Creating a Successful Support Model for the SAS® Enterprise Intelligence Platform

Jerry High, Blue Cross and Blue Shield of Minnesota, Eagan, MN

Abstract

SAS has a robust set of tools that provide business analysts many capabilities ranging from statistics and data mining to data integration and business intelligence. In the early years, prior to version 9, typical analysts would simply use the SAS programming language and write code to obtain the desired results. Support had two aspects: one from the systems administration team where the SAS software was simply installed and made available to the analysts, the other from the SAS community sharing their insights about business problems and coding techniques. Today under release 9, more specifically with the introduction of the SAS Enterprise Intelligence Platform, the capabilities have increased dramatically. So, too, has the need for a proper support model to ensure that today's sophisticated business analysts and other general consumers of information can leverage the power of SAS to the fullest. This paper explores many aspects to consider when developing a successful support model for a SAS environment.

Introduction

A successful support model is important to any company that wants to leverage information for better operating results. Delivery of information doesn't just happen. Analysts use various tools to acquire data and package it into useful information that is actionable. Decision makers of all levels in the organization need access to this actionable information. Effective delivery of this information is obtained through a combination of people, process and technology. A good support model is important to assure that these three components work effectively together.

A successful support model doesn't just happen. Much effort goes into creating one that works well for the user community. It takes time to develop and requires support from the organization within. It takes an understanding of the internal infrastructure followed by carefully identifying the role of the support team. The support team has a responsibility to the company to support the users with solutions that work for them while being careful to abide by the rules defined within the organization.

One thing to keep in mind, one size doesn't fit all. With the SAS Enterprise Intelligence Platform, there are many tools available to the user base. While SAS Enterprise Guide (EG) works for many power users, it may not be well suited for the executive who is looking for overall corporate results in a dashboard. In order to provide good support, the team needs to be well versed in the SAS tools and their capabilities as well as the business and their reporting/analytical needs.

Building a successful support model is an art not a science. What works for one company may impede wide acceptance for another. One knows it's successful when adaptation is growing and the business community is focused on the information obtained from the tools. If the support model is working, nurture it. If it's not working, change it quickly.

This paper begins with understanding the corporate environment and developing an identity for the support group. With that as a starting point, one can establish a core group to begin supporting the SAS environment. Once established, the team will need to define processes and controls to effectively manage the content, user roles and the overall framework. With that in place, it is important to have experts available to provide consultation to the SAS users and developers so that they can utilize the tools effectively. Finally, the team must constantly evaluate their effectiveness to find ways to improve the information delivery throughout the organization.

To further clarify building a successful support team, this paper will present what has been done at Blue Cross and Blue Shield of Minnesota (BCBSM). It is here that specific examples will be discussed that will highlight why it takes a conscious effort to build a good support model.

Corporate Information Goals

Before diving into hiring staff or reorganizing a team from within, one needs to first understand the corporate information goals. This is a broad question, but a very important component to understand. Building a support model that doesn't align with the goals of the organization doesn't have a chance for long term success.

A presumption is that most companies want more information from their data. There is more to this. Is the company looking for operational information for daily management of activities or are they looking to mine data for interesting trends? Do they want standard reports printed on a daily basis that are created by a production batch process or do they want to interact with the data and look at it in different ways that a printed report can't provide? Are they happy with their reporting/analytical capabilities or do they yearn for something better? Understanding answers to questions like this will help tailor the support model.

It's also important to understand the information needs from various areas within the company. One will likely find pockets of high end analysts who have everything they need and others who are starved for information to be successful with their work. It's not uncommon to find some groups who just want their daily/monthly report printed with the standard statistics they have used for years. Some may have the perception that getting a new report or new information is a major effort and are not even aware of the capabilities available today. Many times one finds those pockets of high end analysts are over burdened creating and running (manually) standard reports for other groups who don't have the expertise to mine their own data.

This assessment may also uncover some shortcomings or misperceptions about the information goals. Different segments of the company may be centered on data points or report layout rather than analytical and actionable information. The company could very well be data rich and yet information poor. This can then provide an opportunity to change/improve how data and information flows inside and outside of the organization. In all likelihood, there will be an interest to improve the capabilities of information delivery throughout the organization. With this in hand, one can begin to build a successful support model.

Strategy

With an understanding of the corporate information goals, one can begin to develop a support model strategy. This includes such items as a mission statement and a clear definition of the role this group is to perform. One needs to assure that from technical to business functions that various departments within the company understand the role of this team. The scope of responsibilities will likely determine which division this group reports to. For example, if the support includes components of the framework (e.g. software administration), this team would likely report into the IT division.

To assist in defining the group's role, begin with a list of all of the potential functions/roles that could be performed. Include functions of other departments initially. This allows an evaluation of the touch points or hand-offs that may be needed between groups. A list of functions might be similar to Table 1. More ideas on functional areas can be found in chapter 3 of "Business Intelligence Competency Centers" (see references section) listed at the end of this paper. This can be further enhanced by developing a SIPOC (Suppliers Inputs Process Outputs Customers) model so that all groups understand the roles of the support team. Which formal modeling process is used is not important, only that one clearly defines the roles of the support team.

Function/Role	Responsible Department	Coordination with
System administration	IT Operations	
Software administration	Support team	IT Operations
End user support	Support team	Help Desk
User education	Support team	Consumers
Corporate Data Warehouse	Warehouse team	
BI specific marts	Support team	Consumers & Warehouse team
OLAP cubes	Support team	Consumers
Processes/controls	Support team	Corporate Policies
Security	Support team	IT Security
Object/Code promotion	Support team	Consumers & Change Management
Audits	Support team	Internal Audit
Analytics	Business teams	
Enterprise BI applications	Support team	Business teams
Local BI applications	Business teams	
Ad Hoc analysis	Business teams	
Framework Architecture	Support team	Business & IT teams

Table 1

With the strategy and roles in hand the team can be formed. There is much to know and define about the SAS framework. One should get started with the strategy and be prepared to adapt along the way. As the technology and information strategy evolves, the team will need to evolve as well.

Technology

With the staff formed, one of the first things they will need to do is to understand the tools they have to work with. If one is preparing for deploying the SAS Enterprise Intelligence Platform for the first time, many decisions need to be made from what platform (Operating System) to single server or multi-server framework. These decisions will be influenced by the number of expected users and expectations of the framework. There will be trade-offs with each of the decisions. There is no right answer. For example, a multi-server landscape with the mid-tier separate from the SAS Foundation (SAS Metadata, Workspace, etc.) can help isolate and protect data while introducing complexities such as firewall rules between components. The key is that you have staff members who understand these implications and can work with the proper internal resources, such as the Security team, to assure corporate architecture and policies are adhered to.

For a larger deployment (over 250 users), expect to have a couple of the staff members who specialize in the SAS technology alone. That would include administration of all of the components such as Metadata, the SAS OLAP engine, the SAS Scheduler, and EG. It would also include any other products that may be part of the suite such as the SAS Scalable Performance Data Server (SPDS) or any SAS Access products to foreign databases like DB2 or Oracle.

If this function is to be maintained by another IT department (based on role definitions), be sure they understand fully the magnitude of the support that they are taking on and that they will be responsive to the SAS Support team. The other IT team will need to understand the nature of SAS and the Business Intelligence (BI) products. Many times, even I/O requirements are not fully understood initially by a Systems Administration team.

Aside from the raw technical components there are other administrative functions that can be tackled by the team. This would include functions such as Security. Depending on the company's security policy, one may be allowed to run the Metadata security independently from the centralized function. In other cases (usually larger deployments), the two teams will need to consider if Security Administration should use the SAS Management Console to manage security separately or if there is a centralized LDAP (Lightweight Directory Access Protocol) compliant database that could be read and then imported into the Metadata repository. Another function that needs to be clearly defined is Code/Report Promotions. Again the support team can manage this; policy may require this be done by a centralized team. If the support team is allowed to manage each of these functions, be prepared to provide audit trails to comply with corporate policies.

Processes/Controls

When looking at the technology component, one quickly comes to an understanding that a major component is not the technology at all. As mentioned in the previous examples with security and code promotion, roles may be dictated by corporate policy. This brings to the forefront a basic dilemma, balancing corporate policy with the business need to be more nimble in building reporting and analytical solutions. This can be difficult to relay to a business unit who has historically had SAS programmers who build reports on the fly whenever a request comes forward. Questions and challenges will likely come up when "restrictions" are presented to development.

Depending on the expected exposure/use of the SAS Enterprise Intelligence Platform, the controls could be very simple or very complex. If SAS is being used for a company with 20 analysts, very little controls may be needed. The group of analysts can self dictate much of what is done. On the other hand, if SAS is being used to produce enterprise level reporting solutions for a large company with hundreds of information consumers, many controls will be needed to assure order of the content.

Wayne Eckerson wrote an article called "The Myth of Self-Service BI" (see references section). He describes how the initial desire of a Business Intelligence tool (along with a data warehouse) was to allow users to have "complete and unfettered access to data without IT interference." Eckerson cites an example of a company that had 26,000 different reports for 450 active users. He essentially says that the power users, which normally constitute about 20% of the user base, tend to create many reports from various sources using definitions that tend to vary from department to department. The casual users (the other 80%) tend to find it difficult to find the right report and in many cases will find similar reports with different underlying definitions leaving in question as to which is correct for their needs. He concludes that report governance is important to a successful BI deployment.

Report chaos is one reason for implementing controls to the BI framework. The broader issue is risk. Controls need to be in place when the company would be at risk. If an individual analyst is doing data mining searching for anomalies to research further, the company is at little risk. If an analyst is producing a financial statement for an external client, that data has to be correct. If the data doesn't align with other reports for that client, problems may ensue. If that report has private information and is delivered to the wrong client, lawsuits or fines may ensue.

From the business perspective, if too many rules are in place, they cannot be effective in their use of the tools. Tools with too many controls have little chance of adoption or success. When developing controls in the SAS framework, one will need to look at the risk of the work being done and have flexible controls that relax as the risk drops. To get a sense of the factors to consider when developing controls see figure 1. The main point is that as the risks go up, so should the controls around the SAS framework.

Understanding that different types of work in the SAS framework may present different risk does become a challenge when developing processes and controls. The individual analyst running their own program or project within EG poses little risk. That same analyst may want to create a report using SAS Web Report Studio (WRS) with a SAS Information Map that is already defined and available. What if that analyst isn't aware of or doesn't know where to find that Information Map? Does she create a new one with her own definitions? What if she wants to publish the report through the SAS Information Delivery Portal to an external client? What if the underlying query invokes a bad SQL call to a remote database? All avenues of access need to be considered
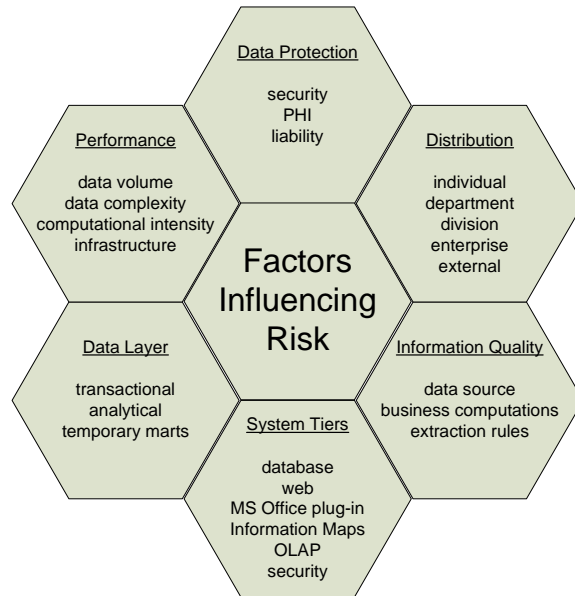
Figure 1

The great thing about the SAS framework is that objects are very portable or interoperable between the various components. The bad thing is it can be a control nightmare. An analyst can create a SAS Stored Process in EG and consume it through the web or the SAS Add-in for Microsoft Office (AMO). If that Stored Process has an explicit query to a remote database using the individual's credentials, what prevents a non-authorized user from opening a spreadsheet and extracting data they shouldn't?

To address these concerns, one could consider multiple SAS servers to segment the type of work. If SAS is installed on a UNIX server, partitioning will likely be available to create LPARS (logical partition) or domains that can separate ad hoc analysts using EG for their personal programs and projects from Enterprise level reporting applications in the BI space. This allows for different controls through different Metadata repositories while keeping the SAS footprint on a single physical server.

For reporting solutions, reports, macros, Stored Processes, Information Maps, OLAP cubes and other objects need to go through a certification process before they are deployed. Again consider the risk factors. As the exposure goes up so should the processes around certification. Again the individual analyst can create and consume their work, however, if they now want to use the newly created Information Map for the entire company that Information Map should go through a review process on the definitions, filters, and query for accuracy. It should also go through a performance review to assure that database and SAS server resources are not adversely impacted from concurrent use of the Information Map. To that end, depending on the size of the organization, one should consider additional test servers (logical or physical) to validate or even develop these objects and reporting solutions.

When all of the processes and controls are defined, they should be published. They can be in a public directory or company website. It's important to not only publish the rules, but the justification of the rules. If the various users understand what motivated the rules in the first place they are much more accepting. They may also raise issues that weren't thought of originally that would suggest modifications of the rules.

To assure compliance one should implement audit logging. At the very least one should have system monitoring to track the system health and find stray/rogue tasks. In some cases, the audit department may require validation that system and information access is restricted to the appropriate people. Tracking system usage also helps with planning for system capacity forecasting and is one metric of the success of your SAS framework.

All this work around the processes and controls may sound onerous. For an IT development team, they live through rules like these on a daily basis. For a business unit that just wants to develop a local reporting system for their own team to use, it truly is onerous. That's why one other dimension comes into play. As mentioned before, controls need to be based on the risk. The team that wants to develop a reporting system for a handful of users should have enough knowledge about the data and the generated reports to publish rather quickly without heavy controls. Again, this may suggest another logical server for the less risky content and development. This "local" development is controlled by limiting the number of users (exposure) and then shifts the responsibility from the support team to the business unit. If an "enterprise" level reporting system fails, the SAS support team is called in to handle the issues. If a local reporting system fails, calls are routed to the business unit that created the content. The SAS support team limits their responsibility to the server availability. Locally developed objects can be fast tracked as long as they don't adversely impact the servers that run them.

As mentioned before, server separation may be one way to assure that ad hoc programs don't adversely affect enterprise level reporting solutions. One last consideration is consumption of objects or reports. As an example, if a Stored Process is developed, has gone through the full certification process and has been deployed as an enterprise object, who should be able to use it? The answer is "anyone" (assuming proper authority). However, the Stored Process that hasn't gone through any certification should be restricted to just the author. This same logic applies to Information Maps, OLAP cubes, macros or any other object. To understand this visually, see figure 2.
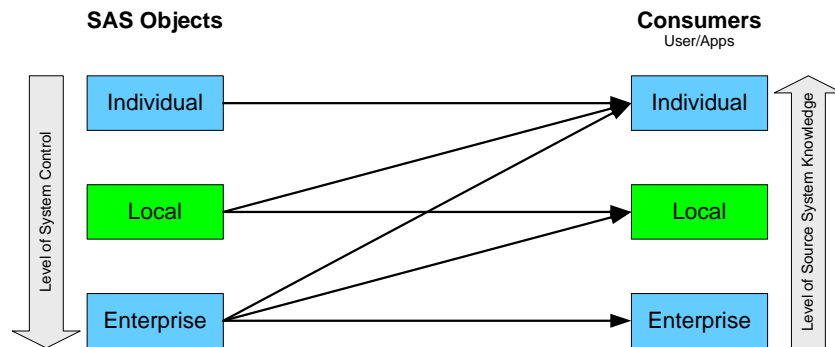

Figure 2

One thing to keep in mind here, if one uses a multi-server strategy to segment the local consumers from the enterprise consumers, then one may need to copy objects from the enterprise server to the local server to provide local access of enterprise objects. This now requires additional maintenance to assure that enterprise macros, stored processes and other objects are available in the local server. This just points out that each decision made will have trade-offs to consider.

The previous discussion should solidify that developing processes and controls is where the majority of the work ensues. In order to keep oneself from being overwhelmed, one should put together a list (see Table 2) of processes and controls that are needed and set out to define them incrementally. These controls should be done cautiously and with care even if it means being more restrictive at first and then loosening them later. Reigning application development practices in is much more difficult then allowing more freedom later down the road. Remember, these controls can differ between the level of exposure (ad hoc, local or enterprise).

Examples of areas that need defined processes and controls				
Security	Systems	Objects	Certification	Utilization/Audits
Users	Disk Space	Macros	Content	System
Roles	File Archiving	Stored Processes	Data	Users
Groups	Code Promotion	Information Maps	Functional	Objects
Objects	Query Time Limits	WRS Reports	Performance	
Data	Support Calls	OLAP Cubes		

Table 2

Education

The SAS Enterprise Intelligence Platform provides much more than the original Foundation framework. Users of Foundation components, who haven't used a BI tool, may not understand the capabilities. With processes and controls in place that may appear more restrictive, they may not easily adopt the BI mindset. It is very important to provide many educational avenues. This can range from simple demonstrations to full training programs done either internally or through professional training classes. In either case supplemental information is needed to educate the user/developer base on the controls and logic behind them.

Education can also take the form of internal user group meetings. The support team should host these on a regular basis. The team should encourage others outside of the team to present their solutions to business situations they have encountered. This should be an environment where questions are welcomed and encouraged. Nothing beats a group of SAS users collaborating to find solutions.

The support team should maintain a web presence (i.e. a department website) within the company. This should contain the processes and controls around the SAS framework, past presentations from user group meetings and describe what products are available within the company. Current SAS related links and planned system changes could also be noted here.

The support team should also have individuals that can consult with various business teams. The objective would be to listen to information delivery problems and recommend solutions or approaches in creating solutions and in some rare cases, suggest other products rather than a SAS BI solution. An example might be if there is a desire to post PDF content generated by other tools. Although SAS can manage external content, it might be better to use a corporate approved content management software package.

Education also takes on a broader meaning here. In some companies, how data is reported may need to be enhanced. As mentioned previously, one may uncover shortcomings when reviewing the corporate information goals. If there are ways to improve on information delivery, the support team needs to show the possibilities and work to enable more users to gain access to analytic and actionable information.

Continuous Quality Improvement

Once the support team is in place, the software is loaded, and controls are in place, it's not complete. All of these should be monitored for effectiveness. There is no support model that can start off working smoothly with everything in place. If user adoption isn't ramping up, explore what is inhibiting its growth. Interview users throughout the organization to understand what they feel is missing or not working for them. It may be that the initial deployment is way off course and there may be a few simple solutions to address the issues and increase user adoption. Don't be defensive, rather embrace the dialog. The engagement process alone has value in keeping close contact with the customers that are served by the team. If the users understand the reasoning behind the initial deployment and the support team understands what aspects are not working for the users, modifications can be made to enhance the way the framework is used.

Thomas Davenport, author of "Competing on Analytics" (see references section) describes how successful companies are using analytics to gain a competitive advantage. On a smaller scale the support model can be successful if they take time to create and monitor metrics. Which standard reports are being used and which ones are not? Are there more users of the BI framework than before? Are users still engaged (using the system) or has their initial interest fallen off? All of these can be measured either directly with system logs or surveys. Even feedback sessions or user group meetings can be venues to obtain valuable insight. Never assume that your support model is working, measure and confirm that it is. If it is not, make changes and measure again.

Another aspect to monitor is the content used in the framework. The entire suite of objects (e.g. macros, Information Maps, SAS datasets) should be tracked to control reporting chaos. If multiple teams have developed similar objects, review to see if they could be combined or consolidated. That holds true even for components that may be outside of the support team's core responsibility. One may find common business rules used across departments that should be integrated into a data warehouse environment rather than each group defining it at run time. One may find common data being joined together from desperate sources into a SAS dataset. Perhaps that data should, again, be integrated into a data warehouse. Addressing items like these can help improve the overall information delivery throughout the company.

Case Study – Blue Cross and Blue Shield of Minnesota

At Blue Cross and Blue Shield of Minnesota (BCBSM), we recognized the power that analytical and actionable information will contribute to the success of our company. In our minds, delivery of that information is an important component to the business. To that end we've named our department Enterprise Information Delivery. Our mission is to "Raise the level of analytic capabilities across the organization". To be clear, we do not perform statistical analysis, rather we have positioned ourselves to help those who analyze data to leverage the SAS suite of tools. We do have developers who do use Java and the SAS BI components to develop enterprise level reporting applications, but the reports developed for mass consumption are suggested via the business stakeholders. Also, even though we administer the SAS software components, there are other IT groups that handle the server hardware and operating systems, security and other system operational functions.

Our support model has three primary functions: Framework Administration, Development Center, and Competency Center. A sample list of the responsibilities of each group is listed in Table 3. It is interesting to note that these three functions are not pure from a staffing point of view. It is essential that each staff member has an understanding of all three areas and in some cases may fill responsibilities across these three groups. As an example, a business unit may want a customized Java interface with specific guided queries. The developer must understand how calls to the system are made and the impact it can have to the underlying framework. That same developer may work directly with the requester to assist in building requirements which would normally be done by the competency center staff. One of the keys to our success is the cross functional skills within the team.

Enterprise Information Delivery (Sample list of responsibilities)		
Framework	**Development Center**	**Competency Center**
SAS framework administration, maintenance and support	Development and maintenance of enterprise reporting applications and objects	Ensures that the company leverages the analytic capabilities

Maintain SAS system health through monitoring and tuning
Administer tools such as:
    SAS Scheduler
    SAS OLAP Engine
    SAS SPDS
    SAS Metadata
    SAS Enterprise Guide
Maintain remote database client connectivity
	Enterprise reporting applications
	Custom Java User Interfaces
	Enterprise objects such as:
	    Macros
	    SAS Stored processes
	    SAS Information Maps
	    Report templates
	    SAS BI data marts
	    SAS based services
		Governance of:
		    Overall architecture
		    Data marts
		    Processes/controls
		    Enterprise reporting applications
		User communications:
		    SAS users group
		    Department web site
		Engineer solutions based on business requirements

Table 3

The framework team is our first contact point for any SAS related issues. Help Desk calls or system alerts hit this group first. They assess the issue to determine if they resolve it or route it to others on our team. A simple inquiry requesting assistance with SAS BI or Foundation features is routed to the competency center. Application related issues go to the development center. Issues that are dealt with in other teams (e.g. security/password issues with SAS or relational databases) are redirected to other IT support teams. Each member of the framework team selected specialties, such as the mid-tier layer, Metadata, or the Scheduler. Working together the team can handle all of our SAS technical issues.

Next is our development team. Here it was important to distinguish our role versus other groups with locally built reporting solutions. This team is focused on reporting applications delivered to larger groups of the organization. The reporting/coding can range from simple WRS reports from Information Maps to complex Stored Processes using explicit SQL queries to large relational databases that have been tuned for web time performance. We even have customized Java web sites that capture parameters and pass the values on to SAS and the underlying Stored Processes that are executed. Many of these applications do have an operational nature to assist line managers with their daily work. When new reporting needs/projects get started, this team works closely with the business to understand the requirements and to tease out needed information that a standard IS development specification document might not include.

The competency center is a multi-skilled group. They must know the underlying technology and how to develop with the various SAS components. They must also understand the functions of the business and various data sources and when to use them. They provide advice for others who develop local reporting solutions and help to solve SAS

programming inquiries. Some "reporting" problems are not actually reporting in nature, but because SAS is a familiar tool, some will look to SAS as the solution. There are a few times when we may actually recommend a non-SAS solution. And finally, although all team members have responsibility to provide input to processes and controls, the competency center has oversight of these artifacts.

The team meets on a weekly basis to track various work efforts like system upgrades and project development efforts. Issues are discussed, priorities are adjusted based on business priorities. For the most part, resolution is left to the group. If needed, they can raise items to management. They have guidelines on expectations. As an example, they can develop support processes in any way they see fit. They just have to keep in mind that they have to live with the on-call support calls that may result. They also know that management will challenge them to always consider the business perspective.

Early and throughout this paper, I have used the terms enterprise, local and ad hoc. Power users who use EG and Foundation as their primary tool to analyze data are what we call ad hoc users. They represent the highly sophisticated users whose workload and analysis varies regularly based on shifting business needs. In the early days, that was all a support team dealt with. When the SAS BI platform was added to our environment we were not all that different than described in Ekerson's article "The Myth of Self-Service BI" (see references section). The initial belief was we could buy the SAS BI suite, install it and let the users develop freely. We quickly learned that this wouldn't be successful. And because we didn't have many processes and controls in place, we started with enterprise level reporting applications where we could leverage already established rules around IT development.

With enterprise reporting applications as our starting point, we were then restricted by non-SAS controls and code management tools, external security repositories, session time-outs rules and content switches. We adapted fairly well through a lot of work to integrate the SAS BI platform with these other tools so that other teams could promote reports and UI changes. We followed the corporate standards and development outside of our team was very limited. No business team wanted to be so encumbered so many never adopted the web based capabilities. Later we investigated more about the current IT controls and found out that as long as we followed the corporate policies and could provide our own defined processes with the required audits in place, we could be allowed to manage the SAS specific components independently. We still follow the standard processes and controls with Java components, but the blended approach has improved our ability to respond to the business. With this blended solution we are now able to set up a more streamlined method for local reporting solutions.

So now we have evolved to enabling small workgroups to build local reporting solutions. These solutions, once set up, will not need to go through a full IT promotion process, migrating reports and other objects through test, integration and certification regions before being deployed to the production network. Rather, there will be a "contract" with those teams to take responsibility for the content and performance of their reporting solution. We will still have oversight and promote their objects, but done more quickly. We're still involved so we have visibility to all solutions so that we can assess opportunities to improve the overall content contained in the BI framework. And any support call that comes to the support team can be routed to the correct business unit when appropriate.

Up next for us is implementing the SPDS. We recently added this to our suite of SAS tools. The goal is that by creating targeted data marts specifically tailored to BI and analytics, we will now have the opportunity to bring desperate data elements together that may not be available in a single source today. Our group is now working on new processes and controls that will allow us to leverage the OLAP engine with drill through capabilities into relevant detailed data on the SPDS. This should add a new dimension to our current capabilities and enhance our information delivery experience.

Conclusion

This paper outlined components of a successful support model. Each organization needs to understand their environment and develop a clear role for the support team. It appears clear that it takes a combination of people, process, and technology to build a successful support model. Of these, well defined processes and controls are usually the last to tackle, but likely most important factor in a successful support team and further a successful BI deployment, especially in larger organizations.

Building a successful model begins with understanding the company's information goals and continues with building a team based on a defined strategy. With the people, processes and technology in place, one needs to continually evaluate these to assure that they are meeting the stated goals. It is important to adapt quickly if the support model is not meeting the organizational needs.

Although a case study was presented, it should be thought of as an example of one possible approach. Each company needs to evaluate their own needs and tailor an approach that suits them best. The scope of this team should be formed to include or exclude functions based on the defined roles. Whatever the scope is defined to be, it

should include a competency center in some form. Whether as a stand alone component or integrated in the duties of the entire team, this function is another key component to exploiting and promoting the SAS BI framework.

Finally, the makeup of the support model, like analytics and information delivery, will always evolve and grow. The key is to embrace the change.

References

Miller, Bräutigam, Gerlach (2006) Business Intelligence Competency Centers. New Jersey: Wiley.
Ekerson, The Myth of Self-Service BI. Retrieved February 12, 2010, from TDWI web site:
http://tdwi.org/Articles/2007/10/18/The-Myth-of-SelfService-BI.aspx.
Davenport, Harris (2007) Competing on Analytics. New York: Harvard Business School Press.

Acknowledgements

Recommended Reading

Business Intelligence Competency Center. Retrieved February 12, 2010 from SAS web site:
http://www.sas.com/consult/bicc.html.
Business Intelligence. Retrieved February 12, 2010 from Better Management web site:
http://www.bettermanagement.com/topic/default.aspx?f=10

Contact Information

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

Jerry A. High
Blue Cross and Blue Shield of Minnesota
3535 Blue Cross Road
Eagan, MN 55122
Work Phone: 651-662-8548
E-mail: jerry_a_high@bluecrossmn.com