FROM MAINFRAMES TO MICROS: EFFECTS ON WORKSHOP DELIVERY

Susan J. Bahr
Academic Computing
Office of Systems and Computer Services
Cuyahoga Community College
2900 Community College Avenue
Cleveland, OH 44115
(216) 241-6261

Introduction

Helping users become self-sufficient is a chief function of User Services. To help the user obtain self-sufficiency, documentation and newsletters are written and promulgated, one-on-one consulting is provided, literacy courses are offered, and workshops are taught.

From my experience, I have learned that users want to know about computers but they don't want to invest any time in learning. They won't open a manual even if it's written for novices. They refuse to do anything except scan through newsletters in an attempt to minimize the time invested. They consider literacy courses a major commitment of time. Consulting seldom works unless it is done as a response to a question. However, because workshops only require the user to show up at a designated time, they seem to be the best means for imparting computer information to the lazy user. And let's face it, the consumer of User

Services is lazy.

These broad generalities are especially true at Cuyahoga Community College (CCC). We distribute hundreds of novice-level manuals to faculty/staff/administrators each term, but even the most literate users won't open them. The manuals mostly serve as decoration for their bookshelves. I even get calls from sophisticated users who marvel at how much I know about our text editor, and are amazed that I learned it all by simply reading the manual.

We publish a newsletter six times a year. Most of the articles are written for people who know very little about computing at CCC. Unfortunately, only a select few of the CCC staff have come to realize that it is an invaluable information source.

We have had a very successful computer literacy course in place for two years, but it is a rare

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1983 ACM 0-89791-116-4/83/010/0169 \$00.75

participant who shows up for all of the ten or more classes. Furthermore, the participants rarely spend much time outside of the class trying to learn the material.

We provide a good deal of consulting support, but those people who do come to see us usually have ill-defined ideas of what they want and high hopes of spoon-feeding.

Because of the CCC staff's love of spoon-fed computer knowledge, workshops seem to be the most successful of all our User Services. Over the past year, the number of people registering for workshops has doubled and now exceeds the number of people the staff can support.

This paper examines the reasons for the successes and failures of this method of user education, especially those reasons which are pertinent to my workshop delivery techniques. These techniques have changed over time in order to accomodate the change in the users' attitudes towards workshops.

Lecture

A few years ago, "Lecture" was the mode of delivery for workshops I taught at Temple University in Philadelphia. When teaching a class in SPSS, for example, the blackboard and handouts were my primary tools. Luckily, SPSS was primarily run in batch from punched card input. So writing lines of JCL or SPSS control cards on the board related directly to the cards that would have to be punched. Back then, the SPSS workshop participants were content with the teaching methods.

But SPSS wasn't the only thing happening at the Temple University

computer center. People were beginning to discover timesharing on the CDC Cyber 174, so workshops were given on the text editor, Senator. These workshops were taught in the lecture mode. We did blackboard simulations of timesharing, distributed copies of timesharing sessions, and fully expected people to walk out of those classes, sign onto timesharing and actually accomplish something. There was hope for them if they followed our sample timesharing sessions, but they were not prepared to handle error messages and figure out the proper recovery procedures.

Surprisingly, people didn't seem to mind, because timesharing was "new" and they still had the option of using cards. So they only came to the workshop if they were willing to struggle through the initial learning process in order to reap the benefits in the long-run.

Demonstration

After Temple, I taught a few workshops at Trinity University in San Antonio. Trinity had one room set up with large monitors that were slaved off of a terminal in the front of the room.

"At last, an on-line demo," I thought.

Two problems immediately became evident. The first was that on-line demos by a computer center person tend to be flawless, so the participants rarely see what to do when mistakes are made. The second problem had to do with Trinity's running CMS with full-screen editors. The terminal in the demo room could only handle the line-editing commands of the full-screen editor; the full-screen commands were presented in the

"Lecture" mode. So once again I was back doing blackboard simulations, but since they didn't miss what they hadn't had, the users were content.

Hands-on

When I later came to Cuyahoga Community College in Cleveland, the users were using timesharing on the Honeywell DPS/1. There were no keypunches in sight.

"Aha!" I thought, "I'll finally be able to teach in a hands-on mode."

I began teaching classes in the terminal rooms on the various campuses. The main topics were MAIL, the electronic mail system, and FRED, the text editor. The users got to have their hands on a terminal. They were able to experience first-hand the joys of having the computer talk back to them, and the horrors of receiving error messages. MAIL began to catch on like wildfire.

Because there are four physical locations of CCC throughout Cuyahoga County, MAIL allows people to communicate quickly with each other, instead of waiting for something to arrive through campus mail or playing "telephone-tag". The other key feature of MAIL is that it is easy to use. You need to know how to log on and type MAIL. Then you need to know the Send command to send a message, the Print command to print messages, the Delete command to get rid of old messages, and the Quit command to get out of MAIL. (You also then need to know how to log off.) It's a relatively simple system to use.

FRED, on the other hand, is too complicated for many users. It's not full-screen and you have to get used to the invisible pointer

pointing at your current line. You cannot just type over mistakes; you must use the Substitute command and type in the wrong version again. There are just too many commands to learn, and errors lead to really cryptic error messages. Teaching FRED is difficult even with the "luxury" of having the terminals needed for the hands-on approach.

During the past few years, users have become increasingly dissatisfied with FRED workshops. With the emergence of arcade and home video games, the average person could destroy an invader form space with the push of a button. When the 24-hour banking machines were installed at many banks in the Cleveland area, the touch of a few more buttons produced money! This caused the average person to think that computers save large returns for small investments of time and effort. So what were they doing in a FRED workshop which required too much time and effort?

The emergence of the video games, 24-hour banking machines and other computer applications available to the general public created well-defined expectations by the workshop participants. They were no longer content to come and accept whatever was taught. The users now came to a workshop with a "Show me!" attitude but added onto that is a touch of "Tell me why I'm here!" and a little "Entertain me!" thrown in. All this was supposed to equal "Teach me!", even though they didn't always act like that was their goal. The users wanted a show, and you wanted them to learn. If learning required too much effort, you got turned out and they got turned off.

The problem, therefore, was to be able to teach them what you wanted them to learn while somehow

managing to give a "really good show". This problem, which is a product of the microcomputer age, continues to plague the workshop leader.

Micros

Last December, we taught a two-day workshop to the administrators at the West campus of CCC. We taught FRED and MAIL hands-on in our terminal room with neither topic being well received. The attention span of the administrators could be likened to that of three-year-olds. They wouldn't sit still for a brief introductory lecture before the hands-on part began. They wanted to walk in, sit down, log on and get into the class without concern for whether or not they would understand what they were doing. They just wanted to do it.

With MAIL, they wanted the "Show me!" and "Entertain me!", but "Teach me!" was not important. The idea of receiving a message from another participant took care of the "Entertain me!". Being able to send a message to someone, satisfied the "Show me!" criteria.

FRED was another story. The "Show me!" part was too hard, so they were not entertained. With these two criteria not being met, "Tell me why I'm here!" became the foremost request. But that never really was satisfied since no application seemed worth the trouble of using FRED.

All hope was not lost. Apples were flourishing at CCC, and the Dean realized they could be used for budgeting. So as part of the two-day workshop, we also taught VisiCalc on the Apple in a hands-on mode. The administrators sat down at the Apples, and had immediate control of the cursor. They were in

command of the computer, not the other way around. After typing a few numbers and formulas, they quickly realized how fast the computer was performing calculations that would otherwise take them hours by hand. The "Tell me why I'm here!" was apparent because they realized how much time they would save by using VisiCalc to do their budget calculations. The "Show me!" and "Entertain me!" were easily taken care of by the friendly microcomputer. During breaks, many of the administrators stayed at their Apples and worked on designing various models they would need in the weeks to come. The "Teach me!" criteria became something they wanted since the other three criteria were fulfilled to their satisfaction.

Conclusion

After seeing the administrators' reactions to the VisiCalc workshop, we are even more aware that every workshop must be taught hands-on. Our awareness of this need is apparent when comparing workshops from last spring where only half were taught hands-on with the workshops this spring in which 13 of 14 were hands-on.

We are making more of an effort to come up with very structured demos which are available to each participant during the workshop. An example is our GRRAMS database manager workshop in which previously prepared databases are available to the participants during the workshop. We make special efforts to overcompensate for the comparative user-unfriendliness of the mainframe whenever we teach mainframe computing topics.

We have also been teaching an

increasing number of microcomputing topics with the hope that microcomputers will be the vehicle by which we get novices hooked. Last spring we had no microcomputing topics taught hands-on; all we had were demonstrations of microcomputer software. Of the 34 workshops which were taught as part of our workshop series this past fall, winter and spring, 11 of the workshops were hands-on microcomputers, while only 2 were microcomputer demonstrations.

Regardless of how well we present these workshops, there are still some drawbacks to this mode of user education. First, the time spent in a workshop is not the same as day-to-day experience with a package. For example, trying to warn users that diskettes can be damaged and that mainframe storage may be better is something they will never believe until their dog chews up the diskette which contains their dissertation. Second, after a workshop is given, users must spend time practicing what they learned. Too often, college staff members attend a workshop on MAIL, never use it, and then sign up a second time to learn it again. This is waste of the computer center's time as well as their own. Third, users must begin to read the manuals we distribute. We write manuals at the novice level, reproduce them and hand them out during the workshops. But users rarely use them as a source for solutions to their computer problems. Fourth, before attending a workshop, the user should have some idea of his/her reasons for attending. They should investigate the topic before coming, and if they discover that they have no use for a particular application, they should not attend the class.