Paper 272-26

Enterprise-Wide SAS Administration is a Full Time Job—Here's How!

Paul D. McDonald, SPIKEware, Inc., Schaumburg, IL

ABSTRACT

There are many great things to do with the SAS Software System. The STAT package is full of toys for statistician's glee! The AF package can make a developer drool, and web-heads can use the IntrNet tools to really make their applications come alive!

This paper is about none of that, nor about any of the other neat stuff or cutting-edge technologies that the SAS Institute is developing. This paper is about making all that stuff work together effectively. Readers will learn how to design build a strong SAS infrastructure across multiple platforms, develop the basics of an in-house SAS Software training system, create an easy way for SAS to work with other applications, and prepare their system for new technologies that come out in the future.

In other words, this paper will show you how to be a SAS administrator and make people think you don't do any work because you do it so well.

Introduction

As a SAS Programming Consultant on various sites over the years, I have found that most sites leave a tremendous amount to be desired when handling SAS administration. Often times I find that my work as a consultant is hindered because of simple changes and upgrades that must first be done before I can work in my area of expertise. After visiting with several people across the industry, I have found that my frustrations are not alone.

This paper has come about largely due to the author's frustrations at various client sites across the country.

The paper is written in two parts:

- PART 1: WHY to have a SAS Administrator
- PART 2: HOW to have a SAS Administrator

<u>EXERCISE</u>:
Call your company from an outside line. Ask to speak to the company's SAS System Administrator. Do it again in two or three weeks and compare the results.

PART I: Why have a SAS Administrator?

The entire goal of a good SAS Administrator is to promote efficiency with the SAS Software System. This efficiency exists on the following levels:

- Programming Efficiency
- System Efficiency
- Training Efficiency
- New Technology: Change Efficiency
- Bottom Line: Financial Efficiency

This paper covers all these aspects of efficiency. But first, a word of caution:

$$(efficiency) \neq (effectiveness)$$

SAS Administrators need to understand that their entire purpose is efficiency, but never sacrifice effectiveness. The needs and methods for effective SAS Programming are discussed in many other papers and books, but is beyond the scope of this document.

Programming Efficiency

Consider the "nail gun" – a wonderful tool for the right occasion. Pictured here is the **Hitachi Nr83a Framing Nailer Round Head** power tool for carpentry work. The product description states:

- *Full round head 2" - 3-1/4" smooth, ring and screw shank*
- *Weight: 7.9 LBS.*
- *Air pressure 70-120 psi 3/8" hose air inlet*
- *Nail capacity is 64-70*
- *Used for floor and wall framing and general heavy duty construction, including sub-flooring, sheathing and heavy fence building.*
- *This nailer can use what is known as 16D short (3-3/8" nails). it is not rated for true 3-1/2" nails.*

This tool has a place in the carpentry field. It can be very efficient when building a house or fixing a roof. It is not recommended for cabinetry, dry wall installation, or for shoeing horses (HINT: see "hammer").

The same is true for SAS. Too many times in SAS programming (and the IT field in general) the wrong tool is used for a job.

In this respect, the SAS Software System is very similar to a Nail Gun. A properly-trained individual or team can use the tool to accomplish things at a rate that is near impossible with other simple tools like COBOL or C++. However, without proper training and in the wrong instance, someone can cause a lot of damage and possibly hurt themselves or someone else.

<u>TRUE STORY OR URBAN LEGEND?</u>
On one project a sales representative sold the client on completing a "data check" for new, updated calculations. The process involved looking at every record of the data and comparing it to its appropriate counterpart from the prior month's data. If every variable on the record was an identical match, the program would not "re-load" the data. If one variable was different, then that record would be loaded. Later on (also sold as an "efficiency") the design called for specific numeric variables to be checked within a certain "fuzzy range" for a significant change.

The end result was that the each variable was checked to see if it should be calculated. 90% of the time it cost much more CPU processing to complete the check, and the other 10% or so of the time the savings for doing the check was negligible.

System Efficiency

Simply put, ask the question: do I have enough hardware to handle the type of work that needs to be done? Am I processing too many records for the system to handle? Can I complete this job at another time of day where more resources are available?
These and other questions are essential to ask in designing your system. Contact your SAS Institute Representative for the latest system requirements for the SAS System.

Remember the physical hardware limitations of your site as well as other system usage. For example, if you are working on an MVS system and only have four tape mounts, do not try to merge five tape data sets together in one data step. Although I'm not sure what would happen, I can tell you this much: "It's not good."

Training Efficiency

Get people to write SAS code effectively and to meet standards. This is not an objective or a goal, it is a commandment. I quote:

"Thou shalt haveth thy standards of methods, and thy methods shall bear the fruit of greatness so that the multitude should reap the blessings and rewards"

--2 Opinions 2:4-5

The absolute easiest way to support and enforce a system of standards is to develop an in house user group. Through such a user group, you will find that the users will truly <u>want</u> to follow standards! Most programmers would love to follow standards at a site, if someone would just tell them what they are. Since most sites do not have these standards (or if they do they are not enforced) that leaves much extra "back-end" work to support an application.

Through various site testing, we have found an in-house user group (IHUG) functions best if it meets bi-weekly on company property for a brown-bag lunch training. This makes the IHUG "easy to start, easy to run, and rewards that will last forever."

<u>POP QUIZ</u>

1. **How often do you have meetings for your in-house user group?**
2. **Do you have an in-house user group?**
3. **Would you like one? (HINT: The answer is YES)**

To find out more about starting an in-house user group (and to get lots and lots of free stuff) contact the SAS Institute and ask for the "User Group Support" team.

There will be more on training later on in this paper in Section II Part B: Developing Training Systems.

<u>TRUE STORY OR URBAN LEGEND?</u>

At a non-profit organization, a programmer was told to print out labels for a mailing list in alphabetical order. He would get in very early and could be found at his desk, peeling labels off the sheet and re-sorting by hand the labels in order as requested. He was thankful when shown that "PROC SORT" thing.

New Technology: Change Efficiency

SAS Version 6.07 (that's a lot of floppy disks to install…)
SAS Version 6.08 (This is cool)
SAS Version 6.12 (This has some neat stuff)
SAS Version 7.0 (what was that one?)
SAS Version 8.0 (Almost got it installed…)
SAS Version 8.1

If the above list of releases from SAS do not bring to the top the true need for control over new technology additions, try this very typical example of short-term history for a typical SAS System site:

- **PLATFORMS**: PC, UNIX
- **SYSTEMS**: Windows 3.1, Win95, Win98, Win2000, HP-UX, SUN-OS
- **SAS VERSIONS**: PC 6.x, PC 8.x, Open Systems 6.x Open Systems 8.x

	N Op Systems	N SAS Systems	N Combinations
PC	4	2	8
UNIX	2	2	4
TOTAL	6	4	12

A total of twelve different combinations of possibilities for SAS Users—and that's if there are no cross-platform applications at the site (meaning a SAS job is either on PC or UNIX but never crosses using FTP, SAS/Connect or other multi-platform utility). With the potential for cross-platform management, that would leave **a potential thirty-two possibilities to manage**.

It is obviously easier to manage ONE possibility as opposed to THIRTY-TWO. The need for new technology change efficiency becomes obvious from this example, and the need compounds as the SAS Software System expands to interact with more outside utilities.

Bottom Line: Financial Efficiency

Just in case it has not become abundantly clear, resources cost money. The managerial question then becomes: "can the money be allocated before it is saved?" To answer this question, I turn to Bill Good of Bill Good Marketing, Inc.

"I am now going to answer the age-old question: 'Which came first, the chicken or the egg?' The answer is: 'It was the chicken.' This means that you've got to hire the help first if you ever want to be able to afford the help. The easiest way to make sure you can never afford help is to never hire any."

--Bill Good, Bill Good Marketing

In this statement, Bill Good was addressing securities and investment salespeople's need to hire assistants. I propose that this same chord rings loud and true for information systems.

<u>TRUE STORY OR URBAN LEGEND?</u>

A client needed HR Benefits data on a large division that they planned to sell to a major corporation for over four billion dollars. Although the data was mixed in with other reports, it was easy to place a "where" statement on the pre-written report and a four billion dollar decision was moved forward by five days.

What measures should you use when deciding how much to pay for system administration? Can it be outsourced to a third party vendor or should it be completed in house by full-time staff? How much time should the administrator devote to the process? Also, be sure to ask around to your SAS users and find out what challenges they face.

Informal surveys have shown that the SAS Technical Representative at a site is either a SAS Programmer who is busy with other activities and doesn't have time for the need, or is a non-SAS technical person who doesn't understand the need in the first place.

When asked about the time allocation of SAS system support, several of the technical contacts who have a SAS background reported as high as 30% of their time was used to support the SAS Software System, and that they wished they could devote more time but they needed to complete other work.

Resources worth having are worth managing.

EXERCISE:
Announce an emergency meeting for all users of SAS Software at your company. Count the number of people who attend, and compare that to the number of licenses your company pays for.

PART II: How to have a SAS Administrator

Okay, so recognize the need for a SAS System Administrator to make a focused effort on managing your SAS resources. The key question becomes: **"NOW WHAT??"**

Here's the steps:

A. DECIDE, COMMIT, and gain BUY-IN to have one
B. Develop a Training System
C. Mold New SAS Coders and Users
D. PC/Network Administration
E. Mainframe Administration

These five steps can be much more difficult than they first appear.

A. DECIDE, COMMITT and GAIN BUY-IN

The bottom line here is that if the leadership of the company does not see the benefit of having a SAS System Administrator, your company will not have a SAS System Administrator. They approve budgets, they write the checks.

Identify key individuals at your company who should make the decision. Contact SAS Technical Support for assistance, and begin a focused plan on creating leadership "buy-in" to the administration program. The exercises in this paper are a good places to start for ideas.

"Everything rises and falls on leadership"

--John C. Maxwell

EXERCISE:
Give the same assignment to three SAS programmers in different departments. Compare the results.

B. Develop a Training System

Here are six simple steps to developing a training system:

1. Create an In-House User Group
2. Put Your People on SAS Mailing Lists and Listservs
3. Have ready access to manuals or manuals online
4. Have ready access to SAS Support on the web
5. Have an Electronic In-House Support System
6. Use The SAS Institute

These six steps are very simple. And can be accomplished with very little capital and effort. The electronic in-house support system can be as simple as a Lotus Notes or Outlook discussion folder for topics, ideas, and quick tips that apply specifically to your site. Additionally, confidentiality is normally not an issue because the conversations are stored on an in-house server.

EXERCISE:
Read the entire section of PROC FREQ in the Procedures Guide and see if you really do know everything about it…

IDEA

Have plenty of material available. Books, SAS Institute training, and more. Subscribe to every newsletter available on SAS programming. Try making your own from time to time.

IDEA

Rip articles out of magazines on programming and drop them on the desks of staff members with a paper-clipped list of everyone in the department with your name at the top—have them scratch their name off and pass the article on to the next person.

IDEA

Send someone to SAS Institute Training. Have them present what they learned the following Friday at lunch.

IDEA

Assign a section of the SUGI26 PROCEEDINGS to each SAS Programmer. Have them review and bring to the IHUG any useful material.

EXERCISE:
Go through your supply of SAS Programming books. Throw out all SAS Version 5 manuals.

C. Mold New SAS Coders and Users

Three simple steps, repeated over and over again, can lead to molding new staff (and old staff) to meet the standards (expect resistance at first). These steps are:

1. Control the Point of Contact for New Staff
2. Set up the New Staff Member's System Immediately and Intently
3. Have a New SAS Staff Training Program (even for the "old hats")

Control the Point of Contact

When a new staff member comes to the company to use SAS Software, the SAS System Administrator must know about it and be in control. Few administrative hurdles are worse than a new person coming in and making changes without regard to the present system. Since this author is usually the one to commit such an offence, let it be known that the consequences to the system can be dramatically damaging.

EXERCISE:
Contact your human resources department and apply for a job at your company as a SAS Programmer.

New Staff Member System Setup

Not only must the SAS Administrator know that the new person is coming to the site, they must set up the system immediately! The fast and focused effort will emphasize with the new staff member the importance of proper SAS Administration at the site and will build a sense of pride in the site itself.

SAS Training

SAS Institute offers a great collection of training courses. Ignore them at your own risk. Use them and gain the reward.

EXERCISE:
Get a copy of the SAS Training Catalog. Take a course you have no interest in.

D. PC/Network SAS Administration

Managing multiple users of the SAS Software System on a PC Network can be a headache. Some sites have as many as 500 (and some have even many more!) installations of PC SAS. This paper shows you how to begin effectively managing an extensive system of PC SAS.

This system assumes the following:

- You are operating in a PC SAS environment
- Each SAS programmer has read-write access to one unique network drive
- SAS is installed locally on each user’s PC

The Secret: AUTOEXEC.SAS

The AUTOEXEC.SAS file is a SAS program that runs when each new SAS session is started. Using this utility, you can then have the SAS supervisor check for and install any required updates and maintenance automatically.

THE LOCAL USER’S OPTIONS

A “DO NOT MAKE ANY CHANGES BELOW THIS LINE” comment should be included, and all administrative statements should be entered after that comment. This way, the user can use the AUTOEXEC.SAS without modifying the administrative capabilities.

CREATING THE SASUSER TAG

Each SAS User needs to have a unique identifier of some type. The easiest way to do this is to create a macro variable &SASUSER using a %LET statement in the AUTOEXEC.SAS.

```
   %let sasuser = PMCDONALD ;
```

This username must be unique among all SAS users on the system.

EXERCISE:
Ask all SAS users at your site to send you a copy of their *autoexec.sas* file.

SETTING UP A MACRO LIBRARY

I have found it useful to have all standard macros stored and compiled on the local drive. I have also found it extraordinarily helpful to create a MACRO library, where all site-standard macros write their datasets. This prevents any overwriting of work datasets of the same dataset name.

```
   libname macro ‘c:\sas\macros’ ;
```

Write your site-standard macros to only create datasets in the MACRO library (unless the dataset is expected output of the macro).

EXERCISE:
Look out on the world wide web for Michael Friendly’s SAS Macro collection.

RUNNING A NETEXEC.SAS

The next step is to have every users’ AUTOEXEC.SAS program call a unique SAS program using the %INCLUDE statement (referred to as NETEXEC.SAS).

```
   %include ‘k:\sasadmin\netexec.sas’ ;
```

In this example, we assume that the drive ‘k’ is a network drive.

The NETEXEC file is then functioning as a “common” AUTOEXEC.SAS program. The NETEXEC.SAS program can then be used to do the following:

- Set up the NETEXEC libraries for the NETEXEC process
- Verify the active status of the SAS User
- Check for the need for any system update
- Perform any needed system update
- Set up site-standard libraries and catalogs
- Release the NETEXEC libraries upon completion

SET UP THE NETEXEC LIBRARIES FOR THE NETEXEC PROCESS

The first step of the NETEXEC.SAS program is to set up any of the NETEXEC libraries required. A simple LIBNAME statement should suffice:

```
   libname netexec ‘k:\sasadmin\data’ ;
```

The NETEXEC library should hold a SAS dataset named NETEXEC.SASUSER. This dataset should contain a list and description of every site-registered SASUSER. The following variables are recommended:

- SASUSER (an equivalent of the &SASUSER macro variable)
- STATUS (ACTIVE/INACTIVE flag for the SASUSER)
- UPDATE1-UPDATEX (one Y/N variable [or date if you prefer] for each of the required system update)
- LAST_DT (last access date)

VERIFY THE ACTIVE STATUS OF THE SAS USER, CHECK FOR THE NEED FOR A SYSTEM UPDATE, AND PERFORM THOSE SYSTEM UPDATES.

This process is simple: create a DATA _NULL_ that scans the NETEXEC.SASUSER table for the status of the user. The source code might look something like this:

```
   data _null_ ;
      set netexec.sasuser nobs=nobs ;
      if sasuser = “&sasuser” then do ;

         if update1 = ‘N’ then do ;
            call execute (“%include
   ‘k:\sasadmin\updates\update1.sas’ ;”) ;
            update1 = ‘Y’ ;
         end ;

         if update2 = ‘N’ then do ;
            call execute (“%include
   ‘k:\sasadmin\updates\update2.sas’ ;”) ;
            update2 = ‘Y’ ;
         end ;

         stop ;
      end ;
      if _n_ = nobs then do ;

         put ‘ERROR: You are not a registered
   SAS User at your site.  Please see the
   administrator.’ ;

      end ;
   run ;
```

Here are the critical parts of this data step:

1. If the entire dataset NETEXEC.SASUSER is processed without finding a user, a non-registered user routine can be activated. This can be anything from a NOTE or WARNING message in the log to shutting down SAS using ABORT ABEND.
2. Each time a new UPDATE is added to the system, a corresponding IF statement must be added to the administrative data step. The CALL EXECUTE call routines are used to INCLUDE appropriate files that will then process the necessary updates.

EXAMPLE: Here is an example of an INCLUDE file to process the update for the update for SAS/ACCESS for MICROSOFT EXEL97. This particular update came as a nice executable file, called SASEXEC97.EXE. A simple X command is used to run the update. The %INCLUDE file would contain the following source code:

```
   options noxwait xsync ;
   x 'k:\sasadmin\updates\sasexec97.exe' ;
```

This particular setup program from the SAS Institute runs nicely—it's a nice little "follow-your-nose" setup routine. However, not all setup routines are that nice. For more complex routines, try writing some Windows HELP files to guide the user.

Other ideas include modifying the user's local AUTOEXEC.SAS for any administrative changes, copying macro catalogs to the local drive, and updating a local drive SASNEWS file.

EXERCISE:
Look at what is in your SASNEWS file.

SET UP SITE-STANDARD LIBRARIES AND CATALOGS
If there are any standard LIBRARIES and CATALOGS for the site, this is the place to set them up with a LIBNAME statement.

RELEASE THE NETEXEC LIBRARIES UPON COMPLETION
Use a LIBNAME CLEAR statement to clear the NETEXEC library for the remainder of the SAS Session, something like this:

```
   libname netexec clear ;
```

Including this statement ensures that SAS USERS will not inadvertently modify any administrative files and datasets.

MAINTENTENCE REPORTING

Simple maintenance reporting can be completed daily, weekly, or as desired by the SAS System Administrator. These reports can be "who is on the system" and "who needs an update."

EXERCISE:
Give the same PC production report to three programmers and ask them to run it. Compare the results.

E. Mainframe Administration

Mainframe Administration is initially much simpler, but long range can have massive implications. The key factors here reside in the changing version of SAS Production Code and the SAS Software System itself.

Issues covering changing versions of the SAS Production Code (meaning, SAS Code written at your site by your staff) are covered in other areas under such topics as "change control" and "migrating to SAS Version X.x" and are beyond the scope of this paper. Such topics tend to be very specific in nature.

EXERCISE:
Ask around for a copy of your company's software standards testing manual or change control documents.

Maintaining the SAS Software itself on a mainframe environment is a different story. It begins by asking several questions that reside in the area of trust:

For example, when a new version of SAS Software shows up at your site, do not replace your current version without completing several tests of critical SAS systems.

An additional example is that when a new version of SAS Software shows up at your site, do not replace your current version without completing several tests of critical SAS systems.

And finally, when a new version of SAS Software shows up at your site, do not replace your current version without completing several tests of critical SAS systems.

(AUTHOR'S NOTE: I know I basically wrote the same thing three times in a row… it's really important!)

TEST TEST TEST

When installing a new version or release of SAS Software (this goes for PC as well as Mainframe) remember that it is thoroughly tested by The SAS Institute. Next, remember that The SAS Institute doesn't have your code and your data. So TEST!

Assuming that a new version of SAS will work seamlessly on your present system can literally lead to doom. Set up a TEST ENVIRONMENT and complete the new installation there. Run PARALLEL TESTING of several key jobs. Compare CPU processing time, output, and the log (DO NOT FORGET THE LOG!) If done correctly, any unforeseen problems can be handled appropriately. Otherwise, the unforeseen challenges will run the risk of causing severe and uncorrectable disaster.

EXERCISE:
Look for documentation of known issues that occurred the last time a SAS System Upgrade was completed at your site.

UPGRADE UPGRADE UPGRADE

For every site that does not test enough and upgrades too soon, there seems to be another site that never upgrades at all! This is a shame—think of all those programmers at SAS Institute slaving, working to make the system better!

Let me be clear: There is a REASON that a TS-Release or new version comes from SAS Institute. The reason is: "It's better." Use it. Make sure you have the safeguards pointed out above in the "TEST TEST TEST" section, but use the most recent release possible.

EXERCISE:
Check the TS-release of your most recent version of SAS. Call SAS Support and find out if you are using the correct release and version.

Conclusion

I have found PC SAS Site Administration (and mainframe administration, for that matter) to be widely ignored. Yet, it is a vital part of providing quality results in the data analysis field.

These ideas can make your site run more smoothly and make supporting a large or small number of SAS users much easier and therefore much more likely to be fulfilled. In turn, the end results and output created by the SAS site will be more robust, more reliable, and less restrained.

And that makes using SAS better for everyone!

Notes

Part II Section D (PC/Network SAS Administration) is based on the paper PC SAS Administration: Making Your Ship Go presented at SUGI25 by the author.

Special thanks to the following:

Bill Good of Bill Good Marketing, Inc. as quoted in "Pick the Cherries Not the Pits" seminar http://www.billgood.com/.

John C. Maxwell of Injoy, Inc. as quoted in various speeches and training seminars. http://www.injoy.com/.

Hitachi and Carpenter's Tools for permission to feature the **Hitachi Nr83a Framing Nailer Round Head**. Carpenter's Tools http://carpentertools.vstorehardware.com/.

Contact Information

Paul McDonald is the CEO/President of SPIKEware, Inc., a SAS consulting company based in Schaumburg, Illinois. He has been coding in SAS since 1993. He holds an AA in Engineering Sciences from Cloud County Community College in Concordia, Kansas, a BA in Physics from Southwestern College in Winfield, Kansas and an MBA in Finance from Keller Graduate School of Management in Chicago, Illinois.

Paul enjoys speaking and presenting, and is ready to entertain as well as educate.

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

Paul D. McDonald
SPIKEware, Inc.
1450 East American Lane Suite 1400
Schaumburg, IL 60173
Phone: (847) 330-4460
Email: PDM@SPIKEware.com
Web: http://www.SPIKEware.com/