

# The Obsolescence of the SAS System Revisited

Grant Blank, Independent Consultant

## ABSTRACT

*At SUGI 10 in 1985, I contended that the SAS System was close to obsolescence and that significant changes were needed for Version 6. Now that Version 6 (i.e. Release 6.06) has been released, this paper reassesses the thesis that the SAS System may be obsolescent. The context of computing has changed dramatically in the 6 years since SUGI 10, and statistics and databases have changed as well. Everything has become more complex, there are many more audiences of users with different needs and different interests. The SAS System continues to serve one large market well (large corporate mainframe users) but, as other groups of users have emerged, it has been unable to repeat its mainframe success. Non-mainframe audiences would be better served by software more closely adapted to their environment and many mainframe users would benefit from easier-to-use and more flexible SAS products. The paper suggests innovative new products that can serve these users.*

## INTRODUCTION

At SUGI 10 in 1985 I delivered a paper titled *The Obsolescence of the SAS System*. The paper contended that the SAS System was in danger of becoming outdated by changes in hardware, software, and user expectations. Major improvements were needed for Version 6. In the past year we have seen the widespread release of Version 6 (i.e. Releases 6.06 and 6.07) for mainframes, minicomputers, and UNIX workstations.<sup>1</sup> It is appropriate to again assess the SAS System. We need to ask, how well does Version 6 serve our needs? What do we want in Version 7?

In a fundamental sense, nothing has changed in the six years since 1985. Good software is still the same. It solves problems simply and flexibly. Flexibility has been key to meeting the needs of users with an immense range of problems and kinds of data. Simplicity means that the SAS System has provided relatively easy ways to solve complicated statistical and database problems. In my view, the success of the SAS System has always been based on a combination of flexibility and simplicity. Outstanding examples of this combination include the DATA step's ability to read, manipulate, and output virtually any kind of data inside a single conceptually unified system, and PROC GLMs ability to unify and simplify analysis of many kinds of linear statistical models.<sup>2</sup>

In another sense, many things have changed in the six years since 1985. The broad changes in machine speed and capacity, in software design and capabilities are well known and need not be detailed here. As machines have become more capable, software design has become more sophisticated. Standards have become higher in such crucial areas as ease of use, ease of learning, and documentation. These changes continue trends that were clearly visible in 1985.

Some consequences of changes in hardware and software have been less widely recognized. Since 1985, faster, cheaper computers have penetrated to many new, different groups of people, leading to a proliferation of audiences.<sup>3</sup> This caused a

rapid increase in specialization to the point where many audiences not only do not talk to each other; in fact, they may not even be aware of each other's existence. A crucial consequence has been to make the world of computing in which SAS Institute operates much more complex.

The question is, what does this more differentiated and complex world mean for SAS software and for us as SAS users? The issues of simplicity and flexibility are more complex than in 1985. We can no longer talk in general terms of simplicity for everyone, but we have to talk about specific audiences of users and their unique needs.

Many small companies have found niches in which they can grow and prosper. For example, we see companies selling market research tabulating software or software to read and generate reports from census data. SAS Institute is not a small company. It competes with many other companies in many different markets. It has to know about and understand many different audiences.

Some anticipate re-emergence of a single standard. We need only wait a few years, they say, and everything will shake out; we will again have one or two dominant packages. This is not realistic. There are too many different audiences with different needs and interests. No software can be all things to all people. We will never return to days like the early 1980s when the SAS System was overwhelmingly dominant. Nonetheless, as I make clear below, I believe that SAS Institute can do better.

These problems also make this paper harder to write. I need much more information to be able to comment on how SAS meets the needs of different audiences. This will be reflected by the limits of my comments. For example, I have nothing to say about the SAS/C® compiler and library products, NeoVisuals® software, or about SYSTEM 2000® Data Management Software. I have not used these products, I don't even know anyone who has.

Nonetheless, as an independent consultant, my experience with the SAS System products is extensive and long-standing. I have used it since 1977 and continue to use it regularly. My consulting requires work on mainframes, minicomputers, workstations, and micros, and I am in contact with the needs and interests of a wide variety of users.

## THE SAS SYSTEM, VERSION 6

To evaluate the SAS System, Release 6.06, we need to know how version 6 differs from version 5. Remember that SAS Institute began as a statistical software vendor, with its first commercial product released as SAS76. Through three additional major releases, SAS79, SAS82, and Version 5, SAS Institute continued to implement its vision of statistical software. By any standard it was a remarkably successful vision. By the time version 5 was released in 1985, the SAS System dominated the mainframe statistics market.

Success brought a new problem: if it was to grow further, SAS Institute had to develop in a new direction. While Release 6.06

contains many new features and enhancements, I see a core theme that reveals the new direction. At the risk of oversimplifying the changes in Release 6.06, I will summarize them. The SAS System is no longer primarily a statistical software package with unusually good report writing and data management capabilities; its new core is built around a database management system (DBMS) with unusually good reporting and statistical analysis capabilities. The new emphasis on database capabilities is the real news of Release 6.06.

An overview of Release 6.06 helps us understand the core issue. But we need to look at the SAS System in more depth to understand how it compares to other software and how it meets the needs of users.

## COMPARISONS TO 1985

At SUGI 10, I contended that the SAS System was weakened by four problems,

1. Limited interactive capabilities.
2. Serious flaws in documentation.
3. Limited ability to extend SAS's capabilities.
4. Lack of DBMS capabilities.

Let's look the changes implemented for Release 6.06 in each of these areas.

**Interactive capabilities.** SAS Institute is clearly moving to make its products more interactive. In Release 6.06, the ANOVA, CATMOD, DATASETS, GLM, PLAN, and REG procedures have some interactive capabilities. Interactive report writing is available in REPORT. SAS/ASSIST® software, SAS/EIS® software, SAS/INSIGHT® software, and the new window interface in the SAS Display Manager System are major steps in the right direction. Extensive help files in SAS/ASSIST software, SAS/FSP software, and elsewhere are a welcome addition. However, many of the same problems remain:

- The syntax of commands is batch-style verbose.
- In most procedures, errors cannot be diagnosed and fixed interactively, the entire procedure has to be rerun.
- Most significantly, as a result of SAS's design, PROCs tend to proliferate. SAS programming usually requires multiple PROC and DATA steps. This makes interaction with the system more clumsy and more complicated than necessary; requiring too many steps and too many passes through the data. This is not the design of a modern interactive system.<sup>4</sup>

Interactive software is one area where standards are much higher now than they were in 1985. Today's appropriate standard of comparison is modern interactive products like Borland International's DBMS Paradox or to Systat's statistical package, SYSTAT. Both are more interactive and easier to use. Despite recent improvements, the SAS System is not competitive with a truly interactive system.

**Documentation:** In 1985, I said that documentation was the weakest part of the SAS System. In the past 5 years, there have been major improvements: badly needed tutorial manuals have been added, and the SAS Companions for the various operating systems have been expanded. Worth particular mention is the excellent Language and Procedures manual (SAS Institute 1989).

Explanations of some perennially confusing topics like data management are much improved.

Documentation is another area where standards have become higher and, despite significant improvements, I have been disappointed. Large portions of the Release 6.06 documentation has become more voluminous but not much better. SAS documentation continues to suffer from the same problems that have plagued it since SAS76. Except for the Introductory Guide, tutorial manuals for products like the macro (SAS Institute 1990a) facility and PROC TABULATE (SAS Institute 1990b) often do not present explanations in the order in which users need them. This forces users to page back and forth, searching for crucial information.

Worst of all, these manuals mislead by omission. By that I mean both tutorial and reference manuals often fail to mention important limits, and examples appear to be chosen to avoid revealing limits to users. Examples of this are common. For one, try to find out how to print the contents of a window. You might also look in the help files. Good luck. A second is the RBREAK command in REPORT. It is presented as a way of adding a blank line to separate report headers from the body of the report. The documentation obscures the fact that it works only on the first or last pages of a report, not on other pages. The accompanying example conceals this because it is a one-page report (SAS Institute 1990d). As a third example, try to find how many characters SAS software will read in a line of source code and what will happen if you exceed that limit. As far as I have been able to find, this is not documented in any written manual. Finally, to take a specific case, the INFILE command documentation never gives the default input line length if a user omits the LRECL option. The general theme of these last two problems is that, for many procedures and commands, the manuals fail to spell out the default values. Documentation would be much better if each and every command contained an explicit paragraph describing defaults.

SAS Institute has produced a complicated software product and it is not easy to document. Nonetheless, the documentation standards in many other companies seem to be higher. Despite some outstanding manuals, SAS Institute's documentation is not as uniformly good as, for example, WordPerfect Corporation's (1990a, 1990b) or Borland's documentation for Quattro Pro (e.g. Borland 1990).

**Extensibility.** PROC PMENU is an important enhancement. I look forward to SAS/EIS® software. These are valuable because they provide easy ways to build a front end for SAS applications. This is useful, but it only goes part of the way. Building the underlying applications is still difficult. The macro language has seen minor enhancements and Screen Control Language has been added to SAS/AF® and SAS/FSP® software. These products have the same weaknesses they have always had: They are hard to learn and not very readable.<sup>5</sup> Just to get an application up and running still requires fairly heavy development costs. As a result, for most users, the SAS System cannot be easily extended.

**DBMS capabilities.** The biggest news in Release 6.06 is enhanced database management capabilities. SAS/CONNECT™ software and the transparent access to non-SAS data provided by the engines that read other database formats are powerful enhancements. For non-procedural applications, PROC SQL eases user access to data. SAS data views and indexes are also valuable new capabilities. Other enhancements to the SAS System have added functions, often at the cost of more complexity. The DBMS enhancements are different; instead of adding complexity, they simplify. SAS Institute has done an excellent job.<sup>6</sup>

The preceding points describe how the weaknesses of the SAS System that were visible in 1985 appear from the vantage point of 1991. We've looked backward, now we turn to the future.

## OTHER AUDIENCES, OTHER PRODUCTS

SAS Institute has built on its mainframe strengths with SAS/CONNECT software and its database strengths with PROC SQL. SAS Institute has other strengths which appear to have gone unrecognized, or at least the Institute has made little effort to capitalize on them. I want to suggest several products that could be incorporated into Version 7. These products build on SAS's strengths but address the needs of audiences which are increasingly less likely to use the SAS System as it currently exists.

**Interactive TABULATE.** PROC TABULATE is notoriously difficult. The fundamental problem for users of PROC TABULATE is not deciding what statistics to calculate. Means and sums are straightforward. The problem is the complicated syntax required to describe the format of the final report. In my experience, producing the numbers doesn't require nearly as much time as editing the format. For example, changing column widths; altering, adding, or deleting labels; changing a format; and printing more or fewer digits to the right of the decimal. Because they do not require rereading an entire dataset, these problems are ideally solved in an interactive procedure. The current batch design of TABULATE requires users to reread the entire dataset just to fix a label. Version 6 provides interactive access to several statistical procedures and to the experimental PROC REPORT.<sup>7</sup> Similar interactive products have been successful for other companies (e.g. Strawberry Software's A-CROSS). TABULATE would be an attractive product to a much wider audience if it were interactive. TABULATE deserves to be better.

SAS/ASSIST software provides menus to access TABULATE. These simplify and they are a step in the right direction. To a person who knows only IBM mainframe software, they probably look wonderful. Interactivity, however, is not defined on mainframes. On Macintoshes, in DOS spreadsheets, and in WordPerfect, changing a column or title requires only a few keystrokes and the revised table appears instantly on screen. That's the way modern, interactive software works. That's what TABULATE needs.

**Query-by-Example DBMS Interface.** Sure many people need SQL<sup>®</sup> and access to huge mainframe databases. SAS Institute does well to make its products available for people who need those things. There are also many people who have different needs and they want databases with a different user interface. They don't know SQL or database theory. They need something much easier to use; more visual, more interactive, and more flexible. These users need a query-by-example. Paradox is an excellent example of a database built on these ideas. A query-by-example front end for SAS Software would reach these people and build on SAS Software's strengths as a database management system.

**SAS/LITE Software.** I would like SAS Institute to rethink how it meets the needs of low-end users. On personal computers, SAS software requires an expensive machine. Many people don't have that much computer. Still others don't need the full SAS software product. Many people work with less than a thousand observations and a couple dozen variables. They may only want a few numbers. Users like this are most productive on a small, unified system rather than a large, diverse collection of products. As a consultant, despite my enthusiasm for SAS software, I have never felt that I served the needs of such users by recommending they use the SAS System. They don't need that much power and they do need an easier-to-use system. For users like this a more compact, easier-to-learn SAS product would be a real advantage. It would also be a way to recruit them to the larger SAS System.

This is an old argument and the Institute has decided against such a product in the past. I would only point out that the audience for SAS/LITE is large. Companies like BASS Institute,

SPSS, Stata, and Systat make tens of millions of dollars each year because SAS Institute never produced a product with acceptable performance on less than a 386 microcomputer with 3 megabytes of memory. JMP<sup>®</sup> software is a creative and welcome addition, but it only goes part of the way. Many people need statistics different than those provided by JMP software and they want a different approach than JMP provides.<sup>8</sup> There is a need and audience for other products that are easy to use and don't require lots of expensive hardware.

## USER INTERFACE DESIGN: METAPHORS AND USEABILITY TESTING

Besides additional products, Version 7 needs to clean up some serious problems in Release 6.06. One of the most problematic aspects of Version 6 from a user perspective is the proliferation of overlapping, partially inconsistent procedures and products. There are two levels of this problem.

On one level the problem is that when there is overlap, SAS Institute offers users little guidance as to the circumstances under which it thinks different products are appropriate. For example, I would like to see a clear statement about when SAS Institute envisions people using the macro facility versus screen control language. A similar question occurs about when to use PROC SQL or the DATA step. These products require significant time and effort to learn. Software written in them may run for 5 years or more. Neither I nor my clients want to go to the trouble and expense of writing software using a facility that the Institute has decided to stop enhancing. Guidelines from SAS Institute would be helpful. SAS Institute provides such guidelines in several areas; for example, the suggestions on when to use indexes (SAS Institute 1990c, p. 224). Such guidelines need to be expanded to cover much more.

On another level the inconsistencies seem to be the result of more serious design problems. I want to be clear about the nature of these problems so I will give two examples out of many that could be chosen.

**Four Ways to Print in Windows.** The first example concerns inconsistencies in printing in windows. In some windows (appointment, calculator, notepad, output, log, and program editor windows) the PRINT command sends the contents of the window to the printer. PRINT does not work in SAS/FSP or SAS/AF windows. In SAS/FSP windows and SAS/AF windows the SPRINT command prints the screen, but not the rest of the logical window. Third, in still other windows (catalog, directory, keys, options, filename, footnotes, help, libname, forms, setinit, titles, and variables windows) the only way to print is to cut and paste into another window that allows printing and to print from there. Finally, the PRINTALL option will print the entire formatted contents of a file from FSEDIT and FSBROWSE, but it can only be issued from the command line, not from within a window. Thus, SAS Institute gives us four very different ways of printing.<sup>9</sup>

Documentation doesn't help because it rarely acknowledges that users might want to print the contents of windows and it doesn't notice that the inconsistencies exist. The inconsistencies put a burden on users that better design would not. Better designs are so simple it's embarrassing: in any window a PRINT command prints the entire contents of the window, an SPRINT command prints the current screen.<sup>10</sup>

**Six Ways to Browse and Edit.** The second, more serious set of inconsistencies concerns the astonishing multitude of ways that SAS software allows users to view and edit objects such as catalogs, data libraries, dataset descriptors, and data. I know of six ways and I may have missed others.

1. The old fashioned, batch way requires using PROC CATALOG, PROC CONTENTS, PROC COPY, or PROC DATASETS.<sup>11</sup>

2. In interactive SAS, the LIBNAME, DIR, CATALOG, and VAR windows allow users to browse and edit librefs, files in a libref, catalog entries, and variables, and to browse data. But they do not allow users to edit data, or the contents of catalog entries. A further limitation is that users cannot enter at a lower level (say, a DIR window) and then go to a higher level (say, a LIBNAME window).
3. SAS/AF programs can be browsed and edited only inside PROC BUILD and nowhere else.
4. Format libraries can only be viewed from the batch output of PROC FMTLIB. They cannot be edited or viewed interactively.
5. Graphics output can only be viewed from PROC GREPLAY.
6. Finally, the ACCESS window in SAS/FSP software allows browsing or editing of catalogs, dataset descriptors, and variables.

Sometimes you can edit, other times you can only browse. Sometimes you can move higher or lower in the SAS hierarchy, sometimes you can't. Sorting out what is allowed, and keeping track of it is a complicated mess and puts another needless burden on users.

A better design would allow entry points at any level; for example, users could look at a catalog, a dataset, a view, graphics output, a notepad, a variable, or data. From any level, users would be able to move freely up or down, browsing or editing as they wished. For example, a user could browse a libref, select a dataset, change the dataset label, edit the data, correcting an observation, then go to a format library to edit and replace one of the formats. All this should be possible smoothly, interactively, using full screens, as an integrated package, and without the software imposing constraints on the user. It should be possible to browse or edit format libraries, graphics output, and SAS/AF programs. The point is not that a single procedure should be able to do all of this, but that there be a single supervisory program to call appropriate SAS procedures to allow browsing or editing anything that users produce. PROC COPY and PROC DATASETS is still necessary for work done in batch but, in an interactive environment, the presence of five other incomplete and inconsistent methods is irritating, unnecessary, and burdensome. It's hard to teach, hard to document, hard to learn, and hard to use. SAS Institute can do better.

**Metaphors: "The SAS Hierarchy Tree".** Given SAS Institute's problem of coordinating the design work of many programmers and analysts, one solution seems particularly appropriate. Like the desktop metaphor of the Macintosh or the notebook metaphor of Go's Pen Point operating system, SAS needs a set of principles that give designers a clear idea of how their final product should look to users. SAS Institute needs a *metaphor*, a metaphor for the SAS hierarchy. One common metaphor for a hierarchy is a tree. I propose the metaphor of the *SAS Hierarchy Tree*.<sup>12</sup> The point is, such things as datasets, graphics output, notepads, and format libraries are not separate, they are connected. And SAS Institute needs to think of them as connected. The common thread is that users want to browse and edit them from any point in the SAS System. SAS software badly needs this sort of unifying theme.

The underlying problem seems to be that SAS institute is not providing guidelines for user interface design. As a result, each programmer or team makes separate decisions about user interface. Most of the decisions make sense in their own terms. The problem is, they are often *different*. The difficulties that this creates can be illustrated by describing how to select the item you want to edit from a list displayed in a window. In some windows, users type an 'e' (for edit) in the selection field, in other windows, users type an 's' (for select). My personal favorite is the VAR window where, if you want to edit a variable label, you type an 'r' (it's not intuitively obvious, but this stands for rename). Some

programmers, remembering that many users have a mouse, allow users to double-click to select an item. Others don't. Users can't construct a simple set of rules that allow them to navigate in the system because there are no simple rules; there are only different decisions made by different people. Without rules, everything is a special case, and users have to memorize every single special case. This makes SAS software confusing to everyone and exceptionally difficult to teach to new users.<sup>13</sup>

Achieving a consistent user interface is a well-known problem in large software products like the SAS System. Other companies overcome the problem with various techniques. The idea of an underlying design metaphor is the best known. Metaphors provide both users and designers with simple rules for how the system should look and feel. The Macintosh desktop metaphor is the most successful of this kind of metaphor.

**Useability Testing.** Another approach is to conduct useability testing. This involves asking a group of users to actually use the product, often in a location where they can be videotaped. The users are asked to comment on all aspects of the user interface, especially how the product works for them compared to how they want it to work. For example, the LIBNAME window lists all currently defined librefs. However, users are not allowed to edit existing librefs (perhaps changing the path) or define new librefs in this window. To change or add a libref users have to submit a command in the program editor window. This may make sense from a programmer's point of view. From my point of view as a user, it makes no sense. To me, the LIBNAME window is the obvious place to go to define librefs. This is exactly the sort of issue that comes up and can be discussed in the context of useability testing.<sup>14</sup> There is little evidence in Release 6.06 that SAS Institute done useability testing.

In cases like the examples above and others, the design of SAS software gets in the way of what users need and want to do. This strengthens the case of those who would like to avoid SAS software. PC software has been so successful because it emphasizes correct priorities: Ease-of-use, well-written documentation, simplification, interactivity. I don't suppose that we can reasonably expect SAS Institute to be Microsoft or Borland International. But we can reasonably point out that Microsoft and Borland have grown big by providing really good software; by many standards, better software than SAS Institute. And we can ask SAS Institute to do better.

## CONCLUSION

There are reasons why the SAS System is not as dominant on smaller systems as it was on IBM mainframes. I believe that much is due to the weaknesses that I have identified. Underlying these weaknesses is a single problem: On small systems especially PCs, other companies are providing better products; products that are easier-to-learn and easier-to-use. Many people will not choose SAS/STAT® software when they have an econometric problem suitable for Stata or an experimental design appropriate for SYSTAT's MGLH. Why use the SAS System for data management if you can put your data in Paradox?

I suggest there are four reasons why people use SAS software on smaller machines.

1. Current users have paid the high cost of learning it and they can use it easily.
2. Their companies may have an investment in SAS software on mainframes and they want to support only a single product.
3. The combination of features in SAS software makes it productive, even if any single capability is not strong. Users can do everything in one system and avoid learning many different kinds of software.

4. Users may have mainframe data and SAS/CONNECT software makes it available to them.

With the exception of the single product (2.), these are all linked to SAS's mainframe strengths. SAS Institute is wisely trying to leverage its mainframe strengths. But, the proportion of work done on mainframes is declining. Increasing numbers of people have no contact with mainframes, and don't want any. They don't want to learn software as difficult as the SAS System. SAS Institute can offer such people JMP and SAS/ASSIST software. JMP is a wonderful, creative product, but "statistical visualization software" has a limited audience. SAS/ASSIST is also an excellent product. As a user interface for SAS Software, its limitations tend to be those of the underlying SAS procedures as I've described above.

The idea of a single system that works identically across all levels of computer hardware (2., above) is extremely valuable. SAS Institute and its users gain much by implementing this approach. However, this carries with it the problem that microcomputer audiences have different expectations for their software than their mainframe cousins. For SAS Institute to be competitive with other microcomputer software, it has to have the level of ease-of-use and consistency in the user interface that microcomputer audiences expect. SAS software is not there yet. Even mainframe audiences would benefit from easier-to-user, more interactive, and more consistent products.

In 1985, my fear was that SAS Institute had lost its sense of direction; that my clients and I would have to find other software to meet our future needs. I no longer feel that way. I like the new database management focus of Release 6.06. In sharp contrast to 1985, this vision of SAS software has the power and depth to carry the institute and its users for the foreseeable future. What I look for in Version 7 is something different than I wanted in Version 6. The biggest weaknesses of SAS software seem to me to be related to what I loosely call the user interface. In Version 7 I look for new products and improvements aimed at increasing ease-of-use and ease-of-learning, more interactivity, clearer documentation, and more consistency.

Innovative products like JMP software and SAS/INSIGHT software are outside the SAS mainstream. The core SAS software has a fundamental problem: Version 6 feels too much like the same old SAS that we first saw in SAS76. This is a mixed blessing. Those who attend SUGI know SAS software well and we love it. But we are not good examples of the future. The future belongs to the millions of people who are not now SAS software users or even computer users. They will become users as computers continue to penetrate new areas of society in the next decades. Their standards are different and their work is different. Their standards will be shaped by products like Quattro Pro 2.0, WordPerfect 5.1, or others with modern, easier, more interactive user interfaces, especially Macintosh products. Other companies manage to produce complicated products with fewer inconsistencies. This is where SAS is falling short. Its products aren't competitive with many non-mainframe audiences. For the future SAS software needs a more interactive products and a more unified design.

---

## NOTES

[1] SAS Institute's version numbering scheme creates confusion. The initial release of the SAS System for PCs was called Version 6. The initial release for UNIX was Release 6.03. However, releases prior to Release 6.06 contained few enhancements over

Version 5 and, SAS Institute's advertising claims notwithstanding, various important features had been removed (like non-integer weights). Prior to Release 6.06, the major justification for a new release number seems to be not user enhancements, but that the product had been ported to C. Release 6.06 marks the first major enhancement since Version 5. To avoid clumsy wording, all my references to Version 6 are to Release 6.06 or later, and references to Release 6.06 also apply to Release 6.07.

[2] Many of the ideas in this paragraph and others are developed more fully in the 1985 paper (Blank 1985).

[3] Those in marketing may be familiar with the concept of audiences under names like markets or market segments. For the purposes of this paper these three concepts are identical.

[4] To some extent, the institute appears to recognize this problem. For Version 6, some procedures have been combined. In some statistical procedures, plots can be produced without leaving the procedure, MEANS and SUMMARY are now identical, and PROC DATASETS has been expanded to incorporate CONTENTS and COPY.

[5] Due to the absence of quoting requirements, screen control language is probably more accessible and easier to read than the macro language. This is, however, only a relative statement. For a careful comparison, see Norton (1991).

[6] Though I wish that missing data could be handled according to SQL standards.

[7] Unlike REPORT, an interactive TABULATE should be intelligent enough to know that changing a label does not require rereading the entire dataset.

[8] I do not intend this as a criticism of JMP software. In my opinion, JMP, especially version 2, is the most innovative, creative product SAS Institute has produced in a decade. Using it as a model, I would like to see more innovation. I would also like a DOS or Windows version of JMP software.

[9] A fifth way is releasing output from the Output Manager window to the printer, but that's a different situation.

[10] Requestor windows or dialog boxes may be needed to allow the user to specify particular options or, since a PRINT command could result in thousands of pages of output, to confirm what the user wanted. These are minor issues, however, compared to the current, fundamentally inconsistent design.

[11] In Version 6, PROC CONTENTS and PROC DATASETS are identical and that PROC DATASETS is interactive. Nonetheless, in batch processing these are the procedures used to view catalogs and data libraries.

[12] Other possible names are the SAS *Arboreal Hierarchy* or, giving it a name sure to appeal to SAS Institute's marketing people, we could emphasize the precedence order of the levels in the system and name it the SAS/CASTE.

[13] In the light of these issues the reaction that this paper received from SAS Institute employees at SUGI 16 is probably significant. I received direct comments on the paper from 3 Institute employees. Two were from the education and training division and they both thanked me for my comments. The third was from marketing and he seemed to be so angry that he misunderstood what I was trying to say. He wanted to know if I had a consulting relationship with a company called Metaphor Computing. I do not and in any case, my concern in this paper is about ideas like desktop metaphors, not about products or companies with the word "metaphor" in their title. The difference in their responses may reflect some of the differences between training and marketing. The training staff have to explain the SAS System to users as it is but marketing has the luxury of explaining SAS software as it should be.

[14] Note that useability testing is not the same as beta testing because it focuses on consistency and ease-of-use in the user interface. Beta testing focuses on finding bugs.

## REFERENCES

Blank, Grant. 1985. The obsolescence of the SAS System. *Proceedings of the Tenth Annual SAS Users Group International Conference* (pp. 279-284). Cary, NC: SAS Institute.

Borland International. 1990. *Quattro Pro Version 2.0 User's Guide*. Scotts Valley, CA: Borland International.

Norton, Andrew A. 1991. Screen control language versus macros in batch environments. *Proceedings of the Sixteenth Annual SAS Users Group International Conference*. Cary, NC: SAS Institute.

SAS Institute, Inc. 1989. *SAS Language and Procedures: Usage, Version 6, First Edition*. Cary, NC: SAS Institute.

SAS Institute, Inc. 1990a. *SAS Guide to Macro Processing, Version 6, First Edition*. Cary, NC: SAS Institute.

SAS Institute, Inc. 1990b. *SAS Guide to TABULATE Processing, Version 6, Second Edition*. Cary, NC: SAS Institute.

SAS Institute, Inc. 1990c. *SAS Language: Reference, Version 6, First Edition*. Cary, NC: SAS Institute.

SAS Institute, Inc. 1990d. *SAS Technical Report P-202. The REPORT Procedure, Experimental Release*. Cary, NC: SAS Institute.

WordPerfect Corporation. 1989a. *WordPerfect for IBM Personal Computers and PC Networks*. Orem, UT: WordPerfect Corporation.

WordPerfect Corporation. 1989b. *WordPerfect Workbook for IBM Personal Computers*. Orem, UT: WordPerfect Corporation.

## ACKNOWLEDGEMENTS.

I would like to thank Andrew A. Norton for conversations about all versions of the SAS System. For comments on other drafts of this

paper I thank Jeff Bass, Rebecca M. Blank, Nancy Denton, Tamara Fischell, Mary Conklin Gomborg, Peter Kretzman, James K. Lichtenstein, J. Mei-Mei Ma, and Andrew A. Norton. Remaining errors or omissions are my responsibility. I thank Trilogy Consulting Corp. for sending Andrew A. Norton to the SUGI 16 conference where he performed his task of turning transparencies during my presentation with high competence.

The ideas and interpretations in this paper are personal views and not those of my clients or any other organization. Readers should know that I have had previous consulting relationships with BASS Institute and SPSS, Inc., and I have a current relationship with Systat, Inc. I have no relationship with any other companies whose names or products I mention in this paper.

Comments or criticisms are welcome and solicited.

Grant Blank  
Independent Consultant  
1450 East 55th Place, #922  
Chicago, IL 60637-1815  
312/947-8194

Internet: GRANT@SAM.SPC.UCHICAGO.EDU

JMP software, SAS Software, SAS/ACCESS software, SAS/ASSIST software, SAS/AF software, SAS/C compiler, SAS/CONNECT software, SAS/EIS software, SAS/FSP software, SAS/INSIGHT software, SAS/STAT software, NeoVisuals software, and SYSTEM 2000 Data Management Software are registered trademarks of SAS Institute Inc., Cary, NC, USA. SAS/LITE software is a potential trademark of SAS Institute Inc., Cary, NC, USA.

SQL is a registered trademark of International Business Machines Corporation.

All other product names and trademarks are the property of their respective owners.