PANEL DISCUSSION OF SAS USERS UNDER VMS OPERATING SYSTEM

- Louise Horny - Program Resources, Inc. - will be speaking, from a user's point of view, on the transitioning process from IBM to VAX
- Brad Reese - Loyola College-graphics
- Gerald Perry - Celanese Corporation-statistics
- Kevin Mallory - Texas Instruments - performance

Louise Horny

I am going to talk about a system at the National Institute of Environmental Health Sciences at the Research Triangle Park. This is a pretty large and impressive place now, but a few years ago we were in temporary buildings and trailers. Our own computing facilities have grown from a PDP1170 to a couple of VAXs. We are a branch of NIH, located in Bethesda, Maryland. Most of our computing was done through the Division of Computer Research and Technology at NIH. We use the IBM 370 in Bethesda, and our turnaround time is very fast. We have have lots of memory, and if something runs out of space, you just add more. But we are interested in a VAX because this capability is in Bethesda, Maryland, and we are in North Carolina. We are at the mercy of phone lines and the weather between here and Bethesda. For other reasons, it is nice to have the system in the location where you are working. It is also nice if you don't have to trust a tape with your data on it to the mail system. These are the reasons why we were very interested in developing our own computer facilities.

As we have grown, we have had even more need for it. We now have two VAXs and hope to be getting a third pretty soon.

Because we are right next door to SAS Institute and were interested in getting the SAS System on raw data set called RAWDATA. Then, when you get into the SAS session and and the logical file name is in the INFILE statement, the SAS System knows where to go. If you are like me and forget things, you might get into the SAS System and realize that you did not define your logical file name. When I made this slide I thought it was defined; in the early version I couldn't do this. This is a statement that allows me to execute a VMS command that defines a logical file name and tells it where to go to find the data. There are several other statements that we mentioned this morning that are now being allowed.

If you are in a SAS DATA step and your data are already in a SAS data set, you would use a SET statement to find the data. And again, the SAS code looks a lot alike, but instead of pointing to a DD statement you have to tell the SAS System where to find the data set. You can do this in several ways. The LIBNAME statement tells the SAS System where your directory is, or you can use a VMS command ($ specifies the VMS command), which tells VMS to set your default directory or work in the directory [USERID.SUBDIR]. With the IBM there was a right way and a wrong way, and that's it. With the VAX you can do the same thing several different ways. I did not realize that, and I used to do all of them. I think there are some things that I could leave out now, but what the SAS System looks for is a data set in your subdirectory called, whatever the name is.SSD. That means it is a SAS data set. The OS data set, which can have several SAS data sets within it, really doesn't exist. You have a separate file for every SAS data set.

There are some other bugs that I like to think of as ladybugs. We are glad to see them. For instance, now we get subscripted arrays, and we have been promised n-dimensional array handling in the future.

A lot of the SAS code looks the same. I learned there wasn't going to be any JCL, and I thought that is wonderful. But then the question is, how do you know where the data are? In the DATA step you still have the INFILE statement. It looks just the same, but instead of pointing to a DD statement, you're pointing to a previously defined logical file name that tells you to look in a subdirectory [USERID.SUBDIR] and look for a the VAX since we have a number of SAS users, we were an alpha test site. We started with Version 4.04, and we just received Version 4.06 about a week ago. I wish I could talk more about 4.06 specifically because there are quite a few differences, but most of what I will be talking about is 4.04. However, I don't want to talk a lot about bugs found on that because it is obsolete.

For instance, I had a problem with PROC FREQ. If you want to specify a number of tables and you have several variables, such as B1 to Bn, then you have to put your parentheses around it. I reported this as a bug. I found out that I just hadn't read the right page of documentation.

With input statements, you have to put a decimal if you are specifying a decimal format for numeric variables.

This slide shows that if you use the sorting order of character variables, the order is different. If you are counting on it being one way--don't, because it is going to turn out the other way. I can't remember which way is which, but the programmer just wants to know which way the order is going to work.

There was a question this morning about transporting SAS code, and one of the things this points out is that you can transport your code from IBM to VAX, but it may not run. There are little, things like this that have to be changed. So, if you bring your code from the IBM system and put it on your VAX, you are going to get error messages.

The SAS System was slow on our system. We run quite a few processes and, again, I am talking more from my experience on Version 4.04. But SAS was slow, and I didn't like to run interactively. I am used to using batch mode on IBM. One reason I did not want to run interactively is that I sometimes process large data sets, and I would get logged off because I ran out of CPU time. So most of the time I used batch mode on the VAX. This was too bad because we were not taking advantage of the interactive capability of the VAX system. Therefore, I have now gone to batch mode and that means I set up a command file that defines where my data set is, sets a default, and tells the batch processor in what directory I want the command file to run. The underlined is the statement and we have a logical called SAS that is basically the command to execute SAS; the command tells the batch processor what the job name is (the .SAS is not necessary). It is going to look for a file ending in .SAS, and then you can add a PRINT command. You start accumulating a lot of files and the VAX does not replace files; it just creates new ones. You have to do some careful file maintenance. You have your SAS command with the SAS statements, a log file, list file, and then a command file if you use batch mode. There is also a job log, which is what you get in the main directory.

We did not like writing our own command files, so we had one of the systems people write a command program for us. We call it SAS Submit. It allows options, like whether you want to be notified when it ends, whether you want a batch log, what you would like to name your list file, what you would like to name your log file, whether you want the list file printed, and so forth. This has helped people who just want to write their job, say SAS Submit, give it the file name, and it runs.

We have heard a lot about the display manager. I tried it, and I think it is a caterpillar because it has the potential to turn into a real butterfly. I have not used the SAS System interactively very much, and I understand that it is much better in the 4.06 Version. I am looking forward to trying it out, getting used to it. We will see how it goes.

The SAS System is improving on the VAX, but it is not the product that we know and love on IBM. It may be that the goal to see no evil is impossible, but I think most of the bugs are getting out. I think it is going to be a good product.

I wanted to plant a question, but I didn't have the nerve. The question was whether or not the graphics were done with the SAS/GRAPH product, and they aren't. The reason the graphics are not done with the SAS/GRAPH product is because we are not able to send to a queued device, and my pen plotters are queued just like a line printer. I have been promised that in a later version I will be able to do it. I could get the graphics on the screen but not to my plotter.

Brad Reese

Before I begin my discussion of the SAS/GRAPH product and some of my experiences with it, I would like to tell you about my environment in terms of the machine and my personal experience with it.

We have a new VAX 11/782. On this machine we have roughly 2000 users (2000 userids) and approximately 1000 faculty members, but this seems to be increasing.

Our 782 has two RE81 disk drives and RM80, which reside in the system and in the SAS System for that matter. I do my graphics using a Tektronix 4105, plot it on the 4695 color printer. We have Tektronix 4012s, 4006, a hard copy unit, and some ADM3As. They seem to work with the 4012 protocol. We did have Version 4.04 in operation, and my total experience is in SAS 4.04.

In the VAX world, Version 4.06 came in with a bad tape. The day before I came up here the correct tape arrived. I came from an IBM CMS and VSI installation, where I used the SAS System extensively, and I was very happy, especially with CMS. Therefore, I think I appreciate the SAS System on the VAX all the more.

I'd like to talk about some of the problems I had with the SAS/GRAPH product using Version 4.04. SAS Institute says that all these bugs have been fixed, and they give a brief outline of the new SAS/GRAPH product features you can expect on Version 4.06. There was a problem with the FORMAT statement within the PROC GPLOT, GMAP, or whatever in Version 4.04. I tried to use GPLOT to show a ratio of terminals in use in overtime in an installation and tried to get my time variable to show up using the TIME. format, but it did not work. After going through several layers of consulting at SAS Institute involving phone calls to and from the Institute, I found out that I had to put the format in the DATA step rather than in the PROC step. I have been told this has been fixed in the latest release. I discovered an interesting bug when I was trying to do the U.S. map (state parks per state), which is an example in the SAS/GRAPH manual. When I put the format in for the PKFMT, which is park format that you create using PROC FORMAT, it went into an infinite loop. Fortunately, I stopped it in about thirty minutes. When I took the format out, however, it worked fine. Of course, I did not have formats. When I moved the format up to the DATA step in that particular one, for some odd reason it did not pick it up.

I also encountered a problem with GCOUNTOUR, the cowboy hat. It worked fine until I inserted the pattern option, and at that point, it sent an incorrect escape sequence back to 4105, and I had to use the reset button to get it to work again.

The good point is that SAS Institute claims these problems have been fixed in Version 4.06.

Let me tell you about some of the features of 4.06. GMAP has a couple of new options, one is called XSIZE= and the other is YSIZE=. These features let you control and adjust the amount of space used by GMAP when you create the graph.

GPLOT has some new PLOT statement options. One of the options, FRAME, encloses the axis region with a line in the axis color. The CFRAME= option lets you specify a color to fill in the area within the axis and fills the axis region with a solid color. The SKIPMISS option causes discontinuity at points with missing values when you join points.

GCHART has some new GBAR and HBAR options. VREF= has been augmented to have more than one number as has HREF=. GCHART also has some new pie options, and these are only applicable to the pie chart. One of these options, MATCHCOLOR, lets you color the text with the same color as the pie slice. There is a GROUP= variable that creates one pie for each value in the GROUP= variable. The ACROSS= option specifies the number of pies across a page. If you have several pies in a group, you can specify the number across. Of course, another option, DOWN, lets you specify the number of pies down a page. You can specify enough pies to make them illegible if you want to.

There are some new pie labeling methods. The NAME= lets you format a value for the pie variable. The VALUE= gives you the frequency, percent, sum, or mean. The PERCENT= lets you determine the percentage of the total pie the slice represents. These methods are OUTSIDE, INSIDE, ARROW, or NONE. OUTSIDE lets you specify the text outside the slice. For example, if you put NAME=OUTSIDE, it places the name outside the slice. INSIDE allows you to put the text inside the slice. ARROW puts a name in the margin and puts an arrow pointing to the slice. NONE specifies no text. There are a couple of new map data sets. One data set is called U.S.CITY and has the location of U.S. cities on both the U.S. and the state maps. U.S. CENTER is the location of the visual center on state maps; it is handy for putting on the state label.

ANNOTATE is probably one of the neater applications. I was looking at bumblebees and so forth. With ANNOTATE you could have drawn those using the SAS/GRAPH product. ANNOTATE allows you to customize your graphics through a lower level interface. If you want to see any of the newer applications demonstrated, you can go upstairs where there is a Tektronix terminal hooked to the VAX in North Carolina.

Gerald Perry

Approximately 2000 chemists and chemical engineers and a few mechanical and electrical engineers make up the technical computing personnel at Celanese. There is a strong inclination toward mathematical modeling, so that gives a different twist to statistics. Currently we have 5 VAXs, and there are 3 more being installed, but I am going to talk about our experience with the VAX 11/780 at our research center in New Jersey. This system has 6 megabytes, 2 RE8081s, and is constrained to 64 concurrent users. At that site, we have about 400 users, but only 64 can be active at any given time.

Mathematical modeling is unique in its ability to use regression analysis to make data fit mathematical equations that you are creating. We took an a priori model and applied random noise to it and fed that to PROC STEPWISE, and, both on the VAX and IBM version, it was relatively disappointing and led us to an interesting idea and a possible extension of SAS statistics. What we are proposing is a PROC CODE that would allow us to take X - X for all the variables, and then when you create interactive terms and high order terms, those terms become independent for the linear variable. Under those circumstances, the models would hold together over much greater amplitudes of error than they would without the PROC CODE. It would be a significant addition, at least for mathematical modeling in the SAS System. Then if you go through the process of having a PROC CODE, you need to have a PROC DECODE because if you look at your coefficients from regression analysis, they don't mean anything in terms of the original variables. Therefore, you need an inverse function.

Another thing we wish we had is an easy way of measuring experimental error in the process of general regression studies without having elaborately planned experiments because that is an expensive process for many of the applications we are doing.

Finally, one last thing that we would like to have is the ability to plot the topographical contours as opposed to areas on a contour for plotting interactive terms in mathematical analysis. Celanese is very high on the SAS System. We are looking at it very enthusiastically; we can't wait until some of the further developments occur and some of the performance enhancements are made. We have been running 4.06 for a couple of weeks, and even at this point, there is a significant increase in performance. If you have a 780, I strongly recommend adding some memory because that will help not only the SAS System but everything else you are doing on it. We have had an immense improvement in the performance of that system over the last year.

Kevin Mallory

I am from Texas Instruments in Dallas. I work for the Corporate Research Lab. I am not a SAS user but rather a VMS person. I learned to use the SAS System for the purpose of doing a benchmark.

I'd like to show you a sample of some data that I used in my benchmark. These are basically remassaged VAX accounting records. This is a sample SAS application; it is very straightforward, just a DATA step and a SORT step at the end. We are creating a few variables out of the data set. The data set had about 8400 records in it, and it was sorted out by that one variable.

Our 4341 group 2 (1.78 x a group 1) has 3380 and 3350 disks, 8 megabytes memory and runs on VMS P release 3. The VAX has a side running VMS 3.4 with 4 megabytes and some interesting SYSGN parameters: WS max of 2000 and a virtual page count of 16384.

For the user name parameters on the VAX, the user had a byte limit of 12,000 working at a default of 150, (the standardship), a working set quota of 200 (100 less than the standardship).

The data I have, which I felt were reliable, only went to a working set size of 1024 and a paging file quota of 16,384 (exactly what our virtual page count is). For the IBM side, we decided to do some reverse calculation because the page sizes are different. One page is 512 bytes. Working set extent is an example of 500 pages, which is a typical non-SAS user size; it would give it about 256K bytes of storage. On the IBM side, it turns out that virtual memory is about 248K or equivalent to 496 VAX pages.

We ran the sample SAS application on the 4341, and the DATA step took .17 of a CPU second, SORT step took .16 of a CPU second, for a total of .37 of a CPU second, or 1 minute and 20 seconds lapsed time with about 2000 or 3000 I/Os.

ACCUMULATE.SAS, with a working set size of 500, took 21 minutes CPU time. I was lucky and got one of the first tapes of Version 4.06, so about a week-and-a-half ago, I reran all of my material and redid all my slides. You are seeing about 21 minutes of CPU time, and look at the number of page faults, and this was done on a quiet system in the middle of the night. The data set and the SORT step, the elapsed time, CPU time, direct I/O, buffered I/Os, and page faults are shown for each step by working set extent. The DATA step for this application took 34 minutes of wall time under Version 4.06. We increased the working set size so we could go up to 900, and we dropped the CPU time down a little bit. We added a lot of memory and you are going to find a significant improvement. The SAS System will take whatever memory you are going to give it, especially for your DATA steps. It would have been interesting to show you those figures because this is a real performance increase. I really have faith in the SAS people; they have done a wonderful job under 4.04.

Editor's note: the remainder of this panel discussion consists of a question and answer session. Unfortunately, the questions are inaudible on our tape, so we are providing the answers with the idea that you can use the information given.

The CPU of the VM system was about .37 of a CPU second and less than one minute of elapsed time. But you have to realize that it is a 4341 group 2 with 3380 disks.

Some other applications we are running on this system that tend to consume a lot of resources are Ingress, Telegraph, Supercomp, HTRI, FRI, and SSI Process, which is a chemical process simulation. We have a number of very intensive mathematical applications running on the VAX. We are an alpha test site. We receive the first of the SAS System and we set up some test cases for each one of the SAS procedures that we use quite frequently. Celanese has also been a beta test site for some time for the IBM version of the SAS System.

We use GPROC GLM most often, and we have tested all of these procedures with data sets that we know. In fact, we compared the IBM system with the portable version, and the results compared favorably in every case. We use GLM for simple data reduction. We have some interest in PROC STEPWISE for other applications of GPLOT and GCOUNTOUR. One of the things we are missing is PROC MATRIX. We have heard several times it is going to be addressed in PROC NLIN, now in Version 4.06

It is not the SORT step that is taking the time; it is the DATA step, and the SAS people are rewriting that in macro to make it a lot faster.

I looked at this with SPM, and I decided to see what I could do to speed up this poor DATA step that was spending all its time in RMS. I could not do much for it because, as you can see, RMS tuning really did not help.

There is a thing called TEST Basics that comes with your portable SAS System that tests your portable SAS installation to make sure it is all right. SAS Institute will ship you a sample log, and they will run the TEST base program on your VAX and then compare the two logs. I tested the base program after it had run on our side, and we ported it over to the IBM. Two minor changes were made, and it ran. There are some nice things in 4.06 that are not in the release of the SAS System that the IBM had. Here is the comparison of CPU time. You can see the figures for the TEST base. VAX VMS has a working set size of 1024 and about a minute of CPU time. In their installation manual, SAS Institute recommends a working set size of 900, but I experimented and 1024 was about the best payback I could find.

This is an example of a VAX SPM report using a working set size of 1024. This report shows that during a sample period of one hour, I started SPM and the SAS System concurrently, and the SAS System was getting 32 percent of the samples and taking 98 percent of its time inside the user image, so it is not really using. The bottleneck is not necessarily VMS.

This is a graph by IPL (Interrupt Priority Level). IPL0 is user mode, so that is where we are spending 37 percent of our time. IPL2 is AST delivery, IPL7 and IPL8 are I/O post. The SPM report tells you by system module where you are spending your time inside VMS. Here is a slide that shows the mode to executive where you get to RMS. This last page is showing you that they really don't break down the RMS code into modules, but it is showing you that a lot of the samples are found inside RMS. This 0 percent is a little bit off because they don't show you the point percentages.

Regarding SAS images, I can get a report about the SAS System by module by collecting these data. The SAS Institute people have expressed an interest in seeing this because we have a version of SPM that they do not have. You can break it down by routine, but unfortunately SPM does not have enough internal virtual memory to do that. This is broken down by link module. About 19 percent of our time is spent in the DATA step interpreter, and when they improve that, we are really going to see a performance increase.

In conclusion, I think SAS Institute is doing a wonderful job. They have really made a tremendous leap between Version 4.04 and Version 4.06. Those of us who like our VAXs are very happy with what the Institute is doing.

Does anyone have any questions for us?

The 1024 is about 512K. We significantly reduced the memory that was available to the 4341 for the benchmark because the default for our system there was significantly larger.

Are there any other questions?

I wasn't fighting a paging and swapping situation. Basically, I was the only person on the system.

The SAS System is relatively painless compared to some of the other things we run. If you do SSI process, for example, where you are simulating a chemical plant that may have four or five distillation columns and separation processes, it takes substantial amounts of time and resources. Adding memory was the most dramatic improvement we got from the VAX. A 4-megabyte 780 is a mistake. You have more power than you have resources to dispatch it.

I agree with you as far as a 4-megabyte VAX is concerned. But, with the 16-megabyte VAX you will find that you spend more time trying to figure out what to do with the resource. When the high-end machine comes out, the ability to have 16 megabytes will be easier, and I think we'll see them. However, the CPU power of the VAX is really not geared to have 16 megabytes yet.

We have run the VAX with six users, and I can't say there is a difference in performance when there are six SAS users versus two or one. We run with several users at a time, but the problem is that as well as the SAS System there are quite a few other processes running at the same time. We find that if you get two or three interactive SAS users, it slows the whole thing down a lot.

You made a comment about a PROC CODE and DECODE. Could you elaborate on that? Was it in relation to the STEPWISE regression, trying to make variables independent of each other?

We were thinking of something that would allow us to use STEPWISE or GLM or any of the subsequent procedures, so you could do something like a data manipulation step procedure in front of any of those procedures to do the coding, and after you do the procedures, you could do a DECODE to get your coefficients back.

If you want to misuse STEPWISE regression to build a model of an X1*BX22*BX3X1, for example, and get as effective a measure as possible, coding would enhance that process significantly.

Previously in 4.04 the working data set was in your directory, and when the process looked for that data set, it took the latest version of what was there; however, it might have been created in a different process and you would get the wrong one. I had a job running in 4.06 and accessed my directory, and the working data set didn't seem to be created in my own directory; it was somewhere else, and I didn't have to worry about it.

I think you will find that problem is fixed in 4.06.

Depending on what you are doing, we found that 8 to 20 megabytes is about optimum performance of a VAX.

With that in mind, you could take the working sets of the people who want to use the SAS System to 1024 or above, and that would help with the CPU time taken.

You must realize that a 750 is not 780 is not a 730. With a 750 you might want to take 6 megabytes of memory, but you are not going to be able to get the same number of people on the SAS System using a 780 with an 8 to 10 megabyte. If you were going to dedicate the system to SAS users you could probably get 6 to 8 people on it. If you take the 750 with 6 megabytes, I would not put 2 or 3 people on the SAS System; I'd put 4. I did some interesting tests with an interactive job load on this same machine. This is a software development machine and has heavy CPU-intensive things. But we have people attempting to text edit and people doing lots of compiles, and you could really notice the drag with just this one particular SAS application.

If I had to make a recommendation without actually having done a benchmark on a 750, I would suggest using 6 to 8 megabytes with 3 or 4 SAS users in addition to your regular load. We are forcing people who want to use the SAS System to use batch, putting it on a lower priority. If you are trying to balance interactive response, you can put people who want to use the SAS System on a priority 3 batch job, and they use memory, but they are quiet about it.