The topic of this paper is a set of speculations about how PC SAS will be used in the workplace. We believe that the porting of the SAS system to the PC DOS environment will be a watershed event for SAS users, and indeed, for business and academic computing organizations in general. It has been estimated that 1 out of every 5 Americans uses a computer of some form at work. This number is expected to climb steadily for the next 15 years. The PC environment is ideally suited for such institutional change, and in fact is leading the way towards increased computer use by non-specialists. The introduction of PC SAS leaves the SAS Institute positioned to take advantage of this growing user base.

In this paper we would like to address several topics. First, we would like to discuss the fate of mainframe SAS. We would like to discuss the adaptations that must be made by information centers to support this product. We would like to predict where and why PC/SAS will be successful and where and why it will not take hold. In all of these topics, we will merely be expressing our own opinions, based on a number of years of experience in the User Services area. Our personal experience with PC SAS has been minimal up to this time, however.

Is the End of Mainframe SAS in Sight?

This question is actually fairly simple to answer. The answer is "we don't know." The only reason we don't feel embarrassed about this is that probably no one else knows either. In point of fact, there is massive confusion in the data processing marketplace as to the mix of work that will be performed in the future on data processing central mainframes, departmental minicomputers, and personal to confusion on this point. They probably don't know the answer to this question - we cite as evidence the brouhaha over faster crystals in the PC/AT.

We believe the fate of mainframe SAS depends on psychological factors as much as on technical issues. In other words, what will PC/SAS feel like? Even if it is much slower than mainframe SAS, the psychological advantage of controlling your own computing destiny may outweigh this disadvantage for users.

Successes for SAS on the PC

SAS on the PC will achieve its greatest usage as the goal of connectivity between larger systems and intelligent workstations is realized. Connectivity will increase the productivity of the information worker. Connectivity will allow the information consumer to use the computer directly, rather than relying on computer professionals.

Let us start with the information worker. As a thought exercise, try to estimate what percentage of time information workers spend on getting data from one process to another. By this we mean getting data which are output from one program into another program. Of course, if the two programs are on two different physical machines, the required time to do the work increases still more. One of the initial stunning advantages of the SAS system was the ease in which you could take the output from one Procedure and pipe it as input into another procedure. We feel that the success of SAS on the PC in any organization will depend on the relationship between the PCs and larger processors. If this relationship is largely seamless, PC SAS will be successful. If the relationship is discontinuous, PC SAS will not be successful.

It seems that both IBM and the SAS Institute recognize the fundamental importance of connectivity. A variety of tools are already available to facilitate connectivity. First, note that downloading corporate databases to an individual's workstation is made easier with the new SAS procedures that send code up to the host, and send data and results back to the workstation. Other factors can facilitate this process including IBM's PC-VMBond and the versions of Rexx that are available under PC-DOS. It is now possible using VM/Bond and Rexx, and SAS/RTERM to build applications that are driven at the PC-DOS level, but which logon to a host, access data, grab host machine cycles where needed, and display results at the workstation. Some machine accounting environments, all of this could be done without the user even knowing that he has logged on to a host. The goal here is to create a seamless environment for the information consumer. No doubt LU 6.2 will make the programmer's job of creating a seamless environment even easier.

Next, let's consider a typical information consumer. For illustration, we would like to
invent a typical information consumer who needs to play with corporate or departmental data using a spreadsheet, or perhaps she wants to produce a table of frequency counts or arithmetic means for a memo. Let us assume that she does not know SAS, nor does she know JCL, CMS, TSO, ISPF, what "LOGON" means, or any of the wonderful acronyms that make programmer's lives meaningful. Instead, she is a financial analyst, a manager, a marketing strategist, or perhaps a president. As she has risen to the top, she has achieved the honor of having a PC/XT or PC/AT put on her desk, and she has started to get hooked into computing through electronic mail, spreadsheets, and being able to touch up memos using a word processing program when her secretary is not in.

As far as our hypothetical information consumer is concerned, the whole mechanism we describe appears to be an ordinary DOS command. The name of this command or BAT file is the only thing she needs to memorize to use the system. That is a good thing, because she is too busy to memorize any more than this. Of course, there is no reason that an AUTOEXEC file cannot be used to avoid even this memorization. At any rate, we present her with a sequence of fill-in-the-blank windows which allow her to describe the data she needs, and the aggregation which is required. Our own preference for the design of such windows is an idea we call Statistics-By-Example, a forms-based approach which is similar to IBM's Query-By-Example. Her request for information generates a SAS job on the appropriate computer. Which machine the data are on, and what the name of the file is, and the LINKs and ACCESSes, or the JCL which is required to grab the data is irrelevant to her. The appropriate SAS code, including TSO or CMS statements is automatically uploaded. The processor may decide to actually perform the statistical analysis on the larger machine, perhaps based on the size of the file. In any case, when the job is done, her workstation beeps at her, (something which can be arranged with PC/Bond) and then she returns to the PC/SAS application and looks at the result.

As an added benefit, we can always show her the SAS commands which were used to generate this result. Then after using the system for the weeks it might take to learn the SAS system. There are two reasons for our doubt. One is that we feel the SAS system has grown inceasingly complex. Second, we think that there is a limit to the number of corporate employees who can actually understand data processing and statistics. We begin with the first issue.

Some Possible Limits to the Success of PC/SAS

PC SAS consists of a window manager grafted around a command language. Even though SAS/AF will allow informal programmers to build fill-in menus and other friendly interfaces, managers will always be able to generate problems that cannot be solved with any given menu-based system. We suspect that managers are not going to invest the weeks it might take to learn the SAS command language. Even if a manager did so, casual users tend to forget keywords and other details unless they frequently use the system.

Here is one of the reasons we say SAS has grown complex. Either of us can look on our shelves and find as much as three linear feet of SAS documentation sitting there. There are currently numerous domains of the system: the original SAS language, the applications, and they can then be translated into a compiled language after user testing.

How Information Centers will Have to Adapt

The proliferation of microcomputing in general, and PC/SAS in particular, will increase the support burden on information center staff. In a real sense, each microcomputer running SAS is now a new data center which has to be supported. As such, we might ask how we can keep all the PC SAS users running the same version? How does one distribute the 11 diskettes of SAS software? Two possible distribution solutions are PC/VM-BOND, and Local Area Networks. (These could be used to distribute the program, but not execute it.) The local support staff would have to write Rexx or BAT language code in order to insure that the terms of the license agreement with the SAS Institute are being met. The support issue becomes more difficult when it is realized that different individuals may configure their machines differently. I might have a crystal to speed up my PROC INL code, and my office mate might have a brand-X graphics adapter for use with SAS/GRAPH. This problem could be made worse if there are different LANs running in different departments. In all these cases, trouble shooting SAS problems could be difficult.
display manager, the macro language, the FS/CALC language, and the SAS/AF application development language. To this we might add PROC IML and the RTERM language. These domains go together in the sense that there are no command collisions, but our overall impression is that these things have a high degree of independence. The SAS system is an integrated system in the sense of ease of getting data from one process to another. In another sense, wouldn't it be more palatable to new and old users if there were much more overlap in keywords, syntax, logic and structure. These are the things that requires investments of time on the parts of users. We do not see the classic PC-user making this type of investment. It is telling that a document necessarily called the SAS User's Guide: Basics, is 1,290 pages. The historical dilemma facing SAS is that users may begin to perceive the system as the world's most complex statistical package, when in reality it is the world's most simple operating system.

The second reason for doubt is based on the premise that many MBAs are not trained to deal with data. Even if they could memorize all the relevant SAS commands, and how to run the display manager besides, just by listening to a cassette tape while they slept, it is not clear that most managers would know what to do with the system. Most managers are not trained in concepts such as statistical analysis, match-merging and table lookups and all the other wonderful things that SAS can do in just a few lines of code. What we will call the "next-step" will be required to bring such people into the SAS system. The next step will resemble artificial intelligence, and will help managers formulate their problems. The next step, perhaps SAS VII or so, will help information users interpret their output, and suggest how they might best graph the data. Expert system technology will be required to achieve this.

Conclusions

We feel that the PC version of SAS will be a major success. The primary users of the system will be information workers and programmers who already know SAS. More and more of these people will want to use PC/SAS as IBM integrates its disparate product line with LU 6.2. Information center programmers will be able to use the system to allow information consumers to generate some ad hoc reports and analyses on their own. This activity will be somewhat labor intensive on the part of the programmer, but not unduly so due to the ease of using SAS/AF. Managers and others who were introduced to computing in the 80's by way of the spreadsheet and other microcomputer applications will not learn PC/SAS in its vanilla form, but rather will get to know it through these padded cells provided for them by programmers. Our pie-in-the-sky SAS system, say version VII, would resemble an expert system more than it resembles the SAS programming language of today.