

Paper 203-2007

Ensuring you Have the Proper Resources for your SAS®9 Applications

Margaret Crevar, SAS Institute Inc., Cary, NC

Tony Brown, SAS Institute Inc., Dallas, TX

ABSTRACT

Several previous SUGI papers address performance-problem troubleshooting with SAS®9 applications. This paper contains information to help you manage and monitor your computer environment to ensure that you have adequate resources to support your SAS®9 applications. The paper briefly describes the infrastructure that is needed for various SAS® applications (from simple SAS jobs to the most complicated SAS®9 Enterprise BI applications). It then shows how to identify which system infrastructure resource areas are most under pressure; and how to continually monitor them by using simple, operating-system tools and more-complex, third-party monitoring applications. The monitoring and resulting resource management advice will help ensure that you can meet the demands of your SAS users.

INTRODUCTION

This paper discusses some of the more common uses of SAS®9 by customers. These uses include legacy batch SAS jobs submitted by long-time users of SAS, long-running ETL flows created by using SAS® Data Integration Studio, and the more complex SAS®9 BI applications, such as SAS® Enterprise Guide® and SAS® Web Report Studio. For each of these application areas, the discussion includes information about which computer resources are likely to be stressed the most and need to be monitored, and some operating-system tools that can be used are given.

Setting “busy” threshold levels for computer resources to help determine the next steps when resources become overloaded are also discussed. These concepts are very similar to the way that System Administrators monitor mainframes to ensure that everyone has the computer resources they need.

You’ll also learn how to use third-party software tools to continuously monitor your hardware, produce reports of daily activity, and send a warning of potential hardware issues to your System Administrator.

LEGACY BATCH SAS JOBS

Usually, legacy batch jobs read data from an external source(s) (which can be anything from text files to relational databases) into a temporary SAS data set in the SAS WORK area. There can be only a couple of steps in length, or there can be hundreds of steps that are required in order to analyze external data source(s) and produce the desired reports and data marts. The usual steps in a SAS job that use the most resources are DATA steps and Procedure steps, such as PROC SORT, PROC SUMMARY, and so on. If statistical procedures are used (for example, the IML, REG, MIXED, GLM, and NLMIXED procedures), significant CPU resources can be consumed.

There are three typical areas that need to be monitored – IO, memory, and CPU. Let’s discuss what things need to be monitored within each area.

In legacy batch jobs, the primary area to monitor is the SAS WORK area, which may be heavily used by the logic in SAS steps. In SAS programs with multiple DATA and PROC steps, it’s common for the steps to Read and Write files to the temporary WORK area. In addition, some steps that manipulate data (that is, sort, aggregate, and model data) create additional temporary files that might be larger than the input data files. A long string of SQL steps can result in many intermediate join files. Furthermore, when batch SAS jobs run concurrently, the number of temporary files is multiplied because *each batch job starts its own SAS process and creates its own sub-directory in the file system that is associated with the WORK area.*

Because these temporary files can put significant pressure on the file system associated with the WORK area, you will need to monitor the WORK area for I/O bottlenecks. This is the leading reason for poor performance with these types of SAS jobs.

You also, have to ensure that the temporary files that are being created do not completely fill the WORK area. If the WORK area is full, the next time that SAS tries to write data to the WORK library, the SAS process fails due to insufficient disk space. As mentioned earlier, by default, all SAS jobs use the same WORK area—if that library is full for one SAS job, it will be full for all SAS jobs. Therefore, it is recommended that you monitor the WORK area to ensure that it does not become 100% full. Some SAS customers want to know when the file system is 85-90% full so that they can decide if they need to extend the WORK area.

The last thing from an I/O perspective is the I/O throughput that is associated with the various file systems. You want to ensure that the disks stay busy (100% busy is okay) without any wait queues. A **chronic 100% busy** disk-set that has a **wait queue of 5 or more processes is not okay**, especially, when there are complaints about slow response. Therefore, it is recommended that you monitor I/O throughput to ensure that wait queues for long periods of time during the day are not building up.

Memory consumption is another resource that you need to monitor. In particular, you need to be aware if your operating environment is swapping/paging from real memory to virtual memory. Swapping/paging results in significant degradation in I/O performance. This generally happens when an application's memory requirement exceeds physical memory and/or because SAS Reads and Writes data via the operating environment's file cache, which uses memory. For example, in order to ensure the best performance, it is recommended that you enable Read-Ahead and Write-Behind/Write-Through daemons. (The term "daemon" might have different meanings on various hardware platforms but, whatever term is used, you want the SAS application to be given a signal that the Write has been committed to cache as opposed to disk). When the Read-Ahead or Write-Behind/Write-Through daemons are enabled, the default kernel parameters on some operating environments signal the systems that they can use almost all the real memory on the computer for file cache. If this signal is given to the operating environment and SAS also needs a large chunk of memory for its processing, swapping/paging can occur.

In this instance, you need to monitor the amount of available physical RAM on the computer. One SAS customer wants to know when the computers at their enterprise drop below 600MB of available memory for long periods of time because, usually, on their system, this is a sign of excessive swapping.

You need to monitor how busy your CPUs become with all the batch processing. Under most Windows and UNIX operating environments, there are no standard queue managers (such as there are on mainframes). Therefore, there is no way to throttle the number of concurrent batch SAS jobs and other applications without writing custom queue managers. You need to determine what level of "CPU busy" is okay for your system. At some sites, SAS customers think that a computer that is running at 90% CPU busy is great. At other sites, SAS customers think that a computer that is running at 90% CPU busy is a "red flag" that indicates it's time to consider adding processors to the computer.

ETL FLOWS

Let's examine how ETL flows that use SAS Data Integration Studio or SAS/Warehouse Administrator[®] are processed. Similar to the legacy SAS batch jobs, long-running ETL flows read data from external sources (the data can be anything from text files to relational databases) into a temporary SAS data set in the WORK area during the extract (E) and transformation (T) phase of processing. Then, a SAS data mart is loaded (L) into a permanent SAS data library. Occasionally, during the ETL flows, data will be written to a permanent SAS data library in order to be able to recover the data. Both ETL tools, SAS Data Integration Studio, and SAS/Warehouse Administrator use a SAS Metadata Repository to keep track of data sources and data dictionary information.

Similar to the legacy SAS batch jobs, the typical areas that you need to monitor for long-running ETL flows are: I/O, memory, and CPU. If you place the SAS Metadata Server on a different computer from your back-end SAS servers, you also need to monitor the network.

In addition, you need to monitor the permanent SAS data libraries to ensure that they are not filling to capacity and that there are no wait queues. The permanent SAS data libraries need to have adequate throughput (usually associated with a lot of spindles). The more disks that the file system contains, the better it is for the heavy (usually, sequential) Read-Write activities that are associated with ETL flows—the same as is described above for the SAS WORK area.

In SAS[®] 9.1.3, a SAS Metadata Repository (which is a separate SAS server) maintains all the metadata that is required for the ETL flows that are associated with SAS Data Integration Studio. This metadata is kept in an in-memory database that is accessible only by the SAS Metadata Server. Because this data is stored in memory, you need to ensure that the computer has adequate memory and that swapping/paging is avoided. Many SAS

customers put the SAS Metadata Server on its own computer in order to ensure that no other applications will try to use the memory in the computer. This now adds a network connection between the ETL flows and the computer that needs to be monitored in order to ensure that there are no bottlenecks.

SAS[®]9 BI APPLICATION TOOLS

SAS ENTERPRISE GUIDE

SAS Enterprise Guide delivers a graphical user interface (GUI) to most of the features in SAS. You can select SAS data sources, extract data, merge different data sources, and produce custom reports. SAS Enterprise Guide uses the metadata store in the SAS Metadata Server to ensure that you have the required permissions to access various data sources. SAS Enterprise Guide creates SAS code in the background, which you submit to a back-end SAS server. This process is different from submitting SAS code as a batch job because these back-end SAS servers are, usually, up-and-running, and they can be shared by multiple users of SAS Enterprise Guide, which is very popular with SAS[®]8 and SAS[®]9 users.

Similar to when you execute batch jobs and ETL flows, the areas that need to be monitored when you use this tool are: I/O (especially the WORK area), memory, CPU, and network using the same methodology as described above.

SAS WEB REPORT STUDIO

In this section, the discussion focuses on additional third-party software that is used with SAS[®]9 BI applications that heavily use the SAS Mid-Tier components, such as SAS Web Report Studio. The SAS[®]9 BI applications require the use of a Web Application Server, such as Apache Tomcat, WebLogic from BEA, or WebSphere from IBM. Therefore, you not only have to monitor the network traffic between the back-end SAS servers and the SAS Metadata Server, but you also need to monitor the additional network traffic to the Web Application Server.

Similar to when you execute batch jobs and ETL flows or use SAS Enterprise Guide, the areas that need to be monitored when you use SAS Web Report Studio are: I/O, memory, CPU, and network. The additional processes in this scenario relate to the SAS Mid-Tier Components that include a Web Application Server that runs several Java processes as well as a WebDAV Server. Because these servers tend to be very memory- and CPU- intensive, it is very common to have these servers run simultaneously on a computer. Therefore, you need to monitor the memory usage to ensure that no swapping/paging occurs. You also need to monitor CPU to ensure that you have enough processor power to support the Java applications.

TOOLS FOR MONITORING

A methodology for solving performance problems with your SAS jobs or applications is documented in the SAS white paper "A Practical Approach to Solving Performance Problems with the SAS System" (Brown 2001). A follow-up paper entitled "Solving SAS Performance Problems: Employing Host-Based Tools" (Brown 2006) goes into more details about how you can solve performance problems by using standard monitoring tools that ship with most of the commonly used operating environments.

If you want to monitor an area for a few minutes at a time, there are some very simple tools that enable snapshot monitoring: on UNIX, use the commands: top, prstat, or topas; on Windows, use TaskManager. Also, there are some third-party tools that you can use; for example, HP OpenView GlancePlus and Solaris Resource Management from Sun Microsystems. These tools have very nice graphical user interfaces, and they have the capability to define "rules" that make the task of monitoring the hardware very easy. Usually, these tools run interactively and do not produce log files.

If you want to monitor an area for longer periods of time, the paper "Solving SAS Performance Problems: Employing Host-Based Tools" (Brown 2006), describes the tools that you can use. In the paper, you'll find information about setting parameters for various tools, such as the Windows Performance Monitor (PerfMon) and the UNIX tools that use the commands: sar, iostat, and vmstat, to monitor system resources as well as interpret the information you have collected. There are sample scripts that can be run where you pass in the collection interval and how many collections you would like and this information is written out to several log files. The log files are interpreted by manually scrolling through the lines of data. IBM has a nice tool for AIX called *nmon* that takes all the standard UNIX monitor data and puts it into high-quality graphs.

HARDWARE MONITORING

If you want to monitor your hardware, continuously, to be sure that you do not run out of a computer resource or that there are no bottlenecks affecting an important computer resource, third-party monitoring tools will make this task easier.

SAS is developing integration packages for HP's OpenView Operations, IBM's Tivoli Enterprise Console, and Microsoft's Operations Manager. Each integration package is customized for a specific product, but the overall objectives for managing and monitoring the SAS[®]9 environment are consistent across the various products. The integration packages will include the following attributes:

- Packaged suite of scripts, tools, monitors, and documentation which describes the installation and integration components for SAS 9.1.3
- Facilities to start, stop, pause, and restart each of the key SAS servers: SAS Metadata Server, SAS Stored Process Server, SAS OLAP Server, SAS Object Spawner
- Availability management facilities to verify the status of key servers and generate events as necessary for server status
- Ability to monitor key SAS server metrics and generate events into the management environment—metrics such as processor utilization, memory allocation, and storage utilization
- Ability to analyze SAS server logs for key actions and generate events into the management environment—actions such as server initialization, user connection events, userID authentication, and server error events

For more details on these integration packages, you can attend the SAS[®] Global Forum 2007 SAS Presents session entitled "Monitor Your SAS[®]9 BI Environment with Enterprise Systems Management Applications and SAS[®] Management Console".

CONCLUSION

This paper explains which performance areas you should monitor and what tools to use. This paper contains information to help you manage and monitor your computer environment to ensure that you have adequate resources to support your SAS[®]9 applications. If you find during this monitoring that you have a computer constraint, here are some guidelines regarding how to fix it.

- If your computer is I/O bound, then you need to consider adding I/O bandwidth. There is a SAS[®] Global Forum 2007 SAS Presents, "Best Practices for Configuring your I/O Subsystem for Your SAS[®]9 Applications", that goes into a lot of details regarding how to improve your I/O bandwidth.
- If your computer is memory constrained, do one of the following:
 - add physical RAM
 - use operating system commands to decrease the amount of memory that can be used for file cache
 - decrease the value for the option MEMSIZE= to reduce the amount of memory that each SAS session can use (if the SAS sessions can function with less memory)
- If your computer is CPU constrained, do one of the following:
 - add more processors
 - use a queue manager to control the number of concurrent SAS sessions

Managing your computer's performance is an ongoing and, sometimes, complicated task. It is essential that you establish performance baselines to help set attainable users' expectations, and to serve as a reference in order to compare performance. The use of tools to establish performance baselines and to monitor system and application performance is crucial in developing, improving, and maintaining high-quality operating environment performance.

RESOURCES

Brown, Tony. 2001. "A Practical Approach to Solving Performance Problems with the SAS System". Cary, NC: SAS Institute Inc. Available at support.sas.com/rnd/scalability/papers/solve_perf.pdf.

Brown, Tony. 2006. "Solving SAS Performance Problems: Employing Host Based Tools". Cary, NC: SAS Institute Inc. Available at support.sas.com/rnd/scalability/papers/TonySUG131_20060403.pdf.

Crevar, Margaret. 2007. "Best Practices for Configuring Your I/O Subsystem for SAS®9 Applications". Technical Papers and Presentations. Cary, NC: SAS Institute Inc. Available at support.sas.com/rnd/papers/index.html.

Hall, John. 2002. "SAS Application Performance Monitoring". Cary, NC: Hewlett Packard Company. Available at www.sas.com/partners/directory/hp/sasapp.pdf.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors:

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

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

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.