and NVMe Transport address and specifies the:

- same Host NQN;
- same NVM Subsystem NQN;
- same Controller ID; and
- either the:
    - same Host Identifier; or
    - a Host Identifier value of 0h, if supported (refer to section <mark>5.16.1.23</mark>).

An association between a host and controller is terminated if:

- the controller is shutdown as described in section 3.6.2;
- a Controller Level Reset occurs;
- the NVMe Transport connection is lost between the host and controller for the Admin Queue or any I/O Queue; or
- an NVMe Transport connection is lost between the host and controller for any I/O Queue and the host or controller does not support individual I/O Queue deletion (refer to section 3.3.2.4).

. . .

3.3.2.2   Queue Creation

. . .

When a Connect command successfully completes, the corresponding Submission and Completion Queues are created. If NVMe in-band authentication is required as indicated in the Connect response, then NVMe in-band authentication shall be performed before the queues may be used to perform other Fabrics commands, Admin commands, or I/O commands.

<New paragraph break>

Once a Connect command for an Admin Queue has completed successfully (and NVMe in-band authentication, if required, has succeeded), only Fabrics commands may be submitted until the controller is ready (CSTS.RDY = 1). Both Fabrics commands and Admin commands may be submitted to the Admin Queue while the controller is ready. A Connect command for an I/O Queue may be submitted after the controller is ready. Once a Connect command for an I/O Queue has completed successfully (and NVMe in-band authentication, if required, has succeeded), I/O commands may be submitted to the queue.

The Connect response contains the controller ID allocated to the host.

<New paragraph break>

~~The Connect response contains the controller ID allocated to the host. All subsequent Connect commands that create an I/O Queue with that controller shall be from the same host, utilize the same NVMe Transport, and have the same Host Identifier, Host NQN, and NVM Subsystem NQN; if any of these conditions are not met, then the Connect command fails.~~

<Rewrite of above paragraph. The only intended change is to allow an I/O Queue connection with either a Host Identifier of 0h (which will be ignored) or with a Host Identifier value that matches the Admin Queue Host Identifier.>

After an Admin Queue is created on a controller, all subsequent Connect commands sent from the same host to that controller, to create an I/O Queue, are required to:

- utilize the same NVMe Transport;
- have the same Host NQN;
- have the same NVM Subsystem NQN; and
- either have the:
  - same Host Identifier value; or
  - a Host Identifier value of 0h, if supported (refer to section 5.16.1.23).

. . .

3.3.3.2.1.1   Generic Command Status Definition

Completion queue entries with a Status Code Type (SCT) of Generic Command Status indicate a status value associated with the command that is generic across many different types of commands.

Figure 94: Status Code – Generic Command Status Values

Value	Description	I/O Command Set Specific	I/O Command Set(s)[1]
	Existing Status Codes		
27h	**Host Identifier Not Initialized**	No	
	Existing Status Codes		
Key: NVM – NVM Command Set ZNS – Zoned Namespace Command Set KV – Key Value Command Set NOTES: This column is blank unless the value is I/O Command Set specific.			

5   Admin Command Set

. . .

5.16.1.23 Discovery Log Page (Log Identifier 70h)

The Discovery Log Page shall only be supported by Discovery controllers. The Discovery Log Page shall not be supported by controllers that expose namespaces for NVMe over PCIe or NVMe over Fabrics. The Discovery Log Page provides an inventory of NVM subsystems with which a host may attempt to form an association. The Discovery Log Page may be specific to the host requesting the log. The Discovery Log Page is persistent across power cycles.

. . .

Figure 1: Get Log Page – Discovery Log Page Entry

Bytes	Description
. . .	
03	**Transport Requirements (TREQ):** Indicates requirements for the NVMe Transport. Bits 7:~~3~~4 are reserved. Bit 3 if set to '1' indicates that the controller supports a Host Identifier value of 0h in a Connect command. If cleared to '0' indicates that the controller does not support a Host Identifier value of 0h in a Connect command. Bit 2 if set to '1' indicates that the controller is capable of disabling SQ flow control. A controller that is capable of disabling SQ flow control may accept or reject a host request to disable SQ flow control. If cleared to '0', then the controller requires use of SQ flow control. Bits 1:0 indicate whether connections shall be made over a fabric secure channel. <table><tr><th>Value</th><th>Definition</th></tr><tr><td>00b</td><td>Not specified</td></tr><tr><td>01b</td><td>Required</td></tr><tr><td>10b</td><td>Not required</td></tr><tr><td>11b</td><td>Reserved</td></tr></table>
. . .	

5.17   Identify command

. . .

5.17.2.1 Identify Controller data structure (CNS 01h)

The Identify Controller data structure (refer to Figure 275) is returned to the host for the controller processing the command.

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

Bytes	I/O[1]	Admin[1]	Disc[1]	Description
				Controller Capabilities and Features
. . .				
99:96	M	M	R	**Controller Attributes (CTRATT):** This field indicates attributes of the controller.

Within the Description for bytes 99:96:

Bits	Description
31:~~16~~19	Reserved
18	**Reservations and Host Identifier Interaction (RHII):** This bit indicates the reservations and Host Identifier interaction support for the controller.

Value	Definition
0b	Not Reported
1b	The Host Identifier is required to be set to a non-zero value for a host to use reservations (refer to section 8.19).

. . .

. . .

. . .

5.27 Set Features command

. . .

5.27.1.25 Host Identifier (Feature Identifier 81h), (Optional[1])

This feature allows the host to register a Host Identifier with the controller. The Host Identifier is used by the controller to determine whether other controllers in the NVM subsystem are associated with the same host. The Host Identifier may be used to designate host elements that access an NVM subsystem independently of each other or for reservations.

The Host Identifier is contained in the data structure indicated in Figure 2. The attributes are specified in Command Dword 11. If a Get Features command is issued for this Feature, the data structure specified in Figure 2 is returned in the data buffer for that command.

<Note: The following is mostly moved text pulled from the PCIe section below 5.27.1.25.1. It has been reordered for better flow.>

A Host Identifier value of 0h indicates that the host associated with the controller is not associated with any other controller in the NVM subsystem. For example, two controllers in an NVM subsystem that both have

---

[1] Mandatory if reservations are supported by the controller as indicated in the ONCS field in the Identify Controller data structure.

a Host Identifier value of 0h indicates that the controllers are associated with different hosts. NVMe over PCIe implementations may support using a Host Identifier value of 0h for the reservations feature (refer to section 8.19). However, reservations and registrations associated with a Host Identifier value of 0h do not persist across a Controller Level Reset since a host that uses a Host Identifier value of 0h is treated as a different host after a Controller Level Reset.

A Set Features command should be used to change a Host Identifier value of 0h to a non-zero value before using streams (refer to section 8.7.3) or using reservations (refer to section 8.19). Information (i.e., streams or reservations) associated with a Host Identifier value of 0h retain the association to that Host Identifier if the Host Identifier value is changed and are not associated with the host that has the non-zero Host Identifier.

The NVM subsystem indicates if reservations are supported with a Host Identifier value of 0h with the RHII bit in the Controller Attributes field of the Identify Controller data structure (refer to figure 275). The NVM subsystem indicates if streams are supported with a Host Identifier value of 0h with the SRNZID bit in the NVM Subsystem Stream Capability field of the streams directive return parameters (refer to Figure 425).

The Host Identifier feature shall not be a saveable feature.

The requirements and use of the Host Identifier feature is dependent on whether the NVMe over PCIe implementation or NVMe over Fabrics implementation are supported. Refer to section 5.27.1.25.1 and section 5.27.1.25.2.

. . .

Figure 2: Host Identifier – Data Structure Entry

Bytes	Description
15:00	**Host Identifier (HOSTID):** This field specifies a 64-bit or 128-bit identifier that uniquely identifies the host associated with the controller within the NVM subsystem. The host provides an 8 byte or 16 byte data structure depending on the value specified in the Enable Extended Host Identifier bit. The value of the Host Identifier used by a host, the method used to select this value, and the method used to ensure uniqueness are outside the scope of this specification. Controllers in an NVM subsystem that have the same Host Identifier are assumed to be associated with the same host and have the same reservation and registration rights. If the current Host Identifier value is cleared to 0h, then the current Host Identifier shall be set to the value specified in this field. If the current Host Identifier value is set to a non-zero value, then the controller shall abort the command with a status code of Command Sequence Error. A Host Identifier value of 0h indicates that the host is not associated with any other controller in the NVM subsystem.

5.27.1.25.1 PCIe Transport Implementations

The Host Identifier is an optional feature when implemented on a controller using a PCIe transport. The controller may support a 64-bit Host Identifier and/or an extended 128-bit Host Identifier. It is recommended that implementations support the extended 128-bit Host Identifier as indicated in the Controller Attributes field in the Identify Controller data structure. The Host Identifier may be modified at any time using a Set Features command causing the controller to be logically remapped from the original host associated with the old Host Identifier to a new host associated with the new Host Identifier.

A Host Identifier value of 0h is a valid value that indicates that the host associated with the controller is not associated with any other controller in the NVM subsystem. Specifically, two controllers in an NVM subsystem that both have a Host Identifier of 0h indicates that the controllers are associated with different hosts. Using a Host Identifier value of 0h is a valid configuration for the reservations feature. However, reservations and registrations associated with a Host Identifier of 0h do not persist across a Controller Level Reset since a host that uses a Host Identifier of 0h is treated as a different host after a Controller Level Reset.

<Note: The text above was moved above in purple to be pulled outside of the PCIe specific section.>

The Host Identifier for PCIe transport implementations shall have a default value of 0h and shall not have a saved value. NVM Express Base Specification revision 2.0 and earlier allowed saving a Host Identifier value, if a non-zero Host Identifier value had been saved then that saved value shall be reset to the default value.

5.27.1.25.2 NVMe over Fabrics Implementations

The Host Identifier is a mandatory feature in NVMe over Fabrics implementations. The Host Identifier shall be an extended 128-bit Host Identifier.

~~The Host Identifier shall be set to a non-zero value in the Fabrics Connect command. The Host Identifier shall not be modified. A Set Features command specifying the Host Identifier Feature shall be aborted with a status code of Command Sequence Error. A Get Features command specifying the Host Identifier Feature shall return the value set in the Fabrics Connect command.~~

. . .

6   Fabrics Command Set

. . .

6.3   Connect Command and Response

. . .

<Note: For fabrics an association is created with the Admin queue in the first connect command and is shared by all I/O Queue connections after that. A Host Identifier is shared across the Admin Queue and all associated I/O Queues.>

The host shall establish an association with a controller and enable the controller before establishing a connection with an I/O Queue of the controller. If the host sends a Connect command specifying a Queue ID for an Admin Queue or I/O Queue which has already been created, then a status code of Command Sequence Error is returned.

A status code of Connect Invalid Parameters is returned for a Connect command if:

- the host sends a Connect command to create an I/O Queue while the controller is disabled;
- the ~~Host Identifier,~~ Host NQN, NVM Subsystem NQN, and the Controller ID values specified for an I/O Queue are not the same as the values specified for the associated Admin Queue in which the association between the host and controller was established;
- the Host Identifier for an I/O Queue is not set to a value of 0h and is not set to the same value as the value specified for the associated Admin Queue in which the association between the host and controller was established;
- the Host NQN or NVM Subsystem NQN values do not match the values that the NVM subsystem is configured to support; or
- ~~•~~ there is a syntax error in the Host NQN or NVM Subsystem NQN value (refer to section 4.4).~~; or~~
- ~~the Host Identifier is cleared to 0h.~~

. . .

Technical input submitted to NVM Express™ is subject to the terms of the NVM Express™ Participant's agreement. Copyright © 2014 to 2021 NVMe™ Corporation.

Figure 3: Connect Command – Data

Bytes	Description
15:00	**Host Identifier (HOSTID):** This field has the same definition as the Host Identifier defined in section 5.27.1.25. ~~The controller shall set the Host Identifier Feature to this value.~~ For a Connect command to create an Admin Queue (i.e., the QID field is cleared to 0h) that completes successfully the controller shall set the current Host Identifier (refer to section 5.27.1.25.2) to this value. For a Connect command to create an I/O queue (i.e., the QID field is set to a non-zero value): • if this field is cleared to 0h the controller shall ignore this field; and • if this field is set to a non-zero value and the value in this field does not match the value in the Host Identifier feature, then the the controller shall abort the command with a status code of Connect Invalid Parameters.
17:16	**Controller ID (CNTLID):** Specifies the controller ID requested. This field corresponds to the Controller ID (CNTLID) value returned in the Identify Controller data structure for a particular controller. If the NVM subsystem uses the dynamic controller model, then the value shall be FFFFh for the Admin Queue and any available controller may be returned. If the NVM subsystem uses the static controller model and the value is FFFEh for the Admin Queue, then any available controller may be returned.
255:18	Reserved
511:256	**NVM Subsystem NVMe Qualified Name (SUBNQN):** NVMe Qualified Name (NQN) that uniquely identifies the NVM subsystem. Refer to section 4.4.
767:512	**Host NVMe Qualified Name (HOSTNQN):** NVMe Qualified Name (NQN) that uniquely identifies the host. Refer to section 4.4.
1023:768	Reserved

. . .

8.6   Device Self-test Operation

. . .

8.7   Directives

8.7.2 Identify (Directive Type 00h)

. . .

8.7.2.1   Directive Receive

This section defines operations used with the Directive Receive command for the Identify Directive.

8.7.2.2.1 Return Parameters (Directive Operation 01h)

. . .

8.7.2.2   Directive Send

This section defines operations used with the Directive Send command for the Identify Directive.

8.7.2.2.1   Enable Directive (Directive Operation 01h)

. . .

For all controllers in an NVM subsystem that have the same non-zero Host Identifier, if a host changes the enable state of any Directive for a shared namespace attached to a controller by a means other than a Controller Level Reset, then that change shall be made to the enable state of that Directive for that namespace attached to any other controller associated with that Host Identifier.

If the Host Identifier value is 0h and the Host Identifier is required to be set to a non-zero value before a host enables a Directive (i.e., the Directive Type is set to Streams and the SRNZID bit in the NVM subsystem Stream Capability field (refer to Figure 425) is set to '1') then the Directive Send command shall be aborted with a status code of Host Identifier Not Initialized.

If an NVM subsystem:

1. supports directives with a Host Identifier value of 0h;
2. streams resources are allocated to a host with a Host Identifier value of 0h; and
3. the Host Identifier is changed to a non-zero value,

then those streams resources remain associated with the Host with a Host Identifier value of 0h and are not associated with the host with the non-zero Host Identifier.

Figure 4: Enable Directive – Command Dword 12

Bits	Description
31:16	Reserved
15:08	**Directive Type (DTYPE):** This field specifies the Directive Type to enable or disable. If this field specifies the Identify Directive (i.e., 00h), then a status code of Invalid Field in Command shall be returned.
07:01	Reserved
00	**Enable Directive (ENDIR):** If set to '1' and the Directive Type is supported, then the Directive is enabled. If cleared to '0', then the Directive is disabled. If this bit is set to '1' for a Directive that is not supported, then a status code of Invalid Field in Command shall be returned.

8.7.3   Streams (Directive Type 01h, Optional)

. . .

8.7.3.1   Directive Receive

This section defines operations used with the Directive Receive command for the Streams Directive.

8.7.3.1.1   Return Parameters (Directive Operation 01h)

The Return Parameter operation returns a data structure that specifies the features and capabilities supported by the Streams Directive, including namespace specific values. The DSPEC field in command Dword 11 is not used for this operation. The data structure returned is defined in Figure 5. If an NSID value of FFFFFFFFh is specified, then the controller returns the NVM subsystem specific values, may return any namespace specific values that are the same for all namespaces (e.g., SWS), and clears all other namespace specific fields to 0h.

Figure 5: Streams Directive – Return Parameters Data Structure

Bytes	Description
	NVM Subsystem Specific Fields
	. . .
06	**NVM Subsystem Stream Capability (NSSC):** This field indicates the stream capabilities of the NVM subsystem.

Within byte 06:

~~Bits 7:1 are reserved.~~

~~Bit 0 indicates whether stream identifiers may be shared by multiple Host Identifiers, or if a stream identifier is associated with a single Host Identifier. If bit 0 is cleared to '0', then the stream identifier is associated with a single non-zero Host Identifier. If bit 0 is set to '1', then the stream identifier may be associated with multiple non-zero Host Identifiers.~~

| Bits	Description
7:2	Reserved
1	**Streams Require Non-Zero Host Identifier (SRNZID):** This bit indicates whether the Host Identifier is required to be set to a non-zero value before the Streams Directive is able to be enabled. If this bit is cleared to '0', then the Host Identifier is not required to be set to a non-zero value before the Streams Directive is able to be enabled. If this bit is set to '1', then the Host Identifier is required to be set to a non-zero value before the Streams Directive is able to be enabled.
0	This bit indicates whether stream identifiers may be shared by multiple Host Identifiers, or if a stream identifier is associated with a single Host Identifier. If this bit is cleared to '0', then the stream identifier is associated with a single non-zero Host Identifier. If this bit is set to '1', then the stream identifier may be associated with multiple non-zero Host Identifiers.

. . .

. . .

8.19 Reservations

NVM Express reservations provide capabilities that may be utilized by two or more hosts to coordinate access to a shared namespace. The protocol and manner in which these capabilities are used is outside the scope of this specification. Incorrect application of these capabilities may corrupt data and/or otherwise impair system operation.

Reservation operation after a division event (refer to section 3.2.4.1) is described in section 3.2.4.2.

A reservation on a namespace restricts hosts access to that namespace. If a host submits a command to a namespace in the presence of a reservation and lacks sufficient rights, then the command is aborted by the controller with a status code of Reservation Conflict. If a host submits a command with the NSID set to FFFFFFFFh in the presence of a reservation on any of the namespaces impacted by that command and that host lacks sufficient rights on all the impacted namespaces, then the command is aborted by the controller with a status code of Reservation Conflict. Capabilities are provided that allow recovery from a reservation on a namespace held by a failing or uncooperative host.

Figure 6: Example Multi-Host System
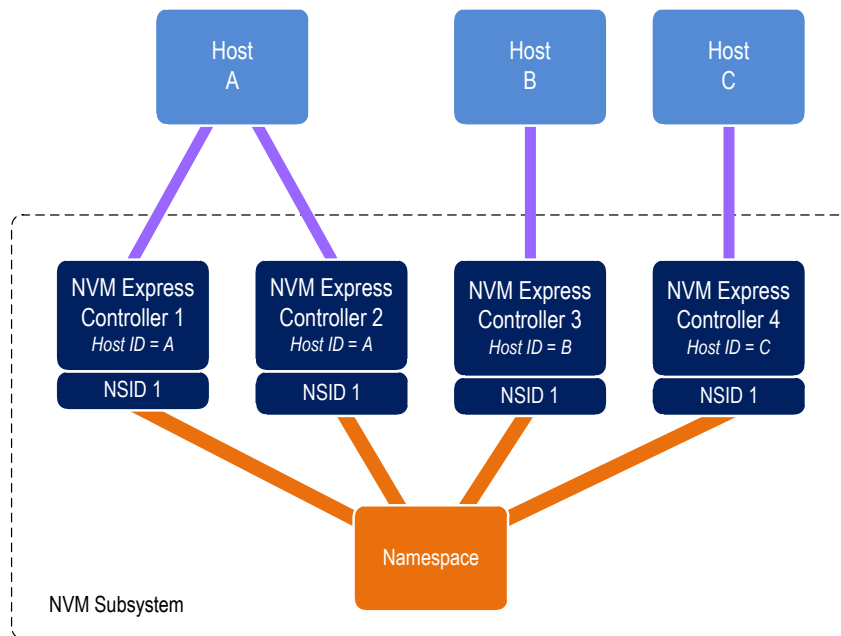
A reservation requires an association between a host and a namespace. As shown in Figure 6, each controller in a multi-path I/O and namespace sharing environment is associated with exactly one host. While it is possible to construct systems where two or more hosts share a single controller, such usage is outside the scope of this specification.

A host may be associated with multiple controllers. In Figure 6 host A is associated with two controllers while hosts B and C are each associated with a single controller. A host registers should register a non-zero Host Identifier (refer to section 5.27.1.25) with each controller with which that host is associated using a Set Features command (refer to section 0) prior to performing any operations associated with reservations. The Host Identifier allows the NVM subsystem to identify controllers associated with the same host and preserve reservation properties across these controllers (i.e., a host issued command has the same reservation rights no matter which controller associated with the host processes the command).

An NVM subsystem may require that the host register a non-zero Host Identifier for a host to use reservations (i.e., the RHII bit is set to '1' in the CTRATT field of the Identify Controller data structure (refer to Figure 275)). If the controller does not support reservations with a Host Identifier value of 0h and a reservation command is received from a host with a Host Identifier value of 0h, then the controller shall abort that reservation command with a status of Host Identifier Not Initialized.

If an NVM subsystem:

1. supports reservations with a Host Identifier value of 0h;
2. registrations or reservations are established by a host with a Host Identifier value of 0h; and
3. the Host Identifier is changed to a non-zero value,

then those registrations or reservations remain associated with the Host with a Host Identifier value of 0h and are not associated with the host with the non-zero Host Identifier.

If the controller does not support reservations with a Host Identifier value of 0h, reservations may have been established by hosts with non-zero Host Identifiers connected to other controllers, and commands from a host with a Host Identifier value of 0h that conflict with a reservation (refer to figure 472 and figure 473) are aborted by the controller with a status code of Reservation Conflict.

Support for reservations by a namespace or controller is optional. A namespace indicates support for reservations by reporting a non-zero value in the Reservation Capabilities (RESCAP) field in the Identify Namespace data structure. A controller indicates support for reservations through the Optional NVM

Command Support (ONCS) field in the Identify Controller data structure (refer to Figure 275). If a host submits a command associated with reservations (i.e., Reservation Report, Reservation Register, Reservation Acquire, and Reservation Release) to a controller or a namespace that do not both support reservations, then the command is aborted by the controller with a status code of Invalid Command Opcode.

. . .