What's New in NVMe® Technology: Ratified Technical Proposals to Enable the Future of Storage

Sponsored by NVM Express organization, the owner of NVMe specifications
Speaker

Mike Allison
Sr. Director NAND Product Planning - Standards

SAMSUNG
Agenda

Included in the NVM Express® (NVMe®) 2.0 Family of Specifications

• Domains and Partitions
• New Protection Information Formats
• Copy Command

Ratified after the release of NVM Express 2.0 Family of Specifications

• TP4034 Disperse Namespaces
• TP4076 Zone Random Write Area
Domains and Divisions

Expanded the specification to account for large NVM subsystems

- **Domains**
  - Smallest indivisible unit that shares state:
    - Power
    - Non-Volatile Storage Capacity
    - Firmware Version
  - Consists of
    - Zero or more controllers
    - Zero or more Endurance Groups

- NVM subsystems with multiple domains are required to support Asymmetric Namespace Access Reporting

- Defines a Division event for the loss of communication to a Domain
New Protection Information Formats

Original 8 byte PI Format
- 16-bit Guard = 16b CRC
- 16-bit Application Tag
- 32-bit Reference Tag
  - Determine sequence of LBAs

### 16b Guard Protection Information

<table>
<thead>
<tr>
<th>Byte</th>
<th>MSB</th>
<th>Guard</th>
<th>LSB</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>MSB</td>
<td>Application Tag</td>
<td>LSB</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>MSB</td>
<td>Reference Tag</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
New Protection Information Formats

New 8 byte PI Format

- 16-bit Guard = 16b CRC
- 16-bit Application Tag
- 32-bit Storage and Reference Space
  - Storage Tag
    - Bit size defined by Storage Tag Size (STS)
  - Logical Block Reference Tag
    - Bit size is the remaining size

16b Guard Protection Information with the STS set to a non-zero value

<table>
<thead>
<tr>
<th>Byte</th>
<th>Guard</th>
<th>Application Tag</th>
<th>Storage and Reference Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>MSB</td>
<td>MSB</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>LSB</td>
<td>LSB</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>MSB</td>
<td>MSB</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>LSB</td>
<td>LSB</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>MSB</td>
<td>MSB</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>LSB</td>
<td>LSB</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
New Protection Information Formats

New 16 byte PI Format

- 32-bit Guard = 32b CRC
- CRC-32C (same as iSCSI and NVMe-MI™ technology)
- 16-bit Application Tag
- 80-bit Storage and Reference Space
  - 64-bit maximum size for
    - Storage Tag
    - Reference Tag

### 32b Guard Protection Information

<table>
<thead>
<tr>
<th>Byte</th>
<th>MSB</th>
<th>LSB</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>MSB</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>LSB</td>
</tr>
</tbody>
</table>
New Protection Information Formats

New 16 byte PI Format

- 64-bit Guard = 64b CRC
  - Defined by the NVM Express® NVM Command Set Specification
- 16-bit Application Tag
- 48-bit Storage and Reference Space

---

64b Guard Protection Information

<table>
<thead>
<tr>
<th>Byte</th>
<th>MSB</th>
<th>LSB</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>Guard</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>MSB</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>LSB</td>
</tr>
</tbody>
</table>
Copy Command

Add a new Copy command

- Does a scatter gather copy of a set of Logical Block Addresses (LBA) ranges (i.e., Source Ranges) written sequentially within the same namespace.

- Submission Queue Entry specifies:
  - the write information (similar to a Write command) except:
    - Number of Source Range Entries
    - Format of Source Range Entries
    - Protection Checking Enablement

- Command Data transfer contains a list of Source Range Entries.
  - Order specifies gathering order on write.
  - On an aborted command, completion entry specifies the lowest Source Range Entry not successfully written.
Copy Command

Source Range Entry 0
Source Range Entry 1
Source Range Entry 2
Source Range Entry 3
Source Range Entry 4

Namespace

Source Range Entry 0
Source Range Entry 1
Source Range Entry 2
Source Range Entry 3
Source Range Entry 4

Starting Destination LBA
Copy Command

Source Range Entry 0
Source Range Entry 1
Source Range Entry 2
Source Range Entry 3
Source Range Entry 4

Starting Destination LBA

Issuer Range Entry

Host does not know if these Source Range Entries are or are not written

Namespace

Source Range Entry 3
Source Range Entry 0
Source Range Entry 1
Source Range Entry 2
Source Range Entry 3
Source Range Entry 4

Error
TP4034 Disperse Namespaces

The intent of this proposal is to allow a namespace to coexist (i.e., shared) across multiple NVM subsystems such that:

- The namespace identifier is unique to each NVM subsystem
- The Non-Qualified Name (NQN) is unique across each NVM subsystem participating in sharing the dispersed namespace
- NGUID/UUID support required to be the same on each NVM Subsystem sharing this namespace

Why?
- Online Data Migration
TP4034 Disperse Namespaces

The intent of this proposal is to allow a namespace to coexist (i.e., shared) across multiple NVM subsystems such that:

- The namespace identifier is unique to each NVM subsystem
- The Non-Qualified Name (NQN) is unique across each NVM subsystem participating in sharing the dispersed namespace
- NGUID/UUID support required to be the same on each NVM Subsystem sharing this namespace

Why?
- Online Data Migration
- Data Replication
TP4034 Disperse Namespaces

The intent of this proposal is to allow a namespace to coexist (i.e., shared) across multiple NVM subsystems such that:

- The namespace identifier is unique to each NVM subsystem
- The Non-Qualified Name (NQN) is unique across each NVM subsystem participating in sharing the dispersed namespace
- NGUID/UUID support required to be the same on each NVM Subsystem sharing this namespace

Why?
- Online Data Migration
- Data Replication
- Full Redundancy
TP4034 Disperse Namespaces

Dispersed Namespace

- Is a shared namespace
- Method for making a dispersed namespace is outside the scope of NVMe® technology
- Controller reports namespace being a Dispersed namespace
- New Dispersed Namespace Participating NVM Subsystem log page (log identifier 17h)
  - Contains the NQN of each NVM Subsystem sharing the namespace
  - Contains a Generation Counter incremented each time the log page changes
  - No event tied to changes in this log page
- Reservations updated to support Dispersed namespaces
TP4076 Zone Random Write Area

- Creates a non-volatile Random Write area cache at a zones’ Write Pointer
- Hosts uses Write commands to initially write LBAs within the ZRWA
- Allows hosts to sequential flush the cache at the Write Pointer on ZRWA flush granularities (ZRWAFG)
TP4076 Zone Random Write Area

- Creates a non-volatile Random Write area cache at a zones’ Write Pointer
- Hosts uses Write commands to initially write LBAs within the ZRWA
- Allows hosts to sequential flush the cache at the Write Pointer on ZRWA flush granularities (ZRWA FG)
- Hosts are allowed to write LBAs any order and non-written LBAs remain deallocated when flushed
TP4076 Zone Random Write Area

- Creates a non-volatile Random Write area cache at a zones’ Write Pointer
- Hosts uses Write commands to initially write LBAs within the ZRWA
- Allows hosts to sequential flush the cache at the Write Pointer on ZRWA flush granularities (ZRWA FG)
- Hosts are allowed to write LBAs any order and non-written LBAs remain deallocated when flushed
- Multiple sequential ZRWA FG may be flushed to the WP
TP4076 Zone Random Write Area

- Creates a non-volatile Random Write area cache at a zones’ Write Pointer
- Hosts uses Write commands to initially write LBAs within the ZRWA
- Allows hosts to sequential flush the cache at the Write Pointer on ZRWA flush granularities (ZRWA FG)
- Hosts are allowed to write LBAs any order and non-written LBAs remain deallocated when flushed
- Multiple sequential ZRWA FG may be flushed to the WP
- LBA writes beyond the size of the ZRWA cause an Implicit Flush

![Diagram of ZRWA and Implicit Flush Range]
TP4076 Zone Random Write Area

- Creates a non-volatile Random Write area cache at a zones’ Write Pointer
- Hosts uses Write commands to initially write LBAs within the ZRWA
- Allows hosts to sequential flush the cache at the Write Pointer on ZRWA flush granularities (ZRWA FG)
- Hosts are allowed to write LBAs any order and non-written LBAs remain deallocated when flushed
- Multiple sequential ZRWA FG may be flushed to the WP
- LBA writes beyond the size of the ZRWA cause an Implicit Flush
Questions?