Paper 205-2007

Architecting SAS in a Modern World: Best Practices for Design, Configuration and Management of SAS® 9

Greg Nelson
ThotWave Technologies, Cary, North Carolina

John Loether
LoetherSpeaks

Abstract

For decades, SAS has been relegated to the world of the mystical, the unknown and perhaps even superstitious. Configuration options, switches and toggles all in conspicuous or perhaps precarious positions, waiting for the day when an expert can take a closer look—or worse, when something breaks.

This paper will show how SAS can be architected with modern Windows and UNIX systems solutions. While current users of SAS 9 can continue to use the software as they (or their fathers) always have, SAS 9 now offers new possibilities for how to use its technology more efficiently as well as how to better manage infrastructure.

We will explore the hardware side of SAS by examining disk options, I/O configurations, backup as well as understanding how powerful multi-way (and multi-core) systems can ease your SAS processing with optimal configurations between processors and memory. Finally, we will discuss when technologies like Citrix (or terminal services) and virtualization (like VMWare) could and should be an option for your organization.

This paper is aimed at the traditional SAS programmer wanting enough information to be dangerous in discussions with IT. For the IT professional who wants the truth about what exactly matters in the world of SAS and what options exist for the best possible configuration.

Introduction

SAS recently celebrated 30 years of providing software for data integration, analytics and business intelligence; and while things have changed dramatically over the past three decades relative to technology, SAS is still the most powerful software package in its class.

For the last three decades SAS has proven itself the leader in solving enterprise class business problems. In March 2004 SAS secured that position by releasing SAS 9 with its integrated metadata repository, multi-tier infrastructure, and increased capability for business analytics. SAS built upon its strong foundation and for some customers the SAS 9 upgrade was simply business as usual.

However we as humans don't really like to change. The idea of not only going through an upgrade, but taking advantage of new opportunities with the technology can be overwhelming at times—even for current SAS customers. The following sections show how SAS 9 works and the critical components of this new architecture that may drive changes in modernizing the hardware platform on which SAS relies. Understanding how SAS 9 works, what matters to SAS in terms of how it operates on modern hardware, how to choose the right platform and the factors that determine the SAS platform you use is essential in getting the most from SAS solutions.

SAS Architecture

A number of goals drove the redesign of SAS 9 architecture, which included:

Manageability – through the management of security (users, groups, roles, access rights), servers (both logical and physical), and metadata (SAS Management Console)

Scalability – implementation of grid-enabled and SMP and multi-core systems management solutions as well as improvements to performance and multi-threading enhancements

Usability– focus on the roles people play and deliver interfaces specifically for those individuals; Persona based clients (Information Maps, Enterprise Guide, SAS Management Console, etc.)

Reliability– creation of failover and clustering environments and reliable backup solutions for 7x24 operations

The SAS 9 architecture relies on a new component, the SAS Metadata Server, to provide an information layer between the programs and the data they access. By providing a single point of access for this kind of information, SAS servers can be located just about anywhere (consolidated or distributed) and on any platform (operating system and hardware), to be accessed by SAS clients. The SAS 9 platform can be configured in any number of ways, including a myriad of server topologies and client offerings.

From here, the SAS client tools take on the form of special purpose applications designed to satisfy the needs of various types of SAS users. Tools used to access information now include a substantially improved Enterprise Guide, the new SAS Add-in for Microsoft Office, and the web-based clients Information Delivery Portal and Web Report Studio, as well as Base SAS components.

Management of SAS metadata is done through the SAS Management Console and Information Map Studio. Management of data can be done through Data Integration Studio in addition to existing SAS products such as SAS/Access and Base SAS.

Each implementation of SAS 9 can be fundamentally the same and different depending on the profile of the business problem being solved. These profiles make use of the new services in SAS 9 in varying degrees.

The figure below depicts one such topology for an enterprise configuration of SAS 9.

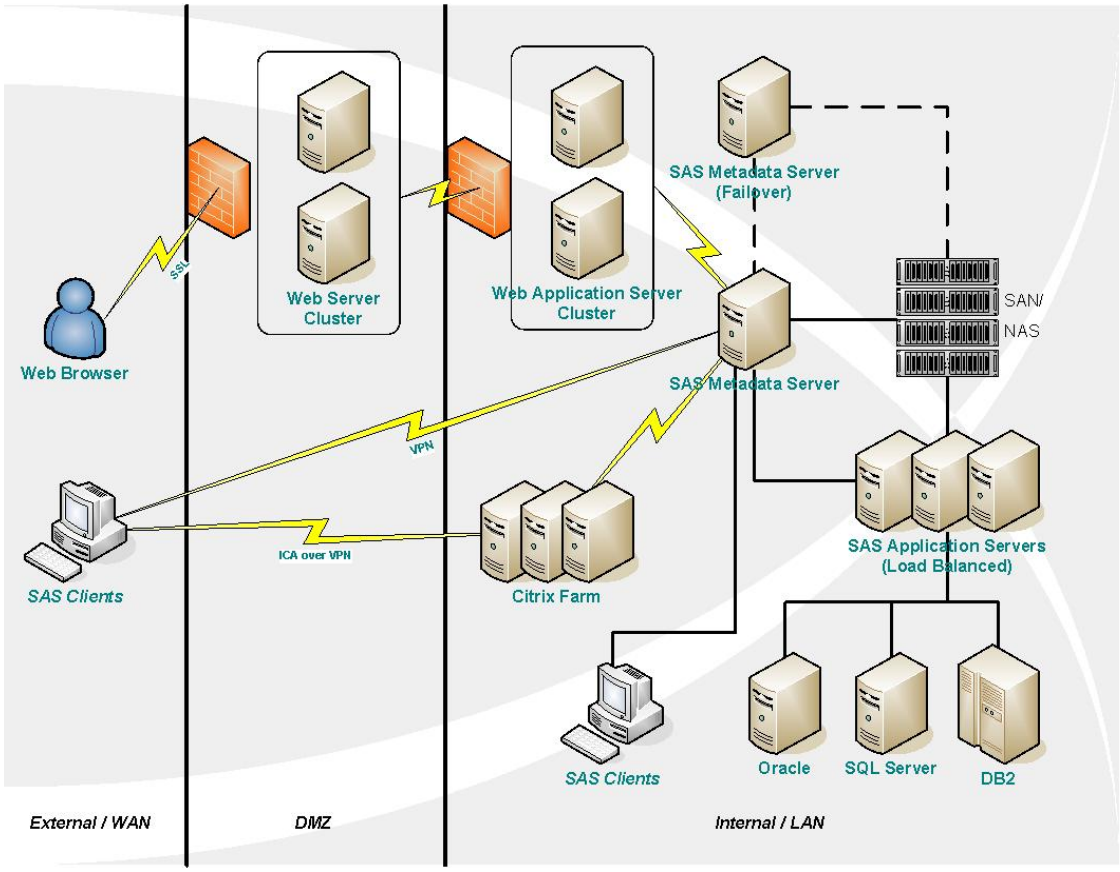

Figure 1. Potential SAS Architecture Profile

In this diagram, typical SAS tool development relies on Enterprise Guide talking to a server-based instance of SAS. This model supports the concept of centralized server resources.

The SAS Institute has made a large investment in these new products and technologies, and these are the future of SAS for some time to come. It must be recognized that some of the new products, such as Information Map Studio, Web Report Studio, and the OLAP server, are aimed at solving general Business Intelligence problem areas.

The figure below shows some of the clients and how they interface with SAS servers.

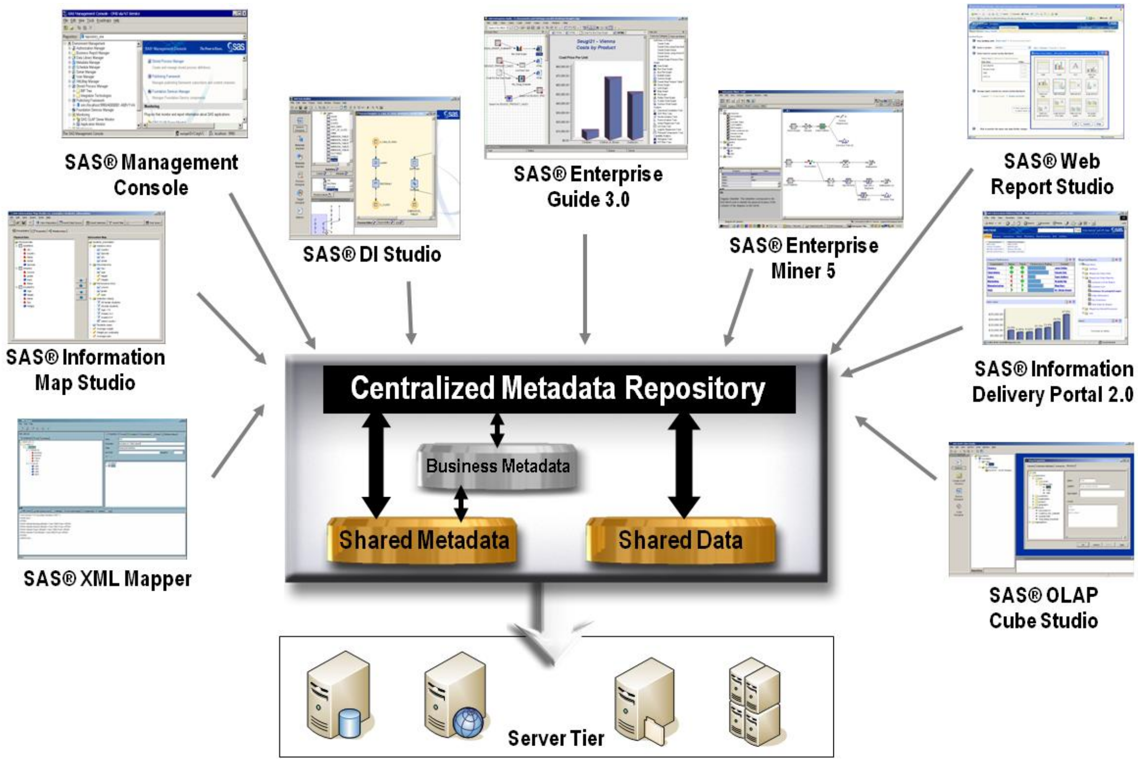

Figure 2. SAS Clients Interacting with the SAS Metadata Server

There are a number of benefits in using the new SAS multi-tiered architecture:

- Delivers immediate productivity benefits to SAS users, particularly those currently working with a variety of SAS interfaces, such as display manager (X-windows, telnet, PC SAS), because of the user-friendly tools appropriate for the job at hand.
- Efficient use of both the desktop and the server. The desktop is used for code editing, reviewing results, and data exchange. The server is used for SAS computations, which are the most CPU and disk intensive operations.
- Excellent support for centralization of SAS, consistent with both the business vision of a global system and the overall IT consolidation strategy.
- SAS Management Console allows management of license information, installation/configuration information, which servers and applications are currently in use, and some ability to monitor SAS servers.

Things that SAS Cares About

When asked, “How can we improve performance?” We as consultants often reply, “It depends.” We know that whether we are talking about the workspace server or the WebDav server, or what types of applications are we running (ETL versus highly analytics tasks, for example), performance is rooted squarely in the architecture selected.

So while, we cannot offer a cookbook recommendation, the best place to start is to ask what is important to SAS—and in terms that we all understand. SAS knows the following factors affect performance:

- Processor speed and type

- Memory
- I/O
- Network
- Management

These are the things (core, foundation, statistical—whatever you want to consider them to be) that SAS cares about.

Processor (CPU)

Heavy-duty analytics often involves floating point calculations in formulae. The Floating Point Data Type and the number of Floating Point Registers also heavily influence accuracy. The following analogy may help explain.

Dr. Seymour Cray built Cray Supercomputers with massive Floating Point (FP) registers and Array Processors. Today, laptops have Duo-Core processors that rival even the largest mainframe computers in terms of processor speed, disk capacity and memory. Over the last several decades of improvements, the goal has become to get more floating point instructions on a single chip, then adding multiple chips to a socket and multiple sockets to the computer. Each step building on the next until a better, faster, and more affordable answer is achieved.

In 2006, what was once the purvey of the supercomputer entered the smaller industry standard chips from Intel and AMD. The desire is more parallel processing with more parallel FP registers so the user gets more out of the architecture. And architectures like the IBM P5, Alpha, and Itanium EPIC keep this horse race going forward.

Along with this desire is to have large SMP counts that can interact with very large memories and at the same time interface and update on chip caches. It should be noted that FP registers should be large (for accuracy), hence the big machines being 64-bit in length for some time. .

There is a difference in both the number of registers and processing units for both integer and floating point. The parallelism and complexity still go to Big Iron (128 CPU SMP, 2 TB Memory, Cell architecture, multi-O/S) making very large computers still the best at what they do. For example, mainframes are still the best at processing information and writing them over large channels to disk.

The figure below notes that CPUs fall into one of several categories starting with how most computers were built historically with one CPU register and one cache. As we move along the continuum, we see not only more CPUs (multi-core) but multi-core and multi-threaded CPUs.

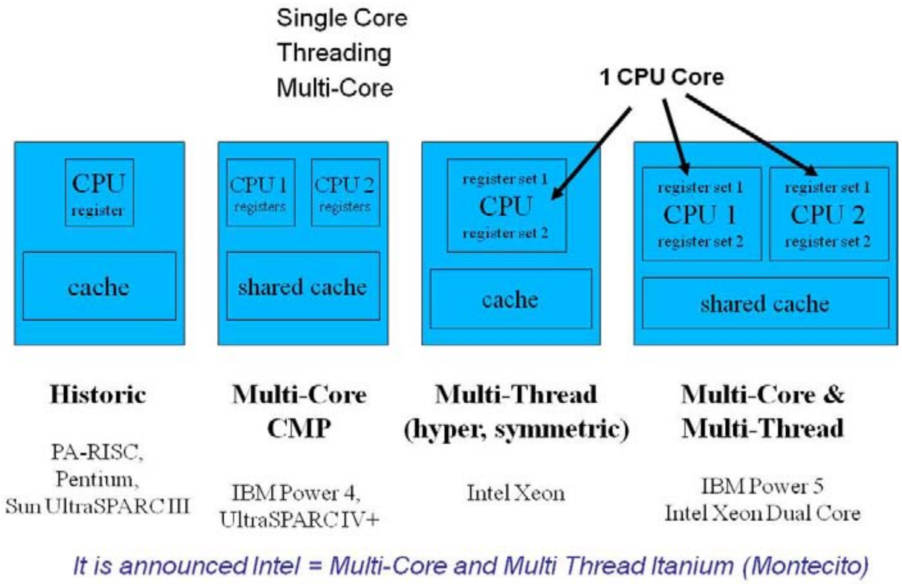

Figure 3. CPUs versus Cores

There are many high-end features in modern systems. For processors in particular, it is important to highlight the very high-end feature set, for example. Itanium has a processor integrated into OpenVMS and HP NonStop (ex-Tandem) systems. In other words, a processor able to sustain the highest levels of availability and deliver the same experience to customers used to well above 10 years of continuous work uninterrupted by either planned or unplanned downtime.

The figure below highlights some high availability features placed into the Itanium architecture. More capability can be added through integration of multiple cores in a cell, and hooks cells together creating machines that very from small to huge with memory sizes from 64 Giga bytes to 2 Terabytes.

You may recognize simple concepts such as ECC cache and CPU data line capability. But also concepts like data poisoning and dynamic processor resiliency. Think of the figure as a who's who in electronic sophistication that also defines the best of the best.

	Itanium*	PA-RISC*	Opteron*	Xeon*	Sun SPARC	IBM POWER4/5
Chip thermal sensors & management	YES	NO	YES	YES	NO	Limited
Full Cache Parity / ECC	YES	YES	SOME	SOME	SOME	YES
CPU Data Bus ECC	YES	YES	NO	NO	YES	YES
†Data Poisoning & Signaling for Error Recovery	YES	NO	NO	NO	NO	YES
‡Enhanced MCA handling & Error Logging	YES	Proprietary Method	Limited	Limited	Limited	Proprietary Method
Dynamic Processor Resiliency	YES	YES	NO	NO	Limited	YES

*Processors used in HP Products

†To prevent an unrecoverable error from propagating and corrupting data, the "Data is Poisoned" in a way that marks it as permanently bad so that the system will either force a reread of that data, or in the worst case forces a crash.

‡Enhanced MCA handling and Error Logging are tools provided to allow for more granularity of error containment. This allows errors to not propagate and to let recovery to have the least impact as possible.

Figure 4. Processor Availability Features

In 2006 and 2007, CPUs have changed from single core to dual core to quad core. This is occurring at both ends of the spectrum, from small blade form factors to large cell based systems. This has caused a large stir in the industry—chips from AMD and Intel have become hybrids capable of doing 32-bit programs as before, but also address 64-bit programs creating healthy memory systems –. And it is not over. Quad core today, but larger cores will be coming in the next three years.

Swimming against this tide is a bad trend. Users are observing an average system use on x86/Windows and Linux machines while getting a 5% CPU utilization on a single core. Efficiency actually goes down as we add cores. To remedy this, we need to make sure the code is recompiled for SMP and hyper-threading. And on big machines that are pure 64-bit, this recompiled code must extend to the hardware architecture of the machine, such as EPIC for the IA-64 architecture.

This leads to threading, multi-processing, and huge parallelism in code execution streams, plus large cache, heavy floating-point, I/O and huge memory or supercomputing. From here it is possible to partition the big machines into multiple pieces with multiple operating systems at the same time. Mainframes no longer run like they used to thus providing the user with a lot of choices. SAS provides choices about how we divide up the architectural components and offers a large array of choices about what hardware to put it on and whether to virtualize the operating system or not.

What is the right choice? Several dual core machines tied together or a bunch of blade machines in a cabinet or a classic Big Memory? Big SMP computers clustered together? Partitioned machines?

The answer lies within the following questions. What is needed at the user/client level? What modules are common for their use? Can a dual-core machine with a decent chunk of memory (GB) handle the data space required? If so, we have found the scale for most people.

Now what about the data sets and databases being manipulated? If they are huge, then a large scale SMP and large memory machine falls into view. If not and it is medium in size (Giga bytes of data), a 4-way machine should be able to handle that part of the server side.

Memory

In addition to pure processing power memory plays a critical role for SAS applications. The big problem in computing is still getting the data in, and getting it out. The size and amount of the data has grown. From Kilobytes to Megabytes to Petabytes in just a few years and it isn't over.

Users will want to house, access and analyze more data this year than they did in the last. Therefore the need for VVLM (Very, Very Large Memory) machines continues. Naturally as memory goes up, the memory feed to the disk storage grows and the sizes of caches are growing dramatically, too.

Therefore, how machines are being connected together is important. The goal is to increase and maintain bandwidth. Where are the data sets? On a SAN? Direct Attached? In memory only? What should be used? Grid computing, cluster computing, or partitioned computing? Regardless of what is chosen the goal is the same: get the data off the disk, into memory, through the processor, and the results to the user—quickly.

The options are endless. Big machines still have the advantage when it comes to pushing megabytes per second (MB/S) into the machine. Large memory allows data to crunch through the fastest. Especially when several users need extractions from the same data set. Also when many users in a statistical environment are using the server, it is ideal to have one with a high floating point capability. On the other side, grid computing that puts several machines in parallel and can provide aggregated horsepower—as long as the access to the data is spread out and not highly concentrated.

The wonderful thing with SAS, it supports all of these options because it is agnostic in its approach. And when applied to the appropriate method, it gives fantastic ROIT and great speed from other applications put into SAS 9.

I/O

Anytime a lot of data needs to be accessed and analyzed it must be read from disk into memory and processed. In the past, expensive disks where hooked up to channel and optimized the data flow to disk to memory processing, retrieving big chunks of data and whisking it into the CPU for the crunch.

Fundamentally that has not changed; however, we have different places where we store the data today. For example on RAIDed disks, and those can be Direct Attached, SAN, Network attached or even memory resident systems. The goal is always get the data into memory—now. And since SAS is ideal to crunch numbers it is essentially a heavy user of the platforms' I/O.

These techniques require systems that can pump a lot of data in and out of the processor. Therefore, in the principle of parallelism in the CPU, 'big iron' systems must have flexible I/O systems that can attach to various style of I/O premises set forth but are essentially capable of connecting BIG I/O pipes for the management of large SMP, large cache, and memory machines.

From a SAS perspective knowing the size of the SAS data set to be processed, gives guidance into how many parallel data pipes are needed. For example, if the entire data set could fit into the memory of a single computer with multiple CPUs, with large FP capability the user would experience the ideal of SAS analytics.

STORAGE TYPES

We often think of storage when we think of I/O because of the transfer of information to and from memory. I/O becomes the bottleneck between the disk and memory. In terms of disks, there are a couple of options and their relative importance to SAS applications to consider.

- Direct Attached (Dedicated) - Predictable performance
- Direct Attached (Shared) - Performance could be impacted by other hosts
- Storage Area Network – SAN - Performance could be impacted by other hosts; Zoning can help with predictability; Sharing and Protection by Host system Failure is an advantage
- Traditional Network Attached Storage (NAS) - Network mounted file system (i.e. NFS); Not for performance use
- Hybrid SAN / NAS Solutions / Clustered File Systems - EMC Highroad Solara, Veritas Clustered File System, IBM SAN-FS; File System Metadata is Via Network, Data is via SAN

So depending on the type of storage needed, the type of storage must be considered when thinking about SAS applications. Think a fast I/O as close to the memory as possible and minimize the distance between the disk and the data being read.

Another factor that will affect performance of SAS applications is the use of RAID. With Redundant Array of Independent (or Inexpensive) Disks (or RAID), there are a number of options:

- **Level 0 --** Striped Disk Array without Fault Tolerance: Provides data striping (spreading out blocks of each file across multiple disk drives) but no redundancy. This improves performance but does not deliver fault tolerance. If one drive fails then all data in the array is lost.
- **Level 1 --** Mirroring and Duplexing: Provides disk mirroring. Level 1 provides twice the read transaction rate of single disks and the same write transaction rate as single disks.
- **Level 5 --** Block Interleaved Distributed Parity: Provides data striping at the byte level and also stripe error correction information. This results in excellent performance and good fault tolerance. Level 5 is one of the most popular implementations of RAID.
- **Level 1+0 –** A Stripe of Mirrors: Not one of the original RAID levels, multiple RAID 1 mirrors are created, and a RAID 0 stripe is created over these. (often confused between RAID 0+1)

Network

As the user interacts with the larger backend machines with all the data, the flow of information to and from Client to Server is rapidly increasing. The days of the Video Terminal are back from a general principle, but the sophistication of the units make a huge difference. Therefore transferring the data from the server to the client results in heavy network traffic. Poor load balance means poor performance.

Consider the following. Does a user need the full function of a PC right at their desk (client) or could some of the functions be aggregated to work on a display system connected to a Citrix server handling many sessions for many users?

With Enterprise Guide, there is a sense you can have it both ways—the usability of the client interface with the power of a server performing all the work. The range of users begins to impact this infrastructure. The advantage is, the user is more mobile than before, the computer equipment is becoming less expensive but attention to the infrastructure is very important. For example, a heavy SAS user is probably going to want their own desktop. But a shared environment can be advantageous to medium load users. Because of all the options, VMware Virtual Desktop or HP's 1:1 PC Blades or Citrix Terminal servers become an interesting dynamic to the discussion.

SAS applications have a dramatic impact on usability performance (how long it takes one to do their work). What we have seen in our own benchmarking tests is that how SAS is accessed (which clients); given everything else staying the same, the SAS clients out-perform Citrix Terminal Server, Windows Remote Desktop and X-Windows by huge factors.

Management

When we think about SAS applications and their increasing complexity, we immediately think about how we support this environment. For SAS applications, there are SAS specific services that need managing and these include:

- Compute Services –SAS/Connect, OLAP Server, Integration Technologies (Object Spawner, Workspace server), SAS Middleware Server
- Application Services –SAS/IntrNet, Load manager, Java Application Servers, Servlet/ JSP, EJB containers (e.g., Web Services)
- Data Services - SAS/Share, SPDS, Relational and object data stores, SAS Metadata Server
- Security and Permissions - Application, user and data security
- Warehouse load process (incl. Scheduling)
- Data reconciliation and quality data
- Log maintenance

In addition, there are infrastructure components that need to be managed. These include:

- Processor load (concurrency and workload)
- SMP, MMP, dual core, quad core, etc.
- Memory
- Disk
- I/O
- Startup and shutdown
- Backup
- Failover, redundancy and load balancing

Some changes are occurring that add complexity to the overall control of the system. Balance of CPU to Memory to I/O has always been there and, what was there, needs to be processed. Keeping the machines busy against load is important. Today the research continues as WLM and GWLM come into the picture. Workload management is how one assigns pieces of the larger machine to handle various loads.

For example, moving and assigning numbers of cells, memory and I/O capacity amongst the jobs being run on big machines. The larger subject matter is under Partitioning, Clustering, and WLM. But we are moving to a more dynamic world in which loads are shifted, even amongst computing nodes.

Here we have combined the ability of a cluster of machines working together and have told each what percentage of the machine they can use. Then the GWLM comes along and redistributes the data. Move Application B to Node 2 and allow Node 1 to take on the whole load of Application A. Thus we continue to push the envelope in big iron on how machines are utilized. The better ROIT goes to those that use the power, rather than sit idle and just burn up electricity.

Partitions and Virtualization

The past few years have brought a greater focus on how to better use the power that already exists within the current architecture. This is reflected in a precept of better ROIT (Return on IT Investment). Often this leads to a discussion on partitions and virtualization. Although not contradictory, their application is applied differently in circumstances and bends their meaning.

The best know virtualization at the X86 level is being done with VMware. VMware allows multiple operating systems to run on its hypervisor, creating multiple instances of an application and O/S running on the same hardware. The O/Ss do not have to be of the same type or versions. One could run WINDOWS NT 51. with XP, with Novell Netware, and Red-Hat Linux all on the same box.

Therefore they are partitioned in that no instance of an O/S can harm another. If the NT 5.1 virtual system blue screens it will not take down all the other instances. Thus there is isolation of instances and each is protected from one another. This is becoming more complex as more Virtualization hardware is being inserted into the chip technology.

But we take a quick look at ROIT and we have taken multiple machines running at only 5% utilization and bumped them up to say 40% utilization, through the use of a hypervisor. This will work with SAS loads given the utilization of the Windows servers is light. If one has say 3 servers all going at 50% or more, it would be silly to virtualize such a configuration.

In regards to big iron, there is a slight change to Partitioning. The physical machine can be split up into different types of partitions—soft or hard for example, security. Users can be isolated or partitioned limiting access to the O/S, other users and applications. The machine can also be dynamically reconfigured in a different way on the next re-load to adjust to changing needs. Discussions on how and when this is done are beyond the scope of this paper, but several vendors have been doing it for years.

The capabilities and complexity of this falls in a large program scope called the VSE (Virtual Server Environment). This leads to architectural discussions and how SAS can take advantage of these great capabilities. Our perspective looks at different ways of slicing up a big machine, but there is also the consideration of Multiple Operating systems. Since SAS is O/S agnostic, this adds another dimension to the equation. Different copies of an O/S can be used to run functions of SAS to a particular advantage.

Examine the graphic below of Multiple Partitions being run on full 64-bit machines such as Integrity, and we see we can mix operating systems. HP-UX, Windows, Linux, and OpenVMS. The pictures below show that a number of flexible solutions are deployed using partition. But further use of even multiple copies of a single O/S can yield great performance gains. Notice the performance gain as multiple processors are applied with Partitioning capability. From 47 hours of work down to 32 minutes!

Virtualization

We have touched on virtualization above and it is best known in the X86 world. But virtualization is also applicable in big iron through the VSE program, but it does have a component similar to what a single machine does with a hypervisor.

A hypervisor (HP_UX) in the case of Integrity is loaded into the machine. Then on top of the hypervisor a guest operating system, such as Windows, or Linux, or HP-UX, or OpenVMS, is added. This is all under rapid development and it means over the next few years a big machine can be sliced up in new ways.

Why do we care? This is not for the faint of heart, but computing science marches on. The old ways are combined with new ways. But the focus is always on better ROIT. It is up to the providers of software like SAS to validate these technical features if they are relevant to their architectures and techniques.

Can SAS be supported in this manner? Currently, we think so, but the issue is one of licensing and how SAS software is purchased. It certainly is not a technical issue.

Choosing the Right Platform

Years ago it was just run SAS on a mainframe and we are done because it took the power and scale of such a machine to effectively run the models of the day. Evolution has brought us the power of a mainframe down to the desk so one can run some modules of SAS right in Excel on a Windows desktop. But can everything be moved to the PC desktop eventually? No.

Scale is still important. SAS can be split up in several ways. Put all on one big machine (like we used to do), or split some of the functions on a backend to do some heavy data lifting and analytics, while the front end drills down and data displays the results from the backend machine. This has been called multi-tier computing or client/server computing.

SAS is O/S neutral and writes for Windows and features rich UNIX systems. The possibilities become broader. Sizing can be a bit more complex, but it also deals with the functionality required and the amount of data being processed. Handling Tera-byte data sets is better addressed on very large memory (2 TB) machines with a lot of I/O bandwidth and processors.

Years ago where a mainframe was required, a small SMP system will do. Scale things down to a few Megabytes of data and a 4-way blade with 64 Gigabytes of memory will do fast processing of that data set. Therefore we mix operating systems, for example Windows client with a BIG Unix system or sometimes a Windows client with a Windows class server.

The list below demonstrates where SAS 9 can be run.

Support Platforms for SAS 9	**Exceptions**
• AIX • Tru64 UNIX • HP-UX • HP-UX for the Itanium Processor	Metadata server & OLAP Server • Tru64 UNIX • Linux for Intel Architecture SPDS • Microsoft Windows for 64-Bit

• Solaris • z/OS • OpenVMS Alpha • Microsoft Windows • Microsoft Windows for 64-Bit Itanium-based Systems • Linux for Intel Architecture • Linux for Itanium-based Systems	Itanium-based Systems • Linux for Intel Architecture • OpenVMS Alpha • z/OS SAS/Access Products • many exceptions Enterprise Miner • more exceptions

It is also important to note that for clients, some mid-tier components and some server tier components require you to read the fine print.

So what's the right platform?

Nearly a decade ago, we sought to answer that question (Nelson and Swirski, 1997) and came up with a number of factors that we think help drive you in one direction or another. These factors are partially technical (size of data and magnitude of the problem), some are what is supportable in your environment (such as in-house expertise and vendor support).

Summary

The good news is there are so many combinations that can be successfully deployed to make SAS perform well. The bad news there *are* so many combinations and SAS as a company remains agnostic, making the decision challenging. The burden goes to the user and those who have done it before. We have been doing shifts with SAS for many years and are sensitive to moving from one version to another and helping companies decide on the right platform. When scenarios are put forth and "Is this possible?" is asked, partnership kicks into high gear and evaluates the options quickly.

Today, working with Partners is becoming realized as more important. While any vendor always sees their systems as the most idyllic for a particular set up, the range of processors has grown and choosing is more complex than years ago. Before, a Mainframe choice was pretty easy to pick. But the power of smaller industry standard processors and their adoptions by companies like IBM, SUN, and HP has given pause to what is simple. Windows systems grow in strength and popularity.

The huge systems still have favor with large I/O, big memory, and cache systems. What may have been the simple choice is now fragmented and it could be possible to scale SAS modules onto a Windows Only solution. And the number of these for the average citizen is shrinking as we witness the processors shrinking (multi-core) over the next years.

Computing power continues to move. What was once only possible on a CRAY can now be done on a PC! The span of data to be manipulated on a mainframe can now be done with a 4-way Intel box. There must be an overall arching plan of how SAS is to be deployed and what modules are key elements or the work is most difficult. The benefit of a tool that can scale to all needs from intensive SAS users to lighter drill down users is the ability to pick a platform that can do it all and do it inexpensively.

Deciding the right SAS solution

With so many choices, determining the best architecture can be a challenge. It is a 32-bit and a 64-bit system working together.

Putting the hardware and software combinations on machines is costly. Let your partners do the work and come up with the recommendations. Yes, it is going to cost you some trust, but one cannot afford to do it on their own anymore.

References:

Nelson and Swirski (1997). Choosing the Right Platform for SASÒ Client/ Server Applications: An evaluation of factors in selecting NT and UNIX (http://www2.sas.com/proceedings/sugi23/Sysarch/p259.pdf)

Biography:

Greg Nelson, President and CEO

Greg has just celebrated his 20th year in the SAS eco-system. He started out as a social psychology student doing statistical analysis then quickly moved into applications development. Greg is the president and CEO of ThotWave Technologies where he supports an entire organization focused on helping customers leverage their investment in SAS. Prior to ThotWave, Greg spent several years in consulting, media and marketing research, database marketing and large systems support. He holds a bachelor's degree in psychology and Ph.D. level work in social psychology and quantitative methods.

John Loether, President and CEO

John Loether is a senior technology consultant for ESS Americas at Hewlett Packard Company. As a technology consultant, he lectures on many areas of computing technology, covering areas of operating systems, hardware platforms, and the ever-expanding range of computing-from today's handheld PCs to super-cluster machines, super computers, and fault-tolerant transaction systems (SMP to MPP).

He started in the early IT arena in 1964 at the University of Wisconsin and with E.I. Dupont. He has held different positions in sales training, marketing, and engineering. Much of his time spent in executing technical consulting sessions with customers on a worldwide basis. Loether has held seminars in Asia, Australia, New Zealand, Japan, England, Europe, Africa, and the Americas.

About ThotWave

ThotWave Technologies, LLC, is a Cary, N.C.-based consultancy and market leader in real-time decision support, specializing in regulated industries such as life sciences, energy and financial services. ThotWave recognizes the difference between simply accessing data and making data work for business. ThotWave commands the juncture of business and technology to help companies improve their operational and strategic performance. Through products, partnerships and services, ThotWave enables businesses to leverage data for faster, more intelligent decision making.

Contact information:

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

Greg Nelson	ThotWave Technologies, LLC.	greg@thotwave.com
John Loether,	Total Tec (formerly of Hewlett-Packard)	johnloether@charter.net

ThotWave Technologies, LLC

2504 Kildaire Farm Road

Cary, NC 27511

(800) 584 2819

http://www.thotwave.com