Roger Dove

This is not a learned treatise on training: rather it is a review of the lessons learned during the implementation of one particular system in one particular setting. It is likely, however, that similar circumstances could be encountered with other implementations and these thoughts are offered in the hope that others can learn from our experience.

The system being introduced was the core module of SOSCIS (Social Service Client Information System), an ICL mainframe application which contains some scope for customisation. This was installed over a period of twelve months in a region with a population of around 200,000 but scattered over a land mass bigger than the whole of Wales. Social work service is provided through fourteen area teams and three hospital teams with area team leaders acting as "gatekeepers" for all client services in their respective areas. There are approximately 7,000 active cases on the books at any one time.

My own involvement in the project began after the decision had been taken on which system to purchase, and when the need for a Project Leader to oversee the implementation had been identified. At that time I was employed as Departmental Training Officer, and my only previous involvement with computer applications was over twenty years ago when I turned my back on industry in order to "work with people". Needless to say, hardware has changed almost beyond recognition in the intervening years, but I was to find that my industrial experience of systems work was a decided asset.

The Project Team was completed by two administrative staff who were seconded to me, initially on a part-time basis but subsequently full-time. These were both much more computer literate than me in contemporary terms, but neither had any formal computer training or experience in implementing systems. As a team, therefore, we had a useful blend of knowledge and experience, but we also had a great deal of learning to do as we went along. The following observations are based on the lessons which we learned in the process, and on the successes and problems which we experienced.

In retrospect we were able to identify six distinct stages which the project went through and which I will describe briefly as follows:

1. System Specification: defining the purpose of the system and the required outcomes.

2. System Design: deciding whether to go for a bespoke system or "off the peg" model.

3. Development: the work which needs to be done to fill in the detail and get a thorough knowledge of how the system will operate is a prerequisite to designing training programmes.

4. Training: must obviously be task centred, but must not lose sight of the personal factor.

5. Installation: when there is a need for support and consolidation on training.
6. Backtracking: where certain fundamental questions need to be addressed:

(a) are the data accurate?
(b) is the system operating efficiently and effectively?
(c) is it delivering the goods as specified at Stage 1?

Our task as a team commenced at Stage 3 of this model with what we called the development of the system which involved:

(a) thoroughly digesting the supplier's manuals;
(b) setting up a test package on which we could simulate the operation of the system and try out our ideas;
(c) conferring with staff on codes and description, and then with the suppliers on the scope for customisation;
(d) designing forms and then testing the first drafts alongside existing paperwork on the manual system;
(e) writing a readable User Guide which was to become the "bible" for users and the basis of our training programmes.

This process took us all of six months and I am quite convinced that the relative ease with which the training and implementation was done owes a great deal to the thoroughness of this development work. The one area in which I think we could have improved was in consulting with staff, and all I would say here is that I see considerable value in involving practitioners who have a detailed knowledge of what happens and what is needed at the coal face. With the best will in the world middle managers often do not and perhaps cannot have this knowledge.

In setting up the training programmes we identified the different needs of operators, who would be responsible for entering and updating data, and users, who would want to access the system for information. We also distinguished between the respective needs of basic grade staff and middle and senior managers.

The actual content of the programmes followed naturally from the content of the User Guide, but a lot of thought was given to the way that the training was delivered. We were conscious of the fact that aptitude would play a large part in the rate of learning and that some people have an emotional block at the very mention of computers. We therefore set out to make it as non-threatening as possible - indeed to make it fun.

One of the implementation team, with no previous experience in training, took on the specific task of developing and running the training programmes, and quickly developed a style which was very well received by those being trained. In particular she tried to personalise the process and, largely for this reason, settled on working with a maximum of three people in any one session.

Operators were given a three-day training in which they were progressively exposed to the complexities of the system, and able to practice on the test system without the slightest fear of doing anything disastrous. They were also given the assurance that support
would be given to consolidate on this training when the system went "live" in their particular area team.

Obviously the time lapse between training and implementation needed to be kept as short as possible, which led us to set up a rolling programme whereby the whole implementation team would move into an area office within two weeks of the local operator(s) being trained, and would play a very active part in setting the system up and getting all current caseloads keyed in. At that stage an assurance was given to the area team that the implementation team would not terminate its involvement until the users were confident in using the systems. Also a lot of good will was generated by the implementation team's willingness to get stuck in to gritty tasks like sorting through current (often inaccurate and out of date) records and keying in client details. In practice this meant that all three of the implementation team were involved in an area team for about 2 to 3 weeks, with support then dropping off to one or two members, to allow the third member to start the training programme with the next team.

The need to consolidate on training in the operator's own work setting was, in our experience, of fundamental importance, and it was very revealing to see how often people who had coped well with the initial training just froze completely when confronted with the live situation.

Although the users of the system (social workers, team leaders, etc.) did not require the same detailed knowledge as the operators, we still felt that they needed a full one-day training, partly to give confidence in accessing the system, and partly to impress upon them the importance of feeding necessary data in the right form to their operators. The basic one-day training was the same for all users, except that middle managers received more in the way of access to management information.

Overall the programme of training and installation went extremely smoothly, with no tears, tantrums, psychosomatic illness, walk-out or verbal or physical onslaughts on the implementation team. This is not to say, however, that there were no lessons to be learned, and I think the most important of these was the need for management at all levels to play a positive and active part in the programme.

Inevitably the introduction of a new system will entail change, and will require decisions to be made on issues which might sometimes be controversial. These are clearly management responsibilities and management should, in my view, take a positive role in getting this message across. In short, the implementation team should not be left on their own to "sell" the system - they will almost certainly have enough on their plates without this!

One small example of how easily these situations can arise occurred when the decision was made not to include foster parents on the computerised client index, the reasoning being that they fall more into the category of colleagues than clients. This did not seem an unreasonable decision, but it did constitute of change from previous practice, and needless to say, did not meet with the approval of some users of the index. As a consequence the implementation team were frequently challenged on issues, such as access to information, and whether the system offered a true reflection of workloads. The tempting response was to say "don't ask us, ask them (management)" but, as "they" were seen as the instigators of the new system the
only effect would have been to undermine the credibility of the system. If this situation is multiplied a few times, then you have an implementation team which feels continually on the defensive and likely to develop something of a siege mentality. The message for me is simple - management need to be actively involved throughout the project.

Approaching the end of the implementation stage it is tempting to believe the task is almost finished - whereas I would argue that there is still a substantial job to be done. In the euphoria of a new system which, hopefully, is a major improvement on the old one, it is easy to overlook the steps which are necessary to ensure its viability. Indeed, if the system being replaced has got into a mess (and this is often one of the reasons for computerising), it could be instructive to look back at the reasons for this. Among them will probably be found things like:

- failure to observe basic disciplines;
- inadequate instruction;
- inadequate supervision;
- poor communication.

Computerisation, in itself, will not remedy these short-comings, although the introduction of a sophisticated system should provide sufficient reason to address them and, unless sufficient attention is given to this, I suggest that you might wake up one morning to the realisation that all that has happened is that a manual shambles has been replaced by a computerised shambles.

I am, therefore, making a strong case for adequate attention and resources to be given to what we have termed the backtracking stage, which should give particular attention to the following areas:

1) Are the data which have been entered during the installation phase accurate and complete? The chances are that in the process of transferring a substantial volume of records from a manual system to a computer system certain things will be overlooked. It is worth going back when the dust has settled!

2) Are the basic disciplines in terms of procedures, accuracy and timing being followed? Is the line management/supervision structure adequate and sufficiently well informed to ensure this?

3) Have the respective offices been weaned off a dependency on the implementation team?

4) Are training routines established to ensure that new staff are properly trained, rather than "picking it up as they go along"?

5) Are procedures and guidelines too open to interpretation to give meaningful statistics: e.g. do all social workers/teams use the same criteria when deciding to close a case?

6) Is the system coming up with the goods as specified at the outset?

I realise that I have interpreted my brief to address the subject of training rather liberally. I do sincerely believe, however, that in a project of this nature, training cannot be considered in isolation, and I hope that our experiences and these observations will prove helpful to others embarking on a similar course.
The author is Training Officer for the Highland Region Social Work Department.