This paper discusses operating a successful Help Desk supporting over 500 active SAS users. We support users working on systems ranging from PCs and Unix workstations to VAX mainframes. In our approach the help desk personnel try to identify the type of user and the nature of the problem quickly and then tailor the response to the user. Responses range from brief explanations of SAS or operating system syntax to extended systems analysis.

The paper also discusses additional activities that may be conducted during those periods when there are no specific requests for assistance. These activities are designed to improve the general level of SAS support by developing generalized solutions to problems and prototype systems using different SAS techniques and products.

A SAS help desk assists users in getting their SAS code to do what they want SAS to do. Our most recent experience with help desks involves our work at the Census Bureau. The Bureau runs their SAS system on several VAX clusters, some MicroVAX systems, a few workstations and several hundred personal computers. There are over five hundred SAS users actively running programs. These users vary greatly in their SAS experience and encounter a wide range of problems that need resolution.

At the Census Bureau, SAS support has been centralized to a single office with five staff members including both Census employees and contractors. Each individual attempts to respond to all types of requests but each has areas of expertise that vary from particular hardware/software platforms to special SAS knowledge.

Most first-time queries occur over the telephone. If we cannot resolve their problem over the telephone, then we ask them to come to our office for more extensive interaction. Customers are handled on a first-come, first-serve basis unless a particular staff member is requested. With from three to five staffers on call, we have not had problems with clients complaining that the phone is always busy. Each staffer has a desk and most work is done either there or at one of the available computers at the office. To help avoid staff burnout, each staffer takes "quiet-time" each day when he or she can study new topics or master familiar ones.

REQUIRED RESOURCES

For the purpose of this discussion, we have separated the required resources into four different categories: management commitment, personnel, hardware, and software.

Management commitment - You and your staff have to have a strong commitment from your management. You get this only by demonstrating that you are performing a service. We provide both periodic reports and logsheets to Census management. The information includes number of calls, number of visits, which hardware platforms were involved, number of SAS installations, number of documents distributed, and number of customers trained.

Personnel - The size of the help desk staff will vary depending on the level of support required and how many users are supported. At the Census Bureau, the experience of the help desk personnel varies widely, ranging from people with over 15 years of data processing experience to people with under five years of experience. This mixture has been both beneficial and effective.

The best help desk personnel enjoy dealing with people. SAS gurus and hackers are not necessarily the right persons for the job. Further, help desk people do not have to have extensive knowledge of the operating systems. They need to know those parts of the operating system that impact SAS, but they don't need to be experts.

Hardware - A help desk should have the same working environment as the users. The effectiveness of the support varies directly with the help desk's ability to reproduce the users' working environment. The hardware doesn't have to be extensive. It could be a terminal in a VAX or IBM mainframe environment. It could be a PC in a microcomputer environment. It could be all of the above. The point is that you should be able to sit down and work interactively with both SAS and the user.

Software - You must have access to the software used by your clients. This includes not only the various SAS products that users will have, but also any third party software with which SAS users interface (network environments, database systems, etc.). Furthermore, you will probably need to have access to a variety of tools that the users do not necessarily need. For example, at the Census, Fastback is important as a mechanism to propagate distributions through the user base.

RANGE OF PROBLEMS

Without question, the primary activity is to resolve user problems in getting SAS code to do what the user thinks he or she wants it to do. This covers difficulties ranging from solving SAS-specific problems to helping in the design of data processing systems.

We classify SAS problems as either syntax, data step programming, or procedure-related. Syntax problems are the easiest to resolve (always keep your reference books handy), but they are sometimes difficult to detect. Use your manuals! Never be ashamed to resort to the manuals—no one expects you to have them memorized. In fact, by referring to the manuals you are sending a message to the users that the manuals are readable and they should review their manuals. As long as you can solve the problem quicker than the user, you are OK.

It is important to remember, the user comes to you only after sitting and staring at the problem for some time. As it happens, if syntax resolution comes quickly, then the user is embarrassed. If syntax resolution comes slowly, then we are embarrassed. Sometimes, if you're lucky, the users solve the problem by just talking with you. Then they are pleased and you did your job by simply listening (an important and often overlooked skill). Help desk personnel must be tolerant of the users and of their mistakes as well as your own mistakes. You must learn to not worry about it.

Fortunately, your users typically share the same perspective towards the SAS system and share the same mistakes. Experience at a specific help desk is the best way to become skilled at resolving syntax problems. We keep a notebook of problems that have been seen repeatedly.

Examining the code is the best approach to resolving syntax problems: looking at printouts, accessing their account while they are at another terminal, or looking over users' shoulders as they work at your terminal. The last approach is most meaningful when the user is actually at the keyboard typing what you dictate. We have developed the practice of using an "OPTIONS OBS=(SMALL NUMBER)" statement whenever we debug SAS code interactively with a user. This reduces the time frame of the debugging process to an acceptable real-time factor.

Our users meet their most significant problems in the data step. Many of them are social scientists not programmers; and the traditional programming aspects of SAS code are most dominant.
in the data step. These problems are more complicated and take more
time to resolve. This leads to one of the main problems of
working a help desk. How long do you work on a problem in front
of the user? When do you send the user back to his or her desk
while you continue to work on the problem? Experience and your
awareness of your own knowledge are the only applicable
guidelines.

Another type of SAS problem relates to specific procedures. In
this case, the user is asking whether a specific PROC produces a
specific result. These types of questions are difficult to answer
without referring to a manual, unless your users concentrate on a
common procedure. At the Census Bureau the most common
procedure is PROC TABULATE. As a result, each of the SAS
support staff has extensive experience with this procedure.

The most time consuming questions are those related to data
processing rather than SAS. These problems require much more
intensive help over a long period. For example, a novice analyst
can come to the Help Desk without ever having designed a data
processing system. As the time spent working with these people
increases, you have an almost irresistible urge. You want to tell
the user to go away for a few days. While they are gone, you
could redesign their data, you could design and code their
programs, and you could process their data for them. All this in a
much shorter time than you would probably spend helping them.
When the user returns, you could hand over the completed
project. Neither your time nor their time would be wasted.

However tempting, you must avoid this at all costs. The Help
Desk is not a consultant shop, it merely helps users find their own
solutions.

In much more interesting situations, you may discuss
exceptional problems with a sophisticated systems analyst.
Often both the user and the help desk individual share extensive
experience with data processing issues. These conversations
focus on specific technical issues. For example, for several
weeks we have been having conversations about SAS
programming techniques for handling a SAS dataset with 250
million observations. In this case, SAS has the same problems
that any programming language would have. Your discussions
center on the techniques and approaches that are optimal for
SAS.

RANGE OF USERS
We always try to be patient and understanding with our clients.
After all, they have been fretting for some time and have made a
series of failed attempts before contacting us. The manuals, if
they have any, read like an advanced textbook in a discipline
that they have never studied before.

We try to portray ourselves as the bridge between the SAS
system and the users. We try to convince them that we are on
their side. We do this by listening to them. We talk out their
problem without hurrying them along. As a user describes his or
her problem we look for clues about what type of SAS user they
are. Are they beginners? Are they looking for a "cheat sheet"
of answers to lead them down the path? Are they social scientists?
interested in finding out what their data mean but not in computing
costs or efficiencies? Are they computer-types who already
know one or more languages and were told to program in SAS?

If the user is a novice, we ask them if they have ever run a SAS
job. This eases the user into talking to us, hopefully about a
topic that he or she knows something about. We have found this
to be a wise investment of a minute or two of our time. The true
savings, however, come for the user who doesn't know how to
submit a job. This discussion will save them frustration and
eventual embarrassment when they have to ask us later for more
elementary information.

If the user is a beginner, it is very important not to criticize the
data organization. They may not have created the datasets that
they have to analyze. We always emphasize how flexible SAS is.
We assure them that we can input all different types of datasets.
We also promote the procedures showing them PROC SORT,
PROC PLOT, PROC PRINT. At this point, we pull out the SAS
manual and show them how to look through it to find what they
need.

A second group of users are analysts: the statisticians,
demographers, and economists. They are busy people who are
now being asked to do their own data manipulation instead of
having a programmer do it. On the whole, these users don't talk
the same language as programmers. Social scientists want to
know "Are people richer or poorer?" "Are there as many men as
women proportionately in all the cells of this age by income
grouping?" Of course, that data is encoded so that male is a type
1 and female is a type 2. What programmers refer to as type 1
records; social scientists refer to as male records. We always try
to talk with the social scientists in their language.

We usually find it best to work closely with the novice user from
the beginning. This helps to ensure success for their first
project. This means that some day they may use SAS again.

As a last resort, we may even write their programming code for
them although they must type it in. If so, we make sure that they
understand it by asking them to explain it. When the program has
run successfully, we show them how to modify it slightly and
run for different categories of analysis. This is usually a great
help to the novice user who wants to master one concept of SAS.
We then explain how directory structures work on the VAX and
urge them to create different directories for different types of
programs.

The last type of user is the computer programmer. If, for
example, they have been primarily a FORTRAN programmer, we
show them how SAS code can resemble FORTRAN so the
conversion process will be less painful. This group of users is,
overall, the most wary. They have been forced in the past to use
many new-fangled packages that promise more than they can
produce.

Help sessions with these users involve describing SAS solutions
for data processing problems. You describe ways to interleave
data steps and procedures to produce the required output.

These users are unreceptive to comments like "You can't do that
in SAS." Yes we say that, even though we also say that anything
you can do in any other language you can do in SAS. You just
can't do it the same way. They focus on the limitations of SAS as
a programming language. They discount the programming
language extensions that SAS does provide, such as dataset
inheritance and PROCs.

One point we have found useful to instill in these users is the use
of PROCs. PROCs can provide either intermediate or final results
rather than hard coding a data step solution to the problem. For
example, the first time programmers use SAS as a programming
language they inevitably use data set do loops to obtain summary
data. A little bit of hand-holding and a discussion of PROC
SUMMARY will do wonders for their programming efficiencies in
SAS and to their appreciation of SAS as a programming language.

We make a practice of calling back all users, regardless of their
level of expertise, to ensure that they have found a solution to
their problem.

WAITING FOR THE PHONE
A happy and doubly helpful help desk is very active between
"Help" calls. Our activities include:

1) Building generalized approaches for specific problems.
These may be SAS and/or operating system problems. This
involves taking the time to work with a problem over several
days. You may take a variety of different approaches either
to solve or to work-around the problem.

For example, our Census clients wanted to use the PC-to-
Host Link to submit jobs from a PC into a VAX/VMS batch
queue and monitor the batch job. We developed several
generic SAS programs which included DCL commands for
determining batch queue status and for displaying VAX/VMS
LOG and US files in the PC output window.

(2) Giving workshops. These can be general ("An Introduction
to SAS") or specific ("The PC-to-Host Link" or "Using SAS in
VAX/VMS"). Our workshops are only about two to three
hours long and are without significant glossy presentation
media. We try to transfer information at a low cost for both
us (the teachers) and them (the students).

(3) Working on special projects. We are always working on one
or more low-priority projects. These are usually prototyping
projects for groups interested in, but not sold on, on SAS.
We use these projects to demonstrate flexibility and ease of
programming to the clients. I prefer to emphasize using
several different approaches to a problem, rather than on
producing a table with a specific format. Thus any single
problem is an excellent excuse for expanding your own
knowledge of SAS.

(4) Writing newsletter articles. If you have a departmental
newsletter, you should start writing articles about SAS for it.
If you don't have a newsletter, you should start one. The
articles don't have to be exciting to all users. We have
recently written articles about SAS startup command files
and producing fancy tables on laser printers. The articles
advertise your existence and verify (hopefully) your
expertise.

CONCLUSION

A successful help desk must be able to identify the type of client
and the type of problem that has been encountered. Both of
these factors play a large role in formulating the correct
response. A successful help desk supervisor must provide a
challenging environment where help desk personnel are always
working on new problems, using new tools, or preparing
workshops for the users.

The authors would like to acknowledge the great assistance of
the other members of the Census Support Staff: Bob Bateman,
Mike Bretz, and Victor Hu as well as members of the ARC staff
that provided comments on this paper especially Kita Garito,
Mark Perdue, and Linda Smith.

For more information, contact:

ARC Professional Services Group
Software Applications and Training Operation
601 Indiana Avenue, N.W., Suite 501
Washington, DC 20004
202/737-2666

SAS is a registered trademark of SAS Institute Inc., Cary, NC,
USA.