Norman Eason's "Maintenance and Asset Management Information Systems"

I can identify five areas of activity that sooner or later need to be addressed:

Developing an information architecture;
Putting the technology in place to support this architecture;
Aligning incentives with the new system;
Drawing on outside resources;
And designing a process to ensure that the other four activities occur.

Robert G Eccles, The Performance Measurement Manifesto, 'Harvard Business Review on Measuring Corporate Performance'

Chapter 14

Successful Procurement for Through-life Effectiveness

Despite the rather grand title of this chapter, it will not be a comprehensive guide to the implementation and management of projects. There are already libraries of books on this subject and I have no intention of adding to their number. It is assumed that the reader will either be familiar with standard methods of project management or will acquaint himself or herself by means of standard textbooks.

This chapter is intended specifically for the implementation of maintenance and asset management information systems, although some of the points made will also be applicable to other areas.

Note that we are not considering procurement in the manner commonly associated with the purchase of software, i.e., purchase it, install it and use it. The fact that we are aiming for through-life effectiveness puts greater demands on the entire exercise. 'Through-life' in this context means the total useful installed life of the acquired system. It requires consideration of the usefulness and delivered value of the system over a relatively long time. The timeframe is very much the personal choice of the procuring organisations as they, one hopes, are in the best position to estimate how long the software will remain in effective use. Unfortunately, this is not always the case.

The timeframe should be, as far as is possible, a **planned** period, with actions taken to ensure that the system remains effective throughout its installed life. Unfortunately, the life of a maintenance or asset management system is often unplanned and is usually terminated by lack of interest or the desire to use a newer, more exciting system. Both of these causes are usually the result of poorly managed software or information. As the software and information are potentially valuable assets, this points to **poor asset management**.

Effectiveness and Change

If we are to achieve successful procurement for through-life effectiveness, we must first consider what this effectiveness is and how the lives of the two prime assets – the system and the resultant information – are determined by decisions made inside and outside the maintenance or asset management departments. These are essential starting points. If you cannot obtain satisfactory answers to these questions, answers that are acknowledged as fact by all related personnel, then you will have based the entire procurement exercise on assumptions that could later prove to be invalid.

Few organisations go to any length to determine these relationships and thus to manage the useful lives of these assets. That this is the case can be seen by the frequent procurement of new systems by many organisations with no apparent ongoing strategy for the data and information that is transferred from one system to the other. Organisations that fail to manage their data and information will almost certainly fail to manage their information systems. If they do not have a plan for both, what is the point of procuring a new information system?

The effective lives of both the system and the information will, of course, be subject to changes caused by changing business requirements. This is, however, no excuse for failure to plan for the management of both their lives. Of course business requirements will change, and many of these changes will be

"Change, after all, is only another word for growth, another synonym for learning.

We can all do it, and enjoy it, if we want to."

Charles HandyProfessor, The London Business School

almost impossible to forecast. Nevertheless, it is really not difficult to provide in some way for these changes and the affect that they will have on the information required by the maintenance or asset management department and on the system upon which that information and its data will be based.

If we consider the information requirements of such unspecified changes, it is possible to estimate how these **could** change as a result of changes in business requirements. This will not be able to be done in a precise manner, but even a fairly crude estimate is much better than none. For example, will any change require a larger quantity of data and, if so, will the system be able to be expanded to cope with it? Will it be necessary at some point to drill down from the information into the data in a manner that is deeper and more widespread than that which is initially required? If so, can the system be easily changed **by the user** to handle this? If organisational changes mean that a change in the grouping of the workforce or the manner of reporting is required, can the **user** adapt the system in order to support this new structure?

These are only a few examples of the very many activities that could be considered in relation to possible changes and which would, by virtue of simple statements in the procurement specification, ensure the continued operability and effectiveness of the system and its information as a result of these changes. Even although no precise match to changing requirements could be expected, this action will have enabled the major options to be catered for and at the very least will have prevented the adaptation of major software or will have prolonged a replacement decision.

Self Assessment

It is essential to determine what type of organisation yours is. Are its policies towards maintenance best defined as **asset management** or does it still maintain an attitude towards maintenance that constrains the operation within a **departmental maintenance** scenario? As we saw in Chapter 2,

"World Class maintenance needs consistency for long periods of time."

Christer Idhammer

Author, 'World Class Maintenance'

whichever of the two types applies will radically change the overall approach to the procurement of a system. This point cannot be emphasised too much, primarily because organisations are just as much involved in deluding themselves as are ordinary individuals. I have seen very many large organisations that claim to be operating an asset management policy when in fact they treat their maintenance department in a very traditional and departmental manner. This is not helped by the habit of such organisations of procuring the most expensive systems because they believe that by doing so their maintenance department will automatically become world class. Such actions are, of course, often encouraged by the vendors of expensive systems who are only too delighted to add another prestigious name to their client list!

As we have seen in many earlier chapters, proclaiming that one is world class and paying for the most expensive system does not result in excellence. This path is a long one that requires many other constituents. Indeed, the act of stating that an organisation is going to be world class without laying the necessary foundations will itself cause many more problems to a maintenance department than if the objective had not been stated. Such an action can only lead to dissatisfaction and dispute. World Class organisations require much more than bold statements and money lavished only on visible areas.

The attitude of those organisations has, however, another, more subtle, effect. Their staff tend to operate within a separate agenda. The staff knows that the objectives are impossible, but they are all aware of the effect that stating this truth to their superiors would have on their own careers. They thus pay lip service to the objectives and methods while ensuring that no blame will be ascribed to them when the objectives are not achieved. This is no way to implement a project, but I have seen very many projects operate in this way. Clearly, it is much better to start off by an honest statement of what can be achieved, what resources are required in order to achieve it, how long it is likely to take and what it is budgeted to cost. This may sound very idealistic, but consider the downside of deluding oneself and one's organisation!

"Every company has two organisational structures: the formal one is written on the charts; the other is the living relationship of the men and women in the organisation."

Harold Geneen

Business Consultant and

Chairman of ITT

Maintenance Strategy

We have considered the relationship between the maintenance or asset management department and the business, recognising that this relationship must be able to change as the business changes. The most obvious way in which this relationship can be expressed and the best way to manage changes is by means of a **maintenance strategy document**. This would seem obvious to most clear-thinking professionals as it provides a medium for discussion and dissemination. However, as I have stated in earlier chapters, I have probably come across more organisations embarking on the procurement of a maintenance or asset management system **without** having first defined and agreed a maintenance strategy than those who have one in place. A maintenance strategy document that has been well thought-out and has controls in place to enable it to change in order to adopt better engineering and operational practices and to move in accordance with changing business needs is **essential** for the commencement of a successful information system procurement exercise.

Ownership of Data and Information

Comparison was made in Chapter 2 between data and the physical assets upon which maintenance is carried out. Readers were encouraged to consider data as another asset and to operate on data in a similar manner to their more familiar assets, considering what foundations should be prepared for the data and how it could be maintained in an optimal manner. In Chapter 11, we took this concept further and considered the ownership of data and information in a similar way to the ownership of physical assets. Our consideration of the differences between departmental maintenance and asset management operations in Chapter 2 brought out the need for asset managers to be the owners or custodians of the **resource** provided by their assets.

If we carry this resource ownership concept over from physical assets to data and information, then it becomes obvious that one

For people to act on the words in vision and strategy statements, those statements must be expressed as an integrated set of objectives and measures, agreed upon by all senior executives, that describe the long-term drivers of success.

Robert S. Kaplan & David P. Norton
The Balanced Scorecard – Measures that Drive Performance

'Harvard Business Review on Measuring Corporate Performance'

of the essential requirements that must be established before the procurement of a new information system is the ownership of the resultant data and information. This is important whether the previous system was a manual operation or another software system. It is also important that this exercise be carried out on an equal basis between the maintenance or asset management department, other departments who provide or use common data or information, and the I.T. department. This latter participant should be fully involved and should have considerable expertise and experience to contribute. However, it is most important that data and information ownership decisions are ultimately made between using departments. These are not software decisions; they would still have to be made in a manual system. However, the rigor with which they should be applied becomes much more important in a software-based system so, whereas it would be possible to neglect them for a manual system, this is not recommended for a software-based system. Apart from the obvious reasons that a software-based system is more costly and results in a more visible system, the fact that ownership rules are not established will inevitably lead to conflict between data and information, lack of faith in the results and eventual deterioration of the system.

The task of defining ownership of data and information is not small. When you consider that this task must also include the definition of ownership of all the variants of data, then this task becomes considerable. So it very seldom gets done or, if it does, it is often done in a piecemeal manner.

If, however, the task is to be completed satisfactorily, then the **ongoing** ownership of the data and information must be taken into consideration. It is often impossible to predict at the start of a project exactly how data and information will be used and thus how it should be owned. As we have seen before, change is inevitable and, as we shall see in the next section, change will result also from the analysis of the data and information. Thus, as we considered earlier when we discussed changes, we must provide here for a **mechanism for change** so that changes in ownership can be properly catered for. This, of course, adds to

"Who is responsible for how measures are taken? Who actually generates the data? Who receives and analyses the data? Who is responsible for changing the rules?"

Robert G. Eccles The Performance Measurement Manifesto, 'Harvard Business Review on Measuring Corporate Performance'

the task of examining ownership before commencement of a project, but it does not take much imagination to consider the consequences of neglecting this ongoing requirement.
In discussing this subject with clients, I have often used the analogy of a passport. A **data passport** or an **information passport** would identify where that data or information was derived from, who owns the data or information, and who uses the data or information. Clearly, this would be an enormous structure to set up and maintain, but the concept is appropriate for the ongoing control of data and information. How it is done and how deeply it would be implemented is the responsibility of each user organisation. Not to incorporate any mechanism would be a mistake; consider the consequences of having no passport!

Recursive Data and Information

I have touched on this subject in earlier chapters, but it is most important to emphasise it here in relation to the procurement exercise. We discussed earlier in this chapter the need to consider the project as more than just the procurement and implementation of the software. We saw that it must include the installation, collection and use of the data and information. Now we must consider the recursive nature of that information.

We saw that the reason for collecting the data and information was for that data and information to be used towards the improvement in the effectiveness and efficiency of the maintenance or asset management department and towards its ability to deliver the resources of its assets to the business. However, we saw in Chapter 2 that data and information were part of a larger structure, the **Data to Wisdom Ladder**. This ladder incorporates a feedback loop that dictates that the data to be collected should be re-examined as a result of the wisdom obtained from the original data. So the task of defining and implementing the information system goes far beyond the collection and use of the data and information. It involves the examination of **the need for further data and information** based on the benefits, lack of benefits, or perceived benefits of further changes.

"A new and radically different view of the meaning and purpose of information is as a measurement on which to base future action rather than as a postmortem and a record of what has already happened."

Peter F. Drucker 'The Information Executives Truly Need

'Harvard Business Review on Measuring Corporate Performance'

As we saw in Chapter 10, the system must be able to cater for this evolution in its use. We are all (hopefully) more knowledgeable after using a system for some time and undoubtedly had we this knowledge when we procured the system we would have done things differently. The purchase and implementation of these systems is very expensive not only in financial terms but in terms of the time spent on them and the possible effect on careers. It is thus essential that consideration be given to the recursive nature of data and information and to take steps to cater for different possible scenarios in the use of the system.

Here again, it is tempting to bypass this task due to the fact that it is difficult to predict the future. But, as we saw in earlier sections of this chapter, it is possible to consider the options required of the system to cater for user-related changes. It is thus possible to minimise the effect and cost of altering the system as a result of these changes. In Chapter 10, we also considered the role of vendors in this activity. For most user organisations, they will in a chain of users supplied by a particular vendor, with many users operating in a more advanced mode and others in a less advanced mode. Thus each vendor **should** have the capability to advise potential users on how the system should be able to cope with more advanced data and information requirements identified as a result of analysing 'first phase' data and information. It is insufficient for them to pass this requirement off with statements such as 'we have additional functions that you can use as you progress' or 'we are always developing new functions'. It is important to first define what functions and facilities you think your organisation would need as a result of improved data and information, then check this out with potential vendors, then check them out again with the more advanced users of each vendor.

One final point should be made with regard to the recursive nature of data and information. It is very common for this 'second level, long-term benefit' of the system to be disregarded by the ultimate users (the maintenance or asset manager and his department), by their executives and even by their Board. If

not prompted to take action on it, the vendor's staff may also disregard it. And they all do so for fundamentally the same two reasons. They believe that they will have removed a difficult and complicating factor that would affect their short-term goals and they also believe that any problems that may occur would appear long after their involvement in the activity, or after the decision to replace the system. What is missing in all these cases is a basic concern for the viability and effectiveness of the data and information generated by the system. It is the lives of these two entities that are important – the information system, albeit important, is subordinate to these. If this fact is borne in mind throughout the project and all possible actions taken to support the through-life effectiveness of data and information, then the project will be heading for success.

Constraints

Before the commencement of the procurement exercise, it is important to identify any constraints that are likely to be placed on the system by rules, regulations and procedures within your organisation. These may be defined for the type and configuration of hardware, the software packages used, or the method of access to the system. They may be unalterable, but that doesn't mean that they should be blindly accepted. I once had a client who was constrained by the I.T. department to use workstations that were specially configured for the organisation. They were, however, tailored to suit the requirements of an office environment and application, with front-end software that operated in a particular way and which could not be altered. For the majority of the use of the newly installed asset management information system, this was not a problem, but this constraint severely limited the asset management department in its future use of the system. For example, they were prevented from integrating any hand-held or remote data capture devices with the system as no facility had been made for such requirements in the definition of the original 'office' specification.

This is a very easy trap to fall into, especially if the 'computing side' of the implementation of a system is left entirely in the

hands of the I.T. department. The situation was made worse for my client by the urgency to complete the project, which was behind schedule by the time that I was called in to perform a Sanity Check (see Chapter 15). By that time, completion of the project (which in their minds was the installation of the software and the loading of the data) was sacrosanct and no changes to the hardware or software were permitted. Thus the original acceptance by the asset management department that the workstations would do the job without them demanding the ability to eventually acquire data from external devices committed them to severe limitation in the use of their expensive new system throughout its installed life.

There was a culture of 'let's get this in and working first and then we'll see about any changes later'. Politics then came into play, with the asset management department being blamed for not requesting external access functionality at the start, while at the same time requiring all departments to comply with the company standard for workstations. Thus the system will never be able to provide all of the facilities that should have been possible and will always be a less than perfect solution to the asset management information requirements.

This example provides a number of lessons. If your knowledge of I.T. is lacking, never assume that advice from the I.T. department, however well intentioned, is correct for your application. It is just as difficult for I.T. personnel to understand maintenance or asset management as it is for you to understand I.T. Assumptions are made on both parts; assumptions cause problems. It is most important that you identify what **could** be done with your new system and then decide for yourself what is possible within the anticipated life of the system. This, of course, should be done within the context of data and information lifecycles as described earlier. Then it is important to describe this **in your own terms** to the I.T. department in the form of a requirements document. You may have to provide some education on maintenance and asset management, but this can only be beneficial. Also, you will be working from your requirements, rather than vice versa.

"It's amazing how much can be accomplished if no one cares who gets the credit."

Dr. W. Edwards Deming

Once you are in communications mode with your I.T. department and they have become aware of the basics of maintenance and asset management, it is advisable to discuss the current I.T. strategy with them and get them to educate you in the areas of I.T. that affect your department. Also, once this is done, the perceived future I.T. strategy and how it affects you can be discussed with a better understanding of each other's needs and constraints. This should break down barriers and considerably help towards a successful implementation of the system. Hopefully it will also reduce some of the politics that is endemic in so many organisations. Removal of internal competitiveness can be a major step towards World Class performance.

The Specification

This is another topic that is large and requires a textbook on its own to do it justice. Also, the subject is as wide as industry and as deep as maintenance or asset management. What is fundamentally important for one organisation is likely to be anathema for others. And the activity will very much depend on where each organisation is on the path between the innocence of a first user of information systems to the excellence of a World Class organisation. Furthermore, organisations invariably have pre-defined structures for specifications in order to fit in with their own particular methods of working. So with these wide and diverse requirements, I shall concentrate on providing a few pointers that are important for maintenance and asset management. In-house teams and/or external consultants can use these in the generation of specifications to suit specific needs.

First of all it is important to establish who is responsible for writing the specification. Will it be written from a maintenance or asset management viewpoint or from an I.T. viewpoint? Or will it be written jointly? Whoever is responsible, have the assumptions inherent in the decision on which way to go been recognised and any perceived disadvantages properly identified and addressed? Are all parties aware of the knowledge that they have gained regarding each other's activity? Is there any animosity between the parties? If so, then it is certain that this will increase rather than decrease throughout the project. Are all parties aware of the long-term nature of the project and prepared to be measured on the delivery of the long-term goals?

How much experience do the specification writers have of the current status of maintenance and asset management information systems? Do they know where their organisation fits in to the spread of systems available, taking into consideration the present knowledge and experience of the workforce of their existing system and their ability to learn and incorporate new techniques? Are there new concepts, methodologies and techniques that should be examined before writing the specification (specifications are often written without awareness of these and their appearance in vendor's proposals invariably confuses the selection process)? If these new concepts, methodologies and techniques have been incorporated into the specification, has this been done in an unbiased manner? (Here again, this can confuse the selection process, as causing an attractive technique from a single supplier to be mandatory in a specification can hide other areas where the vendor's system is less capable).

The specification should be produced in a manner that encompasses all current and future requirements for data and information in a structure that models the operation of the organisation and with functions and functional groupings that comply with the way in which the operation is constrained to work. All of these should be capable of adaptation **by the user**. All requirements should be categorised as *mandatory*, *preferred* or *nice to have*. Also, scenarios should be defined to show how processes and functions must work and interact, also categorised as *mandatory*, *preferred* and *nice to have.* Has an assessment been made as to whether these requirements, processes and functions can reasonably be expected to be met? What is the strategy if no vendor can meet all the mandatory requirements? How does this change the project, its timescales and the ultimate delivery and support of the target data and information?

Who has the authority to sign off the specification? How much does he or she know of the ultimate requirements of maintenance or asset management?

These points will undoubtedly spark off other questions that are pertinent to each individual user organisation. There are degrees of points that will vary in importance from user to user. Therefore there

"How often misused words generate misleading thoughts."

Herbert Spencer 1890 – 1903
'Principles of Ethics'

will be important considerations for particular users that have been missed from this section. This is unavoidable, as the only way to provide a comprehensive list for a particular user organisation is to first understand their specific situation and requirements. However, the reader will have this knowledge about his or her own organisation and should therefore be in a good position to add specific points. The key starting point for this exercise is a questioning attitude and an unwillingness to accept a vendor's or a consultant's viewpoint just because it is trendy or exciting.

Always ask for things in the specification that satisfy your organisation's requirements and make sure that these statements are unambiguous. It is a common ploy of vendors to respond to ambiguous statements in a specification with further ambiguous statements in the hope that the user organisation will accept their validity. I have seen many organisations argue later with their vendor about the meaning of such statements. This usually ends up with a stalemate and distrust between the two parties. However, the vendor will have won the order and will probably be covered legally by the interpretation of his statement!

The Market

The market will have to be investigated. It will be necessary first to find out what systems are available. This can be done via the Internet, but it is then considerably more difficult to categorise them in a manner that will eventually be able to provide a short-list. National maintenance and asset management organisations and institutes should be able to help with this. Consultants should also be able to help, but bear in mind that there are now very many systems on the international and national markets. Also, their abilities are all changing so rapidly that it is impossible for any consultant to be totally aware of the current status of all the systems in any single national market. It is common for consultants to offer a short-list based on the systems that they have worked on or have had knowledge of during their careers. Those offered often tend to be the larger, more expensive systems, usually because the consultants can then claim to have offered the best available. However, these systems may not in fact be the best or the most cost-effective for particular

organisations. It is therefore important to ascertain the basis for a consultant's short-list. Ask which systems he has worked with or seen demonstrated during the last few months. Ask what systems made up the short-list in the last six procurement exercises that he carried out. Ask his clients if necessary. Then, if you are happy with the expertise, the experience and the objectivity of your consultant, use the information contained in this book and the requirements defined in your specification to generate your short-list and eventually select your vendor.

Here are a few points to bear in mind when going through the selection process.

- A scoring procedure should be developed in order to facilitate short-listing. You may well have an in-house procedure for this, or your consultant may have a standard system. Whatever system is used, they all have a basic flaw, i.e., users tend to concentrate on the numbers, assuming that the scoring system, once derived, is correct. This is not the case. A scoring system will help you sort out the short list, but should not be used for the final selection. **No system** will ever get full marks, so it is important to consider in which areas each of the highest scoring systems failed, as well as to consider what other attributes they may have to compensate for these. It is then most important that the impact of these failure areas on your strategy and on your plan for the through-life use of data and information is assessed and appropriate action or reconsideration taken. Most organisations accept these failure points as being unavoidable – and so they are – but they then do nothing about it and thus accept a system that is less than they expected. If necessary, the short-list should be opened up again, recognising the limitations of the scoring system. This is a bold step, but better in the long run than blind acceptance. As a back stop, if all possible systems fail in important areas, then it is **most important** that this is communicated back to the users and the management **so that expectations are managed**.

- The structure, expertise and experience of the team tasked with the selection of a system should be carefully examined, as should any bias, separate agendas or conflicts among team members. The team should be led by a maintenance or asset management

expert with I.T. personnel in a supporting role, rather that vice versa (see later section on The Implementation Team).

- Functions, processes and datasets should be produced and agreed internally to represent the requirements of the system. These should be provided for all demonstrations and a scoring method devised to assess their performance.

- When it comes down to a short-list, the experience of existing users of the selected systems is of major importance. While visits to user sites can provide considerable benefit, this can be a very time-consuming exercise for both the existing user and the prospective purchaser. Much of this time can be saved by telephone conversations, fax and email communication. These methods of communication are often better than face to face discussions at on-site demonstrations as the existing user usually talks more freely about the system and the vendor.

- Never take the vendor's recommendation for a site visit. Ask him for his user list, with the following additional information against each name:-
    - Industry
    - Size of system
    - Length of use of system
    - Details of interfaces with associated systems
    - Contact name/ maintenance or asset management
    - Telephone number, fax and email address for above
    - Contact name/ I.T.
    - Telephone number, fax and email address for above

  You can then choose who you contact, how many you contact and who you wish to visit. When you visit, do not include the vendor in the visit and do not inform him of your visit. I would suggest that you also ask the user to keep quiet about the visit, as it is important that the vendor should not put on any special show for his user or for you because of the visit. Many vendors have unhappy users; it is important that you find any before you make a purchase and perhaps join them!

- What is the mechanism for making a decision? Does everyone on the team get an equal vote? Remember that any scoring system for demonstrations is imperfect. It will help you in the decision process, but will not make the decision for you. How do you assess intangibles such as the clustering and ease of use of functions to form a process? Have any of your end users tried to use the system in earnest? If they have and as a result had reservations, how were these handled? Were they brushed aside as being less important than other aspects of the system? If so, then this indifference is likely to come back to haunt you; it is the end users that will ultimately determine whether or not the system is successful. If the decision process results in less than unanimous agreement, how is this resolved? It is all very well to state that a majority vote will decide or that the team leader or manager will make the final decision. This may be very democratic or respectful of the leader or manager, but the reasons for making the decision and the impact of not accepting the views of the dissenters must be aired. Again, if necessary, the expectations of the end users and management should be reviewed.

The Implementation Team

The composition of the team will be individual to each user organisation. Each will have a particular way that it handles this type of project. As the maintenance or asset management department will rarely participate in such exercises, it is likely that the composition of the team and its method of implementation will be controlled by the I.T. department. It is essential that the maintenance or asset management department doesn't accept this blindly as being the way things are done for such projects. Remember that this will be **your** system and that you will have to live with the system – and its consequences – for some considerable time.

The project should **not** be an I.T. project. Always remember that information technology is a tool; it is the enabler of a facility which is aimed at benefiting the end user. An I.T. department does not know – and can never know – maintenance and asset management as well as your department. Therefore, although they will undoubtedly be a very important part of the project, they should never lead the project. In the same way, they should never be tasked with the selection of a system

"Information specialists are tool makers.

They can tell us what tool to use to hammer nails into a chair.

We need to decide whether we should be upholstering a chair at all"

Robert F. Drucker

The Coming of the New Organisation 'Harvard business Review on Knowledge Management'

or permitted to impose an 'integrated' maintenance package that is part of a large suite of systems. It is the *how* and *why* of maintenance and asset management that is important, not just a knowledge of the functions.

There will be discussions and arguments throughout the implementation project as to how best to carry out tasks and meet each deadline. It is most important that the composition and seniority of the team reflects the overall objective of providing a mechanism for the ongoing support of maintenance or asset management data and information thoughout the planned life of these assets. If this is likely to be jeopardised by the composition and structure of the team, then **it is the wrong team**. It is far better to resolve this problem before the project commences than during the project, when transfer of knowledge and responsibility is much more difficult.

One final point is important regarding the implementation team. As we saw in earlier chapters, most projects of this type concentrate on the purchase and implementation of the software.

If the team is not committed and responsible up until the objectives of the project as defined in the original justification document have been achieved, then there is little hope of ever achieving these objectives.

It is not, of course, essential for all members of the team to be fully occupied on the project until its completion. They must, however, be available and **accountable** up until the time of its completion.

Compromises

We have identified in previous sections of this chapter the need to revisit the overall objectives of the project when it is determined that the requirements in any part of the project are not going to be met. Examples of this were seen to be a less than perfect scoring by the selected system in the rating structure and less than unanimous agreement of the team in the selection decision.

Three further areas of compromise need to be considered. The first of these occurs when the vendor supplies all the functionality that you

require, but the manner in which it is provided does not quite fit into your required method of operation. We touched on this subject when we considered groups of functions and processes, but it is important to re-emphasise the problem here as the differences between what is provided and what you really need (and will be satisfactory for your workforce) may be quite subtle. Therefore, it is important that in any discussions with the vendor no quick decisions are made regarding these compromises. Anything that deviates in **any** way from what you have defined in your specification should be flagged up for further discussion with all relevant members of your team or operation. In a business discussion with a vendor, it is impossible to consider all the possible consequences of such compromises. Any reputable vendor will understand your wish to take time to consider their effect.

The second area of compromise arises when it is agreed that a necessary function or process is absent or does not work in the way in which you require and the solution offered is customisation to suit your requirements. The vendor may or may not agree to incorporate this customisation into his standard product and support it in all new releases. As we saw in Chapter 13, incorporation of new functionality into a vendor's product should **never** be subject to a snap decision. It certainly should never be done in order to win a sale, but this does not stop the majority of vendors from offering it. Look at it from the vendor's viewpoint. His development department **should** be working to a plan and to timescales that will generate the next release of the product. This department is now required to provide a time and budget for the new functionality that will comply with the requirements of the new customer. This takes time to do properly, but in a sales situation sufficient time is rarely provided. The design and estimates must therefore be questionable. Also, this design **should** recognise and incorporate the effect of the new function on the rest of the system and **this is the part that is usually missed if time is short**. Furthermore, if the customisation is accepted, this will have to take precedence over other development work in order to comply with the requirements of the customer. Thus all these factors conspire to degrade the system if the required functionality was not in the development plan for the product. **It is not good for the vendor and therefore it is not good for the customer.** Thus **any customisation**, no matter what promises are made for its incorporation into the product, is a **compromise**. And whenever a

compromise is identified, it is necessary to revisit the requirements, in this case particularly those concerned with the integrity of the vendor and his product.

The third area of compromise concerns the legal contract with the vendor. All negotiations end up with compromise on both sides. With all projects, the completion of the legal aspects of the project is usually the last activity before signing the agreement. How many times have you heard it said about a contract that 'everything has been agreed and only the legal points have to be completed'? Once these are finished, the contract is signed and implementation can begin. How many times have you heard of the contract being revisited by the user department after the legal departments have considered it? Shouldn't the user department have the opportunity to examine the impact of any compromises made during the legal negotiations on the ultimate success of the project? In my experience, this is seldom done and represents a considerable loophole in the entire process.

Implementation

This section is not a treatise on how to run a software project, for two reasons. The first is that there are already many very good documents that cover this important and difficult subject. There are many potential pitfalls in running such a project, so the reader is encouraged to fully prepare himself or herself for the task. The available text is lengthy and comprehensive, and it applies to all software projects, no matter what field the application covers. It is therefore inappropriate to adequately cover this subject within this text.

The second reason is that we are not **only** talking about a software project when we consider the implementation of a maintenance or asset management system. As I have said so many times before, it is important to look beyond the obviously high profile and possibly more exciting activity of implementing the software to the factors that contribute to the ultimate delivery of what you got the funding for. Thus we shall be considering here the factors that are usually omitted from textbooks on software projects and concentrate on important points for the delivery of a maintenance or an asset management solution.

What are the guidelines for the project? If they are all structured around and aimed at the completion of the implementation of the software, then these guidelines are **wrong** and this fact should be relayed to the appropriate management, or the Board if necessary. It doesn't matter whether these same managers or directors have subscribed to the guidelines. That may be a reason for them being reluctant to change the system, but it certainly does not make them right! If necessary, give them a copy of this book with this and the next paragraph highlighted! This mistaken assumption is one of the major causes of the failure of many maintenance or asset management systems to deliver real benefits and justify their existence. For the benefit of those managers who cannot be persuaded to read the rest of this book, I will repeat the advice.

The implementation of a maintenance or asset management information system cannot be considered to be complete or to be successful until all the factors upon which it was originally justified and funded have been delivered. It CANNOT be considered complete upon the successful implementation of the software and the loading of its data.

Now this fact has several implications for the overall project. First of all, as we saw in the 'Team' section, it is absolutely necessary for the project team to continue to be available, although not necessarily involved full-time, throughout the entire project. This may take several years, and it is obvious that many of the team will have left the organisation or will have been re-assigned elsewhere. Thus it is difficult to ensure continuity of availability of staff – or their continued knowledge or interest! Nevertheless, it is essential that a mechanism for support is established **at the start of the project**, with the necessary controls to ensure that the original objectives of the project will be met. Here again, because these are difficult to put in place and there would naturally be a certain amount of disinterest on the part of staff who are now involved in other activities, most organisations have no such mechanisms and thus most projects fail to deliver what was originally expected of them.

So far we have concentrated on the need to retain the knowledge and interest of the user's team. What about the remainder of the implementation team – the vendor's team? There have been many

"New knowledge depends on tapping the tacit and often highly subjective insights, intuitions, and hunches of individual employees and making those insights available for testing and use by the company as a whole."

Ikujiro Nonaka

The Knowledge-Creating Company 'Harvard Business Review on Knowledge Management'

instances of discontinuity in a vendor's team throughout the implementation of their own software, with key resources being taken off to start other new projects. This can be very disruptive to the implementation of the software and to the overall success of the project. It is, of course, possible to see why this is done and to appreciate that the vendor also has a resource problem. However, looked at purely from the viewpoint of the ultimate success of **your** project, it is a potentially serious problem that must be addressed and controlled.

Unfortunately, it is not only the lack of continuity of vendor's staff during the implementation of the software that is the problem. Once the vendor has fully delivered the software, his team is likely to be totally reassigned, although you **should** still be only part way through the implementation of your overall project. Bearing in mind that you will wish to prove that you have delivered all the facilities and information upon which the project was justified, you will most certainly require assistance from the vendor at this time. As this could be some considerable time after the implementation of the software, how will you or your vendor ensure that his current staff have appropriate knowledge of your overall objectives and how the system as implemented can meet these objectives? It is not sufficient for this requirement to be handled purely by a help desk facility that caters for standard software.

Maintenance and asset management information systems are becoming very complex and, as we saw in earlier chapters, the ways in which they can be used are now extremely diverse. Because of this, it is necessary to retain knowledge of a user's complex functionality and interrelationships, and how the software was configured to meet these requirements. This can, of course, be written down in a report that is passed from one consultant to his successor in a vendor organisation. But this usually results in the loss of tacit knowledge, a problem for the user organisation that is usually irreversible and can seriously affect the ultimate success of his project.

How such problems can be averted is difficult to imagine, but their existence is important to recognise and should be considered when doing a risk analysis of the project (see Chapter 16).

"The actions and policies set by management can put an upper limit on quality."

Dr. W. Edwards Deming

If it is at all possible, a team building culture should be introduced which includes both the user's staff and those of the vendor. I have seen this work very well in some organisations, with all staff members becoming a well-knit team, resulting in not only a successful project but also a happy working atmosphere.

We must now consider how the quality of data is affected by the manner of the implementation. In Chapter 3, we saw that the quality of data is likely to deteriorate if no mechanism is put in place to monitor it and take remedial action. The time to consider data quality is at the specification stage, where plans should be put in force that will ensure that the quality of data is ensured throughout its life. Remember that data quality is fundamentally linked to the life cycle of each data element. Therefore, if there is no policy for determining the life cycle of each data element, then there will probably be no retention of quality in the data.

While it is most important to define the levels of quality required of the data, together with the mechanisms necessary to ensure that quality, at the specification stage, it is in the **implementation** stage of the project that action will be necessary to start the data quality process. The implementation stage will determine whether or not quality of data is possible or is forever a lost objective. This is because actions taken – or, more often, not taken at the implementation stage will irreversibly alter the quality possibilities thereafter. The implementation stage is the point at which teams of people – staff from both the user organisation and the vendor – become deeply involved and ultimately responsible for the project. Unless they are **all** highly motivated to deliver a quality system that **inherently** ensures continued data quality, then the ultimate success of the project must be questioned. People make the difference – positively or negatively!

This caution on quality would seem on the face of it to be obvious and thus not worth stating. Yet time and again I have come across organisations that have large and expensive projects in place without any consideration of the need to define how the quality of data will be assured. One large organisation that was on the point of completing the implementation of their

"The quality control issue has more to do with people and motivation and less to do with capital and equipment than one would think. It involves a culture change."

Michael Beer
U.S. Academic

new system told me that they still intended to address data quality, but had not yet found time to get round to it! Without this fundamental requirement in place, one must question the validity of the entire project!

Finally, it is important to highlight an aspect of data quality that is often overlooked. That is the transfer of data from an existing system to a new system. Despite the fact that the reason for procuring a new system is often dissatisfaction with the previous system, data is invariably transferred en mass from the old to the new system. This usually perpetuates the previous problems. When a new system is implemented, it is most essential that some form of data audit is carried out, taking all data streams back to their original source. **All** successful implementations include such an audit. It is a sure sign of trouble for **all** of the data from an existing system to be transferred to the new system without any form of audit.

It would be nice to expect that everything will work first time and that when you transfer your maintenance or asset management staff over from the old system to the new one, there will be no glitches and everyone will be happy. Even if such a scenario were possible, it would still be prudent to cater for whatever problems **could** arise. So what can you do? Let us assume that you have the experience, or access to the experience, of implementing such systems. You would then put all the necessary measures in place to control the project deliverables, to control the staff, and to deliver on time and to budget. You would also perform a risk analysis on the possibility that something could go wrong with the system (see Chapter 16) and put in place actions that would recover the integrity and credibility of the system.

One of these actions would be the examination of everything that could go wrong with the transfer from the old to the new system. The decision on whether or not the technology needs to be changed and the need to maintain current work during the changeover will both affect this transfer. The wise way to handle this is by operating some form of parallel running of the old and new systems. This in itself can cause additional problems, but the effort is usually well worth while considering the risks associated with the transfer. A common way to

handle parallel running is to start off with the new system for a single department, operation or site, then progressively extend the new system to other areas. It is important, however, to handle the progression and use of data by the new system and to be able to test this in relation to that of the old system and the expectations of the new one. Once again, the difficulties associated with this task often result in the operation being neglected or poorly implemented. However, if it is carried out properly, then it more than justifies the effort.

Training of the users is another area that is often left to later on in the project. It should, however, be considered at the start of the project as it is an area that if done badly will almost certainly cause the project to fail. We have considered the motivation of staff many times in earlier chapters and how the lack of attention to their needs or to their questions results in less than successful results. Put more strongly, poorly trained or motivated staff can kill a project, either in an obvious or in an overt manner.

They must be trained in a manner that encourages them to use the system in order to perform their work better – better for the company and, of equal importance, better for them. It is not sufficient to teach them how to use the system. They must be helped to understand the relevance of the system, the data that they are being asked to enter into it, and the information that it provides to their operation. This is not an easy task and it will take some considerable time – and patience! Scheduling standard training courses prepared by the vendor cannot do it. It must involve personnel from the user organisation who know the existing methods and how these are likely to change as a result of implementing the new system.

A common way of addressing this problem is to use in-house personnel who are knowledgeable about the existing process, who have been involved in the development of the requirements of the new system and who are well respected by the workforce. They should be trained on the new system and then used to translate this new expertise into language and terminology appropriate to their colleagues. As they are unlikely to be skilled trainers, they may well require help from the vendor organisation. Once they become more proficient, they should be able to handle any problem by themselves;

this can be of particular advantage later on when the system is in use and staff wish to have refresher courses.

A good way of ensuring that staff can learn the system in their own time and perform their own refresher training is to provide a 'play pit' system. This is a copy of the system with correct data, but which is not 'live', i.e., it is not being used by the maintenance operation. It works in the same way as the real system, but, just as with a child's play pit, no-one can do any damage to the system or to their own reputations! The system should be made available either in a permanent training area or, preferably, accessible from their own work stations. It is amazing how much can be learned when there is no chance that your mistakes can be seen by others! It is also a good way for managers to become familiar with the new system without any embarrassment, although there is considerable merit in having the workforce see their managers going through the same training routines as themselves!

As with all other areas of the implementation, it is essential that the effect of training is assessed throughout the project, and any corrective measures taken as soon as possible. Here again, a risk analysis is appropriate.

A final point that must be made regarding the implementation of the project is to ensure that the expectations of the users are managed. There is absolutely nothing to be gained by keeping the workforce in the dark. This leads to resentment and apathy. However, by taking the time to determine what the users expect from the system and by comparing these expectations with the objectives of the project and its deliverables, this resentment and apathy can be reduced or eliminated. It is important to realise that this activity must not be considered as a one-off process; it must continue throughout the implementation of the project. Things change during a project; the workforce have a right to know what these changes are and how they and their work will be affected by these changes.

Testing and Acceptance

Again we have a large subject on which standard text has been written. Here, however, we are considering not just the testing and

acceptance of the delivered software but the testing and acceptance of the delivered project. This requires comparison of what is actually delivered against what was originally justified and funded. This is a far from easy task because of the considerable time from justification to delivery, the possible changes in staff and responsibilities throughout that time and the various agendas of all those concerned.

While accepting the difficulty in comparison, and recognising that the mix of problems will be different for each organisation, we must nevertheless come up with the stark question 'How do we know for sure that the project has been a success?' Also, if it could be considered a partial success, in what way could it have been improved and is there still scope for this improvement? Unless we test the system against its original justification parameters, we can never be in a position to truly judge its success. Therefore, we can never determine whether or not the project was worth the effort and the expense.

So how will you test your system? What are the key objectives? What are the secondary objectives and what are the minor objectives? How can you judge these? Unfortunately there are no standard answers to these questions as the situation will be different for every organisation. There are many reasons for this, but the most obvious are the differences in culture between organisations, their different starting positions in the implementation of systems, and their different evolutionary capability. However, what is fundamentally important for **all** organisations is the need to consider how to test for successful delivery of the project **at the time that the project is first justified**. In fact, the mechanism for proving the project's success should be part of the original justification exercise.

Key Points in Chapter 14

- 'Through-life' requires consideration of the usefulness and delivered value of the system over its total installed life.
- The life of the system should be a **planned** period, rather than happen as a result of lack of interest or the desire to be modern.
- Unplanned system life points to poor asset management.
- It is essential to determine what the user organisation means by effectiveness and how the maintenance or asset management departments and other departments within the organisation contribute to the lives of the system and its information.
- Failure to determine these relationships will result in a system that doesn't meet the real requirements.
- Organisations that fail to manage their data and information will almost certainly fail to manage their information systems.
- The system should be able to cope with changes, even if these changes are unpredictable. Changes, whatever their causes, can affect the system in a number of predictable ways. These requirements should be included in the requirement specification.
- Self-assessment of the user organisation is essential. Do not procure a system that satisfies the ego but that is wrong for the type of organisation yours really is.
- Data and information ownership decisions must ultimately be made between using departments and not by the I.T. department.
- It is important to determine who is responsible for how measures are taken, who actually generates the data, who receives and analyses the data, and who is responsible for changing the rules.
- The ongoing ownership of the data and information must be defined and a mechanism produced for its control.
- Consider the control of the ownership of data and information in the same way as controlling a passport.
- The data to be collected should be re-examined as a result of the wisdom obtained from the original data.
- Vendors should be able to advise potential users on how the system should be able to cope with more advanced data and information requirements identified as a result of analysing 'first phase' data and information.
- The implementation of a system may be severely constrained by the rules, regulations and procedures of its user organisation.
- Never assume that the I. T. department knows as much about maintenance or asset management as you do. Always question any assumptions made by them or by you.
- Any animosity between the parties tasked with generating the specification will increase rather than decrease throughout the project.

Key Points in Chapter 14 (continued)

- Specification writers must be aware of the long-term nature of the project and be prepared to be measured by the delivery of long-term goals.
- Specification writers must be aware of the current status of maintenance and asset management systems. They do themselves and their company an injustice if the specification is written from a naïve viewpoint.
- Incorporation of new techniques into the specification must be performed in an unbiased manner.
- The specification should enable evolution in use of the system to accommodate various scenarios.
- Requirements in the specification should be categorised as *mandatory, preferred* and *nice to have*. These should cover functions, processes and their interaction.
- It is important that the person tasked with signing off the specification should be fully knowledgeable about the ultimate requirements of maintenance or asset management.
- It is important to always have a questioning attitude and not necessarily to accept the viewpoint of vendors and consultants.
- Be sure to list your user requirements in an unambiguous manner.
- Vendors are well skilled in responding to an ambiguous specification statement with an equally ambiguous answer.
- It is impossible for any consultant to be totally aware of the current status of all the systems in any single national market. Ask any consultant why he offers a particular short -list. Also ask him for the short lists of his previous six consultancy contracts, and check them out with his clients.
- Scoring systems for short-listing cause users to concentrate on the numbers, assuming that the scoring system is foolproof. This is not the case.
- A scoring system will help you sort out the short-list, but should not be used for the final selection.
- It is important to discuss why the highest systems have not gained top marks and what impact that has on the ability of the selected system to **totally** meet your stated requirements.
- If necessary, the short-list should be opened up again, recognising the limitations of the scoring system.

Key Points in Chapter 14 (continued)

- If all possible systems fail in important areas, then it is important that this is communicated back to the users and the management so that expectations are managed.
- A maintenance or asset management expert should lead the selection team with I.T. personnel in a supporting role, rather than vice versa.
- Functions, processes and datasets should be produced and agreed internally to represent the requirements of the system, and should be used for all demonstrations.
- Significant time can be saved by contacting user sites by telephone, fax and email.
- Never take the vendor's recommendation for a site visit.
- It is important that the mechanism for decision is understood by all the selection team, that it is free from any bias, and that it is supportable in the future.
- Less than unanimous agreement poses its own problems. How valid were the disagreed items and what impact do they have on the ultimate success of the project?
- The project team should not be biased towards the I.T. side. The ultimate user is the maintenance or asset management department. Information technology is a tool; it is the enabler of a facility that is aimed at benefiting the end user.
- The *how* and *why* of maintenance and asset management is more important than just a knowledge of the functions.
- If the overall objective of providing a mechanism for the ongoing support of maintenance or asset management data and information throughout the planned life of these assets is likely to be jeopardised by the composition and structure of the implementation team, then **it is the wrong team**.
- Any offering from the vendor that deviates in any way from what you have defined in your specification should be flagged up for further discussion with all relevant members of your team or operation.
- Customisation can have serious negative implications on the vendor's development plans as well as on the use of the system by the user organisation.
- Any customisation is a compromise, thus requiring that a user revisit his requirements.
- The department that ultimately uses a new system should have the opportunity to examine the impact of any compromises made during the legal negotiations on the ultimate success of the project.

- It is essential to ensure that members of the implementation team (including staff from both the vendor and the user) are accessible throughout the entire project and not just for

Key Points in Chapter 14 (continued)

- the software implementation phase.
- Unless the implementation team, consisting of staff from both the user's and the vendor's organisations, are **all** highly motivated to deliver a quality system that **inherently** ensures continued data quality, the ultimate success of the project must be questioned. People make the difference – positively and negatively.
- When a new system is implemented, it is essential that some form of data audit be carried out, taking all data streams back to their original source.
- Parallels running of the old and new systems and the successive transfer from one to the other are essential operations, but ones that are not easy to implement. For this reason, they are often poorly implemented or omitted altogether.
- Poorly trained or motivated staff can kill a project, either in an obvious or in an overt manner.
- It is amazing how much can be learned when there is no chance that your mistakes can be seen by others.
- It is essential that the effect of training is assessed throughout the project, and any corrective measures taken as soon as possible.
- Continuously manage the expectations of the workforce throughout the project.
- Unless the system is tested against its original justification parameters, it will be impossible to truly judge its success.
- The mechanism for proving the project's success should be part of the original justification exercise.