PURESTORAGE®

WHITE PAPER

Deploy Storage Class Memory in Pure FlashArray//X

Manage the balance between application performance and operational efficiency.

Contents

Introduction

Managing the balance between application performance and operational efficiency is a constant struggle for many organizations. Processing information on the server will yield the best application performance, however it typically provides the least efficiency for storing, managing, and maintaining the data. Introducing a storage array into the application architecture provides better efficiencies, but the most mission-critical applications sometimes pay a price in performance. Pure Storage built FlashArray™//X as a high-performance platform to address these needs with world-class, Purity software and modern flash industry standards including NVMe and NVMe-oF. Now, including storage-class memory (SCM) within FlashArray//X provides additional benefits by reducing latency and increasing throughput when reading data from the array.

Storage-class Memory

The purpose of storage-class memory (SCM) is to close the gap between dynamic RAM and storage. Think of storage-class memory as a cross between dynamic RAM and NAND flash. It's a persistent memory technology that, unlike DRAM, will preserve the data it contains during a power cycle much like flash.

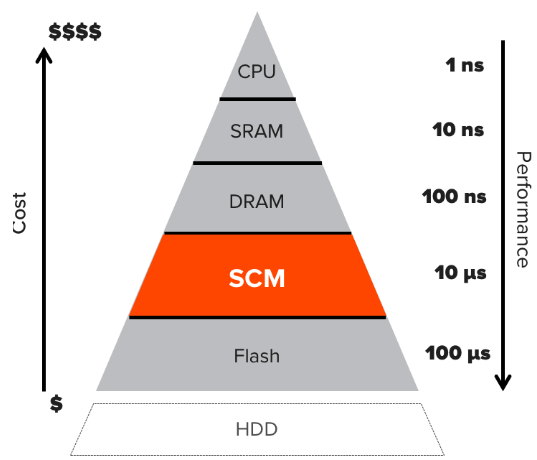

Figure 1. Data-retrieval technologies comparisons

As shown in Figure 1, cost, capacity, and performance are primary differences between flash and storage-class memory. While flash tends to perform in the 100us range, SCM can perform in the 1us to 10us range, depending on form factor.

Figure 2. Intel DC Persistent Memory Module SCM Form factor

The standard non-volatile dual in-line memory module (NVDIMM), or Intel's DC Persistent Memory Module (DCPMM) shown in Figure 2, is a storage-class memory form factor that you can plug into a memory slot on a CPU motherboard that supports such technology. This form factor provides higher capacities than registered DIMM (RDIMM) but lower performance and ability to maintain data through reboots. It is limited in scalability and has tradeoffs in performance vs. scale and functionality. The number of DIMM slots is limited and consumers have to choose between the nanosecond speeds of RDIMMs and the microsecond speed of storage-class memory.

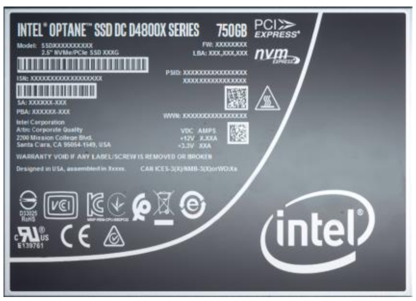

Figure 3. Intel Optane 2.5in NVMe drive SCM Form factor

The other form factor of storage-class memory is a 2.5-inch NVMe-connected drive as shown in Figure 3. This form factor leverages the storage-class memory technology to provide performance improvement of about ten times over NAND flash in the same footprint. Capacity points are a key difference in the SCM drives:

- DCPMM comes in 128GB, 256GB, and 512GB capacities and can provide up to 6TB of persistent memory and 1.5TB of DRAM on a two-socket server utilizing 12 DIMM slots for storage-class memory and 12 DIMM slots for RAM using the 1:4 DIMM to DCPMM ratio recommended.
- The storage-class memory drives come in 375GB, 750GB, and 1.5TB capacities. In theory, you can install as much capacity as you have NVMe drive slots—and installing storage-class memory drives does not impact the memory size ratio.

Figure 4. DirectMemory Modules based on Intel Optane SCM installed in FlashArray//X

DirectMemory Modules and Purity Optimize DirectMemory Cache

Pure is introducing storage-class memory in Pure FlashArray//X70R2 and //X90R2 as DirectMemory Modules (DMMs), leveraging the 750G Intel Optane storage-class memory drive form factor.

The configurations supported are four DMM for 3TB of cache in the //X70R2 and //X90R2 and eight DMMs for 6TB of cache for the //X90R2.

Once inserted into the array easily and non-disruptively, these DMMs become available to Purity as DirectMemory Cache without configuration or tuning necessary. DirectMemory Cache is a component of Purity Optimize that utilizes the DMM as a read-cache for *hot data*. Hot data is a data set that is read multiple times. The DirectMemory Cache keeps the hottest (most-frequently read) data available in the DMMs, while colder data is read from the DirectFlash Modules (DFMs). While the cache algorithm does not involve a specific threshold for the time between subsequent reads to the same data, it must be "short enough" for data to be cached, specifically for large data set workloads that are highly read-oriented.

When something is read from the array, Purity Optimize identifies if the data is hot by placing the data blocks into a short-term cache as they are read. As the short-term cache fills, entries are pushed out through a first-in-first-out (FIFO) queuing mechanism. Once pushed out of the short-term cache, the data becomes a candidate for long-term cache if a read for it occurs while the key is still remembered in a lookback queue. The more often a block of data is read over a period of time, the more likely it will make it into the long-term cache.

Once in the long-term cache, Purity uses a least-recently-used (LRU) algorithm to maintain frequently read data. Like the short-term and lookback queue, this space is finite. As new data is inserted, the "coldest" data (has not been read for the longest time) will age out. For a FlashArray//X without DMMs, this process uses some of the system DRAM, but the cache is on the order of 1/1000th the size of a DirectMemory Cache configuration. For workloads with large data sets, a much smaller subset of the data set can fit in the cache. This means that the likelihood of a cache hit is much lower and the DirectFlash Modules (DFMs) will serve most of the data.

It is important to note that the DMMs are not the primary location for the data and don't count as capacity on the array. All the data is still maintained durably on the DFMs. DirectMemory Cache greatly increases the likelihood of a cache hit by increasing the size of the long-term cache buffer. The DMM uses storage-class memory technology, which has a performance characteristic of about 10us, or 10 times the performance of a DFM. This results in dramatically reduced latency and improved throughput for high-read workloads.

Application Profile

DirectMemory Cache is targeted specifically for read-intensive, high-performance workloads, which is why DMMs are supported only on the highest performing FlashArray//X models. Understanding the workload is the key to knowing whether DMMs will improve application performance.

High-performance, mission-critical databases are the most common type of application workload to benefit from this technology. But not every type of database workload will benefit from DirectMemory Cache. For example, workloads that are primarily write-centric or modify data will not benefit. The workload must be read centric, but a workload that only does a single pass of the data set may not benefit. Workloads that do multiple passes of the same data set will benefit most. This includes data analysis or report jobs that repeatedly read the data set and perform an action on the data.

Data-set size is also an important factor. If the workload data set fits within the 3TB to 6TB range, there is a good probability that the cache hit rate will be high given multiple reads. However, a data set that large may have to compete with other read-intensive applications for the cache space. In the event of multiple workloads and data sets, the cache should be reasonably sized to handle multiple jobs. Even if the total data set size is larger than the cache size, there may be hot spots in the data set which can result in a higher-than-expected hit rate.

The amount of time that the workload spends reading is a third factor requiring consideration. For example, if a job is four hours long, but only 30 minutes is spent reading data, then reducing the latency on those reads may not have a significant overall impact. In this example, if the read latency and throughput cut the overall read time in half, the job would still take 3 hours and 45 minutes. While 15 minutes is 6.25% improvement, it is not a direct translation to a 50% latency reduction

Network latency is another consideration. Read cache improves the read-service performance within the array by 50%. But this improvement is on the order of 100us. When compared to a SAN latency of about 1.2ms, DirectMemory Cache would only provide about 8% overall latency reduction. Translating that into the previous example of 30 minutes of read time puts the overall improvement at about 2 minutes, 24 seconds—about a 1.1% overall improvement. As a result, we expect the best results on workloads for which SAN latency is predictable and much less than 1ms, which means primarily fiber channel or NVMe-oF workloads.

This seems like a lot to consider and can seem overwhelming. Many organizations have a good understanding of their applications and can quickly look at these factors to decide whether DirectMemory Cache is a good fit. Others will require some analysis to decide. For existing customers with workloads running on Pure arrays, Pure teams can use tools to identify the effective hit rate on their array. Long term, Pure plans to enable customers to use Pure1® to model effectiveness.

Summary

Purity Optimize DirectMemory Cache provides a significant read performance boost using storage-class memory. You can easily and non-disruptively add the DMM component to a FlashArray //X70R2 or //X90R2 to provide additional read cache. This read cache translates to up to a 50% performance improvement in read-service latency, assuming a near 100% hit rate. For multi-pass, read-intensive workloads on low-latency SAN networks, DirectMemory Cache can reduce runtimes and improve overall application performance. Pure provides tools to help FlashArray customers determine the effectiveness of DirectMemory Cache.

Additionally, NVMe enables unprecedented performance density required for mixed-workload consolidation in your cloud. FlashArray//X currently supports ultra-dense 18.3TB DirectFlash™ modules, which you can adopt nondisruptively with full performance. In addition, Purity's always-on QoS feature means you can consolidate radically diverse applications without fear. A shared design consolidates data silos; accelerates production, DevOps, and data analytics; and helps organizations pivot to a data-centric architecture.

Pure Storage, Inc.
650 Castro Street, #400
Mountain View, CA 94041

PS-07