Storage Networking Industry Association
Technical White Paper

Hyperscaler Storage

September 12, 2016

Abstract: *Hyperscaler storage customers typically build their own storage systems from commodity components. They have requirements for drives (SSDs and Hard drives) that are in some cases being put into standard interfaces. This paper explores that trend and highlights features of existing and new standards that will meet those requirements.*

USAGE

The SNIA hereby grants permission for individuals to use this document for personal use only, and for corporations and other business entities to use this document for internal use only (including internal copying, distribution, and display) provided that:

1. Any text, diagram, chart, table or definition reproduced shall be reproduced in its entirety with no alteration, and,

2. Any document, printed or electronic, in which material from this document (or any portion hereof) is reproduced shall acknowledge the SNIA copyright on that material, and shall credit the SNIA for granting permission for its reuse.

Other than as explicitly provided above, you may not make any commercial use of this document, sell any or this entire document, or distribute this document to third parties. All rights not explicitly granted are expressly reserved to SNIA.
Permission to use this document for purposes other than those enumerated above may be requested by e-mailing tcmd@snia.org. Please include the identity of the requesting individual and/or company and a brief description of the purpose, nature, and scope of the requested use.

All code fragments, scripts, data tables, and sample code in this SNIA document are made available under the following license:

BSD 3-Clause Software License

Copyright (c) 2016, The Storage Networking Industry Association.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of The Storage Networking Industry Association (SNIA) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

DISCLAIMER

The information contained in this publication is subject to change without notice. The SNIA makes no warranty of any kind with regard to this specification, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The SNIA shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this specification.

Suggestions for revisions should be directed to http://www.snia.org/feedback/.

Foreword

This paper is the work of the SNIA Technical Council. It includes information about features that may not yet be standardized and may change during the process of standardization.

Executive Summary

Large Datacenter customers are known as Hyperscalers. Nearly all of these customers build their own storage systems using Software Defined Storage (SDS) and commodity components, assembled in racks.

Managing this large pool of commodity components has proved challenging and several issues have been identified that could be addressed by new features on these storage components. This paper discusses these requirements and how existing standards and proposed changes to standards might address them.

1 Introduction

Hyperscalers are large enough customers that they can and do request specific features from storage devices via the RFP acquisition process. Drive vendors will add these features in order to sell to these customers, but may differ in how these features are implemented and in how they extend standard interfaces to accommodate them. Software Defined Storage (SDS) products will also benefit from these features as they are added. Many Enterprises are taking advantage of the Hyperscalers techniques by using SDS.

At the 2016 FAST Conference, Eric Brewer (VP of Infrastructure for Google) talked about some of Google's requirements documented in a paper called *Disks for Data Centers*. It is likely that these requirements are not unique to Google as validated by Mike McGrath (Microsoft Azure), and that by addressing some of these requirements, other Hyperscaler customers will benefit.

The Google paper has two sets of requirements that are documented: physical changes (platter size, drive height, etc.) and logical changes to the controller. This paper discusses the later requirements and how they can be addressed by standard interface changes.

2 Hyperscaler Storage Infrastructure

Hyperscale customers typically build their own storage systems from commodity components installed into racks and supplied by vendors termed Original Design Manufacturer (ODM) Direct. Each drive in these systems is assumed to be inherently unreliable so data redundancy across multiple nodes is assumed. This could allow for lower reliability drives if that would save cost. These systems consist of drive trays (Just a Bunch Of Disks – JBOD, and/or Just a Bunch Of Flash – JBOF) and compute servers. On the compute nodes, Software Defined Storage (SDS) is used to create the storage system. Scaling is horizontal by adding additional identical nodes to the existing cluster. Block, File and Object interfaces are available depending on application needs.

This higher level software provides the needed availability through redundant copies of data or through erasure coding across multiple drives. Performance is also achieved by using multiple drives for each

request, serving the data in parallel. Management is performed through the SDS administrative interface and custom datacenter management software that monitors and manages the systems at a datacenter scale. Geographic replication between datacenters is also available for business continuity and disaster resilience.

3 Hyperscaler Requirements

Storage devices in the data center are managed as a collection but are optimized for individual operation. The collective requirements include:

- Higher I/Os per second (IOPs)
- Higher capacity
- Lower tail latency
- Improved Security
- Reduced Total Cost of Ownership

These are compared to the existing drives that they can purchase today. Storage drives were originally designed to be individually resilient (going almost to the extreme to recover any data they are storing) and this is carried forward in the drives of today even though they are now principally components in an overall system,

Overall availability and performance of the collection is done by Software Defined Storage (SDS) by spreading the data across multiple drives both within the data center and geographically across datacenters. However, the drives are optimized to reliably store and retrieve data at the expense of consistent performance in some cases.

Some of the requested features include:

- Control timing over background tasks (when tail latency is an issue)
- Leverage disk's knowledge of details (which blocks are responding slow)
- Prioritize requests, but still allow disk firmware to do the scheduling
- Provide an abstraction layer for these features available from multiple vendors
- Have a Per I/O retry policy (Try really hard, or fail fast)

4 Tail Latency

One of the issues in large deployments of drives is known as Tail Latency. Another FAST 2016 paper *The Tail at Store: A Revelation from Millions of Hours of Disk and SSD Deployments* analyzed:

- storage performance in over 450,000 disks and 4,000 SSDs over 87 days.

- an overall total of 857 million (disk) and 7 million (SSD) drive hours.
- a small proportion of very slow responses from the media:
    - 0.2% of the time, a response to a particular I/O is more that 2x slower than the other I/Os in the same RAID group (e.g. one of the drives was slow in responding)
    - And 0.6% for SSD.
- As a consequence, disk and SSD-based RAID stripes experience at least one slow drive (i.e., storage tail) 1.5% and 2.2% of the time respectively.

Tail Latency is due to SDS spreading the data across multiple drives and issuing requests in parallel. The slowest drive will thus delay the overall response. In most cases, the data from that slow drive is already available to form the response either from another drive with the same data, or via erasure coded redundancy. Striping over a larger number of drives would likely increase the probability of seeing this tail latency.

5 Tail Latency Remediation

A potential solution to tail latency is to provide a retry policy hint during an I/O operation to indicate that the operation is part of a stripe with redundant data from multiple drives and should favor response time over data recovery. The drive would then interpret this policy to minimize retries and return an error rather then spend inordinate time trying to recover the data. The SDS can then supply the missing data from the redundant copies and form the response in a timely manner, eliminating the tail latency.

The custom datacenter management software that is monitoring the drives and their response times may mark drives that are slow in returning the data as offline. The SDS then finds other drives to use in order to restore the redundant copies of data. While this can help in eliminating the tail latency, it may be just one part of the drive that has the slow response time. Taking the drive offline then requires a field replacement, but not immediately. The drive might be replaced during a periodic maintenance pass through the datacenter.

If one part of the drive has slow response times and the drive could detect those slow areas of its media, then it may be able to remap the LBAs to spare media. This action would prevent the datacenter management software from taking the drive offline. Because this would consume the spare media faster than the remapping that is typically only done only on media failure, a means of increasing the amount of spare capacity is needed. A drive vendor might also initially ship drives with additional spare capacity for this purpose.

6 Initiatives that address Hyperscaler Requirements

A number of existing standards and proposed new features are intended to address the requirements of Hyperscale customers. We describe the initiatives and show how they address the requirements below.

Drive Truncation and Storage Element Depopulation

There are two related efforts currently being worked in both T10 (SAS) and T13 (SATA) standards organizations. Repurposing Depopulation is a proposal to enable identification of slow media and indicate to the host that such locations have been identified. The host can then discover these identified regions and request that the drive remove these regions from service. A REMOVE ELEMENT AND TRUNCATE command will then truncate the LBA range, removing the slow media and essentially producing a drive with a reduced capacity. No guarantees are provided regarding data preservation. The SDS would then treat this as a new drive and use it for data as if it was just added.

Data Preserving Depopulation also enables identification of slow media, but enables the host to preserve the data that is in the LBA locations that will be truncated, moving it to an alternate location. Valid data that is located in the non-truncated LBA space is preserved. This is useful for storage systems with a fixed size RAID or other striping where the drive will need to be rebuilt as part of that stripe, speeding the rebuild process.

Streams

Streams is a concept that associates multiple blocks with an upper level construct such as a file or object. It is likely that all these blocks will be deleted as a group and therefore knowledge of this grouping can reduce garbage collection of unused write locations. SSDs can consolidate the LBAs associated with the stream into one or more write blocks. When used with the TRIM command or UNMAP command, entire write blocks can then be erased. This improves performance and minimizes write amplification due to garbage collection. This also improves the life of the device. For the NVMe interface this is part of the Directives proposal targeted at version 1.3. For SAS, this is supported by the WRITE STREAM command among others. In SBC-4 this is the Stream Control sub-clause 4.24. For SATA, proposals are being processed.

Advanced Background Operations

Drives do perform operations asynchronously from host requests. These operations may include:

- Garbage Collection
- Scrubbing
- Remapping
- Cache flushes
- Continuous Self Test

These operations may delay an I/O operation from a host and lead to tail latency. By giving the host some ability to affect the scheduling of these operations, these background operations may be scheduled at a time that reduces the impact on I/O operations, reducing the impact of this effect on tail latency.

Advanced Background Operation proposals are targeted for the NVMe interfaces. For NVMe this work is a proposal targeted at version 1.3. For SAS, this work has added a BACKGROUND CONTROL command as

defined in SBC-4 Background Operation Control. For SATA, this work is defined in ACS-4 Advanced Background Operations feature set.

Open Channel SSDs

An Open Channel SSD is a solid state drive which does not have a full firmware Flash Translation Layer (FTL), but instead presents the full physical address space of the storage media to a host. The FTL is then run on the host and may communicate through an interface such as NVMe. An open source implementation of the required FTL is typically used to achieve wear leveling, garbage collection and sparing. This gives the Hyperscale customer control over the access to the physical media allowing it to control the FTL processes. The SDS is then able to manage tail latency. The LightNVM project has software to enable these types of devices. Using this software, the bad block management, metadata persistence and extensions to support atomic I/O are still required to be handled by firmware on the device. Other approaches may divide the operations between host and device differently.

Rapid RAID Rebuild

For SAS and SATA, when a drive is put into rebuild assist mode (it can be enabled/disabled as needed), the drive does a couple of things differently. It doesn't try so hard to get the data – because in this mode, it is assumed that the host has another copy of the data available somewhere else. This mode can be disabled on a per I/O basis (so that complex error recovery may be done for this I/O).

When an error is detected, the drive tells the host not only about the error on the requested block, but it tells the host about the other errors in a related contiguous chunk (the LBA of the next "good" block). This enables the host to not bother trying to access any of those "already known to be bad" blocks, and go ahead and get the data from the other source for those blocks.

I/O Hints

SAS and SATA support some hints using Logical Block Markup (LBM). Examples of hints that can be used include:

- Subsequent I/O Hint – prioritizes this I/O with regard to other I/Os
- Read Sequentiality – probability of subsequent reads to this LBA range
    - Enables intelligent cache pre-fetch
    - This allows the data to be removed from the cache after it is read
- Write Sequentiality – probability of subsequent writes to this LBA range

In SAS, there can be up to 64 LBMs and each LBM contains a combination of hints. Each I/O may reference an LBM.

In SATA the Data Set Management command may assign an LBM to a range of LBAs.

These hints can be used by the SDS to enable the drive to better manage workloads, possibly achieving higher throughput or IOPs.

7 Summary

The SNIA has recognized the importance of addressing these requirements and involving Hyperscalers in the standards activities that can address them. Through SNIA-based open source efforts and standards coordination, it is hoped that the storage industry can be more agile in responding to identified requirements with future updates and new standards where needed.

The current process of sending requirements directly to vendors to implement as part of the RFP response leads to fragmented implementations that may diverge significantly from the final standardized versions thus delaying the availability of these features across all Hyperscaler and Enterprise customers.