

Deploying flash drives in EMC's DMX to save power, cooling and improve performance

Last Update: Jan 05, 2009 | 10:12

Originating Author: **Nicholas Allen**

Editor's Note: Since EMC's announcement, several suppliers have introduced or promised to introduce flash drives into storage arrays, including NetApp, Sun, LSI, Hitachi, IBM and others.

Can you really achieve the savings and performance benefits promised with flash drives in a DMX?

First let's look at the flash drive specifications and claims.

STEC Zeus-IOPS Drive Specifications:

- Random transactional performance in excess of 52,000 IOPS sustained,
- Over 200 times faster transactional performance than a 15K RPM enterprise class disk drives,
- Less than half the power consumption of 15K RPM HDD's,
- Sustained random or sequential large blocks transfers to 200 MB/s,
- Read/write transactional performance 225MB/s / 107MB/s,
- Read/write sequential performance (MB/s) 200MB/s / 100MB/s,
- 5 year warranty.

EMC Claims for STEC Flash Drives in a DMX

- 10x faster response time,
- 30x IOPS improvement,
- 98% less power per IO,
- 38% less power per drive,
- No moving parts for high reliability.

Second, let's look at what EMC has done to accommodate these flash drives:

EMC tweaks to DMX to optimize Flash Drives Since 2003 EMC has been enhancing the Symmetrix DMX in preparation for flash drives. These enhancements include:

- **RAID-5:** while "mirror everything" used to be the way-of-Symmetrix, you just can't justify the cost for every application any more, and it's probably overkill for enterprise flash drives.
- **TimeFinder/Snaps:** Space-saving snapshots. With the cost of SSD, you don't want to make any more copies of your data than absolutely needed. The recent Asynchronous Copy on First Write enhancements ensure that the Snaps have minimal impact on the response times of the primary volumes on the flash drives.
- **Modular Packaging:** Symmetrix DMX-3 and DMX-4 are "enterprise-modular" arrays, allowing for almost unlimited flexibility of configuration - you can have one "quadrant" supporting as many as 600 drives for maximum capacity, or you can have a quadrant optimized for performance with as few as 32 drives. This approach now lets you dedicate a quadrant to flash drives to maximize their performance (you'll still need the

32 regular disk drives in that quadrant to support DMX's PowerVault, but you can use the space on those drives for other things as well).

- **Cache Partitioning:** With flash drives, you don't really need a lot of cache for reads, but you do want to have a modicum of cache for pending writes. In an interesting twist, you might actually want to decrease cache to a bare minimum for read-intensive applications. Dynamic Cache Partitioning helps to ensure that your memory is used where it's needed most, even as the system dynamically reallocates based on actual workloads.
- **Symmetrix Priority Controls:** Similarly, you want to be sure that the flash drives receive the appropriate relative priority to everything else in the system, and internally the DMX uses the underlying mechanisms to protect "normal" disk drives from starvation caused by the hyper-responsive flash drives.
- **Virtual Provisioning:** This one's probably obvious, but with the cost of flash drives, you really want to buy as little of it as possible, so thin provisioning is almost imperative to maximize utilization. Over-provisioning allows for future growth with a minimum of hassle - just add another group of flash drives to the pool before expanding your databases.
- **Switched Infrastructure:** In addition to the inherent fault-isolation and reliability improvements afforded by the new point-to-point DMX-4 back-end, it also serves to minimize the latency overhead for the flash drives. While the overhead of an arbitrated loop is minimal and practically undetectable for a regular hard drive, even a little latency is noticeable with flash drives. And if/when future enterprise-class flash drives hit the market with a SATA interface instead of Fibre Channel; the DMX-4 is ready.
- **Asynchronous Replication:** while clearly justifiable on the merits of being able to replicate data a significantly longer distance than possible with synchronous replication, asynchronous replication is expected to be the preferred method of protecting data stored on flash drives, for a very simple reason: after you've paid to attain minimal response times, the last thing you're probably going to do is add another millisecond or two of transmission time to your writes.
- **SRDF/S Response Time improvements:** But if your application does require synchronous replication, you'll want the fastest possible response times, so the enhancements made in the latest microcode levels could well make a lot of difference for flash drives.
- **Write Folding:** With effective write performance that pretty much matches read latencies, there's not a lot to be gained performance wise - by caching writes to the disk. But, buffering writes can help reduce the wear and tear on the drive. The longer DMX can delay sending writes to the drive, the higher the probability that a subsequent write supersedes an earlier one. This "write folding" is a key foundation of reducing the amount of data SRDF/A has to transmit, and it will have a similar effect on reducing the amount of "writes" a flash drive has to deal with.
- **Minimized Code Paths** -- when the source of a read is a flash drive, any code to determine whether read-optimization algorithms should be engaged should be skipped. The code path must also be minimized so as to be able to handle flash drive response times – microseconds versus milliseconds

- **Turn Off Sequential Prefetch** -- knowing that the flash drive itself has already fetched the "rest of the track" into its SDRAM buffer should it be needed.
- **Turn Off I/O Re-ordering** -- since there's no rotational latency or seek times to optimize with flash drives.
- **Rebuild all the drives at once** – in the rare event of a flash drive failure(s), all the drives are rebuilt at once instead of sequentially, since there's no real performance difference or overhead for totally random vs. sequential requests.
- **Ensure flash drives don't starve the hard drives** – Use DMX's Priority Controls Feature logic to ensure that "lesser drives" aren't starved.

And, third, let's carefully read EMC's deployment advice.

EMC Deployment Advice for Flash Drives in DMX: The greatest improvements will be seen with higher cache read-miss workloads, owing to the lack of rotational and seek latency in flash drives. Flash drives are most beneficial with random read misses (RRM). If the RRM percentage is low, flash drives may show less benefit, since writes and sequential reads/writes already leverage DMX cache to achieve the lowest possible response times. For example, if the read hit percentage is high (> 95%) as compared to read misses, such as in workloads of decision support systems (DSS) or streaming media, improvements provided by flash drives will not likely be enough to be cost-effective.

Action Item: Users should identify suitable applications and consider replacing 10-15 hard drives with one flash drive to save power, cooling and drastically improve performance. Users should also focus on the cost per I/O rather than the cost per gigabyte. EMC should provide configuration planning tools that help identify applications and balance the ratio of flash drives to hard drives.