



Bertrand Dufasne
Kerstin Blum
Uwe Dubberke

DS8000: Introducing Solid State Drives

Solid State Drives (SSDs) are high-IOPS class enterprise storage devices targeted at business critical production applications that can benefit from high level of fast-access storage.

In addition to better IOPS performance, Solid State Drives offer a number of potential benefits over electromechanical Hard Disk Drives, including better reliability, lower power consumption, less heat generation, and lower acoustical noise.

This paper explores the characteristics of Solid State Drives, available for the DS8000 Storage Systems. This paper was written for the DS8100 and DS8300 systems with microcode release 4.2.

Solid State Drives overview

Besides an ever increasing demand for more storage capacity, many businesses constantly require faster storage performance for their high-end, business critical applications.

The hard disk drive technology has improved dramatically over the years, sustaining more IOPS and increased throughput while providing higher storage density at a lower price. However, because of its moving and spinning parts, there are inherent physical limitations to the response times (access time and latency) that an HDD can achieve.

These limitations and the continuing demand for better response times and more throughput have created the need for a different storage technology that avoids seeks and rotational delays. This technology, known as Solid State Drives, can provide access times measured in microseconds.

Solid State Drives use semiconductor devices (solid state memory) to store data and have no moving parts. SSDs are in fact not new and have existed for a while. They were initially based on Dynamic random access memory (DRAM) and often referred to as RAM-disks. Although they had very good performance, they were also extremely expensive and most businesses could not justify the cost.

The technology has evolved and SSDs are now based on non-volatile flash memory, as is used in USB flash drives and camera memory cards. Flash memory is much cheaper than DRAM although still more costly per GB than high-end HDDs. The technology is still improving and the cost of SSDs continues to drop, allowing them to compete more strongly with electromechanical disks.

NAND-flash based SSD

A flash memory stores the information in an array of flash cells. Each cell consists of a single floating-gate transistor (as depicted in Figure 1) that can store electrons.

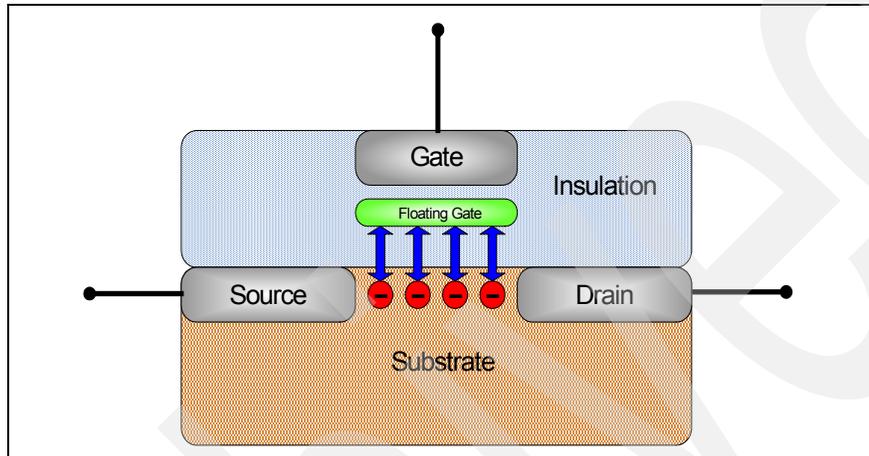


Figure 1 Flash transistor cell

Placing a charge (electrons) on the floating-gate is called programming or writing, whereas removing the charge from the floating-gate is called erasing.

The current commercial NAND-flash has two varieties of flash cells:

- ▶ **Single Level Cell Flash (SLC-Flash)**

SLC-Flash technology manipulates the charge on the floating gate of the flash transistor cell to allow representation of two (voltage) states, which translates to a single bit per cell. The bit value is a 0 (written state) or a 1 (erased state),

A Single Level Cell flash typically allows about 100,000 writes per cell. The number of writes to the same cell is limited because each write operation wears out the insulation of the floating gate.

- ▶ **Multi Level Cell Flash (MLC-Flash)**

MLC-Flash is designed to allow a more precise amount of charge on the floating gate of the transistor to represent four different states, thereby translating to two bits of information per cell and therefore a higher bit density is possible.

A Multi Level Cell only allows about 10,000 writes per cell and the MLC-Flash.

In summary, although an MLC-Flash can store more information, the lifetime of the SLC-Flash is about ten times higher than the MLC-Flash. In addition the MLC-Flash is slower in writes per cell than the SLC-Flash. For those reasons, the Solid State Drives available for the DS8000® use a NAND-Flash with Single Level Cell (SLC) technology.

Note: Solid State Drives available for the DS8000 use a NAND-Flash with Single Level Cell (SLC) technology.

Data stored in the SLC-flash remains valid for about ten years, without power.

This type of flash memory can only be electronically erased or reprogrammed in large blocks (as opposed to small blocks, such as a byte of data, for instance).

The NAND-Flash (the logical “Not And” operation) as used in the DS8000 is working page-based (sector-based) and block-based. A page typically has a size of 512, 2048, 4096, or 8192 bytes. Each page has also a spare area of 64 bytes that is reserved for Error Correction Code, or other internal operations information. Pages are then grouped into blocks. A page is the smallest unit that can be read or written.

Figure 2 shows a NAND-Flash diagram.

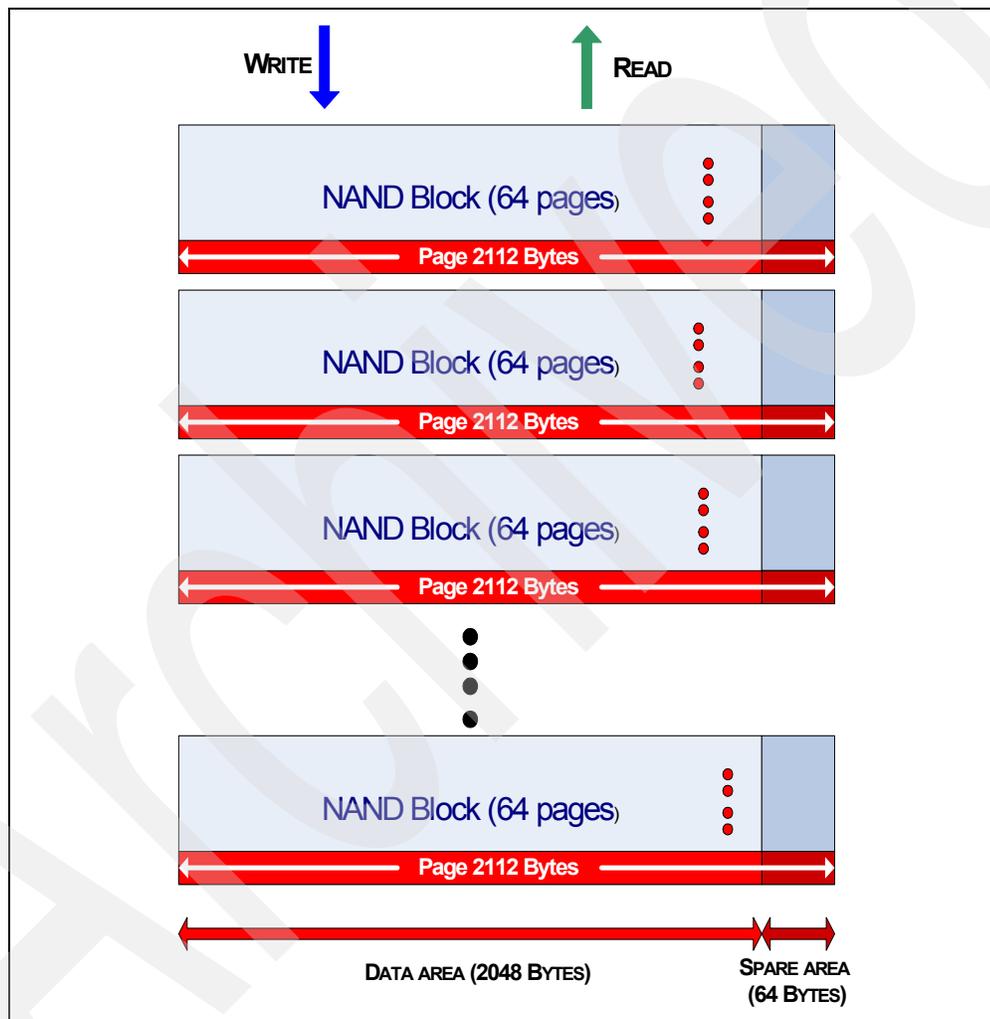


Figure 2 Example of a NAND-Flash memory diagram

Pages that contain data cannot be directly overwritten with new data, they must be erased first before they can be reused. The page itself cannot be erased because of the grouping into blocks, so the block needs to be erased (erasure procedure takes typically 1.5 to 2.0 ms). For each write operation, a copy of the block is needed and is performed at that time. The block that was used before is blocked until it is erased.

Figure 3 shows a write operation and the erase process after the write is completed. When data in a block has to be changed (overwritten), the change is not written in the original block (Block A). Instead the new changed data is copied in a new block (Block B). Then the original

block (Block A) is erased as mentioned before. After this erasure, the block (Block A) is available for use again.

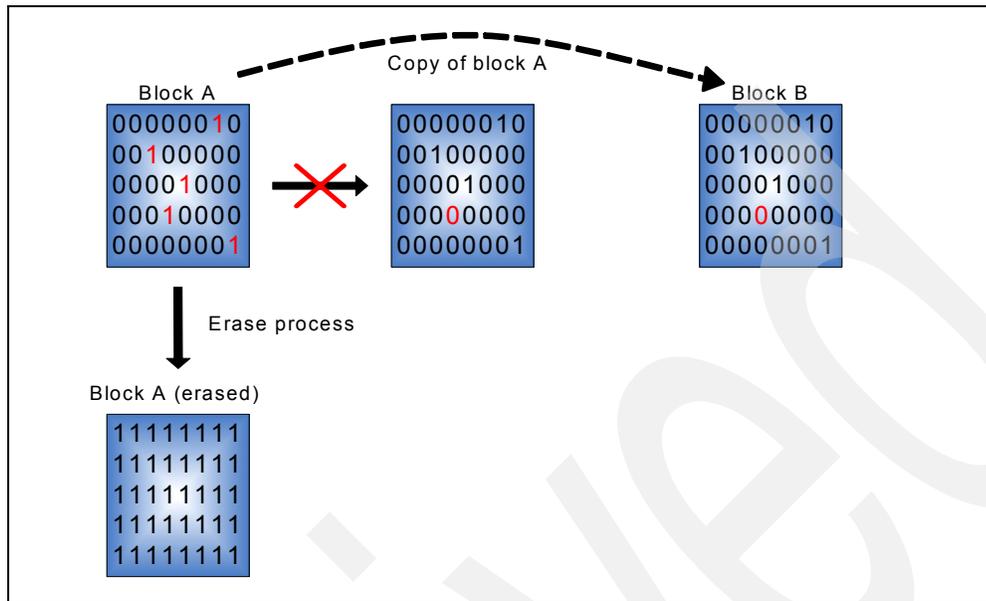


Figure 3 Overwriting and deleting flash blocks

SSD endurance

The life of the flash is limited by the number of write/erase operations that can be performed on the flash. To extend the lifetime of the drive and to ensure integrity of the data on the drive, SSDs as used in the DS8000 have built-in dynamic/static Wear-Leveling and Bad-Block mapping algorithms. This is further complemented by Over-Provisioning and Error Detection Code / Error Correction Code algorithms to ensure data reliability.

These algorithms are implemented in various electronic components of the drive as functionally depicted in Figure 4. The algorithms are set at manufacturing and cannot be tuned by the user.

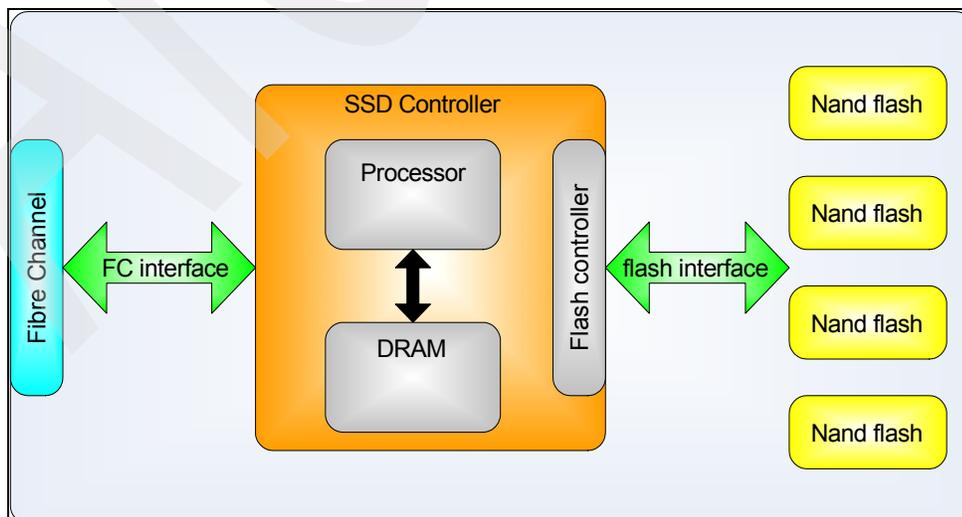


Figure 4 Functional diagram

► **Wear-Leveling algorithm**

The most common place to implement the Wear-Leveling is in the NAND-Flash controller. The goal of Wear-Leveling is to ensure that the same memory blocks are not accessed too often. With this mechanism the flash-controller distributes the erase and write cycles across all the flash memory blocks.

In fact, there are two types of Wear-Leveling algorithms that come into play: dynamic Wear-Leveling and the static Wear-Leveling. With the dynamic Wear-Leveling the flash-controller spreads out the write access over the free or released blocks. As a result of this, the blocks which are used more often are worn out faster than the others. Therefore, it is complemented with a static Wear-Leveling algorithm that moves data not read or changed so often, to blocks that are already strongly worn out.

In summary, Wear-Leveling eliminates excessive writes to the same physical flash memory location.

Note that the use of DRAM cache (as depicted in Figure 4) also helps minimize the effects of any write “hot-spot.”

► **Bad-Block algorithm**

The Bad-Block algorithm detects faulty blocks during operations and flag them. These flagged blocks are excluded from the rotation and replaced with good blocks, so that the data does not go into bad blocks.

► **Over-Provisioning algorithm**

In combination with the Wear-Leveling algorithm, Over-Provisioning is used to further alleviate the write endurance issues. This means that an SSD really has up to 75% more capacity than actually usable. This extra capacity is used by the Wear-Leveling process to extend the lifetime of the drive. In other words, a drive of a given usable capacity could potentially offer more by reducing the Over-Provisioning, but this would compromise the lifetime of the drive.

► **Error Detection Code and Error Correcting Code algorithm**

The Error Detection Code and Error Correcting Code (EDC/ECC) algorithm maintains data reliability by allowing single or multiple bit corrections to the data stored. If the data is corrupted due to aging or during the programming process, the EDC/ECC compensates for the errors to ensure the delivery of accurate data to the host application.

SSD advantages

This section summarizes the major advantages offered by SSDs when compared to spinning, magnetic hard disk drives. As already mentioned, the simple fact that there are no moving parts (disk platters, magnetic heads, motor) in an SSD results in:

► **Faster data access and throughput**

The access to the data with an SSD is faster because again, there is no read/write head to move and no magnetic platters need to spin up (no latency). On an SSD the data can be read almost immediately.

The SSD has up to 100x more throughput and 10x better response time than 15K RPM spinning disks.

► **Better reliability**

Again, the lack of moving and rotating parts almost eliminates the risk of mechanical failure. SSDs have the ability to tolerate extreme shocks, higher altitude, vibration, and extremes of temperature. However, they can still fail and must be RAID protected like traditional drives.

► **Less power consumption**

Because there is no power for the motor required to spin up the magnetic platters and to move the heads, the drive uses less energy than a traditional hard disk drive. Each SSD uses about half of the power of a 15K RPM HDD.

The savings can be substantial if a few SSDs can replace many HDDs to deliver the same performance. This is particularly true for applications that were forced, for performance reasons, to use large quantities of HDDs to get as many spindles as they could, while only using a small portion of the disk capacity.

Besides power consumption savings, the overall system would weigh much less because SSDs are already much lighter than HDDs.

SSD implementation in DS8000

Starting with Licensed Machine Code 5.40.20.xx, the IBM® DS8000 series provides support for Solid State Drives (SSDs). Initially drives are available in 73 GB and 146 GB capacities and are Single Level Cell-Flash (SLC-Flash) drives with dynamic and static Wear-Leveling, Bad-Block mapping, and Error Detection Code/Error Correction Code (EDC/ECC).

These high-IOPS enterprise class storage devices are classified as Tier 0 storage according to the following tiers of enterprise storage systems:

Tier 0:	Enterprise SSD (Fibre Channel SSD, SAS SSD - serial attached SCSI SSD)
Tier 1:	Enterprise Hard Disk Drive - HDD (Fibre Channel HDD, SAS HDD)
Tier 2:	SATA SSD and SATA HDD (SATA - Serial ATA)
Tier 3:	Tape

The Solid State Drives are optional to the DS8000 series and are available with feature numbers 6xxx.

SSDs have the same form factor as the regular spinning drives available for the DS8000 and they come pre-installed in a regular DS8000 disk enclosure, half populated with 8 drives.

Initial configuration recommendations

Use the following guidance for initially configuring a DS8000 with SSDs:

- For new (manufacturing order) DS8000 systems to have SSDs installed, the system must be ordered with RPQ 8S1027. SSDs are available, also via RPQ on field systems, to ensure proper placement for best performance.
- We recommend that you configure 16 SSDs per Device Adapter pair, with a maximum of 32. The reason is to maintain adequate performance without saturating the DA. The SSDs supported in the DS8000 are so fast that the DA can become the performance bottleneck on some random workloads. We therefore recommend that you use just 16 SSDs per DA pair to get the maximum performance from each SSD. (The DS8000 supports up to 8 DA pairs.)
- Considering that the DS8000 can have a maximum of 8 Device Adapter pairs installed, the current maximum number of SSDs in a DS8000 should be limited to 128.
- SSDs can be intermixed with traditional drives, either on the same DA pair, or on separate DA pairs within the DS8000. We recommend not intermixing SSD and HDD on the same DA pair, as this would negatively impact the performance benefit of having SSDs
- When intermixed with spinning drives (HDDs), the total number of drives (SSDs and HDDs combined) is limited to 256 in model 931, or 512 in models 932 and 9B2, which limits the

expansion frame configurations to 3 Frames (one Base Frame and two Expansion Frames).

- ▶ IBM recommends RAID 5 for SSD drives. RAID 6 and RAID 10 are not recommended and initially not supported on the DS8000 for SSD drives. RAID 6 consumes too much expensive flash space for little benefit (and lower performance), and RAID 10 provides little performance improvement because SSDs are already so responsive to random reads.
- ▶ SSDs are not available as Standby Capacity on Demand disk drives.

Figure 5 shows the placement of drive enclosures in a DS8000 system fully populated with SSDs.

Figure 6 shows a machine with SSDs and HDDs. This is a sample configuration for best performance. Other configurations are possible.

Note: Other configurations can be considered on an RPQ basis.

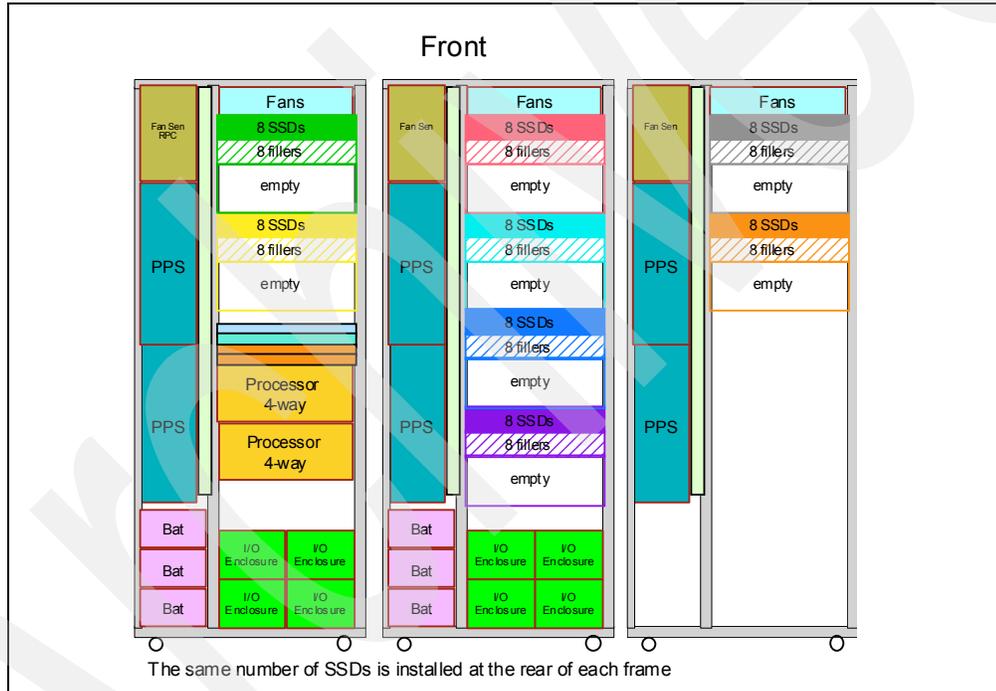


Figure 5 DS8000 fully populated with SSDs

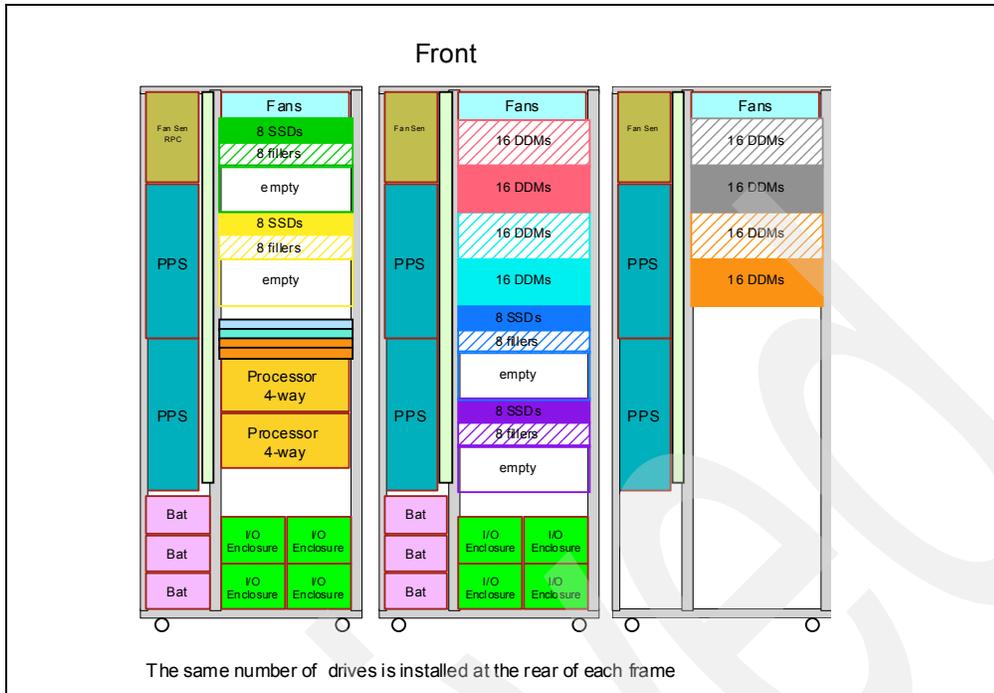


Figure 6 Machine mixed with SSDs and HDDs

Note: The customer should avoid intermixing extents of SSDs with other drives. Remember this when using storage pool striping-extent rotation or multi rank extent pool.

DS GUI and DSCLI changes

Creating arrays, ranks, extent pools, and volumes based on SSDs is the same process as with regular disks. Some minor changes were made to some DS GUI panels and DS CLI commands output to reflect the presence of SSDs in the system. Next, we provide some illustrations.

Note, however, that even though the process is unchanged, there are special considerations and best practices to follow when creating arrays, ranks, and volumes if you want to keep the benefits of SSDs. These considerations and best practices are discussed in “Considerations for DS8000 with SSDs” on page 11.

Figure 7 shows an example of a Create Array panel on a DS8000 with SSDs. You can see that the drives are recognized as SSDs, and are characterized with an RPM value of 65K (although there is no rotating part); This is just a way to clearly distinguish them from other regular disks.

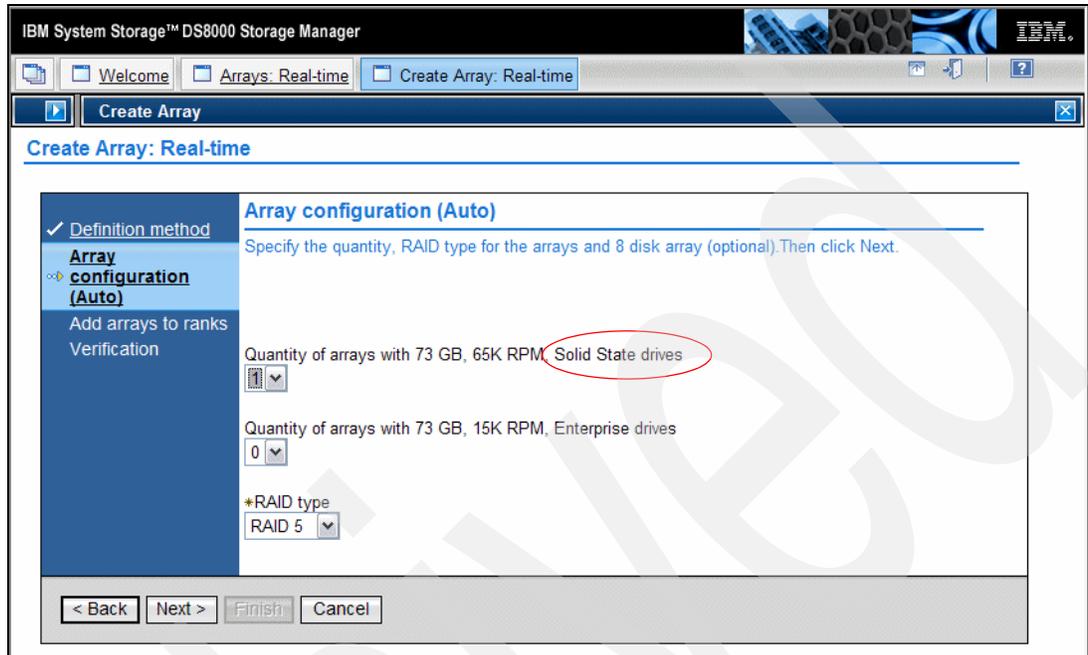


Figure 7 Create Array - Array configuration window

The possible DDM types are:

- ▶ ENT = Enterprise drives, represents high speed Fibre Channel disk drives
- ▶ NL = Nearline™ drives, represents ATA (FATA) disk drives
- ▶ SATA = Specifies high capacity SATA disk drives.
- ▶ SSD = Solid State Drives

RAID type: The supported / available RAID types are listed. For SSDs, RAID5 only is supported.

Figure 8 shows the newly created arrays configured with SSDs. The DDM Class column indicates *Solid State* and the DDM RPM column reflects the *65K* fictitious value.

Select	Number	Data State	RAID	Status	Rank	DA Pair	DDM GB	DDM RPM (K)	DDM Class	Encryption Group
<input type="checkbox"/>	A0	Normal	RAID 5 (6+P+S)	Unassigned		2	73	65 Solid State	None	
<input type="checkbox"/>	A1	Normal	RAID 5 (6+P+S)	Assigned	R0	2	73	65 Solid State	None	

Figure 8 Create Array - Created arrays

Figure 9 shows the Create New Extent Pools dialog, with an example of SSDs in the Drive Class field.

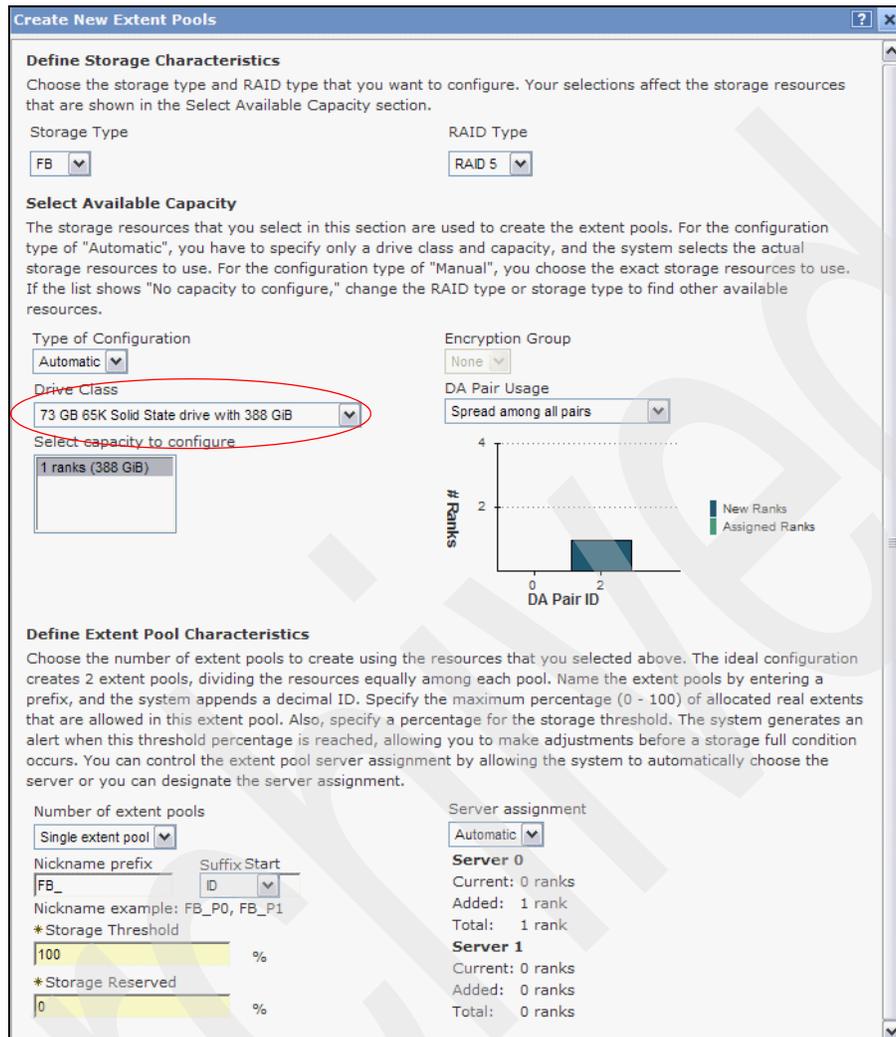


Figure 9 Create New Extent Pools - Define Characteristics window

The `lsarraysite` command, illustrated in Example 1, indicates the presence of SSDs with the new SSD value in the Array Disk class column and also a fictitious RPM of 65K in the rpm column to create the arrays.

Example 1 Listing array sites

```

dscli> lsarraysite -dev IBM.2107-75DG621 -state unassigned -l
Date/Time: March 30, 2009 5:28:07 PM CEST IBM DSCLI Version: 5.4.20.27 DS: IBM.2107-75DG621
arsite DA Pair dkap (10^9B) diskrpm State      Array diskclass encrypt
-----
S5      4          146.0  15000 Unassigned -    ENT      unsupported
S6      4          146.0  15000 Unassigned -    ENT      unsupported
S7      4          146.0  65000 Unassigned -    SSD      unsupported
S8      4          146.0  65000 Unassigned -    SSD      unsupported
S9      5          146.0  15000 Unassigned -    ENT      unsupported
S10     5          146.0  15000 Unassigned -    ENT      unsupported
S11     5          146.0  65000 Unassigned -    SSD      unsupported
S12     5          146.0  65000 Unassigned -    SSD      unsupported
S13     6          146.0  15000 Unassigned -    ENT      unsupported
    
```

S14	6	146.0	15000	Unassigned	-	ENT	unsupported
S15	6	146.0	65000	Unassigned	-	SSD	unsupported
S16	6	146.0	65000	Unassigned	-	SSD	unsupported
S17	7	146.0	15000	Unassigned	-	ENT	unsupported
S18	7	146.0	15000	Unassigned	-	ENT	unsupported
S19	7	146.0	65000	Unassigned	-	SSD	unsupported
S20	7	146.0	65000	Unassigned	-	SSD	unsupported

Example 2 illustrates the `lsarray` command output when SSDs are present.

Example 2 Verifying the array

```

dscli> lsarray -l
Date/Time: March 30, 2009 5:35:13 PM CEST IBM DSCLI Version: 5.4.20.27 DS: IBM.2107-75DG621
Array State      Data RAIDtype  arsite Rank DA Pair DDMcap (10^9B) diskclass encrypt
=====
A0  Assigned  Normal 5 (6+P+S) S3      R0    0          146.0 SSD      unsupported
A1  Assigned  Normal 5 (6+P+S) S23     R1    2          146.0 SSD      unsupported
A2  Assigned  Normal 5 (6+P+S) S4      R2    0          146.0 SSD      unsupported
A3  Assigned  Normal 5 (6+P+S) S24     R3    2          146.0 SSD      unsupported
A4  Unassigned Normal 5 (6+P+S) S7      -     4          146.0 SSD      unsupported
A5  Unassigned Normal 5 (6+P+S) S8      -     4          146.0 SSD      unsupported

```

Considerations for DS8000 with SSDs

This section reviews special considerations and usage recommendations when SSDs are used in a DS8000:

► **Copy Services:**

Copy Services operations are not specifically affected by SSD. Our only recommendation (as is the case with HDDs as well) is not to use a slower device as the target of a copy Services function.

► **Sparing and rebuild time:**

The sparing algorithm is unchanged for DS8000 equipped with SSDs. The rebuild time for an SSD is, however, much faster than with regular HDDs of the same capacity. For an idle 6+P SSD array, we have measured a rebuild time that is around 10-15% faster than a 6+P array with 15K RPM HDDs.

► **SSD usage:**

SSDs cost more per unit of storage than HDDs but the gap in price should narrow over time. Also SSDs offer only relatively small capacity per drive in comparison to HDDs. You thus want to make sure that they are used (reserved) for the most business critical applications and that they are used under conditions that can guarantee optimum performance.

Use the following guidelines when deciding:

- SSDs should currently remain targeted at applications with high access workloads (IOPS). Applications with high IO rates and poor cache hit ratios are ideal candidates for SSDs while applications with a good cache hit ratios or low IO rates should just remain on spinning drives. Also, sequential workloads (as opposed to random) should stay on spinning disks.
- Given the current high cost per GB for SSDs, use a mix of SSDs and HDDs (not on the same DA) in the DS8000, such that data from less critical applications can be stored on HDDs (you can use FC and SATA drives).
- There is no additional benefit to using SSDs with remote copy services. In general, if SSDs are used for remote copy source volumes they should also be used for the remote copy target volumes. If not, then the secondary HDD based targets may become the bottleneck in the system. This is especially problematic for synchronous replication (Metro Mirror) as delays will be pushed back to applications. For asynchronous replication (Global Mirror) you may see an increase in recovery point objective (RPO) if the throughput to the primary far exceeds the secondary capability. The important thing here is to do the appropriate capacity planning before placing SSDs into a remote copy environment.
- SSDs may be used with FlashCopy either for source or target volumes. If SSDs are used for source volumes while HDDs are used for the secondary, it is a good idea to do the FlashCopy with background copy and during a period when the write rate to source volumes does not exceed the capability of the targets. Additionally, although SSDs would likely perform well as a Space Efficient FlashCopy (SEFLC) target repository, it does not fit with the basic premise of the technology. SEFLC is intended for cost reduction by not fully provisioning the FlashCopy target space. Since SSDs are costly it is likely that fully provisioned HDD space would be less expensive and perform almost as well.
- Use 16 SSDs per DA pair.

If you run an I/O driver to a single SSD that is set up for direct attach, it can do up to 30k IOPS. The DS8300 can do about 20k IOPS across seven or eight disks on one device adapter. This is close to double on a device adapter pair. The throughput can scale up as more device adapter pairs are used for SSDs. Even the most active production data cannot exploit 30k IOPS on a 73 MB or 146 GB disk.

In other words, the SSDs used in the DS8000 are so fast that the Device Adapter becomes the performance bottleneck on random workloads. For sequential workloads the DA was already the bottleneck with traditional HDDs (hence using SSDs for sequential IO would not be a good investment).

- Use optimum host attachment options.

The functionality provided by a storage server such as the DS8000 through its virtualization layers that make the particularities of disk hardware transparent to applications is critical for the usability of the system. However, this functionality adds to the response time you would normally get from an SSD. It is important not to further impact the response time that you use the host attachment options that guarantee the best performance. In particular, consider using zHPF (High Performance FICON®) when attaching to System z® as well as enabling HyperPAV.

Note: You need sufficient amount of SSD to gain the best performance benefit. Indeed, applications such as databases can have high access rates, which would make them good candidates for deployment on SSD. However, the same database might require a large storage capacity. In this case you thus need a sufficient amount of SSD capacity to get the best performance improvement.

- For specific information on SSD usage with DB2®, refer to the IBM RedGuide, *Ready to Access DB2 for z/OS Data on Solid-State Drives*, REDP-4537.

For more SSD performance related information, refer to the following white paper: *IBM System Storage DS8000 with SSDs: An In-Depth Look at SSD Performance in the DS8000*, at:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101466>

To give you an indication of the relative SSD performance compared to HDD, we include here two charts that you can also find in the foregoing publication.

The results in Figure 10 were measured by a DB2 I/O benchmark. They show random 4KB read throughput and response times. SSD response times are very low across the curve. They are lower than the minimum HDD response time for all data points.

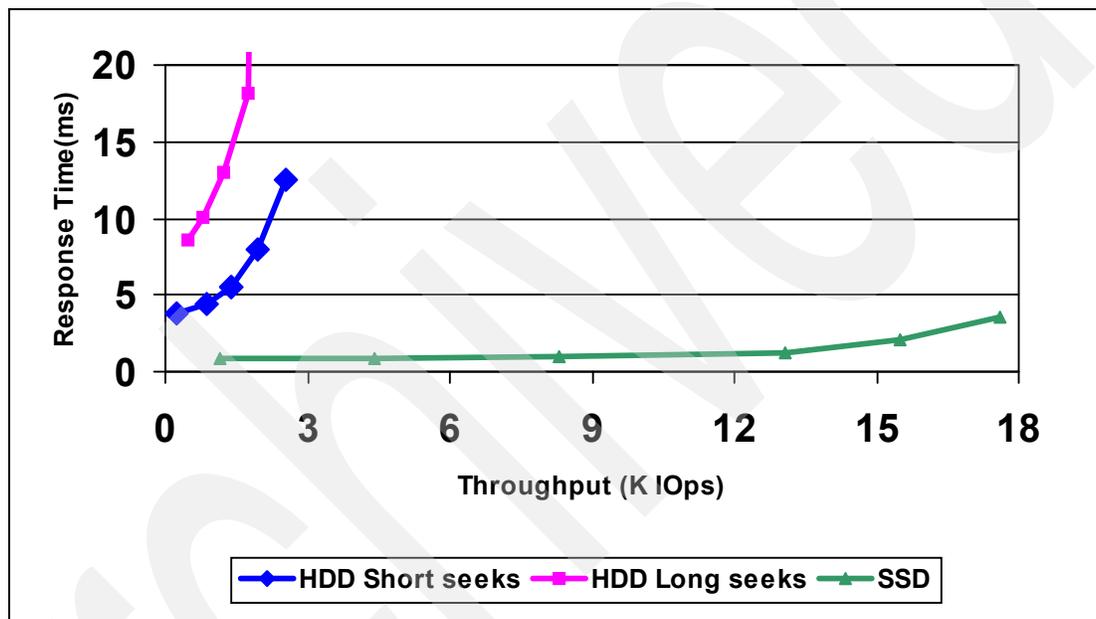


Figure 10 DB2 on CKD Random Read Throughput/Response Time Curve

Figure 11 shows that SSDs provide about the same improvement on random writes as they do on random reads. Both the read and write SSD tests do about 20K IOPS to the SSD rank

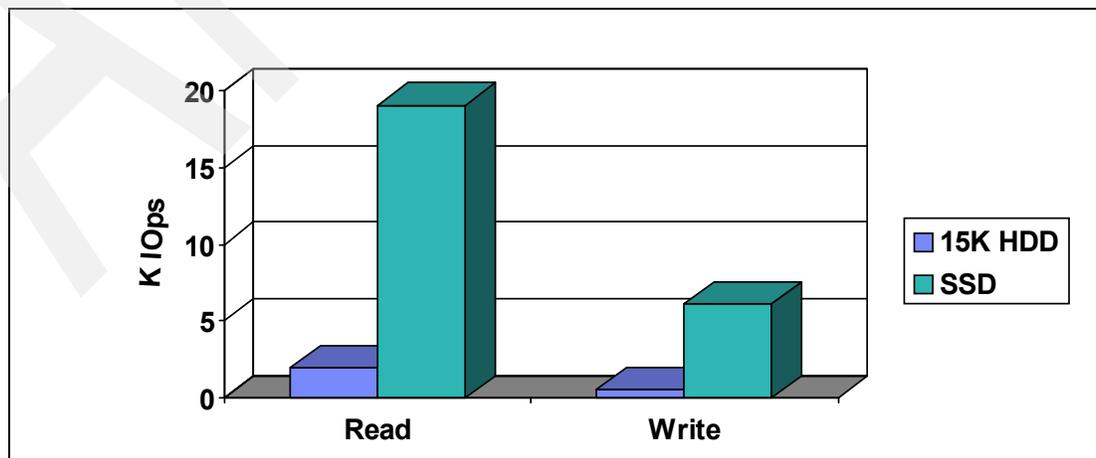


Figure 11 Open 4KB random IO: SSD vs HDD on one RAID5 rank

Finding SSD candidate workloads

In this section we briefly describe tools that can be used to identify workloads (applications) that would be good candidates for placement on SSD.

More details on z/OS® SSD instrumentation and tooling can be found in the article, *Stop spinning your storage wheels: z/OS Support for solid state drives in the DS8000 storage subsystem in z/OS*, Hot Topics Newsletter, Issue 20 at:

<http://publibz.boulder.ibm.com/epubs/pdf/e0z2n191.pdf>

DFSMS

Data Facility Storage Management Subsystem (DFSMS) provides additional I/O statistics in the SMF records that can help you identify good candidates for placement on SSDs.

System z provides a breakdown of the time spent executing I/O operations. The times are stored into the I/O measurement word whenever an I/O interrupt occurs. One of the measurements surfaced is the device disconnect time (DISC). DISC is a measure of the time spent resolving cache misses for READ operations. It also measures disconnect time on WRITES to synchronous Metro Mirror devices. SSDs are good candidates for applications experiencing many cache misses (such as with random reads). However, SSD technology does not reduce the elapsed times because of the use of synchronous replication technologies

DFSMS collects disconnect time statistics that are summarized and reported at the data set level. DFSMS has been enhanced to separate DISC time for READ operations from WRITE operations. This should help in identifying application workloads that would benefit the most from SSDs.

To benefit from the enhancements listed in this section, z/OS V1R8 and above is required with following APARs:

- ▶ OA25559 Dev Support - Enabling APAR
- ▶ OA25603 CMM
- ▶ OA25842 SMS
- ▶ OA25688 SMF Macro
- ▶ OA25724 – DEVSERV
- ▶ OA25590/OA25596 – Media Manager

SMF record updates

New I/O statistics in the System Management Facilities (SMF) records are as follows:

- ▶ SMF 42 Subtype 6:

This records DASD data set level I/O statistics. I/O response and service time components are recorded in multiples of 128 micro-seconds for the Data Set I/O Statistics section:

- S42DSRDD (Average disconnect time for reads)
- S42DSRDT (Total number of read operations)

- ▶ SMF 74 Subtype 5

The DS8000 provides the ability to obtain cache statistics for every volume in the storage subsystem. These measurements include the count of the number of operations from DASD cache to the back-end storage, the number of random operations, the number of sequential reads and sequential writes, the time to execute those operations, and the number of bytes transferred. These statistics are placed in the SMF 74 subtype 5 record.

For more information, refer to the publication *MVS System Management Facilities (SMF)*, SA22-7630.

Other DFSMS changes

Changes have also been made to the DEVSERV command and DEVTYPE macro for SSD support:

► DEVSERV QDASD

The DEVSERV QDASD command can be used to display the number of configured cylinders for a device or range of devices, including device model 3390 model A. The first character of the UNIT number represents the subchannel set number. When the ATTR parameter is added, DEVSERV QDASD displays device attributes, including if the device is a Solid State Drive, for example:

```
ds qd,d300,attr
IEE459I 16.59.01 DEVSERV QDASD 613          C
UNIT VOLSER SCUTYPE DEVTYPE          CYL  SSID SCU-SERIAL DEV-SERIAL EFC
ATTRIBUTE/FEATURE          YES/NO    ATTRIBUTE/FEATURE          YES/NO
D300 TK9085 2107921 2107900          65520 2401 0175-02411 0175-02411 *OK
SOLID STATE DRIVES          Y          ENCRYPTION          N
```

► DEVTYPE macro

INFO=DASD

- Solid State Drives (byte 8, bit 6)
- Data Encrypted device (byte 8, bit 7)

FLASHDA SAS

The FLASHDA is a tool based on SAS software to manage the transition to Solid State Drives (SSDs). The tool provides DASD and data set usage reports using SAS code to analyze SMF 42 subtype 6 and SMF 74 subtype 5 records explained in the previous section to help identify volumes and data sets that are good candidates to reside on SSD.

For more information, see:

<http://www.ibm.com/systems/z/os/zos/downloads/flashda.html>

The FLASHDA user guide is available at:

<http://publibz.boulder.ibm.com/zoslib/pdf/flashda.pdf>

Omegamon XE

OMEGAMON® XE is a suite of products that monitors and manages system and network applications on a variety of operating systems. These products allow you to monitor the availability and performance of systems and resources in an enterprise system from one or more designated workstations, and provide reports that can be used to track trends and troubleshoot problems. Reports are displayed on the user interface and can be modified to include historical data if historical data collection is configured.

Migrating from HDD to SSD

When you have identified data that would benefit from placement on SSD, you can plan migrating data from HDD to SSD. The following tools and utilities can help.

z/OS Dataset Mobility Facility

The z/OS Dataset Mobility Facility (zDMF), previously known as Logical Data Migration Facility (LDMF), is a host-based (mainframe) software that can migrate data sets non-disruptively across virtually all the major hardware vendors and between different disk capacities, including EAV, and provide proven compatibility with IBM System z. Note that an application restart might be required to complete the migration process if the migrated data sets are persistently held by the application.

Transparent Data Migration Facility

Clients that need to migrate data without disrupting business applications, regardless of platform, distance or vendor hardware can use the Transparent Data Migration Facility (TDMF®). This is a host-based unified data migration solution for both open systems and mainframe environments.

Note: For more information about zDMF and TDMF migration, check the IBM Redbooks® publication, *Migrating to IBM System Storage DS8000*, SG24-7432.

TDMF's key unique differentiator is its ability to non-disruptively migrate data while applications are still online for business use. In addition, TDMF balances I/O activity to optimize application performance during data migrations.

DB2 Online Reorg

The DB2 Utility for reorganizing data is called REORG. DB2 Online Reorg moves table spaces from one volume to another.

DFSMSdss and DFSMShsm

DFSMSdss with IBM FlashCopy or DFSMShsm moves data sets with an application restart.

DFSMS policy-based storage management

DFSMS maintains a table for each supported device that provides the millisecond response time (MSR) values for random access read and writes (also known as direct reads and writes), and sequential reads and writes. Separate MSR values are provided for cache access and disk access (referred to as native). SMS takes the MSR values (0 to 999) provided by each device table and inserts each into one of 28 discrete bands.

When a data set allocation request is made, SMS attempts to match the requesting MSR of the storage class associated with the data set with one of the 28 MSR bands.

As part of the DFSMS support, a new device performance capability table exists for volumes backed by Solid State Drives. A user who wants to favor SSD selection over non-SSD can assign a storage class with a direct MSR value of 1 and a direct bias of R (and not specify any values for sequential MSR or sequential bias).

For example, to direct storage allocation away from SSD, specify the following and do not specify values for sequential MSR or sequential bias:

DIRECT MSR = 10, DIRECT BIAS = R

Storage modeling and analysis tools

Disk Magic™ is a performance modeling tool that can help you predict the expected performance of a DS8000 with a specific configuration running a specific workload. In the short term, the tool has been enhanced to model the performance of SSDs. It is possible to evaluate placement of specific workloads on SSDs while other workloads remain on HDDs.

Future enhancements are planned to help the user determine which workloads or volumes are best to move from HDDs to SSDs. As with any modeling tool, the quality of the output depends on the accuracy of the input data describing the workload characteristics. Customers should contact their IBM Storage Sales Specialist or IBM Business Partner if they are considering SSDs for their DS8000 and would like to have a performance modeling study.

TotalStorage Productivity Center for Data

IBM TotalStorage® Productivity Center for Data (TPC for Data) is a comprehensive file and capacity management solution for heterogeneous storage environments.

TPC for Data facilitates consolidated data collection and reporting of servers at the disk and file level. Several reports can be produced to help you understand the I/O workload distribution among volumes as well as workload characteristics.

For detailed information on TPC for Data, refer to the IBM Redbooks publication, *Deployment Guide Series: TotalStorage Productivity Center for Data*, SG24-7140.

For DS8000 specific information related to TPC for Data, refer to the IBM Redbooks publication, *DS8000 Performance Monitoring and Tuning*, SG24-7146

The team that wrote this IBM Redpaper publication

This IBM Redpaper publication was produced by a team of specialists from around the world working at the International Technical Support Organization, San Jose Center.

Bert Dufrasne is an IBM Certified Consulting I/T Specialist and Project Leader for System Storage™ disk products at the International Technical Support Organization, San Jose Center. He has worked at IBM in various I/T areas. Bert has written many IBM Redbooks publications and has also developed and taught technical workshops. Before joining the ITSO, he worked for IBM Global Services as an Application Architect in retail, banking, telecommunications, and healthcare industries. He holds a Masters degree in Electrical Engineering from the Polytechnic Faculty of Mons (Belgium).

Kerstin Blum is an accredited Senior Product Service Specialist at the ESCC (European Storage Competence Center) in Mainz, Germany. She joined IBM in 1999 and since then has worked as a Product Field Engineer (last level support) on various ESS generations and DS8000, and focuses on SSA, ISS, DDM, and MES. She was assigned for 18 months as the EMEA field support interface with the ESS/DS8000 development and test teams in Tucson, AZ. She holds a degree in Electrical Engineering, with a specialization in Communications Engineering, from the University of Applied Sciences of Wiesbaden (Germany).

Uwe Dubberke is an IBM Certified Specialist for High End Disk Solutions, working as a field specialist (RDS) in DASD and SAN products in Germany. He joined IBM in 1990 and supported different high end customers as an Account CE. He has also worked as an SE and, since 1999, he supports the EMEA Central Region Hardware Support Center in Mainz as a virtual member. He also has been a virtual member of the SAN Support Group in Mainz since 2005. He holds a degree in Electrical Engineering, with a specialization in communications engineering, from the University of Applied Sciences of Gelsenkirchen, Germany.

Thanks to the following people for their contributions to this project:

John Bynum
IBM US

Cheng-Chung Song, Ben Donie, Lee La Frese, Jeffrey Berger, Leslie Sutton, Ron Knowlden,
IBM US

Brian Sherman
IBM Canada

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document REDP-4522-00 was created or updated on June 13, 2011.



Send us your comments in one of the following ways:

- ▶ Use the online **Contact us** review Redbooks publications form found at:
ibm.com/redbooks
- ▶ Send your comments in an e-mail to:
redbooks@us.ibm.com
- ▶ Mail your comments to:
IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400 U.S.A.



Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

DB2®	OMEGAMON®	System z®
DS8000®	Redbooks®	TDMF®
FICON®	Redbooks (logo)  ®	TotalStorage®
FlashCopy®	Solid®	z/OS®
IBM®	System Storage™	

The following terms are trademarks of other companies:

Disk Magic, and the IntelliMagic logo are trademarks of IntelliMagic BV in the United States, other countries, or both.

Nearline, SLC, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.