Addressing Cold and Archive Storage Requirements with Optical storage

Terry Schroeder
VP Business Development / Archive Solution Architect
Legal slide

The information provided by Sony is confidential and Sony makes no representations and undertakes no obligations with respect to such information. Information such as anticipated specifications, features and availability is provided for non binding discussion only.
So Why Optical Now?

- Solution Engineered and Optimized for Data Center requirements & PB sized data problems
- Native WORM for Regulatory Compliance Archive & Data Preservation
- Ransomware Protection
- Green Tech for flexible DC deployment options
- Non-Magnetic and Non-Contact Recording/Reading
  - Media survives EMP / Flooding
- Eliminate majority of data migration requirements and $$$
- Data Integrity - Low error rate media, media encoding verification and Erasure Coding
- Media life 100+ years *

* Estimated from Sony’s accelerated life-test based on ISO specification
PetaSite EX

Enterprise Archive Library
(w/Archive Gateway/S3 Metadata Services)

Monitoring and Performance

Operations

S3 / Rest API

S3 REST API Guide

Gen-3 500GB
Optical Media

Archival Disc

Optical Disc Archive

5.5TB
Enterprise Archive Media Cartridge

Sony
Media and Drive Technology

**Disc Structure**
- Dual Sided – Triple Layer
- Inorganic alloy substrate
- Protective cover layer

**Disc Reliability**
- 100+ Year *
- Longer Life under ideal conditions

---

**Sony Optical Drive Technology**
- 8 CH dual sided drive
- Simultaneous dual sided recording
- Drive Transfer Rate
  - Average Write: 188MB/s (1.5Gb/s)
  - Average Read: 375MB/s (3Gb/s)
- FC Interface (8Gb)
- High reliability laser

---

*Estimated from Sony's accelerated life-test based on ISO specification
Sony Enterprise Archive Solution (PetaSite EX)

AWS Compatible S3 Clients
- Peer connected Storage
- HSM / Storage tiering / Policy driven
- Application Stacks (Archive use cases)

Enterprise Management Platform (EMP)
- Solution Dashboard
- GUI and REST Interfaces
- Archive Management
- Provisioning, Data, User, etc.
- Alert / Fault Management
- Performance Management
- Operational Management

Enterprise Archive Gateway (EAG)
- S3 Endpoint
- Object store

Sony Enterprise Library
- Cartridge Slots: 912 – 8,544
- 4.8PB ~ 47PB
- 5.5 TB Optical Cartridge introduction
- Up to 10 Expansion Units
- Up to 4 Drive Units
  - Drive Qty.: 21 for single drive module
  - Maximum configuration = 84 drives
- IE port: up to 80 cartridge slots
- MTTR 1 hour
- Enterprise Redundancy

S3 Protocol

HTTPS / Rest / SNMP
PetaSite EX Configuration Options *(summary)*

- Single Library, Single Name space, cold archive
- Object Store Appliance option (Max 21 Drives)
- Multi-Library, Single or Multi-Name space
- Erasure Code (EC) does not span libraries
- Large scale-out cold archive use cases
- Archival Storage Partitions (ASP) can be scattered across multiple libraries.
- Multi-Library, Single Name space
- EC’ed Clustered libraries 4*
- Performance class solution supporting greater random I/O workloads and improved TTFB

Solution Resiliency and Data Availability

* Cluster does not imply “fail over”, the libraries are clustered for improved performance and data availability.
Concepts to be covered

1. Rule of 4, Storage Allocation, PetaSite EX – Storage Constructs
2. Some easy operations math
3. S3 Client optimizations to address latency
4. Bucket Parallelism – Optimizing performance for large “Put” operations
5. Optimizing “Get” Operations
Rule of 4, Storage Allocation, PetaSite EX – Storage Constructs

- “Put” operations are performed across a minimum of 4 optical disc.
- All Data is EC’ed (Recommended for archive/cold storage operations)
- The minimum storage allocation is 4 Cartridges - 22 TB Raw or 16.5 TB of EC’ed data.

Unique to PetaSite EX - Storage constructs

- Enterprise Access Gateway / EAG
  - Object Store solution optimized for Optical library
- Archival Storage Partition / ASP –
  - The logical construct of management units for the whole archival storage volume
  - Set policies for size of archive, levels of protection(EC), availability or encryption, allocation strategies and read/write priorities
- Object Repository / OR(Bucket) –
  - The construct of a bucket as introduced by Amazon® is equivalent to an Object Repository of an archival storage partition
  - Defines policies for object retention / destruction and user access (users, groups, read only, write, etc.)
- Protected Volume Array / PVA –
  - The physical storage construct utilized to allocate media from the library to an ASP and protection level (EC) of the data
- Protected Disc Array / PDA
  - Set of 4 disc medium spread across a PVA that has been allocated to an ASP.
Easy Operations Math – S3 Client – \textit{max\_concurrent\_requests}

Optimizing S3 client configuration for “Put” operations

N = Optimal number of parallel client operations for 1 PDA sometimes referred to as \textit{max\_concurrent\_requests} in the S3 client configuration.

X = 12GB = Cache full threshold where data begins writing to the Library, fixed value

Y = 24GB = Maximum size of RAM Cache available for each PDA, fixed value

Z = Object size (2GB in this example)

\[(X + Y)/2/Z = N\]

\[(12+24)/2/2GB = 9\]

The result of 9 means that from your S3 Client infrastructure you should always have 9 parallel Put operations running during the job. If we were using AWS’s S3 CLI we could configured 1 client with \textit{max\_concurrent\_requests} = 9 or if we had 2 clients operating then the recommended setting would be \textit{max\_concurrent\_requests} = 5(round up).
S3 Client Optimizations – Clients and Network

Each S3 client instance rarely can use all the underlying node's bandwidth or CPU, so better performance can often be achieved with multiple S3 client instances running. There are other things to be considered, like how much I/O an S3 client can generate and CPU utilization. No two S3 clients are alike.

It is recommended that your S3 client and PetaSite EX infrastructure exist within the same LAN and should support a minimum of 10Gbe network.
S3 Client Optimizations - Throttling

EAG, unlike most Object Store solutions must deal with the inherent latency of a library architecture, namely robotics. As such tuning the client behavior is important to optimize job performance and execution.

We already called out one client optimization, `max_concurrent_requests`, but there are additional behaviors and settings that really address the issue of latency.

Throttling - 503 Slow down/retry – In general, all S3 clients are designed to enter a throttling operation when a 503 response is received. Most likely this will be the result of robotics loading the requested PVA and or other requests/jobs that could be ahead of your request. Do not assume that your S3 client responds correctly, you should perform testing to ensure correct behavior.
S3 Client Optimizations – Retry and Timeouts

Retries – The maximum number of retries is a S3 client configurable option. It is recommend to determine a sufficiently safe value by comparing system resources with the use cases and the throttling methods, but by default we recommend a large setting like 20-25.

Timeouts - The client timeout value (upper limit on the time to wait for a response) can be set in most cases. We would recommend that the timeout value should be set to about 10 minutes.

With these two settings established, a job could wait 200-250 minutes before the job would fail, allowing plenty of time for other operations to complete in the library.
Put operation optimization - 307 (temporary redirection) and 100-Continue

To support a high transfer rate of data to the library and drives the EAG service presents multiple interface nodes to the client network. From a client perspective a single S3 endpoint (FQDN) is provided. To ensure that a client is talking to the right IFN the EAG service utilizes the http 307 Temporary redirection response to direct the client to the proper IFN that can address the client request. This is all standard stuff except if.....

Almost all S3 Clients add an “expect 100-continue” header when issuing a request. If you don't have it, be sure to enable it. 100 Continue is used for handshaking before data transfer, and prevents data transfer to the wrong destination in a system where the above-mentioned 307 Temporary Redirection might be returned. The client does not start transferring the payload until a 100 Continue response is returned, and waits for 307 to return. If this handshake is not performed, the data transfer will start to the wrong IFN and the 307 must be handled after the first transfer was completed and retransmitted to the correct IFN, the result being unnecessary network bandwidth being consumed and lost time.
Bucket Parallelism

Bucket Parallelism enables more bandwidth/library resources to be available for ingesting large data sets in shorter periods of time

- With BP enabled, all PVA’s are loaded when the first “Put” operation is received
- Objects are scattered across PVA’s when written with BP
- Understand that you are provisioning a large volume of storage with this feature, but it enables scalability for large Put operations
- Water fall effect, the bandwidth of a single PVA must be exceeded before Objects will be written to the next PVA
  - Not providing enough Client performance/load will result in library resources being consumed that will be idle until the write operation is complete.
## From a single PVA to BP – Performance Example

<table>
<thead>
<tr>
<th>PVA loaded &amp; Drive &amp; Cartridge Ct.</th>
<th>Single drive</th>
<th>Write Speed</th>
<th>EC Overhead</th>
<th>PVA Size 3/4</th>
<th>Hr's to Write</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP Setting</td>
<td>Drive MB's</td>
<td>TB's</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>188 Avg.</td>
<td>752</td>
<td>564</td>
<td>16.5</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>1504</td>
<td>1128</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>2256</td>
<td>1692</td>
<td>66</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>3008</td>
<td>2256</td>
<td>82.5</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>3760</td>
<td>2820</td>
<td>99</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>24</td>
<td>4512</td>
<td>3384</td>
<td>115.5</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>28</td>
<td>5264</td>
<td>3948</td>
<td>132</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>32</td>
<td>6016</td>
<td>4512</td>
<td>148.5</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>36</td>
<td>6768</td>
<td>5076</td>
<td>165</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>40</td>
<td>7520</td>
<td>5640</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Optimizing Get operations – It’s sequential

For Get or read operations, it is necessary to devise techniques that minimize unnecessary seeks or idle drive times. The goal is to perform read patterns that are truly sequential like GET object1, object 2, etc.. The next best option is to make the read pattern as sequential as possible. To achieve this, we must understand the order in which the objects were written to the media. This can be easily done with the “list-objects” command which by default returns the objects by the time stamp in which they were written.

By default, “Get” operations are parallel. Multiple PVA’s will be loaded in parallel if required to address a “Get” operation.

It is recommended that you increase the number of max_concurrent_requests from the “Easy Math” section for better “Get” performance. This may have an implication on your S3 clients robustness from a networking standpoint. Secondly, if you wrote data using the Bucket Parallelism feature, utilizing more clients will improve your read performance due to the scattered nature of the written objects.

For truly random reads or small reads across many media, it is difficult to fully optimize performance because of the required media changes. But the best practice should still be followed, read the objects in the order they were written.
PetaSite EX and TreeFrog

TreeFrog (HPSS Storage Broker Feature)
- Storage Management software utilizing peer coupled storage where data is directed to other Peer storage endpoints.
- Built with Apache jclouds® a Multi-Cloud toolkit enabling broad cloud technology support in TreeFrog.
- Utilizes Open SIRF (data preservation format). Stored Treefrog data containers (managed datasets) include a SIRF magic object, a catalog and user data fragments. Data fragments contain user data objects and files.

Sony Enterprise Library

TreeFrog and PetaSite EX
- Currently Sony is coordinating interoperability testing with the HPSS Collaboration team in Houston.
- Proof of concept jclouds S3 bucket read/write connectivity testing completed in 2Q19.
- Sony and the HPSS Collaboration team in Houston intend to complete TreeFrog interoperability testing with PetaSite EX and subsequently provide deployment support for the PetaSite EX solution.