Getting the User Community Involved
in Education and Information Sharing

Russell Newhouse, Parke-Davis, Ann Arbor, MI

ABSTRACT

Education and information sharing is not a one-way street. In order to receive effective education, and to receive necessary information, the user community needs to communicate its needs to the providers. But it doesn't need to stop there. The user community can also initiate educational efforts and become the conduit for spreading information. This paper will discuss how we got the user community involved in the education and information sharing process at Parke-Davis, some examples of user-led educational efforts, and some of the lessons that we have learned.

INTRODUCTION

The SAS® user community at Parke-Davis is a diverse lot from a number of different departments, ranging from Application Developers and Statisticians working on clinical study data, to Chemists and other scientists working on preclinical data. It includes power users who are very knowledgeable of almost every facet of the SAS System to users who barely know how to generate a data dump using the PRINT procedure.

The IS department in charge of maintaining SAS contains some SAS users, but is basically not SAS-savvy. It is in charge of the entire computer environment at Parke-Davis.

Historically, it has been up to the IS department to keep the user community informed on the current and future state of SAS, and to provide some education on how to use SAS in our computer environment. Unfortunately, the information and education provided has not always been adequate or designed with the various types of users at Parke-Davis in mind. For example, there have been instances when a new product has been purchased, such as SAS/EIS®, but the user community is not informed that this product has been purchased or how to access and use the new product. So when it comes time to consider renewing SAS licenses, the IS department decides to drop the new product because nobody is using it.

Instances like this make the user community feel that they are not being kept up-to-date on how to use the SAS environment at Parke-Davis. And to make things even more frustrating, the company is forcing the users to leave the "friendly" MVS environment they've been working in for a number of years and move to Unix and NT. Needless to say, the user community is crying out for more involvement in the education and information sharing process.

That's not to mean that education has been completely lacking, or that the user community has had no input into education. But education has tended to be limited to bringing in "canned" training classes, like the SAS Institute training classes. The main trouble with these have been that they are limited to a small number of users, tend to be expensive, and aren't always geared towards what we do and how we do business.

Nor does it mean that the user community was being intentionally ignored. With such a diverse, and largely unorganized, group of users, it was hard to determine not only what the user community needed but even who to ask.

USER TEAMS

It was realized by the IS department and a number of managers in charge of departments containing SAS users that the user community needed a better way to formulate and communicate their needs. It was decided that the best way to achieve these goals was to create a

series of cross-functional (and cross-departmental) user teams.

The user teams would consist of regular users from various departments and backgrounds. While the overall scope of a team would depend on their particular objective, it would be able to make recommendations to management about what user's needs were, and recommendation of actions to be taken by the IS department, management, and even the users themselves. The teams would also serve as a communication conduit back to the user community.

THE UNIX IMPLEMENTATION TEAM

The first team to be formed was the Unix Implementation Team, or UIT (Parke-Davis seems to be crazy about acronyms). Its scope was to discuss the problems that the first set of users who had moved to Unix have had dealing with the new operating system, how the operating system had been implemented at Parke-Davis, and with problems specific to SAS on Unix. While it looked at a number of issues, such as a need to improve print services and the need for more robust Unix servers, the major recurring theme it identified was the need to better educate the user.

The team identified three education topics that needed to be addressed: educating the user on how to use Unix as implemented at Parke-Davis, educating the user on the differences between Unix and MVS, and educating the user on not only how to use SAS on Unix but also the best way to use SAS on Unix. The last two topics were considered to be very important, since it was apparent from the first set of Unix users that we had developed some bad programming habits while we were on MVS.

Since MVS hides a lot of memory management and input/output management from the user, we never really had the need to develop efficient programming and testing techniques. There really was not that much difference in the amount of time needed to push a data set with a large number of variables through a DATA step or a data set with only the bare minimum needed

variables. And testing programs with a large amount of data didn't seem to slow down submission time or affect any other users. Unfortunately, we found out quickly that the same wasn't true in Unix.

It only took a couple of users to bog down all SAS users, and bring everything to a screeching halt. A lot of that did have to do with how the original SAS Unix server was set up and how work space (or the lack thereof) was being allocated, but most of it had to do with the bad habits we were bringing over from MVS, since grossly inefficient programming could bring down even a well-tuned Unix server. While people were used to being able to run large simulations on MVS with no perceived effect on other users, the same was not true on Unix. We needed to educate people on efficient programming techniques.

CODER'S CORNER

While the UIT continued to talk about education, I took it upon myself to create the first user-initiated education effort to come out of the needs discussed by the UIT: a semi-regular electronic newsletter with a name stolen from SUGI - Coder's Corner. It's initial charter was to try to educate my fellow statisticians in some more efficient coding techniques.

While we have a number of statisticians who are quite SAS-savvy, we also have a number who only know what they need to know to get the information that they require. They really don't know much about such SAS features as DROP or KEEP statements, and certainly nothing about WHERE statements. Most have basic SAS knowledge, but seem to be stuck in Version 5; they haven't kept up with changes made in Version 6. They don't really think about the I/O impact of sending the same raw categorical data set through multiple analysis procedures, like FREQ and CATMOD, compared to preprocessing and compressing the data set and using a WEIGHT statement in FREQ and CATMOD. This was the audience that was probably more inefficient and more out-of-date

compared to other power users. So the newsletter was initially designed with them in mind.

Each issue focuses on just one topic, or a couple of related topics, so that the reader won't be overloaded with too much new information. And each issue is kept short, I try to keep it around the size of the typical SUGI paper, something that can be read in 15 - 20 minutes.

I also try to avoid covering only complicated topics. I know that there was a lot of things that seem very basic to me, but that a number of other statisticians, especially those who haven't needed to do as much data manipulation, wouldn't have much experience in. So I warned my audience that for some the topics might seem very basic, but they still might pick something up that they forgot about. Among some of the early topics were:

- KEEP and DROP statements,
- WHERE statement,
- IN operator,
- avoiding sorts,
- preprocessing data, and
- using FIRST. and LAST.

The initial response to the newsletter was quite favorable. As expected, I got a lot of comments from people about how they learned something new, or finally understood how to use a certain feature. I also got a number of requests to expand the distribution list. So now the newsletter goes out to the "data out" SAS user community - the statisticians and the clinical programmers (roughly a circulation of 75 readers). And as I started to run out of efficiency tips, the topics expanded to cover other SAS features and tips.

The newsletter is distributed electronically as a PDF file through our e-mail system. Back-issues are made available through our company intranet. They come out in a semi-regular basis, I try for one a week, but that depends on my work load and the number of interesting topics I have in mind. In each issue, I ask the readers to feel free to suggest topics, or submit problems. I

have gotten some good issues out of the reader feedback.

The main problem in putting out the newsletter is time. Typically, it takes me only about four hours to write an issue, more if it is a topic I'm learning at the same time I'm writing it. But this time is squeezed into my regular work load. So at times when I'm extremely busy, which is very common in the clinical research business, there might be quite a long period between issues. My readers understand this, however, and are willing to wait.

INFORMATION SHARING SEMINARS

One education method that came directly out of the recommendations of the UIT, and the education subteam that was spawned by the UIT, is a series of quarterly information sharing seminars presented by the clinical programming group. Each seminar is an hour and a half in length, and consists of a number of short presentations by clinical programmers. The topics covered in each seminar tend to be related, but don't have to be.

The seminars have covered SAS topics, such as efficiency tips or using print drivers to create HTML or RTF formatted files, but they are not limited to just SAS. The first seminar presented a number of topics related to effectively using Unix and using SAS on Unix, a new task for a number of our users. And future seminars will probably cover such new topics as getting data out of our new data base system. The goal of the seminars is to share experiences, not just in SAS, but in all aspects of what programmers need to know to get their job done.

Since this initiative just started, it is still too early to judge its success. The first two meetings were well attended, but it will be interesting to see if the momentum is maintained.

SAS STEERING COMMITTEE

The second team that was formed is the SAS Steering Committee (SSC). To quote the mission statement, "The SAS Steering

Committee will act as the Voice of the Customer on SAS issues at Parke-Davis. The committee will meet and analyze issues, concerns, and directions for SAS use at Parke-Davis. The committee will formulate recommendations for Parke-Davis Management based on the results of their analysis using concordance decision making." Unlike the UIT, the SSC is concerned about SAS and only SAS, on any and all platforms: NT, Unix, and MVS.

What's not in the mission statement is what has become one of the major purposes of the SSC, to keep the user community up-to-date with the current state of SAS, and the future of SAS, at Parke-Davis. To further these purposes, the SSC has been looking at various methods to "spread the word" about SAS at Parke-Davis.

The first method used is the creation of a SSC page on the company intranet. The SSC page presents a lot of information about SAS, including:

- the supported versions of SAS on each platform (including the specific TS release, information needed for validation purposes),
- the default version of SAS on each platform (i.e., the version you get if you just type "SAS"),
- minutes of SSC meetings,
- proposed plans for the future of SAS at Parke-Davis,
- links to information on SAS such as:
    - web tools,
    - Release 6.12, and
    - SAS information and tips from various sources, both from within and outside the company.

Another method to spread information about SAS that is being considered is holding a "SAS Fair". This half-day affair will be similar to the demo area at SUGI. An area will be set-up, with a number of computers (and demonstrators) available to demonstrate a certain feature at each machine. Users will be able to wander through at any time, and see whichever feature they desire. Along with these ongoing, on-demand, demonstrations, a demonstration theater will be set up to present key features of Release 6.12 to a small audience in short presentations.

The main purpose of the fair is to "evangelize" Release 6.12, introduce new features and packages (like the web tools), and encourage the users to try out the new features. It is hoped that this will take care of a recurring problem we have had with new products in the past.

When a new product had been installed before, say SAS/EIS, there was rarely an announcement from the IS department about the existence of the new product, no presentations on what the new package could do, and no training on the new package. That meant that after licensing it for a year or two, the IS department would wonder why nobody is using the new package, and would drop it from the license. There have been occasions when a little time afterwards, somebody would come up with a need for this package, and when they asked the IS department about it, they would reply "Well, we had it before and nobody used it. Why should we get it now?"

By introducing the new products at the SAS Fair, especially the web tools (which the SSC members see as potentially having the biggest impact on how we do business), we can try to generate user interest, and speed up the use of the products.

WHY NO USER'S GROUP?

One of the questions that I'm sure people reading this paper are asking is why we haven't formed a SAS User's Group at Parke-Davis? Traditionally, a User's Group serves the same type of education and communication role as Coder's Corner, the information sharing sessions, and the SAS fair. The SSC considered the issue of starting up a User's Group, but decided against it (in fact, it was during this discussion that the idea of the SAS Fair came about).

Parke-Davis did have a company SAS User's Group a number of years ago. It was started by a group of users from various departments who

wanted to better spread the word about SAS at Parke-Davis. It started with great enthusiasm, and well-attended User's Group meetings. But it didn't take long for that enthusiasm to wane, as it became obvious how much work running a User's Group can be, especially by a small group of people with no dedicated officers. It became more and more difficult to fill an hour and a half meeting every three months, and attendance started to dwindle to the core group of power users who didn't need as much help with SAS as other members of the user community. When the person who was pretty much in charge of running the meetings went on maternity leave, the User's Group faded from existence.

So based partly on this past experience, the SSC felt that it was not desirable to restart the User's Group. But the SSC also felt that a User's Group is really not the best way to reach all users. It seemed to us that user groups tend to attract all types of users in the early stages, but in the long run tend to attract mostly power users. And while there is nothing wrong with power users getting together to try to push the envelope, it doesn't really help our mission to educate and inform all users. The SSC would support anybody who wanted to start a User's Group, or increase participation in the Michigan SAS User's Group (which a number of our power users are members of), but we felt that the SSC was not the appropriate group to organize and run a User's Group.

LESSONS LEARNED

We are still very early in this process, so it is hard to come to some concrete conclusions about what has worked and what hasn't. But there are already some lessons that we have learned.

The first, and probably the most important, is the need for supportive Management. You need Management that understands that education and information sharing is important to the success of the company, that it shouldn't just be an afterthought, and it shouldn't be abandoned due to time and budget constraints. They should understand that decisions should be explained, that people are more willing to accept a "negative" decision if there is a sound explanation instead of being presented as a dictum. And management, especially in IS, should be willing to listen to the users and not just brush them off.

While this is important, it is also one of the hardest things to get. We've been very lucky at Parke-Davis the last few years in that we have a management group that is very supportive, and it is hard to say what has created it. I personally think that it is a result of two things. One is the fact that a lot of our current middle management consists of people who have come up through the ranks at Parke-Davis. They don't want to repeat the mistakes they've seen happen in the past. The second is the current emphasis on the team philosophy, which includes management as a player and not just as a coach. Most of the colleagues at Parke-Davis have gone through some sort of team building training, and it has improved communication.

We've learned that for an education effort to be successful, the users need to commit themselves to wanting to learn. Users need to be active in the learning process, they can't learn by osmosis. No matter how many ways you create to spread information, it means nothing if the user is unwilling to participate. We have had instances where we have spread the word about a change using many different methods, and still have some users who say "nobody told me about this." So one of our core values has become education, and this is part of a colleague's performance evaluation. If a manager sees that one of their direct-reports is unwilling to learn, they will discuss it with that person, and if that attitude continues, it may impact on their performance appraisal and compensation.

We've learned that the educational method used depends on the educational need that is to be met. One method does not fit every need. Sometimes all-day seminars work, other times it is best to use short, focused training methods. And the process should not be static. The method that worked six months ago might no

longer be adequate. Training should keep pace with the users. So we try our best to fit the method to the need. And if it doesn't seem to be working, we try something else.

And finally, we did learn that the user community needs to get involved in the education and information sharing process. Nobody better knows the needs of the user community than the users themselves. They don't have to take complete charge of training, in fact they should leave a lot of it to the education experts, but they need to at least participate in the definition of the requirements of training. They need to identify what their educational needs are, and give that information to the people who can do something about it. They should also be encouraged to initiate whatever training they are qualified to do themselves. It is counterproductive if a company's training group feel that they are the sole provider of education. There are quite often needs that they cannot meet or provide in a timely manner, and are best left to the user community.

You may contact the author at the following address:

Russell Newhouse
Senior Statistician I
Parke-Davis Pharmaceutical Research
2800 Plymouth Road
Ann Arbor, MI 48105

or by e-mail at:

newhour@aa.wl.com