Frequently Asked Questions Regarding Storage Configurations

Margaret Crevar and Tony Brown, SAS Institute Inc.

Last Updated: July 2011

OVERVIEW

The SAS Global Forum 2007 paper "Best Practices for Configuring your IO Subsystem for SAS®9 Applications" provides general guidelines for configuring I/O subsystems for your SAS applications. Since that paper was published in 2007, we have received many very good questions from the SAS field that were beyond the original paper, so we have written this paper to address those questions. Because the questions are widely varied, we thought the best way to address them is in a frequently asked questions (FAQ) format. In addition, our goal is continually update this paper as we receive additional questions. To ensure that you have the most recent copy of the paper, be sure to reference the Last Updated date above.

FREQUENTLY ASKED QUESTIONS

What do we mean by file system? A *file system* (is a method of storing and organizing files and where they are placed for storage and retrieval. Some of the most common file systems are NTFS (for Windows), ZFS (for Solaris), JFS2 (AIX), EXT4 (for Linux), XFS (XFS for Linux), JFS (for HP-UX), and VxFS from Veritas (available on many different operating systems).

A file system includes metadata that the operating system uses to identify where to write files. An additional benefit of a file system is that it can exploit a system-level cache to help speed up access to data. Performance is enhanced because the cache accesses frequently used data, thus bypassing the latency that is associated with direct access to the physical disk drive.

In UNIX operating environments, two types of file systems are predominant: indirect, block-based file systems (for example, UFS) and extent-based file systems (for example, VxFS and JFS). The main difference in the two types is the way that the metadata and the data of the file are organized. Given file sizes and access patterns, the two types have different performance characterizations. *Indirect, block-based file systems* perform well for small files. *Extent-based file systems* might perform better on large files, particularly with sequential access, because there are fewer indirections to access the data.

File systems are further classified as journaling and non-journaling file systems. Here again, the choice of file system affects performance. A *journaling file system* logs the metadata operation to a separate intent log, so you can determine the integrity of the file system from the log. A *non-journaling file system* does not have this log, which means that you must read the entire file system to determine its integrity.

In addition to the local file systems mentioned, there is also a file system known as a clustered file system. This type of system manages and synchronizes access to the file across multiple hosts. Therefore, you must use a clustered file system, to share data in a single file system across multiple computers (or operating-system instances). You can also share data in a file system among multiple computers by using NFS or CIFS methodologies. However, these methodologies are not optimal for SAS applications because SAS uses file locks by default when it accesses files. When you use NFS, file locks that are turned on disable caching as well as Read-ahead and Write-behind processing, which, in turn, adversely affects performance unless your version of NFS supports local locking.

On the latest models of virtualized storage arrays, we typically encounter file systems placed across a LUN(s) that are striped across many (and sometimes) all the spindles in the array. This striping enables a single file-system manifestation to be placed physically and logically on a large number of disk spindles to distribute the workload and to increase throughput or input/output operations per second (IOPs) by aggregating the resources of multiple spindles. The physical spindles underneath these LUN stripes are most often pre-configured (and unchangeable) in a RAID5 safety configuration.

What is meant by "heavy I/O load?"? For the purposes of this paper, *heavy I/O* means sequentially reading files whose physical size is either close to or exceeds the amount of physical memory in the computer in size and where I/O rates need to be at least 100 MB/sec or preferably 50-75 MB/sec per CPU core in your system to process the data in the timeframe required by the SAS users. In addition, this term can mean having multiple SAS sessions/users in which the total amount of data that is accessed simultaneously is close to or exceeds the amount of memory in the computer.

What RAID level is optimal for SAS? While SAS can operate on any RAID level, you should choose the safest and most affordable RAID level that satisfies your performance/safety requirements. As mentioned previously, modern disk arrays often come with a limited number of supported RAID configurations. Today's storage-array configurations have a sufficient amount of internal cache alleviate RAID parity penalties and to enable Read and Write grouping for performance efficiency.

A RAID10 (a striped (RAID0) configuration is a well known configuration that is also fully mirrored (RAID1) with regard to SAS file systems. RAID10 offers the ultimate in safety and performance. However, from a pricing perspective, it nearly doubles the number of physical spindles that are required to do the same work as a RAID5 configuration. The performance gains are not significant enough to justify the cost of the additional spindles.

What are the recommended stripe size values for creating LUNs and logical volumes in SAS? SAS typically recommends a minimum value of **64K**, or **128K**, with as large as **256K** when very-heavy sequential I/O processes operate against files that are 100 GB or more in size.

Note: For optimal performance, when you make stripe-size values as large as the recommended values, you must also set the SAS buffer size (via the BUFSIZE= system option) to the same value or to ½ or ¼ on very large SAS file(s). This formula works well as long as the primary access method for the SAS file is sequential. However, if your primary access method for these large SAS files is random, you should not set SAS buffer size to **128K** or **256K**. Instead, keep the size at **32K** or **64K**.

What is the recommended stripe size for logical disk striping? If you use logical volume striping, then the stripe size of the logical volume should be geometrically in tune with the physical-disk stripe size (for example, the same size or a higher multiple). Note that some virtual storage arrays currently regulate the logical stripe size and you cannot change it. Work with your operating system and storage manufacturer to determine what you have.

How many disks are required per file system? This number depends on the storage array that is used, the type of file system, the I/O throughput that is required by the SAS application, and the capacity that is required to house the permanent or the WORK data operations. SAS suggests that a single file system should be able to support at least 100 MB/sec or preferably 50-75 MB/sec per CPU core in your system of I/O throughput. In most RAID5 cases, with the minimum configuration, this means physically striping across at least seven disks as well as across a parity disk. If you are running on a storage system that is "striped-everything" then all LUNs will be striped across all disks. In this case the number of spindles is dictated to you by the storage manufacturer when it is assembled. This may have ramifications discussed later in this paper.

So the storage-array physical and logical configuration may need to be changed to meet the I/O throughput requirements of the SAS application. It is not unusual for SAN administrators to have to employ stripe space across more actual disks than are required for capacity in order to obtain sufficient spindle aggregation to satisfy the necessary I/O throughput.

Does SAS work well with new technology storage? SAS files are just file-system files, so SAS functions on all of the latest storage technologies. Despite this fact, newer types of storage have their pros and cons, especially for the large-block, sequential access patterns that SAS typically employs. The following list explains the different types of physical drives that are available and how SAS works with them. For more information about these drives, see the "Appendix."

- Serial Advanced Technology Attachment (SATA) drives: SATA drives are quite popular because they have a large footprint (for example, 500 GB, 1 TB, or 2 TB per device) and they are relatively cheap. However, their slower spin rate and higher seek time typically do not make them a good fit for SAS, especially for the SAS WORK area.

- Serial Attached SCSI (SAS) drives: SAS drives are a better fit for SAS applications, especially when they are used as internal drives, or in direct-attached storage or low-end storage arrays.

- Dynamic RAM (DRAM) devices: These devices store data in memory and yield as fast as nanosecond-range performance for data access and writing, which works extremely well with current CPU speeds. SAS has had good field performance experience with these devices, but their cost can be hard to justify.

- Solid-State drives (SSD, also referred to as flash drives): Solid-state drives are popular currently as well. Technology with these has improved since they were first introduced and we are seeing very good read and random-write rates to these devices. These changes make these drives very attractive in terms of very high performance for SAS WORK file systems. There are ramifications for sequential write performance explained in the Appendix.

- Network Storage Appliances: These devices are self-contained storage arrays that come preconfigured with a considerable amount of physical disk space. Our experience in working with customers has shown that these appliances are a good fit for smaller, very static application and data profiles. Generally, they do not perform as well with the high volume of I/O of many concurrent (10+) SAS users that demands fast I/O with large volumes of data (files of 25 GB or more). Be aware that you must have enough network bandwidth to support these arrays.

How many spindles (disks drives) are needed to achieve 100 MB/sec I/O throughput in a multiple-user SAS environment? The answer to this question depends on the underlying storage. At a minimum, it requires seven disks with 15MB/sec I/O throughput that are striped together (or eight disks in a 7 + 1 RAID5 configuration).

As stated previously, file systems are generally composed of LUNs that are striped across many physical disks. In this case, throughput and IOPs are garnished by definition. A physical disk stripe aggregates the throughput of all the disks it stripes across, and the IOPs available to the stripe are dictated by the aggregated IOPs rating of the individual disk. Because many LUNs can be striped across the same physical disks, they must divide the IOPs and throughput available. In this case an individual SAS file system shares these resources with other file systems and is not guaranteed to receive what it needs if collective resources of the spindles cannot meet the collective needs of the LUNs. At times, even SANs with LUN stripes across 250 spindles have not performed adequately. It is not just a matter of involving enough spindles; it is also a matter of ensuring that adequate disk resources (in terms of throughput and IOPs) are available for the entire SAN workload that is placed on the spindles.

You should avoid using just a few large disk drives (such as the new SATA disks that come in 500 GB and 1TB sizes) to create your file systems. While a single 1TB SATA drive might be what you need for physical disk space, a very few single physical disks will not give you the necessary I/O throughput rate or IOPs to support your SAS application.

When the SAS workload is represented by many SAS processes performing large sequential I/O, is it best to set the storage array for large sequential I/O by using the I/O throughput measurement or by using an IOPs measurement? Before we get too involved with the answer, we need to explain the primary difference between a throughput orientation and an IOPs orientation. With I/O bandwidth, you assume that there is only one writer or reader thread that is working at a time and that the data that is being read from or written to is in sequential order on disk. When this happens, you do not have to take into account disk latency (that is, from constant disk head movement, finding the data on the disk coupled with the latency that is associated with the rotational speed of the disk).

So, why do we care about latency? When you have multiple SAS processes that are reading and writing large volumes of data at the same time, the odds are not likely that all of the data being read will reside in the same sector of the volume. As a result, the disk head must move around a lot to find the data that is requested. On a single hard drive, this disk seek time becomes a very large factor in the total time required to retrieve the data. As you add hard drives to your storage array, this time might be reduced, but it will never be eliminated.

In addition to the throughput rate we can also use IOPs to help determine if there is an I/O bottleneck on a SAS file system. The IOPs can be obtained from most hardware monitor tools. If this number is close to the known IOPs capacity of the underlying resources for the file system you are using, then we are approaching a computer-resource constraint with the customer's I/O subsystem. This is another way to show IT administrators that their I/O subsystem is stressed; thereby causing the performance issues their SAS users are experiencing.

We are trying to develop a general rule regarding the minimum IOPs that are required for SAS applications. This number is dependent upon not only the storage array layout, but also the SAS buffer size that is used. While determining a general number is difficult, we have found that current SAN systems that employ striped-everything approaches, the IOPs numbers have usually been adequate. As noted above this is dependent upon the entire SAN workload and overall device utilization.

Now that the data stores for SAS customers are getting larger (greater than 500 GB in many cases for single files), is there any additional information to pass on to storage administrators? In the past, we have suggested that when SAS data files become tens of gigabytes in size, you should consider changing the default BUFSIZE= value for these large files to **32K, 64K,** or even **128K**, provided you also can set the stripe size to **64K** or **128K** for the file system that is being used..

However, we now see SAS data files to be hundreds of gigabytes in size. When this happens and these files are accessed by several SAS processes, setting the stripe size to **256K** as well as to setting the BUFSIZE= value for the SAS data file to **128K** or **256K** will help your throughput.

Note: We strongly suggest that you only set the BUFSIZE= value this high for SAS data files that you know are going to be very large (for example, those files greater than 100 GB) and for which the primary access method is sequential. We do not recommend that you globally set the BUFSIZE= value this large because it can introduce performance issues with files that are less than 10 GB.

BUFSIZE does not apply to utility or index files created by SAS. There are other SAS options for these. In addition, certain SAS procedures require you to use their PROC statement parameters to increase the page/buffer size of the files they create.

Are there any tips for maximizing I/O throughput from an operating-system perspective? Yes, but few of those tips have been documented in papers. We have listed them here, by operating system:

- **Windows operating environments:**
    - **Windows 2003 (either 32-bit or 64-bit/all editions**)**—**This operating system can have stability issues when you run any applications with heavy-I/O. SAS is an application that typically has heavy I/O. It is important to be aware of this issue when you deploy SAS on Windows 2003, where I/O is a critical component of the

system requirements. The main symptom is that the SAS job fails with this error: **ERROR: An I/O error has occurred on file <libref.datasetname.DATA>.** More details about this issue are documented in SAS Note 36664, "Potential issues with heavy SAS® I/O workloads on Windows 2003, 32- or 64-bit operating systems." If you encounter this error, we strongly recommend that you upgrade to Windows 2008. As a workaround, you can turn on the SGIO option within SAS or you can use the Windows 2003 **setcache** option to limit the amount of memory that can be used for file cache.

- **Windows 2008 —**This release of Windows resolves the I/O issues discussed in the Windows 2003 section. However, we uncovered another issue where the Windows 2008 system becomes very "sluggish" due to all the available memory being used. To overcome this issue, SAS has created a hotfix for SAS 9.2M3 (problem number 39615). You will need to apply that SAS hotfix and set the Windows environment variable that is discussed in the SAS Technical Support note.

  In extreme situations, you may need to also use SAS tools (SGIO) to force SAS I/O function into a direct I/O operation.

  One other thing to mention is to make sure that any file systems that are created by Windows 2003 are realigned with the recommended Windows 2008 alignment settings in order to avoid performance degradation. More details on the above can be found in the "Configuring and Tuning Guidelines for SAS9 in Microsoft Windows Server 2008" paper.

- **VMware software—**Recent testing done by the SAS Enterprise Excellence Center (EEC) Benchmarking lab shows minimal performance degradation when you add VMware software on top of existing Windows systems (servers and storage). An analysis of customers with poor performance using VMware monitors indicate that it is generally caused by a reduced I/O throughput rate, especially when you are running heavy I/O applications (including both SAS applications and third-party applications). In addition, VMware engineers note that if you use network mapped file systems with its software, you must be prepared for performance degradation in your I/O throughput rates.
- **Linux operating system—**As a result of working with Red Hat on recent SAS EEC benchmarks, we strongly recommend that if you invoke large volumes of I/O, you should only run SAS 9.2 on a 64-bit Linux X64 computer using Red Hat Enterprise Linux 5.3 or higher. We also strongly recommend that you only create XFS file systems. Starting with Red Hat Enterprise Linux 6, we will change our file-system recommendation to either XFS or EXT4 for heavy I/O processing under Linux. Details on how to configure Red Hat 5 and 6 for use with SAS 9.2 can be found at http://support.sas.com/resources/papers/tnote/tnote_performance.html.
- **Solaris operating system—**Starting with Oracle Solaris 10, the file system of choice is ZFS. We have had very good experience with this file system compared to the UFS file systems that Sun Microsystems used with earlier releases of Solaris. A tuning recommendation for Solaris users is to ensure that the ZFS Internal Log (ZIL) file resides on a file system that has separate physical disks from any SAS-related file system. If the ZIL file resides on the same physical disk as the file system, the heavy journaling activity that tends

to occur can cause performance issues. This tip was provided by a Solaris performance expert from Sun Microsystems.

- **AIX operating system—**Over the years, we have worked with IBM to document various AIX tuning tips. These tips are available on the SAS Partners/Partner Directory: White Papers Web page. That page contains links to papers for both AIX 5L, AIX 6 and AIX7. The most important aspects to note here are that JFS2 is the preferred file system, and you should change some of the AIX virtual memory settings to avoid degrading SAS performance that can result from an aggressive file cache daemon. Several SAS customers have experienced over 2 GB/sec of I/O throughput with AIX systems and very large SAN arrays. **Note:** If you are running AIX 6.1, we strongly encourage you to mount the file system that is associated with SAS WORK (SAS temporary files) with the LOG=_NULL_ parameter.
- **HP-UX operating system—**Over the years, we have worked with Hewlett-Packard (HP) to document various tuning tips for HP-UX Integrity servers. These tips are available in the HP white paper "Taking SAS to the enterprise: Kernel configuration guidelines for SAS 9 on HP-UX 11i v3." In addition to this paper, HP recommends that you use the Veritas VxFs tool to create the file systems that are used by the SAS® System. Suggested mount options for use with the file system associated with SAS WORK are as follows:

  ```
  /dev/vgWork/work /work vxfs
  noatime,tmplog,convosync=delay,mincache=tmpcache,datainlog 0 2
  ```

  Because, by default, SAS WORK contains only temporary contents that are unusable after a reboot or a crash, the system should make no effort to ensure the integrity of those contents. This improves I/O performance at the increased risk of data corruption in the event of a system crash. If you are using SAS' checkpoint/restart, you may want to use different parameters.

  Also, HP recommends that you increase the default block-size values of **1K** to **8K** on your file systems (unless you are creating files that are less than 1K in size).

Are there any tips for maximizing I/O throughput from a storage vendor perspective? Over the past few years, working with various storage vendors to overcome performance issues experienced by SAS customers has helped us to develop the various tuning tips listed in this paper. The following list provides some additional tips that we have learned. In some of those cases, what we have learned is that no additional tuning is required.

- EMC Corporation - Tip details about the following products from EMC are expected for a future update of this paper.
  - CLARiiON
  - Symterix
- IBM Corporation—IBM's current storage arrays typically need no additional tuning if you follow the other tips in this paper. See the note on virtualized I/O below.
- Hitachi Ltd.—You should use the Hitachi Dynamic Link Manager (HDLM) for multipathing, and do not Hitachi Dynamic Provisioning. We have found that, on these arrays, a stripe size of **256KB** is the best for SAS.
- NetApp, Inc.—NetApp's current storage arrays typically need no additional tuning if you follow the other tips in this paper.

Does SAS work well with virtualized I/O? Many of the new storage arrays on the market (the HP Enterprise Virtual Array (EVA) and the IBM XIV Storage System) can be configured as virtual storage arrays. SAS® software functions well with virtualized storage. Still, the key point is whether the file system can provide the I/O throughput rates that are required for SAS applications. **Note:** It has been our experience that by virtualizing I/O, latency is added that can reduce the observed I/O throughput of the storage array.

Are there limitations to the amount of IO that can be pushed via the operating-system file cache? Limitations are different on various operating systems, but with today's fast storage array, you are only limited by what the underlying I/O subsystem can support. For more information about the limitations, caveats, tuning, and throughput characteristics of your operating system, contact your operating-system vendor.

Are there any ways to avoid file caching and to read directly from disks with SAS? As we have discussed in the paper "Best Practices for Configuring your IO Subsystem for SAS®9 Applications," SAS performs I/O via the operating system's file cache, by default. This process is efficient when the files are small (less than 1 GB) and are used often within a SAS job or by multiple users. This method tends to keep highly used file pages in memory, so performance is very fast. However, the current workloads used by many SAS customers now involve accessing files that can be tens or even hundreds of gigabytes in size. When you access files this size in a sequential manner, you can avoid a lot of pressure on the memory subsystem by avoiding the file cache. This is especially helpful when the computer starts swapping from physical memory to virtual memory.

There are several ways to avoid using the file cache with SAS. Starting with SAS 9.1 under Windows, you can use the SAS SGIO option. In SAS 9.2 under UNIX and Linux, commands are available on a SAS library level or a data-file level that instruct SAS to avoid using file cache. These two methods for avoiding the file cache work very well in cases where you are working with very large files, but they do not work well when you repeatedly read the same small file. For more details about both of these methods, see the SAS Global Forum 2009 paper "Improving SAS® I/O Throughput by Avoiding the Operating System File Cache."

Operating-system commands also are available that restrict the amount of memory that can be used for the file cache. For AIX, these commands are documented in the AIX tuning papers that are available on the SAS Partners/Partner Directory: White Papers Web page. These commands really help out AIX 5L and AIX 6. For Windows 2003, you can use the **setcache** option to determine the percentage of memory that can be used for the file cache. For Windows 2008, you can use the DynaCache tool to determine the percent of memory that can be used for file cache.

Avoiding the use of the file cache is often referred to as "invoking a direct I/O access method." While this can improve performance results when very large SAS files are accessed sequentially, it can also cause extremely poor performance when smaller files are accessed that are better off cached in memory. So we do not recommend avoiding the file cache on a global basis. Instead, it should be used based on the particular SAS application or based on the use of large SAS files.

Other things that might influence how to configure file systems for SAS. It is easy to just create a single, large file system for all of your permanent SAS data files, but you need to consider several questions before you do this.

- Does this data need to be backed-up? It is difficult to back up a file system that is multiple terabytes in size.
- Will you need to increase the size of the file system? If so, then the way you put together multiple LUNs to create the file system is very important. With a striped LUN, expanding the file system might require either physically re-laying a larger LUN set on the disk (incurring down time, risk, and expense) or concatenating additional LUN stripes to the existing LUNs for increased space. Concatenated LUNs can have adverse performance implications. Therefore, as you are planning your LUN layouts, work with your storage administrator to determine future growth considerations.

Diagnosing performance issues with your I/O subsystem. Over 80% of the SAS customers with which the SAS Performance Lab staff works can trace their performance issues back to underlying I/O subsystem setup. This is one of the main reasons for writing this paper and for writing "Best Practices for Configuring your IO Subsystem for SAS®9 Applications." We have offered several papers previously that discuss how to monitor your computer to determine if a computer resource is bottlenecked. These papers are available on the Scalability and Performance Papers Web page on the SAS Customer Support site. In addition to the topics discussed in these papers, you should also check the following items:

- If a SAS process is running overnight (that is, not during normal business hours), check to see if there are other applications or tasks running at the same time. We frequently see a corporate back-up application running at the same time. Running applications simultaneously can really influence the I/O bandwidth to the underlying physical hard disks that constitute the file system that SAS is accessing.
- If a SAS process is running during business hours, check to see what other types of applications might be sharing the underlying physical hard drives that constitute the file system that SAS is accessing. If these other applications are performing very small and/or random I/O processes (that is,. they are online transaction processing (OLTP) type applications), then the storage arrays might favor those I/O requests over the large I/O requests that SAS typically runs. When this happens, you should work with the storage administrator to see either SAS or the other application can be moved to a different set of physical disks.
- Under Windows, make sure that your real-time antivirus-scanning software has a list of all of the different types of data files that SAS can create so that it will not scan those files.

CONCLUSION

As you can see, setting up file systems for heavy SAS application use is a broad subject that just a few guidelines cannot cover. If your SAS application is enterprise-wide, time critical or mission critical, will be accessed by hundreds of users, or will be processing 1 TB or more of data, then we strongly encourage you to work with SAS Institute to complete a detailed technical assessment so that we can recommend the best I/O infrastructure to properly support your SAS applications.

APPENDIX

This section provides more detailed information about the new technology drives.

- Serial Advanced Technology Attachment (SATA) drives—SATA drives extend the old ATA (IDE) parallel technology to the next level with its new I/O method using a high speed serial bus to transfer data. The fast, new serial bus is eight times faster than the old parallel ATA bus, and future SATA drives will be even faster. Current SATA drives have a spin rate of 7,200 rpm with a typical seek time of 9.5 milliseconds. This makes them considerably slower than the more expensive SAS drives mentioned below. The mean-time-between-failure rates (MTBF) for SATA drives are not quite as good as Serial Attached SCSI (SAS) drives, either. They are much less expensive, though, and they have gained popularity for that.

  SATA drives also come in high capacities of up to 1.5- to 2-terabyte drives. It is important to note that SAS applications perform best on larger numbers of smaller drives that result in a high-spindle throughput than on fewer, larger drives. So you should not rush out to buy large SATA drives for use with SAS. Stick with the smaller drives and employ a lot of them in a stripe set.

  While they are definitely a step up from the old ATA/IDE drives, SATA drives are still slower than SAS and fiber-channel attached devices. Their cost makes them ideal for SAS shops on a budget; you can buy more and create a longer stripe to improve performance somewhat. They are also widely used as backup and archival drives.

- Serial Attached SCSI (SAS) drives—In much the same way that SATA drives serialized the old parallel ATA/IDE technology, SAS drives have serialized SCSI technology. These drives use a full-bandwidth approach, without splitting the bandwidth for multiple devices on a chain. By using multiple wide-port devices, that bandwidth is increased by simultaneously using multiple 300-megabit channels. This yields a high-performance drive set. SATA backplanes can support a very high number of devices per chain (up to 16,384 per SCSI interface). The backplane can also support SATA devices at a lowered bandwidth (the maximum SATA can sustain - of 150 MB/sec versus 300 for SAS devices). So it conveniently enables you to mix and match SAS and SATA devices. However, you cannot add SAS devices to a SATA backplane.

  SAS devices have a spin rate of 15,000 rpm with a fast seek time of 3.9 milliseconds, much faster than the SATA devices. They boast a higher reliability rate, but they are also much more expensive. So they are ideal for SAS application usage for very high-performance needs.

  What about fiber-channel drives in most SANS today? SAS drive performance is about comparable with the fiber-channel attached drives in most SANs today, with much higher performance (doubling the megabytes per second) coming in the near future. Talk with your preferred storage vendor about their storage futures and what you need to consider for your future storage investments.

- Dynamic RAM (DRAM) devices—These devices store data in memory and yield nanosecond-range performance for access and writing, which works extremely well with current CPU speeds. However, DRAM devices are volatile, and they lose the binary charge that holds the data in a fraction of a second when power is lost. This is unacceptable for data that needs to be persistent.

  DRAM devices offer a fast alternative for SAS WORK file systems because they are used only temporarily, and working files stored there are not recoverable in any event of failure. The problem with using these drives even as SAS WORK file systems is that they can fill up, leaving your jobs unrecoverable. They also are more expensive than traditional hard drives. SAS customers who need extremely fast response time for SAS WORK file systems might find that the risks involved and the extra expense are worth the extremely fast response time.

- Solid State drives (SSD, also referred to as flash drives)—Solid State drives are currently the most expensive storage alternative. They are exceedingly faster than traditional disk drives (accessing data takes microseconds instead of milliseconds). Thus, they have phenomenal Read performance and, usually, excellent random-Write performance. Sequential write performance is not as accelerated and may tip the cost/performance scales.

CONTACT INFORMATION

Many thanks for reading through this paper. Your comments and questions are valued and encouraged. Contact the authors at:

Margaret Crevar
SAS Institute Inc.
100 SAS Campus Drive, R-2427
Cary, NC 27513
Work Phone: 919-531-7095
E-mail: Margaret.Crevar@sas.com

Tony Brown
SAS Institute Inc.
Millennium I
15455 North Dallas Parkway
Suite 1200, Room 1350
Dallas, TX 75001
Work Phone: 214-977-3919 x1655
E-mail: Tony.Brown@sas.com

SPECIAL THANKS

Special thanks go to everyone who reviewed this paper, in particular to Tom Keefer and Leigh Ihnen.

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies. Copyright © 2010 SAS Institute Inc., Cary, NC, USA. All rights reserved.