

SAS as the Information Systems Developers Tool

Linda Lilla and Paul Pochardt, Piedmont Municipal Power Agency, Greer, SC

Abstract

This paper chronicles how a small (2 person) IS Department has utilized SAS® to produce a complete information delivery system. The general ledger is the core of the Financial Management and Reporting System with budgeting, fixed assets, payroll, journal vouchers, accounts payable, billing and accounts receivable feeding into it. This paper will go into some detail as to how this system was designed, along with other systems in our operation, using SAS. We will briefly relate our experiences developing a SAS based system on one hardware platform and porting it to another.

Introduction

SAS is well known and has built its reputation as a top of the line statistical analysis package. But what does it have to offer us applications programmers? When my boss, a SAS "data-step" expert, started talking about all the wonderful new features that the next release (6.06) of SAS would have, and suggested we use SAS as our database and programming language for a fully integrated financial management system, I decided I'd better find out if, within two or three standard deviations, he was out of his mind.

As an ex-Cobol programmer, I initially balked at the thought. My analytical mind went into a tail spin at the thought of all those data steps, not to mention the required screen interaction. We are a two person Information Systems Department, so our resources are limited and it was clear that if we went with Cobol, we would need to use contract programmers. Even with that we would still have the problem of selecting a database. It is important to mention that as we were negotiating these decisions, we were running on a Prime, which at the time was the target of a hostile takeover. This made portability across operating systems very important.

Our Wish List

Along with portability, we had several other basic items we considered necessary. At the top of our list, just under portability was competent, knowledgeable technical support. Having experienced the technical support (and I use the term loosely) of other software companies, our experiences with SAS were truly to be appreciated. When SAS said they would do something, they did. You always got an answer in a reasonable amount of time. Our experience with other support was to be told it could not be done - figure it out ourselves - call them back and tell them how to do it. Other things on our list that led us to choose SAS as our development environment were the ability to share information between departments, a convenient means of data entry, verification and retrieval, along with the ability to graphically display information. We needed an integrated database to eliminate the need to convert data between products, something that was feature rich, and easy to use, something that would provide a real programming language.

With the introduction of Screen Control Language®, SAS became a complete fourth generation programming language

and competitive database. SCL gave us the basic tools we needed: drill down menus, the ability to open and close multiple files, and screen building capabilities. These were augmented with a multitude of other features making it possible for one person to design, code, test, and implement a sizable system.

Where We Are

Piedmont Municipal Power Agency has a fully integrated Financial Management and Reporting System (FMRS) entirely designed in a flexible, manageable, maintainable way using SAS. It features a drill down menuing system where accounting personnel are guided step by step through each task. The monthly financial reporting module features a user definable and modifiable report generator. With the general ledger at the center, accounts receivable, accounts payable, payroll, journal entries, budgets, investments and financial reports combine to produce information necessary for our finance department to function. But how did we get there?

Where We Were

The first step in achieving this goal was to look at what we had and determine why it was not meeting our needs as a Joint Action Agency. PMPA had previously purchased a turn key system on the Prime computer. The Finance Department soon discovered it did not meet the needs of our particular type of accounting. We have our own chart of accounts, but we must also conform to the FERC (Federal Energy Regulatory Commission) chart of accounts. In addition to this we cannot close the previous year until we receive true-ups in April of the current year. Yet we must produce financials for each month in the interim. We also have a stand alone investment system and a payroll system that needs to post monthly to the general ledger. Our complex billing system was already written in SAS, so it required only modifications to produce transactions to post to the general ledger. In addition to dollar amounts, our Finance Director wanted to post and maintain year to date figures for kW and kWh's. These numbers are budgeted and reported just as dollar amounts.

How We Got There

Once we determined what PMPA needed, the systems design work began. Because everything had to work together, it was important to determine what fed information into what and which information was needed before another step could start. Our Finance Director defined a new chart of accounts, and each of these accounts had to be mapped to a FERC account number. Budget figures are broken down to the account level, while a range of accounts are summarized to produce monthly Financial Statements and Quarterly Reports. Once a month was "closed", it was necessary to keep the transactions in detail, permitting us to post additional transactions to a past month and re-close months in succession or even re-close a year to carry forward the balance. The ability to post to any selected month and year is accomplished in SCL by allowing the operator to enter the month and year to post. This is then concatenated to the prefix "tran" and that particular file (tran0193, tran0294, etc.) is opened. If the file does not exist, it is created using a model

file. The end of year general ledger file is archived as gl1993, etc. and can similarly be adjusted and re-closed generating a new glfile to start 1994. The ability to have several files open and accessible increased the usability of SAS exponentially.

On the Prime we were not able to use SAS macros, so along with SCL, we discovered macros. These saved a lot of duplicative code through out the system. One of our main uses of macros is to verify account numbers as they are entered onto the screen. The macro checks to see if the number entered is a valid account number and if it isn't, sends an error message to the screen and places the cursor on the account number in error. A macro is also used for automatic posting to a subsidiary ledger. If an account number is entered that normally posts to a subsidiary ledger a macro is called that pops up a small window and lets the operator indicated whether to post this amount or not.

Because of the number of reports required by our Finance Department, the link statement became one frequently used. Top of page procedures were set up as subroutines and "linked to" when the linecount maximum was reached. Link was also used to clear the screen, reset arrays and perform other repetitive tasks.

Generating checks with SAS was a new experience, but the Accounts Payable system runs smoothly and gives our accountants plenty of flexibility. They can enter check information at any time, and even go back and revise it before printing and posting. PMPA generates manual checks (typed) between check runs. These are entered into the system for recording and posting. When it is time to print checks, a report is generated listing the checks to print and control totals. At this point, the option is available to print, or cancel and make necessary adjustments. Once checks are printed, there is still an option to reprint if there is a problem before posting takes place and an auto void routine to void a string of checks if the printer decides to eat them. To augment the check routine, the Finance Department has the ability to clear checks and reconcile the bank statement, get a listing of all outstanding checks, and request a list of all checks written to a particular vendor, or a list of all checks written in a certain time frame.

Don't Forget the Old

Leaving the fancy new stuff aside we had to resort to using the "data-step" expert for help in our report generating routine. Our monthly financial statements seemed to change with some regularity, adding and deleting lines each month. This always resulted in IS involvement and a program change. We decided it was time to give the Finance Department their independence; so we designed a quasi-report generator using SAS. The columns across remained stable, but the lines changed. A SAS dataset was used to define each line on the report. The description to print on the report is entered. One option is to enter the number of spaces to indent the description. Other options include the ability to underline the amount, skip a line before printing, print a dollar sign and reverse sign before printing (for credit accounts). A total level is entered to automate the subtotals and final totals. Also left flexible are the general ledger accounts that are summarized to make this line in a financial. Up to five ranges of numbers can be entered and the "like" statement is used to match on certain digits of an account number. The Finance Department has access to these datasets through an edit screen to make

their revisions. This report definition file is read with a data step which produces self generating code to select accounts off the general ledger according to information in the dataset. A miracle (and several merges) occurs to get the budget figures for these accounts and format the lines according to the report definition, and the financials (against all probability) are produced on our laser printer, suitable for publication.

And We Ported It

As mentioned previously, we began developing our FMRS on the Prime, being fully aware it would be ported to another hardware environment. After investigating LAN's and several hardware vendors, we chose IBM's RS/6000®. SAS software and the RS/6000 seemed to compliment each other. We tested SAS's portability between the Prime and RS/6000 by PROC CPORT®-ing a small system from our Prime and PROC CIMPORT®-ing it onto an RS/6000 at our local IBM office. It worked just as advertised. Datasets and catalogs converted flawlessly - the only changes we needed to make were the path names in libname and filename statements.

To facilitate a smooth conversion, we had both the Prime and RS/6000 operating simultaneously. Development was halted and catalogs and programs were ported to the new system. The final conversion of datasets was done over a weekend along with the re-wiring of the PC's. Our staff came in on Monday morning, turned on their PC's, logged on with the same ID and password, and saw the same SAS menu screens in their applications. Except for the increased processing speed of the RS/6000, SAS kept everything looking the same.

Other Uses of SAS

PMPA operates a Municipal Load Management System along with a Demand Side Management System to reduce our cities' demand at the time of Duke Power Company's peak. PMPA's cities are billed coincident with the Duke peak, so usage reduction at this time saves our cities over one million dollars each year. The SAS system serves as the basis for each of these systems

In Conclusion

We don't use all the products available from SAS, but we do employ quite an impressive suite of them. Our use of these products has allowed our IS staff of two to fulfill the informational needs of a \$800 million a year utility. Without SAS, we estimate that a minimum of six people in our Information Systems Department would be necessary to develop and support our Agency's needs.

SAS, PROC CPORT, PROC CIMPORT, SCL are registered trademarks of SAS Institute Inc. in the USA and other countries.
IBM and RS/6000 are registered trademarks of International Business Machines Corporation.

The authors may be contacted at:

Piedmont Municipal Power Agency
121 Village Drive
Greer, SC 29651
Tel:(803)877-9632