Paper 374-2011

A Practical Approach to Re-Architecting a SAS® Deployment

Jim Fenton, SAS Institute, Inc., Golden, CO, USA
Robert Ladd, SAS Institute, Inc., Phoenix, AZ, USA
Gary Spakes, SAS Institute, Inc., Cary, NC, USA

ABSTRACT

In today's business environment, enterprise computing deployments must be able to handle the challenges that companies face while adhering to IT standards. With the SAS platform being a multi-tiered environment consisting of components residing on client desktops, middle-tier Web servers, compute servers, and data assets, SAS customers are looking for ways to modernize their environment without breaking the bank. When looking to modernize your SAS environment, there are many things to consider. This paper takes a practical approach to re-architecting SAS environments. It identifies typical architecture scenarios while taking into consideration that each customer has unique needs. This paper can be used as a reference when looking to modernize your SAS environment.

INTRODUCTION

Traditional SAS deployments typically fall into one of three categories: desktop, server-based, or a combination of the two. These environments give developers and analysts the ability to create valued information quickly. While this information is vital to success, these environments are coming under more scrutiny as company standards and policies are more strictly enforced.

A common goal is to create a computing environment that can handle the challenges businesses face including increased data volumes, the need for more complex analysis, tighter processing windows, shorter development cycles, governance, and adherence to regulations. Often these environments are designed to meet the short term need and then require re-engineering.

To build an environment that is dynamic, organizations need SAS deployments that are flexible, scalable, able to improve performance, and increase productivity. This paper discusses typical SAS deployments including the SAS® Business Analytics Framework, additional SAS technologies that help modernize these deployments, and using these technologies together to create a competitive edge.

TRADITIONAL SAS DEPLOYMENT METHODS

For years, Foundation SAS has been deployed in the enterprise utilizing one of three methods:

1. Desktop SAS Deployment
   - This is the least expensive deployment method.
   - The SAS analyst has direct access to data from the enterprise for desktop analysis.
   - Analytical datasets are typically stored local to the desktop, or on a network file share.
2. Server SAS (Foundation)
   - A terminal emulator is used to connect and launch SAS on the server.
   - The SAS analyst has direct access to data from the enterprise to analyze on the server.
   - It is more expensive than a desktop solution, but computing capacity is increased to make this deployment faster and more efficient for large data volumes.
3. Desktop and Server Combination
   - Foundation SAS is installed on the desktop and on a server.
   - The SAS analyst launches the desktop SAS deployment to execute analytical processes locally and/or remotely on the server.
   - The ability to interact between local and remote SAS is enhanced by the ability to write code locally, execute it locally or remotely, and upload or download data.
   - The SAS analyst can work locally if not connected to the network.

Figure 1 illustrates these traditional SAS deployments with data movement represented by the arrow width.

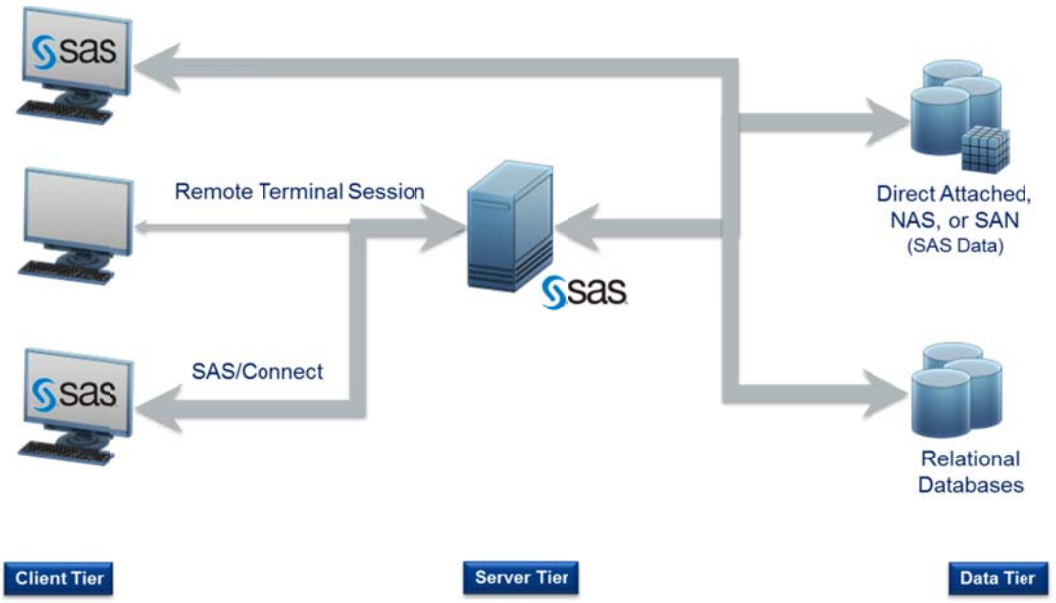

Figure 1. Traditional SAS Deployment Methods with Data Movement Across the Enterprise

While traditional deployments are still viable, they present challenges for both analysts and IT.

- The SAS analyst is limited to writing code. There is no code generating capability within the SAS® Display Manager.
- There is no visual process flow to help understand the detail analytical steps or an efficient way to navigate to a specific step where modifications are needed.
- All data must be copied to the Foundation SAS engine, whether on the desktop or on a server, before any analysis can take place.
- If SAS is on the desktop, data is moved across the enterprise network. Transferring large data volumes across the network reduces the analyst's productivity and impacts network performance.
- Desktop processing presents a scalability issue with data stored and processed on a local disk.
- Data used for analytics becomes stale over time.
- Data stored outside the system of record presents operation and security risks.
- In large deployments, there are multiple SAS licenses to be managed and maintained which can become an administrative burden.

SAS BUSINESS ANALYTICS FRAMEWORK

The SAS Business Analytics Framework (Figure 2) provides a tier-based architecture for analyzing data with both desktop and web-based interfaces. This framework offers a central point of control for access, configuration, and consistency. The fit-to-task desktop interfaces aid the SAS analyst with code generating wizards and a program editor with integrated syntax help for auto-generating code to analyze data. These interfaces can interact with each other in a collaborative analytical environment. In this deployment, data remains secure in the server environment and only moves to the SAS Business Analytics Server when needed. There is one SAS license to maintain for all users. Consolidating data and streamlining processes promotes efficiency and enables collaboration among SAS analysts.

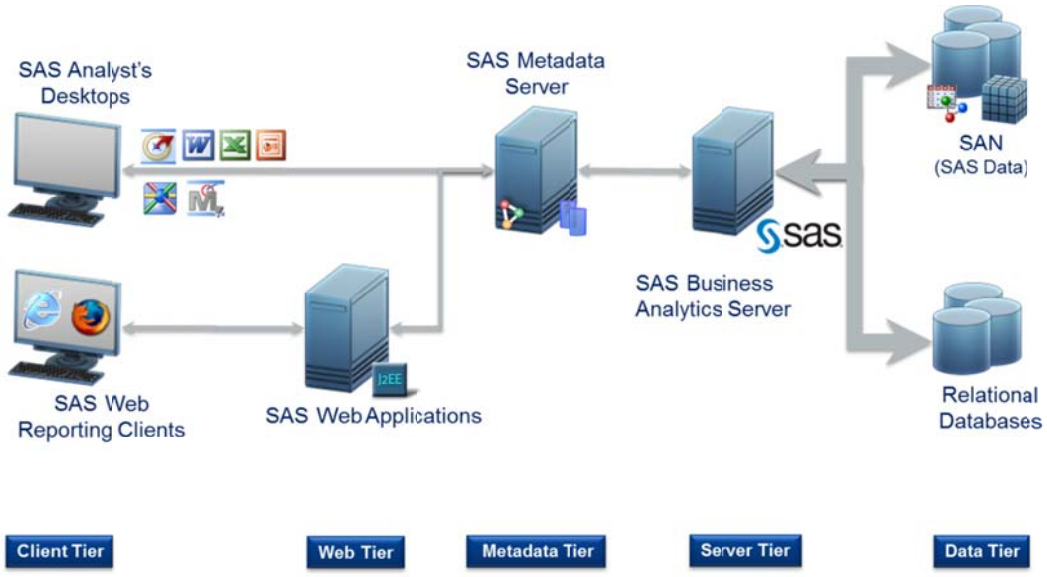

Figure 2. The SAS Business Analytics Framework

Many of these deployments utilize large SMP environments and customers are looking for cost effective options to scale and improve processes. The Business Analytics Framework is designed to be extensible, to align with other SAS technologies that can address scalability, reduce data movement, minimize data duplication, and improve workload management. These technology enhancements are covered in the following sections.

WORKLOAD DISTRIBUTION AND HIGH AVAILABILITY

Up to this point, the discussion has focused mainly on single SAS server deployments. Many large enterprises have more than one SAS server. These servers can be configured to work together to more evenly distribute the analytical workload and provide a highly available SAS analytical framework. A scale-out approach provides flexibility to add capacity as needed along with the ability to distribute workload across the environment.

There are several options for leveraging multiple SAS servers in a distributed environment. The first involves load balancing, where the workload is distributed across multiple SAS servers based on basic workload distribution logic. The second is a more robust workload and server management solution involving the SAS® Grid Manager.

LOAD BALANCING MULTIPLE SAS SERVERS

The SAS Business Analytics Framework utilizes SAS clients to develop process flows and code to be executed in SAS workspaces on the server. These workspaces can be distributed across one or more physical compute nodes defined in the SAS metadata. These servers form a cluster of available resources for executing work from the SAS clients. Once a SAS workspace is created and assigned to a SAS client session, it remains in effect until the client session terminates.

While this section focuses on workspace servers, it's important to note that the SAS® Stored Process Server, the SAS® OLAP Server, and SAS® Pooled Workspace Server take advantage of additional load balancing algorithms. For more information on these servers and load balancing options, see the *SAS® 9.2 Intelligence Platform: Application Server Administration Guide*.

Balancing the SAS workspace load involves a simple cost-based algorithm that defines a baseline cost for each workspace session and a maximum cost (total workspaces) permitted on each physical machine. The maximum cost per server parameter can be set at different thresholds to allow servers with different capacities to participate in the load balancing cluster. When a new SAS workspace is requested, it is sent to the server with the least number of clients connected to it. If all costs are equal, the workload is slotted to the first server in the list.

Incorporating multiple servers into a distributed environment requires connecting the backend data and database resources to all SAS servers in the cluster. The data still travels across the enterprise to the SAS servers for processing. Load balancing provides some high availability for the SAS workspaces and SAS® Object Spawner when multiple physical SAS server nodes are active and available.

The load balancing algorithm does not take into consideration how busy the server is at the time the workspace slot is determined. It is possible to place a new workspace session on a server that could be I/O, CPU or memory constrained. Figure 3 illustrates a load-balancing approach for a multi-server architecture.

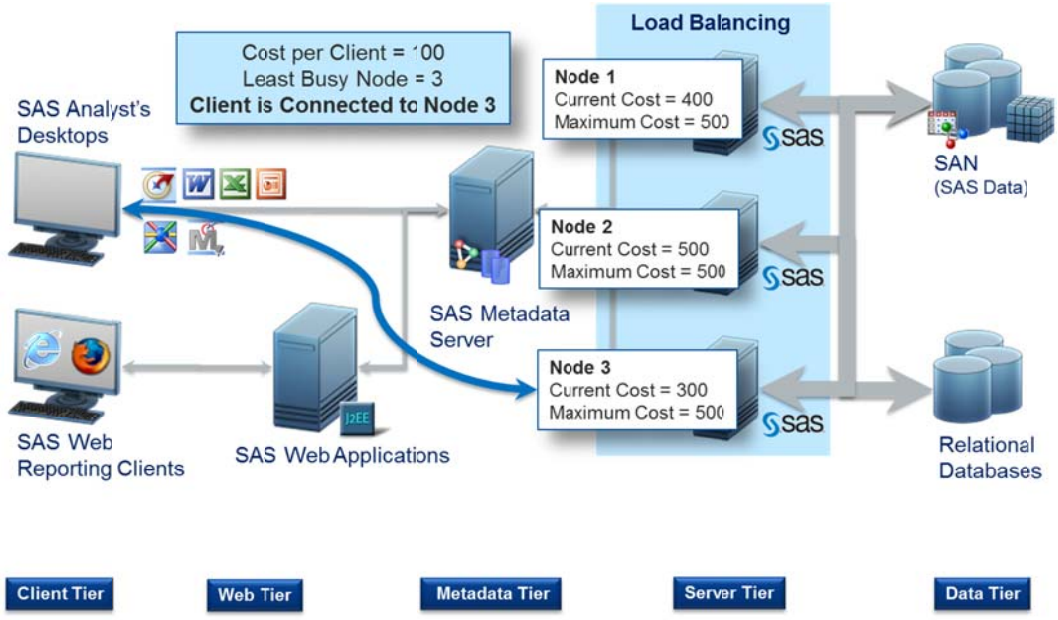

Figure 3. Load Balancing Existing SAS Servers for Workload Distribution

SAS BUSINESS ANALYTICS GRID FRAMEWORK

The SAS® Business Analytics Grid Framework provides a high available, robust workload distribution environment for executing SAS code using batch processes, SAS client interfaces, or SAS Display Manager across physical SAS servers. Figure 4 represents a high available SAS Grid environment where SAS computing tasks are distributed across multiple compute nodes in a network. SAS Grid characteristics include dynamic workload balancing, workload management, policy enforcement, and efficient resource allocation for SAS products and solutions. These are some of the many reasons why organizations are embracing this strategic technology.

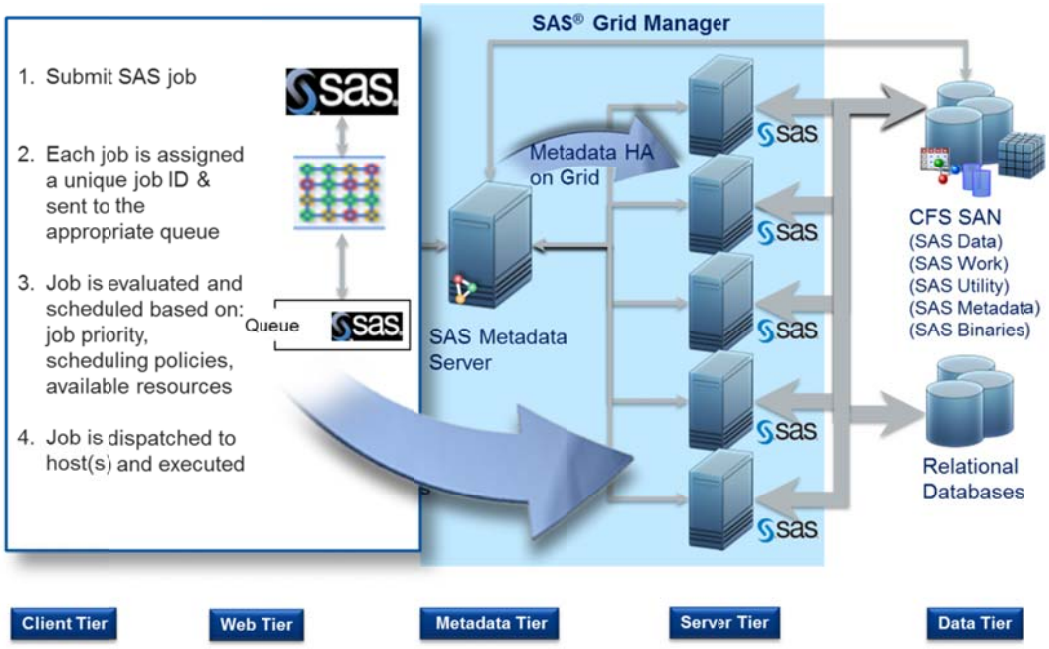

Figure 4. High Availability SAS Business Analytics Grid Framework

SAS Grid provides high availability for compute nodes, SAS servers, and SAS applications. From these perspectives, there is no longer a single point of failure. If a compute node goes down, there are others in the cluster that can take on the process. The same goes for SAS servers, such as the SAS® Metadata Server and SAS Object Spawner. If a SAS server process goes down, the grid platform has the ability to restart it on the same node or restart it on a different node. SAS applications can utilize check point restart or full job restart/re-queue if a job fails due to network or server failure.

SAS jobs are dynamically distributed across the cluster based on server availability, consumption, and job parameters. This provides a more efficient environment, taking into account jobs and server availability before dispatching the work. For non-sequential jobs, the processes can be broken apart to run across multiple compute nodes in parallel for substantial acceleration of the entire job or application.

Having the ability to put fences around a computing environment is a critical component in modernizing a SAS deployment, as it provides robust methods to regulate the use of the shared resources to deliver information faster and more efficiently. Workload management helps enforce company policies and provides a more effective processing environment. Some tactics utilized for workload management include job slots, queues, thresholds, and dependencies.

Job slots are specific to the number of processes that are allowed to run concurrently on a compute node. This protects the server from being overloaded and controls how jobs are spread across the computing environment. Queues play a critical role as a cluster-wide container for jobs. All jobs submitted to the grid go to a queue until scheduled and dispatched for execution. Queues provide control for processes within the grid environment. For example, a queue can be assigned a priority, resource limits, exception policies, scheduling policies, dispatch windows, and much more. Within the grid environment, thresholds such as CPU utilization, memory utilization, and I/O consumption can be set to limit how the server is utilized.

From an infrastructure perspective, adding capacity when needed is much more cost effective than planning years out. The ability to absorb and manage growing analytical needs and user community growth is critical as business dynamics change and challenges arise.

By reducing typical processing barriers, results are accelerated and decisions can be made faster. This opens the door to handle new growth opportunities and the ability for SAS analysts to engage in more complex and detailed analyses. Jobs that take days can now be done in hours, delivering timely information at the point of need and providing the ability to expand the analysis to areas that otherwise would not be considered.

REDUCING DATA MOVEMENT ACROSS THE ENTERPRISE

With increasing data volumes, it is a critical objective to reduce the movement and replication of data across the enterprise. SAS has developed and continues to design software solutions that reduce data movement. SAS® In-Database processing transitions the work done by the SAS engine into the relational database engine. SAS® Scalable Performance Data Server is a data server solution designed to work with SAS datasets.

SAS SCALABLE PERFORMANCE DATA SERVER

SAS Scalable Performance Data Server (SPDS) is a SAS software component designed to help SAS analysts working with large data volumes. This multi-threaded data server engine is optimized for SAS data and SAS analytical processes. The components of this server work to significantly improve the processing time of SAS data.

The data housed in this server has a smaller footprint than standard SAS datasets. The table data, metadata, and indexes are distributed across multiple file systems for optimal throughput. The data is indexed using a unique hybrid indexing schema, making this a scalable analytical data engine capable of processing large SAS data volumes faster than traditional methods. SAS Scalable Performance Data Server processes data inside this multi-threaded data server engine when querying, ordering, aggregating, and sub-setting. This ensures that the minimal data amount is returned to the SAS server. Transient SAS work data can be directed to this server to take advantage of the multi-threaded data server engine. Figure 5 depicts a typical architecture when SAS Scalable Performance Data Server is deployed in a SAS Business Analytics Framework.

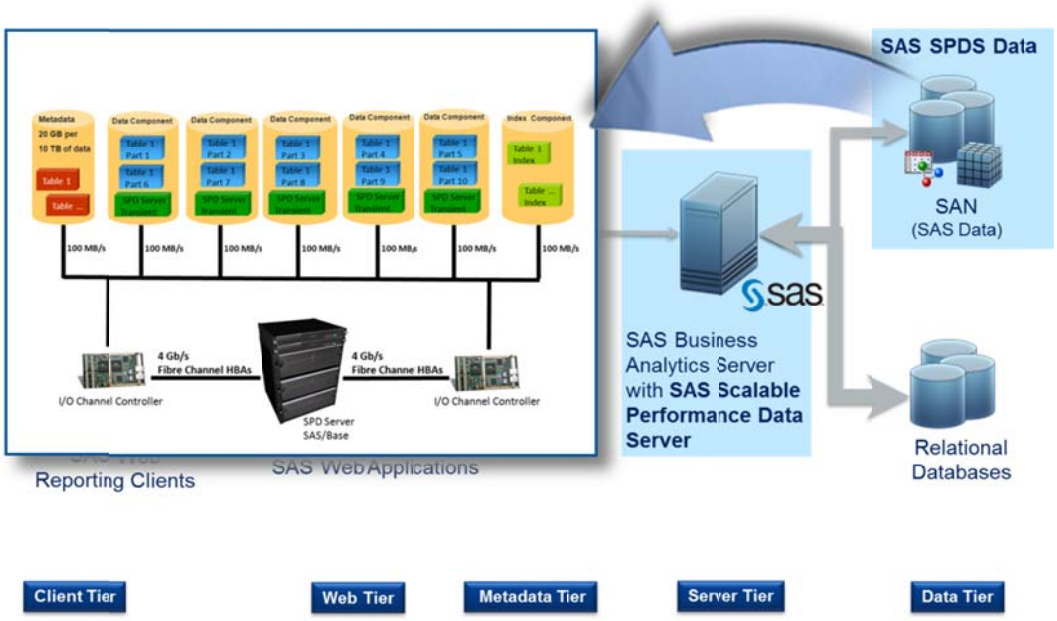

Figure 5. Optimizing SAS Data Movement with SAS Scalable Performance Data Server Processing

Large SAS datasets can be represented by smaller datasets joined together to appear as one. An example would be a separate dataset representing each US state, clustered to appear as one table containing all US states to the SAS analyst. When a query is executed, just the state datasets involved in the query are accessed. The query is executed in parallel and the results are aggregated within the SAS Scalable Performance Data Server prior to returning to the SAS session. This accelerates data processing as only the cluster members involved in the query are accessed.

SAS Scalable Performance Data Server contains a management and security component to secure and monitor access to data. Data from multiple organizations can be combined into this server and access limited as appropriate. Data access can be granted or denied down to the column level to reduce data duplication, yet allow access to the resources that the SAS analyst needs. This eliminates the need for multiple copies of similar data containing only a few distinct fields based on specific user, security, or analytical needs.

SAS IN-DATABASE PROCESSING

SAS In-Database processing leverages a relational database for what it does best, efficient SQL processing requests on massive parallel processing (MPP) architectures. SAS In-Database processing works to either move the complete SAS analytical process into the database or leverage the database to complete as much of the SAS analytical process that it can. Various SAS In-Database solutions are available for Teradata, Netezza, IBM DB2, AsterData, Greenplum, and Oracle database management systems. These solutions are in different development stages for each database and could include the SAS® Scoring Accelerator, SAS® Analytics Accelerator, Enhanced Base SAS® Procedures, and SAS® Format Publishing. Figure 6 depicts the SAS In-Database architecture with in-database scoring and SAS Formats.

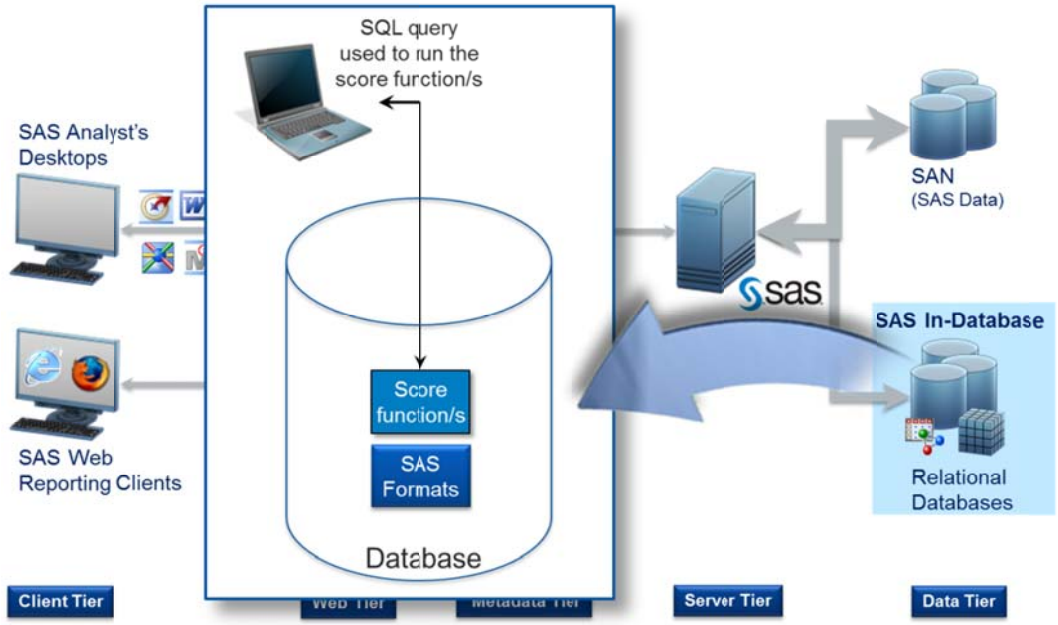

Figure 6. SAS In-Database Processing

Enhanced Base Procedures are SAS procedures that are automatically converted to SQL statements and submitted in the database. The frequency procedure (FREQ), summary (SUMMARY & MEANS), rank (RANK), tabulate (TABULATE), and sort (SORT) are SAS procedure examples that can be executed with just the result set returned to SAS for final output generation. This capability is enabled thru the SAS® Access Engine to the respective database.

The SAS Scoring Accelerator is designed to publish SAS Enterprise Miner models into the database as SAS Functions to score the data inside the database. Once the SAS model is published to the database, it is executed using native database SQL statements. This eliminates the need to export the data from the database, score it in SAS, then load the scored results back into the database or re-write the model logic in a different language. These functions operate at the ROW level. Individual records (scoring on the fly) or entire database tables can be scored using these functions.

The SAS Analytics Accelerator extends the modeling processes into the database. The data acquisition for modeling procedures like regression (REG), principal components (PRINCOMP), correlation (CANCORR), clustering analysis (VARCLUS), factor analysis (FACTOR), cross multiplication (SCORE), and forecasting (TIMESERIES) can be processed inside the database. The data preparations are completed inside the database with the requisite analytical "building blocks" needed for the analytical procedure returned to the SAS server.

SAS Format Publishing takes custom SAS Formats and makes them available to SAS programs as database objects. When a SQL step is executed and invokes a SAS Format, the operation is completed in the database rather than the data being extracted for SAS to complete the process and apply the format.

These in-database solutions significantly improve the SAS analytical process. The data no longer has to move across the enterprise before any analysis can take place as it leverages the database massive parallel processing architecture to complete the work faster. Jobs that took days can now be done in hours providing the ability to expand the analysis to areas that otherwise would not be considered.

BRINGING IT ALL TOGETHER

When looking to modernize a SAS environment, understanding where you are and where you would like to go is an important step in the process. Because business dynamics change rapidly, it is important to align with the business while considering architecture approaches that provide scalability and account for data duplication and movement. Figure 7 illustrates SAS Business Analytics Framework with all the technologies discussed in this paper.

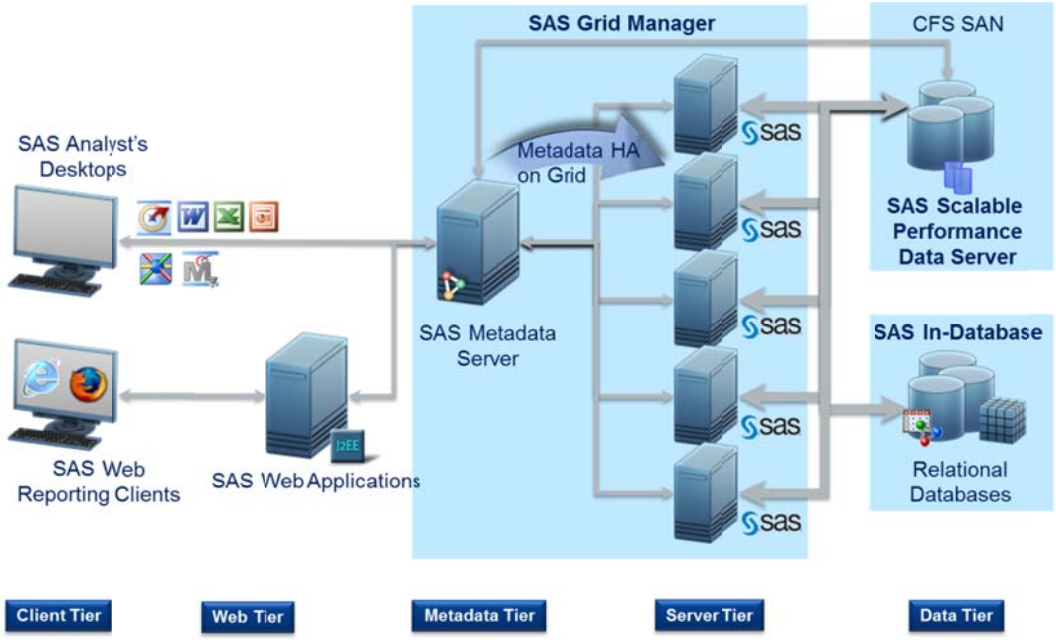

Figure 7. SAS Business Analytics Framework Leveraging In-Database Processing and SAS Grid Manager

Load balancing and SAS Grid complement each other to distribute workloads in a cluster of SAS servers. While grid helps intelligently distribute the workload across the different compute nodes, load balancing helps distribute the initial workspace sessions across the different compute nodes. As a stand-alone feature, load balancing is limited to simple algorithms for distributing SAS client sessions. With SAS Grid, a new algorithm utilizes the information from the entire grid environment to determine the least loaded machine to initiate the SAS client workspaces. The benefit is that server capacity is taken into account before a new workspace is introduced.

Incorporating SAS Scalable Performance Data Server into a SAS Grid environment delivers the ability to process SAS data quicker. This multi-threaded data server is a significant performance value over traditional single threaded SAS I/O processing. The ability to engage permanent SAS data along with transient SAS work data makes this solution a must for organizations processing large data volumes for analytics.

The last piece is SAS Grid and SAS In-Database technologies working together. SAS Grid dynamically manages and distributes SAS workloads to reduce processing time while in-database processing leverages the MPP database architectures for data management and analytics without having to duplicate or move the data. These technologies provide flexible scaling to deal with increase data demands, increased user demands, more complex analysis, and increased demand for more value. SAS Grid also provides the ability to enforce, track, and monitor policies while remaining highly available. Bringing all these technologies together creates a scalable and reliable environment for fast and efficient processing for both SAS and relational data.

The SAS Business Analytics Framework is designed to adapt as the organization changes. This paper discussed four common technologies that provide "snap-on" building blocks to enhance the SAS Business Analytics Framework: load balancing, SAS Grid, SAS Scalable Performance Data Server, and SAS In-Database. Separately, these technologies can significantly improve business processes. Together they have the ability to alter how business is done all together.

REFERENCES

SAS Institute Inc. 2010. *SAS® 9.2 Intelligence Platform: System Administration Guide, Second Edition*. Cary, NC: SAS Institute Inc. Available at http://support.sas.com/documentation/cdl/en/bisag/64088/PDF/default/bisag.pdf

SAS Institute Inc. 2010. *SAS® 9.2 Intelligence Platform: Application Server Administration Guide.* Cary, NC: SAS Institute Inc. Available at http://support.sas.com/documentation/cdl/en/biasag/61237/PDF/default/biasag.pdf

RECOMMENDED READING

- SAS® 9.2 Intelligence Platform, http://support.sas.com/documentation/onlinedoc/intellplatform/index.html#intell92
- SAS® In-Database Technology, http://support.sas.com/documentation/onlinedoc/indbtech/index.html
- SAS® Scalable Performance Data Server, http://support.sas.com/documentation/onlinedoc/spds/index.html
- Grid Computing in SAS® 9.2, http://support.sas.com/documentation/onlinedoc/gridmgr/index.html

CONTACT INFORMATION

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

Jim Fenton
SAS Institute Inc.
Denver, CO 80401
Work Phone: (919) 531-9761
E-mail: jim.fenton@sas.com

Robert Ladd
SAS Institute Inc.
Phoenix, AZ 85012
Work Phone: (602) 265-1616
E-mail: robert.ladd@sas.com

Gary Spakes
SAS Institute Inc.
Cary, NC 27513
Work Phone: (919) 531-5305
E-mail: gary.spakes@sas.com