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

Before you can solve a problem, you first have to realise that you have one!

Chapter Five

The Nature of the Problem

By this stage, readers should be in no doubt that the development, selection, implementation and use of maintenance and asset management systems are far from easy activities. The problem, however, seems to be that many of those who have **not** read the previous four chapters appear to have the impression that an information system for maintenance management is a relatively simple structure with fairly basic functionality designed to carry out a fairly low-level activity.

Obviously, such an attitude could be expected from those in organisations that carry out a **departmental maintenance** policy who are not directly involved with maintenance. It is also an attitude that would be expected from those outside industry with no knowledge of maintenance but with perhaps some indirect influence on the activity, e.g., lawyers, venture capitalists and financial auditors.

To perhaps a lesser extent, but no less relevant to the problem, the suppliers of those integrated software systems to whom maintenance is just another application area, fall into this category.

This chapter has particular relevance to these people. It lists a number of reasons why the maintenance and

This chapter will provide an insight to those who did not realise that there is a problem in providing appropriate information systems for maintenance and asset management.

asset management activities should not only be taken seriously, but should be provided with the necessary information handling environment to enable them to contribute effectively to the business objectives of the organisation.

Although the rest of this book is targeted at those who recognise that there is a problem and need help in solving the problem, this chapter will provide an insight to those who did not realise that there is a problem in providing appropriate information systems for maintenance and asset management.

For those who already understand that there is a problem, this chapter analyses the nature of the problem, identifying factors that complicate the solution and are not readily recognised by potential users. These factors have been the cause of many unsuccessful implementations of maintenance systems, although for obvious reasons the users do not often admit their failures. Some of these failure scenarios are described in Chapter 15.

Does the image match reality?

Before we address the complicating factors in detail, it is worth identifying a problem area that affects most prospective users of maintenance and asset management information systems. This is the need to make a decision about what type of organisation you are and what you aspire to be.

This may seem to be a relatively easy decision to make as most organisations publicly promote what type of organisation they are and most identify where they intend to be in the future. However, we all know of organisations – perhaps our own – whose public identity is considerably different from that seen by the staff. This is where the problem occurs.

"it would seem totally illogical for an organisation to procure an information system which is beyond the capability of the operating activity to use"

cf. "Strategic Asset Management", Issue # 2 (Jan 29 1999) "Never buy an Asset Information System that is Cleverer than You are!" Pt 2 of a Five part series entitled "What to know before you choose your AIS"

Available now on the key word searchable CD "Strategic Asset Management, 1999-2002"

Details:
info@amqi.com

If the public image is of a progressive organisation that uses the latest tools in order to maintain and improve its apparent high level of efficiency and the true situation is a maintenance operation that is under-resourced, de-skilled and de-motivated, then the installation of a new information system will be a waste of money. This would be true whether the implementation is for a maintenance or an asset management system.

Nevertheless, time and time again I have come across organisations that install advanced systems without coming to terms with what type of organisation they are. They believe their own publicity, and they are often encouraged in this belief by vendors who are happy to recommend more advanced systems for 'an obviously advanced organisation'.

Does operating capacity match system requirements?

If you examine this problem laterally, it would seem totally illogical for an organisation to procure an information system which is beyond the capability of the operating activity to use and for which there is no associated operational improvement plan. Such action does not take into consideration the human and sociological aspects of the decision mechanism.

Again, from experience, I can recall many major organisations that fall into this category. A typical approach is to claim that the organisation is world class and to buy one of the most expensive systems, believing that it must therefore be the

Ego gratification is one of the worst traps devised to ensnare the successful businessman.

Harold Geneen,Business consultant and chairman of ITT

best. An accompanying assumption is that the implementation of the system will, by itself, improve the maintenance operation!

Now this may seem to be a ludicrous situation for any organisation, whether large or small, to get itself into. As we shall see in Chapters 15 and 17, it is not only a common occurrence among organisations, it is often repeated by the same organisation each time it procures a maintenance system! A maintenance or asset management system should not be procured in order to polish an individual or corporate ego. It should be procured to aid the maintenance or asset management activity **after the current and future requirements of that activity have been analysed and any pre-implementation restructuring has been carried out**.

Complicating Factors

Many years ago, during the very early days of the application of computers to maintenance management, I was very concerned at the way that the software was being developed for this new market. It started with one or two supplier producing what would be considered now as very basic and embryonic systems. Initially, these systems had a minimal asset register and the ability to produce and return job cards. As with all early software systems, these packages, and others that later joined them on the market, progressed by adding functionality to the products (often as a direct result of developing customised specials for their early customers).

We saw in Chapter 4 how this leap-frogging of functionality eventually led to more flexible

systems in an attempt at satisfying different industries. The early systems suppliers, however, tended to select particular ways of performing functions that seemed the most popular and thus the more readily saleable.

Customisation of standard packages leads to further customisation as requirements change and resultant lock-in of the user to the supplier

My concern with this manner of development resulted from having previously carried out a wide and deep survey of the ways in which organisations performed their maintenance. I could not see how any one organisation could accommodate the then 'standard' packages without considerable compromise on their part, which would inevitably lead to deviation from their original objectives, a considerable requirement for retraining of the workforce, and a potential loss, rather than a gain, in efficiency. A further problem was the possible need for substantial customisation of the software to avoid operational changes, leading to the need for further customisation as requirements changed and the resultant lock-in of the user to the supplier.

To my mind, nobody from either the supplier or the user organisations had really analysed the requirements of the marketplace.

To my mind, nobody from either the supplier or the user organisations had really analysed the requirements of the marketplace. The suppliers were competing with each other with an incremental approach to functionality while at the same time the users had no effective mechanism for recognising the essential differences between their requirements. There was a general assumption that a function such as 'plan work' was common to all maintenance operations and it was only once organisations started to use these functions that they found out that the supplier's assumptions were not the same as their own assumptions!

Results from market survey

As a result of my survey, I was able to publish a list of the differences that were likely to occur between the maintenance requirements of different organisations. Some of these differences have been mentioned in earlier chapters, but all are included in the following list for completeness. Because of the widespread differences that I found as a result of the survey, I called this list **Areas of Divergence**.

AREAS OF DIVERGENCE

CODES
FUNCTIONS
HARDWARE
CULTURE
EVOLUTION
INTERCONNECTION

These **Areas of Divergence** were the basis of several papers that I gave on the subject of information systems during the early Eighties. In this way, I was able to obtain feedback on the perceived requirements and difficulties of developing, implementing and using information systems for maintenance.

Survey analysis endorsed by hundreds of user organisations

After a two or three-year period, I found that this analysis was fully endorsed by hundreds of user organisations, leading me to investigate the software tools necessary to accommodate such a range of requirements. The tools that I eventually defined were Fourth Generation Languages and Relational Database Management Systems (RDBMS), which quickly became the standard offering in the industry. Incidentally, I found this method of user feedback to be similarly valuable

when I subsequently introduced the concepts of **Asset Care** and **Asset Management** to the market.

Need for objective assessment of industry requirements

I firmly believe that it is essential for the health of an industry to regularly make an **objective assessment** of the requirements of the industry and have the results of that assessment endorsed by an in-depth survey of the market. This is extremely important for a rapidly growing activity such as the development, implementation and use of maintenance and asset management information systems.

I appreciate that this is not an easy task. It would, for example, be increasingly difficult to fund this in a way that would ensure total impartiality (I shall be covering the motives, objectives and agendas of vendors and users in Chapters 13 and 17). However, if it is not done, then the software industry, rather than the users, will drive the functionality and features of systems.

Without market analysis, the software industry, rather than the users, will drive the functionality and features of systems.

It is naïve to consider that this will be totally in tune with the requirements of users. It would be more cynical – but more realistic – to recognise that vendors compete against each other for market share and profitability by offering attractive packages of **what they think the users will buy**. This is not necessarily **what the users need**.

There have been milestones in the development and use of maintenance and asset management system, e.g., with the introduction of databases, Asset Care and Asset Management, where vendors have reacted positively to the results of

The evolutionary approach v. the revolutionary approach to system development

objective assessments of the true market requirements. However, between such milestones the tendency has been for vendors to supply what they think will give them an edge over their competition, rather than what the users really need. We can think of this approach by the vendors as an **evolutionary approach** and the objective assessments as a **revolutionary approach**.

So what's wrong with the evolutionary approach?

Nothing – from the vendor's viewpoint! It enables them to add functionality and features to their software (possibly as a result of customisation work performed for an existing user) that they believe will give them a lead over their competitors. It's also good for their users, provided that the new functionality and features are generic and compatible with the ways in which their organisations are constrained to work (we shall be examining this aspect in more detail later). However, it is a relatively slow way in which to push forward the boundaries of the application of information technology to maintenance and asset management.

As we have seen in previous chapters, **change** is now a major factor in these application areas. Change requires data and information to support the knowledge and wisdom necessary for its management. The repository for data and information is the procured information system. This has to be **dynamic**, it has to be **appropriate to the ongoing need**, and it has to make use of every resource that could enable it to perform effectively **for the user**.

Sell solutions, not just products.

Klaus M.Leisinger
Departmental Director
of Ciba-Geigy Ltd.
New York Times,
1988

Without this capability, as we have seen before, *a software system is essentially a mechanism for selling rather than a mechanism for using.*

The reaction of vendors to this will be to cite the many cases where they have a good, ongoing relationship with user organisations. I would not wish to detract from these worthwhile initiatives. However, the fact remains that *unless a revolutionary objective appraisal of the market requirements and differences is made on a regular basis, vendors tend to slip into an evolutionary strategy of WHAT THEY CAN SELL rather than WHAT THE USER NEEDS.*

Returning to our analysis of the differing requirements of user organisations, it is appropriate to re-assess our **Areas of Divergence** in relation to current and perceived future requirements. At the time of writing, the original list of divergent areas is almost twenty years old, so it would be reasonable to expect that things have changed during that time.

Changes in the last 20 years

Four major new factors in the last 20 years affect development and implementation

The four major new factors affecting the development, implementation and use of maintenance information systems are as follows: -

- the introduction of asset management
- the considerable increase in competitiveness within a world market of user organisations
- the phenomenal advances in hardware and software technology
- the introduction of methodologies for maintenance and asset management

'Methodologies' has grown in importance and relevance

Perhaps unsurprisingly, because of the widespread and in-depth work that produced the original list, these factors have made little change to the list.

The introduction of asset management and the increase in competitiveness are still covered by the term 'culture', which naturally becomes an increasingly important factor. Advances in technology change the nature of 'hardware'. The original factor was the tendency two decades ago for user organisations to be tied in to the products of a particular hardware manufacturer; thus the users could only consider maintenance software which was developed using a language and utilising an operating system which was supported by their hardware manufacturers. Such scenarios are now long past, but some would say that the problem is different, rather than easier. There are other forms of 'lock-in' and there is a much greater choice of technology, causing a real **divergence** of requirements. We shall therefore re-name this area 'Technology'.

The last factor – methodologies – has grown in importance and relevance since the original list of Areas of Divergence was published. Certain methodologies existed in the seventies, but were not widespread or sufficiently well known to be included in the list.

I shall not list the methodologies when this topic is considered. In line with the objective of keeping the contents of this book relevant irrespective of changing technology and trends, many of the current methodologies, with their accompanying Three Letter Acronyms, will pass into history, to be replaced by others, for their own period of popularity. I realise, however, that there will be

There is no data on the future.

Laurel Cutler
US vice-chairman of
FBC/Leber Katz
Partners Inc. Magazine
1987

many methodologies that will survive for very many years, but there are too many unknown factors to try to predict which ones will survive, so they will have to remain as anonymous as their shorter-lived competitors.

We shall therefore add a seventh area – Methodologies – to our list of Areas of Divergence. The list now becomes as follows: -

AREAS OF DIVERGENCE

CODES
FUNCTIONS
TECHNOLOGY
CULTURE
EVOLUTION
INTERCONNECTION
METHODOLOGIES

An understanding of the spread of divergence of each of these areas will make a fundamental difference to the way in which a user requirement list is produced, a vendor short-list is derived, a vendor is selected and an on-going relationship is monitored. The alternative is to list **current and estimated future** user requirements and select a vendor and package against these requirements.

By using the Areas of Divergence, it is possible to investigate requirements generically and examine vendor strategies and products in an in-depth manner. As stated in Chapter 4, the selection, implementation and use of a maintenance or asset management information system is now not only a very costly exercise, it can be a very **risky** exercise for those involved with it. As there is no data on the future, it is imperative that all possible factors are taken into consideration. An analysis of differences

in business requirements and use of maintenance and asset management systems is thus relevant to all potential users.

Despite the continuous relevance of this list for two decades, there is no assurance that it will remain correct for the next two decades. The method of attempting to objectively identify differences in requirements will, however, always be relevant and can therefore be applied at any time. When potential users consider each topic area, they should spend sufficient time to satisfy themselves that they understand **all** the variables for that area **at the time of their selection and for as long as they can into the future**.

They should then consider their own requirements and identify where they fit in to the generic list of variables. If they don't appear to fit in generically, then they should be able to define why they have a unique requirement and how it differs from the closest generic variable. Naturally, one would hope that vendors would be doing this already!

The next few chapters will describe each of the Areas of Divergence.

Key Points in Chapter 5

- Companies must recognise what type of organisation they are and where they intend to be in the future.
- Failure to recognise this can lead to unsuccessful system implementation.
- A maintenance or asset management system should not be procured in order to polish an individual or corporate ego.
- It is essential to make a regular objective assessment of the requirements of any industry.
- Vendors compete by offering attractive packages of **what they think the users will buy**. This is not necessarily **what the users want**.
- The information system must be able to handle **change**.
- The information system must be **dynamic** and **appropriate to the ongoing need**.
- Systems that are not dynamic are essentially **mechanisms for SELLING rather than mechanisms for USING**.
- The problem areas that have differentiated user organisations from each other – the **Areas of Divergence** – have changed very little in the past twenty years, so there is every reason to believe that they will continue to be problematic.
- Differences in **hardware strategy** have now been transformed into differences in the application of **technology**.
- Differences and changes in **methodology** now require this topic to be added as another **Area of Divergence**.
- The **Areas of Divergence** are now **CODES, FUNCTIONS, TECHNOLOGY, CULTURE, EVOLUTION, INTERCONNECTION** and **METHODOLOGIES**.
- It is essential that a potential user understands all the possible variables of each topic area before selecting a system.

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

What you need to know about asset numbering systems to ensure your new asset information system caters for your asset and business needs

Chapter 6

Codes

Codes refer to the way an organisation describes itself on an information system.

It covers the identification of the organisation's assets within an asset register, the description and short-hand used to describe the operations performed on these assets, and the manner in which it prefers to analyse the performance of the assets and of those resources operating upon it.

More than the definition of number systems and acronyms, it involves the definition of how these items relate to each other. All organisations carry with them a considerable amount of history, modus operandi, and a whole host of specifics that are inappropriate, difficult, or expensive, to change.

It may be ideal for the user organisation to adapt its codes, structures and relationships to comply with those of the preferred information system, but in most cases this is just not possible or desirable. But it is astonishing how little this is considered when procuring a system.

I have come across many organisations that found out after the implementation of their system that it was not possible to adequately describe their plant and assets

on the system (see Chapter 15). They therefore had either to pay for changes to the system structure (an expensive exercise) or live with the problem and the compromises that it imposed. This always results in a less than expected return on investment in relation to the procurement of the system.

Modelling

"If you cannot adequately model your business, then you must question the relevance of the operations you perform on the model and the information derived from them."

This area is concerned with modelling the operation of the plant and assets on the information system. Clearly, if the model is wrong then the relevance of the information system will be suspect and the possibility for incorrect data, information, knowledge and wisdom is increased. There are so many other factors that can cause the resultant information to be suspect that it would be foolish to introduce such a fundamental problem as well.

If you cannot adequately model your business, then you must question the relevance of the operations you perform on the model and the information derived from them.

I shall attempt to illustrate the problem with reference to an asset register. Suppose we have a well-established operation such as an electrical distribution utility. The utility wishes to procure an asset management information system to enable the organisation to improve the effectiveness and efficiency of its assets and those who support the assets.

A fundamental requirement is that the nature of each asset, its relationship to other assets (either by type or by physical or logical connection), and its relevance, criticality and position within the business hierarchy should be able to be emulated on the system in a manner which is understandable to all who have to use it.

A further requirement is that all of the relationships should be able to be changed at any time, by a user

with the appropriate authority and password, in order to meet the changing requirements of the business.

Once the model has been entered into the system, it must be possible for it to emulate all the operations that need to be performed on the assets or groups of assets. A common requirement in an electricity utility is to be able to isolate a group of assets in order to perform work on them. For example, if an outage occurs, then it is obvious that the work that is scheduled for assets within the outage should be brought forward in order to utilise the available downtime. Now this will only be possible if the end points of the outage are able to be defined on the information system and its associated functionality can enable all the assets within the outage to be identified and any outstanding work to be flagged up.

The System must be able to emulate all of the operations that need to be performed on the assets or groups of asset

This structure is quite different from the asset structure of a process or manufacturing plant, although there are certain aspects that are similar. The differences, however, are sufficient to cause a software system that was designed for a manufacturing plant to be quite wrong for an electrical utility. They may, however, appear to be similar at the time that the system is demonstrated to the prospective user. It is essential that sufficient care be taken at this stage to ensure compliance with the plant requirements. This usually means that a considerable amount of effort has to be put in to identify important structures and data, and to have these emulated and demonstrated by the vendor. Effort expended at this stage is worth while. The effects of not spending this effort can be very costly and embarrassing!

Take the time to ensure compliance with plant requirements—it will more than repay the effort!

Depth and Roll-up of Structure

The depth of the structure defines the number of its levels. Most maintenance and asset information

Too much roll-up can result in considerable processing overhead as well as long reports—*if you don't want it, specify it!*

systems can handle any number of asset levels in their asset hierarchy, so that they can emulate the depth of a user's structure quite adequately. However, in practice, it can become quite tedious, time-consuming and unnecessary to go down to very low levels and allocate jobs, and their history, to low-level equipment within a plant. It may be necessary to go down to this level for some types of plant and equipment, but users should consider the effort required – and the overhead in computing resource and time – of doing so. This may be such as to negate the perceived benefits of a deep search.

Roll-up is the linkage of assets within a parent/child relationship. When work is carried out at a low level, a decision needs to be made as to whether the history of that work should be listed along with the history of the parent asset, grandparent asset, etc. This can result in considerable processing overhead as well as long reports. However, it is a requirement that some organisations have, *so if you don't want it, you must define this in your requirement specification*. It is not sufficient to satisfy yourself that the vendor's asset structure can emulate your structure. It is important to determine how it works and whether or not you are prepared to live with it. Remember that such a function may perform quite differently in real life and with real data to that which is demonstrated at a presentation. It is very easy to lose objectivity at presentations. The rule must always be to question what is demonstrated in relation to your own real needs.

Business Processes

The package should be able to emulate not only your physical asset structure, but also the business processes of your organisation.

Structures need to reflect your processes and the location of your assets

This means that if yours is a process plant, the function of each asset or group of assets should be able to be modelled, right up to and including the function of the overall plant. Also, the asset structure should be related to, but be separate from, this functional hierarchy, so the assets or groups of assets are able to be moved from one function to another, taking their own history with them while at the same time leaving relevant history behind with their previous functional position. Now this facility may seem to be unnecessary for those organisations, such as water utilities, whose assets never seem to move. However, for such organisations it is important to make sure that this static state is correct for all their assets. If they have any assets that move from one function to another, then they should consider a separation of the asset register from the functional structure. This argument also holds true for the numbering of the assets and functional positions.

A third structure concerns the locations of the assets or functions. Again, the example of the water utility with fixed assets may point towards an organisation doing without a separate location register and incorporating location details within the asset numbering system. However, it is for the organisation to decide what information and analysis capability they are likely to lose by doing so. As a general rule of thumb, it is best to have separate asset, function and location registers, and to relate function and location to each important asset. Locations can be further complicated by their roll-up. For example, does the bottom level location code contain the location codes of all parent levels? Also, a coding system for a building may have considerably different structural requirements from the coding system for a manufacturing plant or an estate. It is essential that a prospective user analyses his requirements and makes sure that all his structures are catered for.

Should an asset be described by its type, its function or its position?

We have now considered three coding types that are used to emulate the structure of a plant or operation. For most organisations, these three structures will be sufficient to describe their activities and, in the case of water utilities, would appear to be more than enough. I would, however, like to make two further points about structures.

Structural Confusion

First of all, there would appear to be considerable confusion in many industrial organisations and utilities regarding whether an asset is described by its type, its function or its position. This usually results from no original fundamental thinking or direction taking place at the time that the assets were originally numbered, together with a lack of control thereafter. The resulting 'Asset Register' is then a combination of location, function and asset type codes which are difficult for anyone to comprehend who has not been entrenched in the plant operation for a number of years. This makes it very difficult to transfer such a structure onto a new information system. I have come across several organisations who found that they had this problem after they had purchased a new information system and when they were trying to transfer their Asset Register to it. Clearly, it would have been better, and have avoided many serious problems if they had considered their structures before selecting their system. This, however, seems to be a problem for many organisations. They go ahead and start procuring a system without first considering whether or not their existing structures are correct for themselves and appropriate for transferring to a new system.

This is as much a sociological problem as a structural and historical problem; people tend to want to get on with the interesting parts rather than spend time on

the boring and highly political parts! But if effort is not put into this beforehand, then it is certain that at least as much effort will have to be put into the activity later, unless luck proves that the structure was correct in the first place.

Furthermore, this later effort will not have been budgeted for, it will need to be expended during the implementation project, it will skew both cost and time estimates and it may show that the wrong information system was about to be installed! In short, it is likely to be highly embarrassing for all concerned.

Yet, had the personnel involved asked themselves whether or not they were being honest with themselves and their organisations at the start, they could have avoided all these additional problems! They could have ensured that they were ready for system selection, that they understood why their structure and information requirements were as stated, and realised what they could and could not change.

Failing to prepare is preparing to fail.

Dr. W. Edwards

Additional Structures

The final point that I would like to make about asset structures is that the three structures – type (i.e., traditional asset register hierarchy), function and location – may not be sufficient to satisfy the organisation's requirements for information. This takes some explaining, as it is a requirement that many organisations do not know they have! This is the need to group assets independent of type, function and location and then to perform work and accrue history on the group and on each individual asset.

Supposing assets in such a group were related only by the fact that they had all been designated as dangerous. Another group (which may contain some, but not necessarily all of the assets belonging to the

Coping with additional structures and other groupings

'dangerous' group) may be designated to have potential environmental impact. Yet another group may have no other relationship than that they share a common power supply (a very common occurrence where processes or production systems are changed within an existing plant). For organisations with such asset groupings, it would be beneficial to group work and possible history, on this basis. It would also be beneficial for the selected information system to have the ability for the user to define such structures and for these structures to be able to operate in a hierarchical fashion. It may be difficult for an organisation to anticipate how such a facility will be used at the time that they procure a system. However, it is essential to remember that the procured system will be the foundation for all future data and that the system and its structure must cater for future improvements in operation. These improvements will almost certainly be unknown at the time of procurement and will remain so until the information system produces results, but don't limit your future options!

Other Groupings

The structures problem is not limited to assets. Increasingly, work is carried out by teams that consist of different relationships of separate and combined skills. The reporting requirements for the teams, the team members, the job and job hierarchy, and the asset, function, location or group of assets can be complex. Any complex requirements that are essential to the operation of the user organisation must be considered and stated as a requirement of the information system before its selection. It is too late to state these requirements after procurement; then it will cost money, time and embarrassment to adapt the system – if this is, in fact, possible! Remember also that these structures should be able to cater for contract as well as internal labour, and that a Schedule of Rates facility may be necessary.

It is insufficient to simply have the system demonstrated to you at a presentation

As we shall see later on when we consider the actual procurement exercise, it is essential for the procuring organisation to apply whatever effort is necessary into satisfying itself that the selected system handles these structures correctly. It is insufficient to have these demonstrated to you at a presentation. With current technology, anything can be demonstrated, and the situation is only likely to improve for the vendor and work to the possible disadvantage of the user organisation. *This is a good argument for refusing to divulge all your requirements before a demonstration and for requiring some facilities to be demonstrated without prior warning. And don't accept a promise to respond to the requirement in the future if you believe that what you requested should already have been considered as a possible generic requirement by a knowledgeable vendor.*

Asset Numbering

The coding problem is, however, not just about structures. There is a whole subject concerning the numbering of assets. It is not possible to cover all the ramifications of asset numbering in this book, as the possibilities are endless. Several books and papers have been written on the subject, but will not be referenced here, as they tend to be each author's view of asset numbering, based upon a particular industry. The most common starting point, however, is to prefix each asset number with a mnemonic which is easily identified as the asset type. Whether you then go on to add a sequential number to this mnemonic for each asset type or for all assets will be up to individual preference and historical factors within each organisation. Another differentiator is whether or not the system generates the asset number. If it always generates the asset number automatically and you already have all your assets numbered, then you may

have some problems with the system! Good systems should, however, provide for both requirements.

Length of Asset Number Field

The length of the asset number field in a system can be a further problem for users. This problem is seldom now a case of insufficient characters in the field; rather, it tends to be the opposite, with a large number of characters assigned in order to cope with any eventuality. This achieves its objective regarding the ability to handle any number of characters, but it can be extremely messy and confusing to use. Validation of such a field may also be a problem, and, as we saw in Chapter 3, if the data is unable to be properly validated, then considerable data quality problems can ensue. This leads to suspect information. So, as with all other aspects of the procurement exercise, don't just accept the data fields at face value, i.e., that they are large enough to hold your data. Delve into how you will be required to use these fields and how your ultimate use of the system and expected benefits could be affected. This applies to all important fields. All such small factors are important in achieving a successful system. Put another way, insufficient attention to important details can negate the expected benefits of purchase of the system.

insufficient attention to important details can negate the expected benefits of purchase of the system

Job Numbering

Job numbers can be a similar problem to that of asset numbers. However, in practice, most systems can handle most types of job numbering structures. Some of these can, however, be quite pedantic, especially when the job number field is long in order to cater for all possibilities and all you need is six digits! As we shall see in a Chapter 13, the compromise between flexibility and ease of use will make the difference between an acceptable and an awkward information system from the user's viewpoint.

Fault Codes

The handling of fault codes can be more difficult. This is an area where organisations should be able to progress from the simple control of work on assets and the condition of assets to a much more comprehensive, and thus more complex, asset management environment. At the simple end of the scale, there may be no recording of fault codes, or the requirement may be able to be switched on or off depending on the criticality or importance of the asset or function. At the other end of the scale, it is common to have structured fault, action, cause and consequence codes which can be applied not only to an asset type but also to the function and criticality of the asset. These can often be linked in a manner that allows expertise to be embedded in the structure. Clearly, not only will the specific needs of a user organisation need to be catered for, but it will also be necessary to switch on and off facilities as the user progresses in capability.

Use of Text

The final point to consider regarding codes is the use of text in the system. While it is generally recommended that codes should be used as far as possible, it is almost impossible to completely eliminate text from the recording mechanism. Text is difficult to analyse, especially if it is miss-spelt. Also, how much text do you need, and can this be limited by the user organisation or does the supplier limit it? What facility is there for progressing with information retrieval; can you, for example, choose to add free text in a controlled manner once the users have mastered the use of codes on the system? Could you also incorporate multimedia in the form of pictures, sound or video? In other words, how far has the vendor considered the requirement and does it fit in with where your organisation could be in the future?

Key Points in Chapter 6

- Codes and structures relate to the model on the information system of the operation of an organisation's plant and assets.
- If you cannot adequately model your business, then you must question the relevance of the operations you perform on the model and the information derived from it.
- It is essential to identify important codes and structures, and to have them not only demonstrated by the prospective software supplier but also proved to work in the manner required by the user organisation.
- Asset hierarchies can theoretically be provided for any number of levels. However, each additional level brings with it problems of complexity, computing resource and time overhead that may be unacceptable to the user organisation. Don't just accept the multiple levels offered by the supplier; define what levels you need and examine the overhead presented by them.
- Asset structures, functional structures and location structures should all be separate, but related to each other. It is a common mistake to mix these up (there is usually a historical reason for this) and to transfer this confusion to the information system.
- It is essential that coding and structural discussions and definitions are made before embarking on the selection of an information system, otherwise such discussions and decisions are likely to take place in a rushed manner at an inconvenient time.
- Failing to prepare is preparing to fail.
- Even if an immediate requirement for the facility does not appear to be obvious, user organisations would be well advised to ask for the facility to group assets by any defined criterion, in order to perform work and record history against such a group.
- Job structures and in-house/ contract relationships can be complex. Make sure they are understood, defined and the boundaries of acceptance specified before selection of an information system.
- Asset numbering systems vary widely. Acceptance of a user organisation's historical asset numbering system may not be possible on many information systems and conversion to a new system may be costly – and inappropriate.
- The Asset Number fields provided in systems may be considerably longer than necessary, causing confusion and difficulty in validation.
- Insufficient attention to important details can negate the expected benefits of purchase of the system.
- Poor validation of data leads to poor data quality and suspect resultant information.
- Fault codes can be simple or complex, and can have complex relationships. These should be able to be switched on or off, depending on the needs and expertise of the user organisation.
- Use of free text should be controlled.