

TITLE: "SYSTEM TRAINING TIPS"

by Tim Bryce

Managing Director

M. Bryce & Associates (MBA)

P.O. Box 1637

Palm Harbor, FL 34682-1637

United States

Tel: 727/786-4567

E-Mail: timb001@phmainstreet.com

Yahoo! IM: littleleaguerng

WWW: <http://www.phmainstreet.com/mba/>

Since 1971: "Software for the finest computer - the Mind"

*"You cannot treat a patient if he doesn't know
he is sick."
- Bryce's Law*

INTRODUCTION

Okay, you've built this wonderful new "state of the art" system. Now comes the hard part: startup. To do so, you will have to train the user community in how to effectively use the system. Do not underestimate this task. Now is not the time to grasp defeat from the jaws of victory. Let me give you an example; years ago my wife was a contract administrator for a large jet engine manufacturer in the Midwest. Basically, her department was concerned with ordering parts for the jet engine assembly lines and making sure they were delivered on time. Most of this was done using parts catalogs, index cards, and a lot of telephone calls. It may have been a bit clumsy, but it worked. Wanting to expedite this process, the systems department invented a new Materials Management System (MMS) which was to be used to order and track parts, as well as to interface with the company's Finance System to evaluate engine costs. After the systems department built the MMS, they called for a training session to introduce all of the contract administrators to the new system. At this time, they gave everyone a new TI Silent 700 computer (which featured acoustic couplers at the time) and a list of cryptic commands on to how to use it. The whole session lasted about one hour and did more to confuse the users than to educate them. So much so, that on the Monday when the MMS was to be started, nobody used it and, instead, went back to their manual procedures.

Humans are creatures of habit and, as such, training the users on how to implement a new system must be handled carefully. And it is more than just having good educational skills, its about breaking habits and creating new ones. For example, years ago when companies were beginning to migrate away from DOS to operating systems with Graphical User Interfaces (e.g., Windows and

OS/2), systems people were amazed to find the user community resisting the migration. Surely the new GUI-based operating systems were easier to use and understand, right? Maybe, and maybe not. A lot of users felt comfortable using their favorite word processors and spreadsheets under DOS. Why then, they questioned, should they be forced to move to something else? Good question, one that was seldom answered by I.T. departments. In this case, the answer was: first, to bring a uniform consistency (look and feel) to all applications on the PC, thereby simplifying the learning and implementation of programs, and second; industry trends (away from character based operating systems).

MY BACKGROUND

For the last three decades I have been teaching methodologies, be it related to systems planning, systems development, data base or project management. Frankly, I believe teaching the development staff how to use a methodology is more difficult than teaching users how to use a new system. Interestingly, systems people, who are supposed to be the agents of change, are remarkably the most resistant to it. Nonetheless, I have learned four important lessons from this experience which is applicable to training users in systems:

1. **Provide a tutorial which describes the rationale for the new system**, and defines its concepts and terminology. In other words, sell the system. Be careful to couch the presentation in terms the users will understand, and because of this, avoid introducing technical jargon as much as possible. You want to enlist support for the system, not provide excuses for not using it. Also, all questions should be welcomed and not ridiculed. User questions may appear trivial and foolish at times but your intention is to overcome all objections.

2. **Provide "Hands On" Training** - true it is important to give an academic explanation of how the overall system works, but it is also important for users to actually "touch and feel" the system. They may not come away from the training class as experts, but at least you will have overcome their fear of the new system.

For complicated processes, I tend to offer manually implemented exercises so the students would gain an appreciation for the need for automation. For example, in order for a person to appreciate a calculator, they should first have an appreciation of basic math. If the users understand the processes being automated, they will be more inclined to trust the new system.

(continued on page 2)

(continued from page 1)

3. **Recruit Management Support** - the implementation of a new system, like a methodology, requires the unwavering support of management. If the user community senses the slightest lack of support for the system, they will use it as an excuse not to use it. Because of this, I strongly encourage a representative from management to introduce the speaker/trainer and to monitor the training proceedings.

4. **Create excitement to use the new system** - this can be achieved several ways; for example, promotional buttons, pins, a kickoff party, endorsements, etc. Think of it as releasing a new product. Another option is to train the user community in stages, e.g., taking key "germ carriers" and making them proficient in the system before others, then let them spread the word to the rest of the user community. A little attention to key users can go a long way.

GENERAL TIPS

Aside from the system tips listed above, there are some general tips you should be cognizant of as a trainer:

1. **Be organized** - prepare a well thought out agenda and stick to it. I'm also a big believer of the military approach whereby you *"tell them what you're going to tell them, tell them, then tell them what you've told them."*

2. **Know your audience** - understand their intelligence level and interests, and design a training program around them, not the technology. Yes, you want them to move forward on technology, but you cannot afford to alienate them. By working within their limitations you will be able to accomplish more.

3. **Dress and speak authoritatively** - your appearance and how you present yourself says a lot to the audience about your system. If you dress and act like a geek, the user community might look upon this as another hare-brained scheme by the I.T. department. The appearance and presentation of the trainer reflects the credibility of not only the speaker, but of the system as well. Do it first class and earn the respect of the students.

4. **Stimulate the students; don't put them to sleep** - keep the training positive and upbeat; inject humor where necessary. Allow periodic breaks, but keep them short and sweet.

5. **Select a suitable venue** - hopefully something where the attendees will not be distracted and allow them to focus on your subject.

6. **Provide supplemental training aids** - such as reference cards or perhaps a CD/DVD with a multimedia presentation (e.g., MS PowerPoint, Lotus Freelance, or a podcast). Blogs and discussion groups (list servers) are also useful to act as a clearinghouse for answering questions.

7. **Critique the training program** - allow the students to evaluate the training course. Their feedback will hint as to your success and may point out problem areas that need to be addressed.

CONCLUSION

As someone charged with a key role in changing the status quo, the systems trainer must first understand that his audience does not necessarily want to change. Some of the users will welcome change as they are aware of the shortcomings of the current system. However, others have adapted and feel comfortable using the current system regardless of its shortcomings and, as such, will resist change. The trainer is left with the task of convincing the users not only is the current system inadequate, but that he has a superior alternative to replace it. After all, you cannot treat a patient if he doesn't know he is sick.

Just like your systems development efforts, training requires planning, organization, execution and review. Too often I have seen companies underestimate the training effort and put forth only minimal effort to properly train the user community. Such shallow thinking inevitably leads to disaster later during system startup. I have even seen disgruntled users sabotage the best of systems simply because they didn't understand it or it was not presented well.

The system trainer's mission, therefore, is to explain, demonstrate, and convince the users how the new system will not only benefit the company, but the users as well; and communicate it in terms the user community will understand. Remember, a verbosity of technical jargon impresses nobody but yourself.

END

About the Author

Tim Bryce is the Managing Director of M. Bryce & Associates (MBA) of Palm Harbor, Florida and has 30 years of experience in the field of Information Resource Management (IRM). He is available for training and consulting on an international basis.

"PRIDE" Special Subject Bulletins can be found at:

<http://www.phmainstreet.com/mba/mbass.htm>

They are also available through the "PRIDE Methodologies for IRM Discussion Group" at:

<http://groups.yahoo.com/group/mbapride/>

You are welcome to join this group if you are so inclined.

The "Management Visions" Internet audio broadcast is available at:

<http://www.phmainstreet.com/mba/mv.htm>

Also, be sure to read Tim's Blog at:

<http://blogs.ittoolbox.com/pm/irm/>

"PRIDE" is the registered trademark of M. Bryce & Associates (MBA) and can be found on the Internet at:

<http://www.phmainstreet.com/mba/pride/>

Copyright © MBA 2006. All rights reserved.



Since 1971: *"Software for the finest computer - the Mind"*