New VST Levels Let You Further Optimize Flash Use for Performance and Cost

Jay White
Technical Marketing Engineer

Chittur Narayankumar
Technical Marketing Engineer

Virtual Storage Tiering (VST) is the NetApp approach to automated storage tiering (AST). AST technologies help data centers benefit from the improved performance of Flash-based media while minimizing the cost and complexity. Flash-based devices such as solid-state disks (SSDs) and controller-based Flash can complete 25 to 100 times more random read operations per second than the fastest hard disk drives (HDDs), but that performance comes at a premium of 15 to 20 times higher cost per gigabyte.

Rather than permanently placing an entire dataset on expensive Flash media, VST automatically identifies and stores hot data blocks in Flash while storing cold data on slower, lower-cost media. NetApp has put a lot of time and energy into understanding the challenges that AST must address in order to architect an optimal solution.

With two recent product additions to VST, NetApp now offers end-to-end Flash options spanning from the client application through the disk subsystem.

- **Controller level.** Storage controller-based Flash—NetApp® Flash Cache—retains hot, random read data. (You can learn more about the algorithms used by Flash Cache and other details in a previous Tech OnTap® article.)
- **Disk-subsystem level.** NetApp Flash Pool technology uses a hybrid model with a combination of SSDs and HDDs in a NetApp aggregate. Hot random read data is cached and repetitive write data is automatically stored on SSDs.
- **Server level.** NetApp Flash Accel™ technology extends VST into the server. It uses any server-side Flash device (PCI-e Flash card or SSD) as a local cache that off-loads I/O from networks and back-end storage to deliver optimum I/O efficiency to your busiest applications while freeing up server CPU and memory resources.

Explore

Cluster-Mode and Virtual Storage Tiering

All the VST technologies described in this article work with Data ONTAP 8 operating in Cluster-Mode to deliver the ultimate in scalability, flexibility, and performance. This issue of Tech OnTap features an article by Vaughn Stewart on virtualizing business-critical applications using Cluster-Mode. Check out all the recent Tech OnTap articles on Cluster-Mode.

- Enterprise-Ready Scale-Out with Data ONTAP 8 Cluster-Mode. Provides a good overview of Cluster-Mode technology and capabilities.
- PeakColo Accelerates Cloud with Cluster-Mode. Describes how a cloud service provider uses Cluster-Mode to achieve strategic advantage.
- Cluster-Mode Performance and Scalability. Explains how a cluster of 24 FAS6240 nodes delivered over 1.5 million SPECsfs2008_nfs.v3 ops/sec.
- FAS6200 Cluster Delivers Exceptional Block I/O Performance. Describes the high throughput and low latency of a 6-node FAS6240 cluster running the block-oriented SPC-1 benchmark.

All three levels continue to offer the full advantages of VST, including:

- **Real-time promotion of hot data with high granularity.** Hot data enters VST immediately, and its 4KB granularity means that it uses Flash-based media very efficiently.
- **Easy to deploy and simple to manage.** VST works with your existing data volumes and LUNs. It requires no complicated or disruptive changes to your storage environment. There is no need to set policies, thresholds, or time windows for data movement.
- **Fully integrated.** VST is fully integrated with the NetApp Unified Storage Architecture, which means that you can use it with any NAS or SAN storage protocol with no changes.

This article describes the disk subsystemâ€“level and server-level VST options using NetApp Flash Pool and Flash Accel technology and provides general guidelines on when and where to deploy each of the three levels. If you're not already familiar with Flash Cache, check out the original Flash Cache article for details.

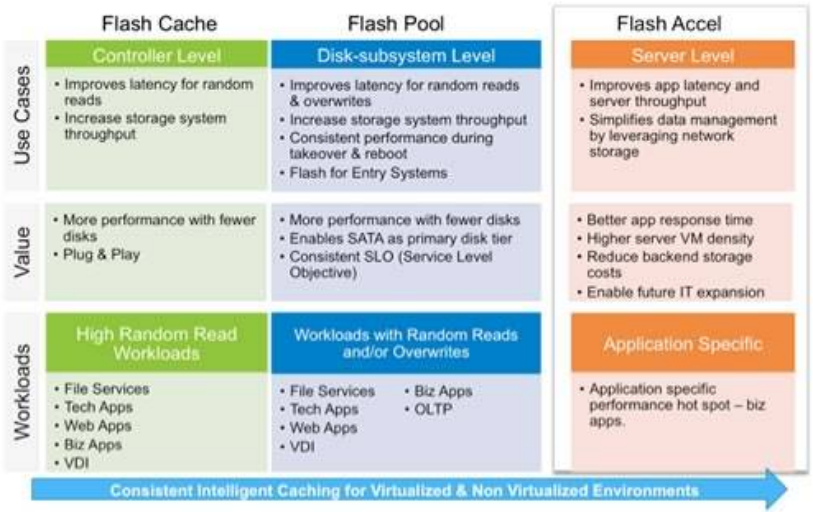

Figure 1) NetApp Virtual Storage Tier now operates at different levels within your infrastructure, allowing you to better optimize your use of Flash.

Flash Pool

A NetApp Flash Pool works at the level of the NetApp aggregate. (An aggregate is a collection of RAID groups.) A Flash Pool is created simply by adding a RAID group composed of solid-state disks (SSDs) to an existing 64-bit aggregate, creating a hybrid disk array that gets the best from both technologies. The SSDs are used to store random reads and repetitive random writes (overwrites) for the volumes within the aggregate, off-loading this work from hard disk drives (HDDs). As a result you can achieve the same level of performance (with better overall latency) using fewer disk spindles or using capacity-oriented disks rather than performance-oriented disks. Flash Pool gives you the latency and throughput advantages of SSD and the mass storage capacity of HDD.

The disk subsystemâ€“level Flash Pool approach offers a number of advantages.

- **Persistence.** Because it is implemented in the disk layer, Flash Pools persist and remain operational when a takeover event occurs. In an HA configuration, if one controller goes offline for a planned or unplanned outage, the other controller takes over its aggregates and volumes, including Flash Pools. RAID provides resiliency to protect the data within the Flash Pool.
- **Random read and random overwrite caching.** From the perspective of an HDD, the most "expensive" activities are random reads and random overwrites of existing blocks. Flash Pool technology off-loads these operations to SSDs. Caching overwrites populates the Flash Pool with blocks that are likely to be reread, and prevents short-lived writes from being written to HDD.

- **Deduplication awareness.** Flash Pool technology is fully deduplication aware. A deduplicated block may have many references, such as when many nearly identical virtual machine instances are deduplicated. Although a deduplicated block is accessed through multiple references, only one instance of that block is kept in SSD. Less Flash is needed to accommodate a given workload as a result of this efficiency. This effect is sometimes referred to as cache amplification.
- **Support for FAS2200 series.** Because of their compact size, NetApp FAS2200 series controllers do not support controller-level VST, but they can utilize Flash Pool technology.

How Flash Pool works

To understand how Flash Pool technology works, you need to understand the processes for identifying and delivering random reads and random overwrites to SSD. The first time a block is read it is read into storage controller memory from disk, and the read event is categorized as random or sequential. As blocks that are categorized as random age out of controller memory they are written to SSD. Subsequent reads of the same block are then satisfied from SSD.

For writes, Data ONTAP is write optimized by design. It uses an efficient NVRAM to journal incoming write requests so that they can be acknowledged to the writer without delay. Writes are collected and written to disk in full stripes whenever possible, driving optimal performance from the underlying RAID implementation and HDDs by turning a collection of writes into sequential write activity.

The goal with Flash Pool is to off-load I/Os from HDD while enabling blocks that are likely to be reread or rewritten to end up on SSDs. Large sequential writes are handled efficiently by HDDs. Keeping them on SSDs would be a suboptimal use of resources. Random writes, and particularly blocks that are being repeatedly overwritten, turn out to be the ideal candidates to target to Flash Pool SSDs. Flash Pool populates SSDs with blocks that are likely to be read and blocks that are written repeatedly.

When a write request is received, Data ONTAP verifies that the write is random rather than sequential and that the previous write to the same block location was also random. If so, that write goes to SSD.

How blocks are evicted from a Flash Pool

Data ONTAP® technology maintains a heat map (stored on SSD for persistence) that keeps track of how "hot" each block is. Reads enter the Flash Pool at "neutral." A subsequent read elevates the temperature of the block to "warm" and then to "hot." Writes also enter the Flash Pool at "neutral." Subsequent overwrites don't elevate the temperature of the block, however.

When available SSD space runs low, Data ONTAP begins running an eviction scanner that decrements the temperature of each block on each pass. For example, "hot" blocks become "warm," "warm" blocks become "neutral," and "neutral" blocks become "cold." If a block is read or overwritten between scanner passes, its temperature is again incremented—"hot" remains the maximum for reads and "neutral" the maximum for overwrites. If a "cold" block is not read or overwritten, it is decremented to a temperature of "evict" on the next scanner pass. At this point "read" blocks are evicted while overwrite blocks are scheduled to be written to HDD.

This mechanism enables only hot data to remain in a Flash Pool when it becomes full. Flash Pool adjusts dynamically to retain hot data, and the amount of a Flash Pool dedicated to reads versus overwrites depends solely on the particulars of the workloads using the pool.

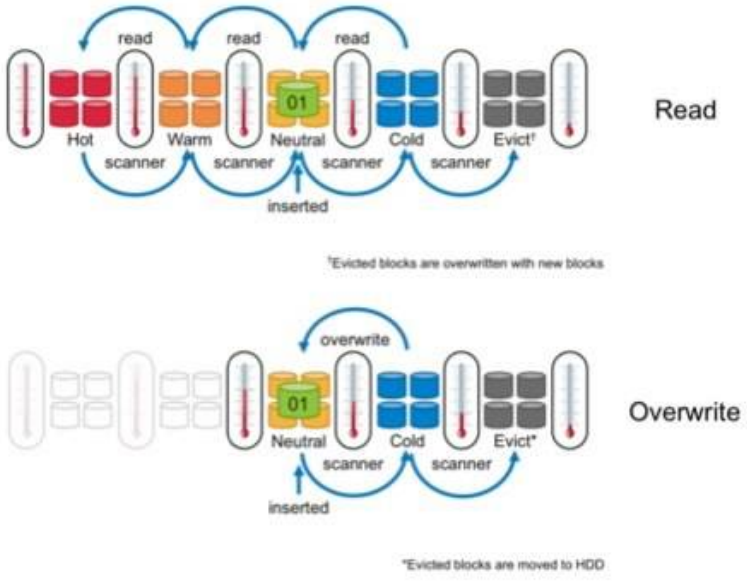

Figure 2) Blocks are evicted from a Flash Pool based on a heat map. Once the pool is full, an eviction scanner decrements the "temperature" of each block on each pass. Blocks are evicted when they reach a temperature of "evict." Accesses between scanner passes increment the temperature of a block, so "hot" data remains in the Flash Pool.

Flash Pool performance

Although we haven't published any benchmarks yet using Flash Pool technology, NetApp has undertaken some comparative before-and-after studies using an OLTP workload to illustrate the potential impact. Starting from the same FAS6210 base configuration, we implemented Flash Pool, optimized in one case for cost per IOPS and in the second for cost per GB of storage. Results are shown in Figure 3. Note that both cases result in a significant improvement in overall latency, which can have a bigger impact on perceived performance than total IOPS in many cases.

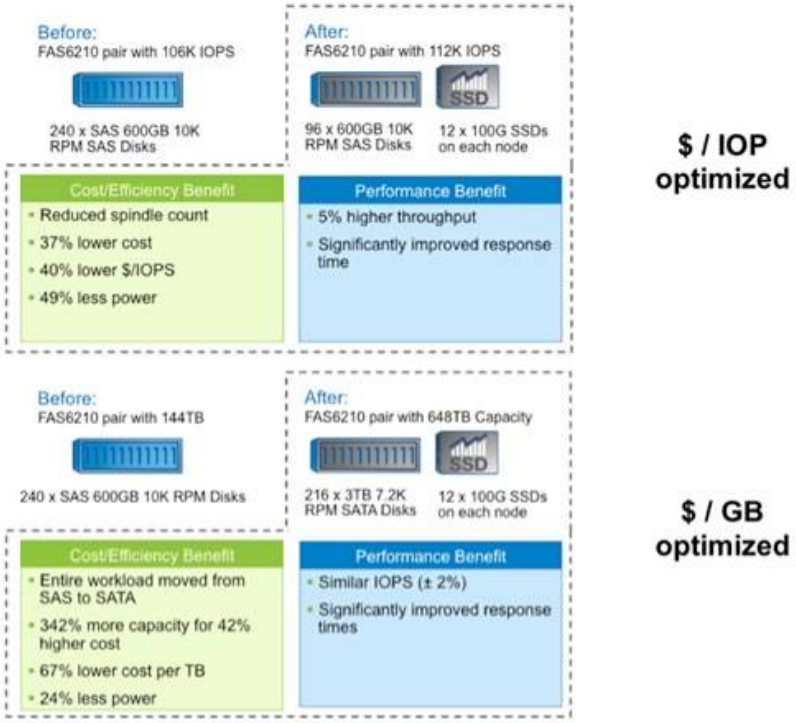

Figure 3) Impact of Flash Pool on cost/efficiency and performance.

Table 1) Flash Pool requirements and options.

Flash Pool Requirements and Options	
Data ONTAP Version	Data ONTAP 8.1.1 or later, 7-Mode and Cluster-Mode
Tuning Options (per volume)	
Read	Random-read (default) Meta: Metadata only Random-read-write: Populates read pool with random reads AND writes None: Disables read caching for a volume
Write	Random-write (default) None: Disables write caching for volume
Supported Platforms	FAS22x0, FAS3240/3270, FAS3160/3170, FAS60x0, FAS62x0, and V-Series using NetApp disk and SSD only

To learn more about deploying and using NetApp Flash Pool technology, check out NetApp TR-4070: Flash Pool Design and Implementation Guide.

Flash Accel

NetApp Flash Accel software was announced in August 2012 and will be available in late 2012. Flash Accel is designed to extend the benefits of NetApp VST across the network to encompass the server itself. Having local Flash devices on a server means that you've got direct-attached storage that you have to manage. This creates potential problems with data protection and isolates silos of data. Server caching with Flash Accel eliminates these problems, and offers a number of advantages.

- **Dedicate Flash to enhance performance of a particular application.** Flash Accel lets you pinpoint Flash use for the benefit of one or a few applications while eliminating the disadvantages of local storage, increasing throughput by up to 80%, and reducing transaction latency by up to 90%.
- **Hardware agnostic.** Flash Accel will work with any enterprise-class Flash device(s) (PCI-e card or SSD) that you have on your server. NetApp has also signed an agreement with Fusion-io to resell its ioMemory products for those who don't have a preexisting device. We've also expanded our Alliance Partner ecosystem to include a variety of server-caching partners. (See the recent press release for details.)
- **Persistent and durable.** Data stored in the Flash Accel cache is able to persist through a server reboot. The cache even remains durable to events such as failures and blue screens.
- **Unique cache coherency.** When an event such as a restore changes data on back-end storage, other caching solutions resort to dumping the entire server cache, resulting in a long period of reduced performance while it is repopulated. NetApp Flash Accel is able to identify and evict just the blocks that have changed, preserving performance.
- **Increases VM density**. Because VMs and applications run more smoothly and spend less time blocked waiting for resources, you can actually increase the number of VMs per server—5 to 10 additional VMs is typical.
- **Improves efficiency of back-end storage.** Tests show that Flash Accel improves the efficiency of back-end storage by 40% versus the same configuration and workload without Flash Accel enabled. This reduces the resources required on back-end storage and frees up resources to support other workloads.
- **Low overhead.** Flash Accel requires only about 0.5% of the memory resources of the ESX host.
- **Data protection.** Data stored in a server-side cache is also stored on NetApp storage, where it can be protected using standard NetApp methods.

The first release of Flash Accel works with VMware® vSphere® 5.0 or higher and Windows® VMs only. Future releases will expand support to include additional VMs, other hypervisors, and bare metal.

How Flash Accel works

Flash Accel consists of three components:

NetApp Flash Accel Management Console (FMC). Configuration and management of Flash Accel is accomplished using a virtual appliance, which runs on vSphere. This management console allows you to:

- Install and configure the ESX hypervisor plug-in driver.
- Install and configure guest Flash Accel agents.
- Discover Flash SSD devices on ESX hosts.
- Configure one or more SSD or other Flash devices on ESX hosts for use by Flash Accel.
- Enable/disable caching on host.
- Resize the cache on a guest VM.
- Report on current cache state and performance metrics.

Flash Accel hypervisor plug-in (installed on the ESX host). The hypervisor plug-in is installed on an ESX host and establishes control over locally attached devices (such as SSDs) and storage array paths according to the configuration you define using FMC. The plug-in creates logical devices and presents them to the ESX storage stack as SCSI devices. Logical devices created on multiple ESX hosts with the same WWN allow ESX to treat a device as a shared device so that VMs using these devices can participate in vMotion® and VMware HA operations. In addition to being able to migrate the VMs, the hypervisor plug-in provides management of the Flash device and can enable dynamic resource sharing and cache block deduplication.

Flash Accel agent in Windows VM. A user-level agent is implemented for Windows guest VMs. This agent:

- Passes configuration to the filter driver
- Enables/disables caching of one or more devices or an entire VM
- Communicates performance metrics to Flash Accel Management Console
- Integrates with other data management software like SnapDrive® and SnapManager® technologies

The service agent exports a Web service to FMC and communicates with the drive via Windows PowerShell™ cmdlets.

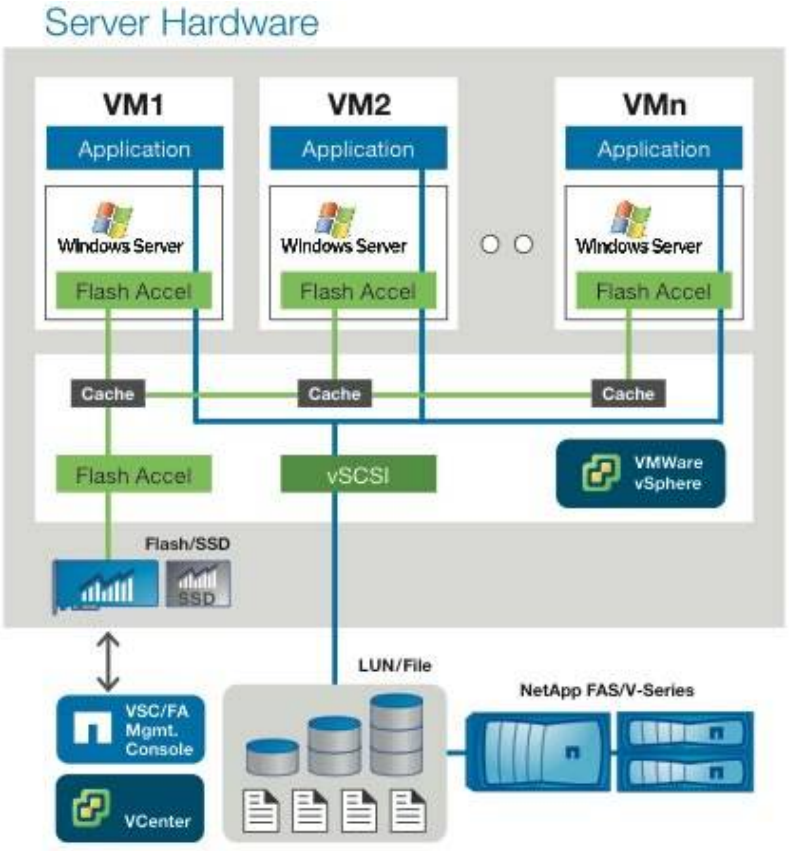

Figure 4) Flash Accel includes agents that run in each VM and a plug-in for VMware vSphere, and it is controlled from Flash Accel Management Console. It can use any PCI-e Flash card or SSD available on an ESX host.

As Figure 4 illustrates, Flash Accel uses local Flash resources on an ESX server to provide a caching layer for Windows virtual machines. The Flash device can be shared among multiple VMs on an ESX host, giving each VM its own local cache.

All reads from the VM are cached locally for reuse, off-loading future reads from back-end storage. Writes are written through to back-end storage but available for rereading from cache.

The Flash Accel cache has two key areas: cache operations and storage manager.

- The **cache operations** layer is responsible for implementing the interfaces for sending I/O requests through the cache; this includes translating incoming I/O requests into a number of 4KB I/O requests to or from the cache and/or the primary storage server. The cache operations layer is entirely implemented in the Windows filter driver.
- The **storage manager** is responsible for the layout of the metadata and cached data blocks on Flash and the implementation of persistence. This module is called only by the cache operations layer. The storage manager resides within the filter driver and the hypervisor initializes, configures, and manages the Flash device.

Data coherency is the most important feature of Flash Accel. If back-end data is changed without notifying Flash Accel, it is possible to have the cache data and the back-end storage data out of sync. This would result in incorrect data being returned to the application/end user from the cache, which would cause data corruption. There are two situations in which data coherency is an issue.

- **Online data modification where data is modified in band.** Flash Accel checks for incoherency when there is a device mount/unmount/boot by comparing cached metadata with that from the storage system to spot incoherency and invalidate blocks as appropriate. An example of this would be a SnapRestore®

operation of application data on NetApp storage. In between the checks, there is no incoherency issue because Data ONTAP will not modify data when a VM is actively using it. Out-of-band modification (in which the administrator updates a running VM by some means the storage is not aware of) is not supported.

- **Offline data modification** (for example, VMDK/LUN restore). Flash Accel takes the same action of comparing cached metadata with data on back-end storage and invalidating blocks as needed. An example is using SnapRestore to restore an entire VM.

The advantage of Flash Accel in this type of situation is that it only invalidates blocks that are different while retaining all blocks that haven't changed. When situations like this arise, other available solutions completely drop all cached data and rewarm the entire cache. This may take a few hours to days depending on the data, during which time performance is degraded.

Flash Accel performance

We compared the performance of the same configuration with and without Flash Accel using JetStress, which simulates the disk I/O load created by Microsoft® Exchange. The addition of Flash Accel resulted in approximately a 77% improvement in I/O performance for both reads and writes. Since application reads were primarily satisfied by Flash Accel, the back-end storage was less occupied by reads and could therefore deliver better write performance, resulting in significant application performance improvements overall. Results are shown in Figure 5.

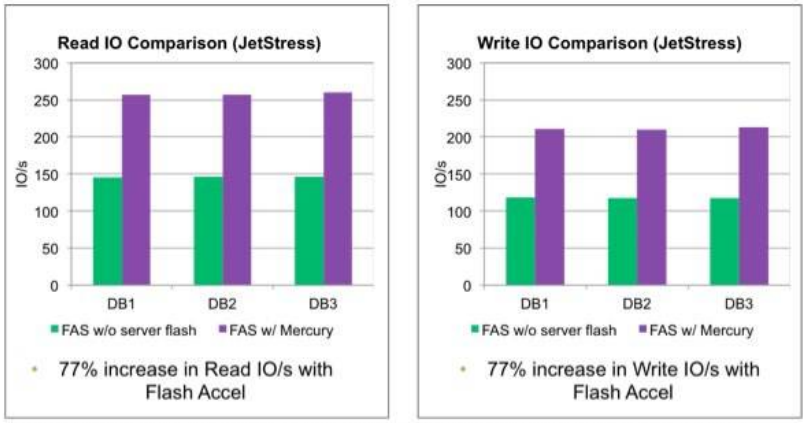

Figure 5) Flash Accel increases read and write I/O by approximately 77% using JetStress to simulate an Exchange workload.

Choosing VST Options

Choosing the best VST level or levels is really about getting the most return for your investment in Flash by accelerating all the workloads that need to be sped up for the lowest cost.

- **Server level (Flash Accel).** Provides acceleration for one or more VMs running on a particular ESX host.
- **Disk subsystem level (Flash Pool).** Provides acceleration for workloads on a per-aggregate basis.
- **Controller level (Flash Cache).** Accelerates all workloads associated with a storage controller.

In other words, within a shared storage infrastructure you get the most workload specificity at the server level and the least at the controller level. If you need to accelerate one workload, server-level VST is a good choice. If you need to accelerate all your workloads (and possibly switch from performance-oriented to capacity-oriented disks), choose disk subsystem level or controller level.

For new deployments, we suggest starting with either Flash Cache or Flash Pool technology and then adding Flash Accel if needed to provide further performance enhancement for the most latency-sensitive applications.

When it comes to choosing between Flash Cache and Flash Pool, the following bullets summarize the similarities and differences.

- Both Flash Pool and Flash Cache provide caching for random reads and both are fully deduplication aware for maximum space efficiency.
- Flash Pool is installed and supports workloads on a per-aggregate basis. Flash Cache applies to all workloads on a controller.
- Flash Cache is plug and play, whereas Flash Pool requires some simple configuration and is then self-managed.
- Flash Pool:
    - Off-loads I/Os to SSD for repetitive random writes
    - Is RAID protected
    - Provides consistent performance after takeover events
    - Supports the entire FAS portfolio including the FAS2200 series

In general, Flash Pool is a good choice for mission-critical applications because the benefit persists after takeover events. It's also preferred for applications that have high overwrite rates and is the only option available on the FAS2200 series. Because of its proximity to main memory, Flash Cache may offer advantages for high-performance file services.

While you can install both Flash Pool and Flash Cache on the same storage system, in general there isn't a big advantage to doing so. Data blocks from an aggregate that has Flash Pool enabled are never cached in Flash Cache.

Conclusion

With the introduction of Flash Pool and Flash Accel to VST, NetApp gives you two new methods to optimize I/O performance using Flash. As a general guideline, it helps to remember that:

- Flash Cache makes everything faster.
- Flash Pool makes an aggregate faster.
- Flash Accel makes an application faster.

You can combine levels to optimize overall performance while minimizing your investment. Whichever options you choose, once VST is installed there's virtually nothing to manage. You can fine-tune your deployment if needed, but the defaults work well in most cases and the benefits are significant and measurable.

By Chittur Narayankumar and Jay White, Sr. Technical Marketing Engineers

Kumar has been with NetApp for over 11 years and is currently part of the Flash Accel Group. He has authored several technical reports and Solution Builder documents related to messaging and collaboration on NetApp storage.

Jay is a Technical Marketing engineer with the Data ONTAP Group responsible for Flash Pool, system performance, and high-file-count environments. He has authored several technical reports and FAQs related to NetApp storage subsystems, resiliency, RAID, and more.

Quick Links

- Tech OnTap Community
- Archive
- User Groups
- PDF
