Managing Technical Architecture

BUTLER COX FOUNDATION

Research Report 83, October 1991

BUTLER COX
FOUNDATION

Managing Technical Architecture

Research Report 83, October 1991

Report synopsis

A technical architecture defines a planned set of computer facilities (the 'technical infrastructure') that can provide the connectivity and flexibility that information systems will increasingly require. In this report, we explain why a technical architecture is needed, how to go about defining one, how to justify the subsequent infrastructure investments, and how to gain conformance to ensure that everyone complies with the architecture.

Published by Butler Cox plc
Butler Cox House
12 Bloomsbury Square
London WC1A 2LL
England

Availability of reports

Members of the Butler Cox Foundation receive copies of each report and management summary upon publication; additional copies and copies of earlier publications may be obtained by members from Butler Cox.

Photoset and printed in Great Britain by Flexiprint Ltd., Lancing, Sussex.

BUTLER COX FOUNDATION

Managing Technical Architecture

Research Report 83, October 1991

Contents

A management summary of this report has been published separately and distributed to all members of the Butler Cox Foundation.

Scope of the research and research team

The views and recommendations in this report are based on a combination of research and the practical experience of Butler Cox consultants in the field of technical architecture.

An extensive, year-long programme of research was carried out, starting in August 1990. It was initiated by the document describing the scope of the research that was sent to Foundation members, 131 of whom responded. The replies highlighted systems directors' main areas of concern on the topic of technical architecture. This was followed by focus groups in Germany, the Netherlands and the United Kingdom, where the issues of main concern were discussed at length. In addition, 33 interviews were carried out with companies in Australia, Belgium, Canada, Germany, Italy, the Netherlands, Sweden, Switzerland and the United Kingdom. The interviews emphasised the benefits to be obtained from a technical architecture and the problems of gaining compliance with it. We also carried out a literature search to review the concepts being developed by experts, particularly in the United States.

Butler Cox's consultancy experience in national and multinational technical-architecture assignments was reviewed, and this was used to structure the approach to developing a technical architecture outlined in Chapter 3.

The research was led by Declan Good, Butler Cox's specialist in IT strategy, whose consultancy work has involved advising multinationals on the definition of appropriate technical architectures. He was assisted by Tony Manley, who has also worked in this field, and was co-author of Report 72, Managing Multivendor Environments. Expert guidance was provided by David Flint and Martin Langham, both of whom have extensive experience of technical-architecture design.

Chapter 1

Technical architecture is a major element of systems strategy

Many business managers cannot understand why computer systems in different parts of the organisation are unable to communicate effectively. Why does it require so much effort to consolidate information about products, sales, stocks and customers? Why is it so difficult to connect the organisation's computer systems directly to customers' and suppliers' computers? Why is it sometimes quicker and easier to rekey the information produced by one computer system so that it can be used by another? The answer to these and similar questions is usually that, in the past, there has been no common framework that will enable different computer systems to be interconnected and no set of rules that are understood, agreed and adhered to across the organisation to ensure that the applications will be built so that they can work together.

Part of the solution lies in the design of the applications portfolio and in the steps taken to ensure consistent use of data across the business. As important, however, is the need for a planned set of computer facilities (the 'technical infrastructure') that can provide the connectivity and flexibility needed and that can adapt to the evolving requirements of the business.

A technical architecture is a plan for a technical infrastructure

A technical architecture is a plan for a technical infrastructure. The architecture (sometimes known as the technical strategy) defines the components of the infrastructure, how they will work together, where they are to be located, when and how they will be used, and the procedures for making changes to the infrastructure once it has been installed. A technical architecture makes it possible to build a coherent technical infrastructure that provides the connectivity and flexibility from which substantial business benefits can be derived. Figure 1.1, overleaf, indicates the range of business benefits that can arise from defining a technical architecture.

It provides connectivity and flexibility, leading to substantial business benefits

The technical architecture is not concerned with the applications systems themselves, but it does embrace the hardware on which they will run and the systems software components such as database management systems. In addition to technical rules and standards, the architecture should include an overall plan, or topology, of the main elements of the proposed infrastructure and a migration plan for its development. To be really valuable, the architecture must be designed to anticipate the possible needs of the business. It is not concerned just with today's requirements, but as far as possible, with safeguarding the future.

The technical architecture is therefore one of the main elements of an organisation's overall systems strategy – the other three elements being the applications strategy, the data architecture and the definition of systems management roles and systems resources.

Figure 1.1 A technical architecture provides significant business benefits

A charter airline took steps to make its technical architecture consistent with its business objectives. Business benefits have been achieved because the resulting technical infrastructure has facilitated:
- A reduction in the cost of cross-functional processes.
- Speeding up of information flow across functional boundaries.
- Access to a repository of company information.

After a false start, a public utility deferred the definition of its technical architecture until business objectives were known and agreed. The resulting benefits to the business were that:
- Data from a variety of functions could be accessed by one workstation.
- Access to this data was simple to operate.

A chemicals group's aim is to facilitate operating units' use of IT by supplying essential communications and standards. The technical architecture has enabled the units to achieve synergy by exchanging information.

The existence of a technical architecture in another chemicals company is enabling it to change its business processes. The chief benefits are that:
- Global systems can be introduced — for example, for issuing customer/product codes from one source.
- Customer service can be improved.
- Intersite communications can be improved.

The technical architecture in a drinks company has enabled it to install common systems around the world. The main benefits are:
- Economic provision of a full range of systems services to all sites.
- Quick, cheap and effective transmission of reports from remote sites.

An oil company aims to define the minimum mandatory standards necessary to handle critical common data, while promoting, rather than mandating, other aspects of systems policy. The chief business benefit will arise from the ability to prepare good-quality information based on common data.

The applications strategy has, in the past, been derived mainly from the known needs of the business – both the current needs and those that can currently be forecast. It has also taken account of both the use of IT by competitors and knowledge of emerging technologies. A growing number of organisations, however, will be adopting a process approach to managing the business and will be redesigning (or 're-engineering') their business processes. In such organisations, the applications strategy should be defined in the context of a thorough review of technical capabilities, organisation structure and business objectives to achieve an effective balance between all four. Achieving this 'strategic alignment' will be the subject of a future Foundation report.

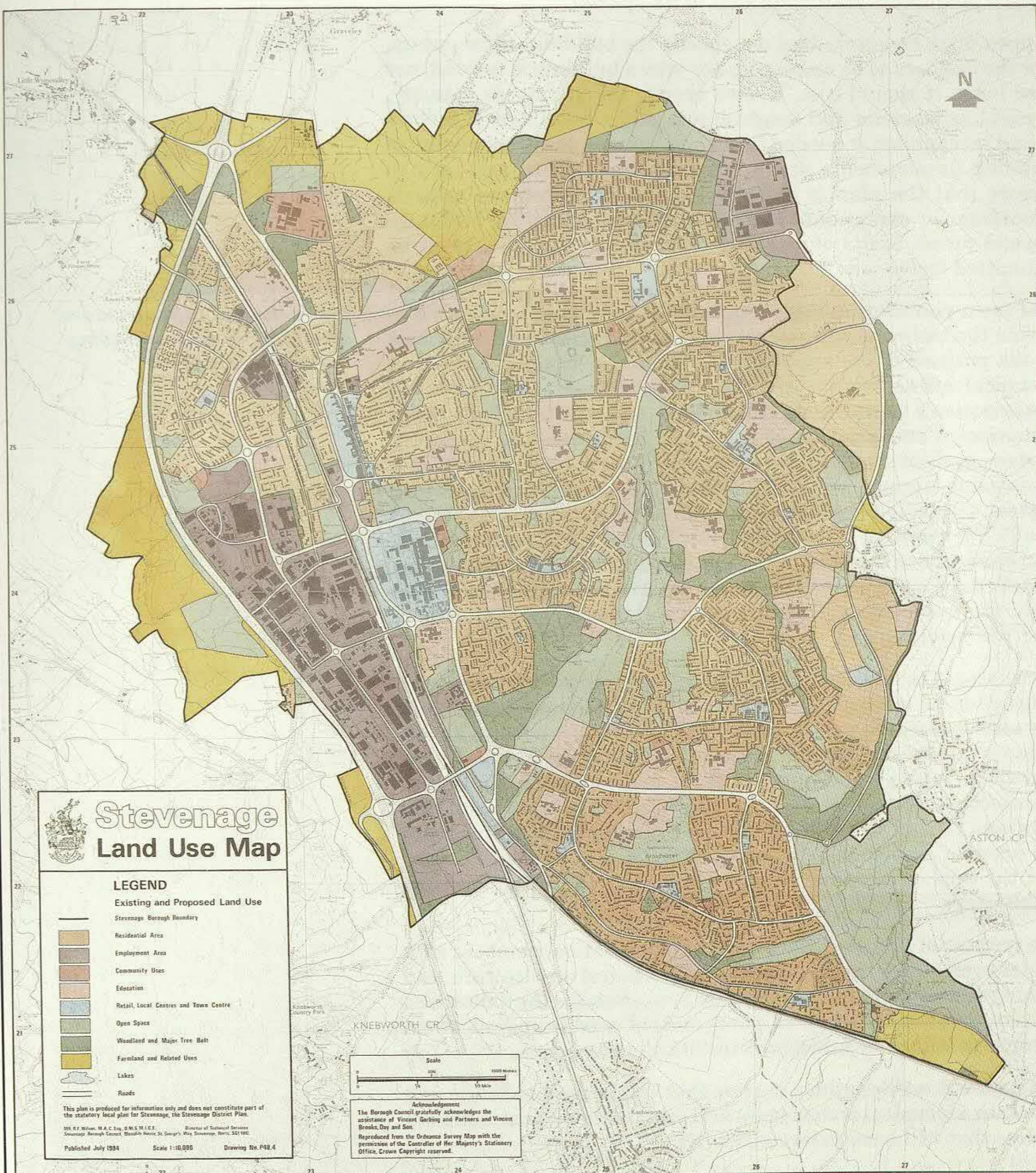
The technical architecture derives from the applications strategy – it must accommodate the known and expected business needs. It must also be designed so that it can accommodate emerging technologies and the installed base of technology (that is, the existing infrastructure), to ensure that the most effective use is made of existing investments.

The technical architecture must accommodate known and expected business needs, emerging technologies and the existing infrastructure

The purpose and scope of a technical architecture are analogous to those of a town plan. The town-planning concept introduces the idea of zones and of an overall topology (see Figure 1.2). The town planner defines the purpose of the various zones of the town

Figure 1.2 A technical architecture is analogous to a town plan

Shown below is the Stevenage town plan. Stevenage was the first 'new town' built for London overspill after World War II, and the plan shows the topology and intended use of each zone. The overall design was driven by the 'Neighbourhood Principle', devised in the 1920s by an American architect, Clarence Parry. This principle applies the concept of a natural catchment area of community facilities such as schools and shops, which represents most people's primary sense of identification. Information technology technical architectures are driven by analogous principles, articulated as the systems management principles. (See Chapter 3.)



(residential, employment, education, open space and so on) and specifies the mix of use within these zones, the limits of the zones, and even the rules for heritage buildings. The next step is to decide on the options for the major components of the physical infrastructure – roads, hospitals, schools, shopping centres and so on. If the town is small enough, the plan can be kept very simple. If it is a city, the plan becomes complex and will have an elaborate set of documents and procedures associated with it – it may even have a permanent staff dedicated to its upkeep.

A town plan, however, does not specify the buildings to be put up, nor the builders to be employed, nor why a building is needed, nor how much it should cost. It does specify the standards, density, general appearance and height limits of the buildings. It follows that a mechanism is needed (with appropriate powers) to monitor building developments and apply sanctions where necessary to ensure that the plan is adhered to, and to operate an appeals procedure for exceptions. Plans for extending existing buildings beyond preset limits of size and location, for example, must be submitted for review.

The technical infrastructure should be divided according to the purpose of the applications

The town-planning analogy suggests that it should be possible to divide the technical infrastructure in some logical way according to the purpose of the applications. The major components of the technical infrastructure are the IT 'platforms' upon which the organisation's applications will be built and run. (An IT platform comprises a combination of computer hardware and operating system, and optionally, other systems software; they range from a mainframe running under the control of a proprietary operating system, teleprocessing monitor and database management system to a PC (or compatible) running DOS, Windows 3.0 and NetWare.) The applications that will be built once the infrastructure has been installed are analogous to the buildings in a town. As with a town plan, the technical architecture does not specify the applications that will be developed.

Perceiving a technical architecture as analogous to a town plan helps systems managers to understand the purpose and scope of the architecture. In particular, it emphasises the need to have a vision of the future – both in terms of technological developments and of the organisation's use of IT. It makes explicit the role of architecture in facilitating the introduction of new technical solutions, as and when these become available. It also illustrates the need for procedures that will ensure that the infrastructure, and the applications built on it, are developed in line with the architecture.

The purpose of this report is to provide Foundation members with guidance on how to set about defining a technical architecture, and gaining commitment to it. In Chapter 2, we describe the limitations of existing technical infrastructures and the need to upgrade them to conform with a well designed technical architecture.

In Chapter 3, we describe a process that can be used to define and maintain a technical architecture and some of the critical technical choices that will have to be considered.

In the final chapter, we consider the issues that will arise once the architecture has been defined. These are concerned with justifying the consequent investments in infrastructure, gaining commitment to the architecture from the business and ensuring conformance with it. Eventually, this means addressing a recurring question – how can business managers be persuaded to accept what seems to them to be a second-best solution that conforms with the agreed architecture rather than a better local solution that might cost less and deliver greater benefits or deliver them more quickly?

Chapter 2

The need for a technical architecture

The difficulties faced by many organisations today are caused by the fact that their existing technical infrastructures cannot provide the connectivity and flexibility that are now required. The only way to achieve the interconnectivity and flexibility that organisations will increasingly be demanding of their systems is to build an infrastructure that conforms with a defined architecture.

Existing technical infrastructures are now inadequate

In the past, an organisation's technical infrastructure has often been based on the products of one supplier, or a small number of suppliers. In effect, user organisations have adopted the proprietary technical architectures defined by suppliers. Until recently, tracking a supplier's technical strategy was a viable option for user organisations. Today, however, it is increasingly recognised that no one proprietary architecture can provide access to the full range of software facilities and cost-effective hardware that is now available. Moreover, the trend to open systems is undermining the dominance of proprietary architectures.

No one supplier's proprietary architecture can satisfy all requirements

In defining a proprietary technical architecture, each supplier's objective has been to establish a favourable market position relative to his competitors', while building on his individual technical strengths and taking account of the full value of the installed base. Some suppliers (Digital for one) have been skilful (or lucky) enough to make decisions that have stood them in good stead over generations of technology. Others (primarily IBM) have traditionally led from a position of market dominance. The remainder have followed the market leaders or, as Apple has done, set completely new standards, and (so far) lived to tell the tale.

User organisations, however, have very different objectives in defining a technical architecture. Their primary focus is on business needs, and technology remains the means to an end, rather than an end in itself. Their technical architectures have to be designed to reconcile business needs with the realities of available technology, and to take account of the legacies of the past in the form of investments in the existing technical infrastructure and applications.

For user organisations, technical architectures have to be designed to reconcile business needs with technology

During the 1980s, the move to devolve responsibility for systems to individual business units accelerated, and this has also made it increasingly difficult to standardise on one vendor's proprietary architecture. In many organisations, there were few overall rules for making technical choices, and where there were, they were not enforced. The difficulty is compounded by the limited progress towards truly devolved systems responsibilities in many organisations. In Report 81, *Managing the Devolution of Systems Responsibilities*, we described the typical systems function today as having reached the stage of 'hierarchical devolution' – where some systems responsibilities are delegated from the centre, but the central systems unit still regards itself as the controlling influence over IT and is reluctant to give this up. The devolved systems units react against this, but lacking a framework of rules, tend to operate autonomously.

As a consequence, individual applications have been developed in a piecemeal way, without an overall strategy defining how they will link together. Moreover, these 'islands of automation' have often been developed on different platforms. These platforms have been chosen because they were the best for the job in hand, rather than because they were a good fit with an overall technical architecture. A further effect of this is that staff who are familiar with the computer systems in one business unit have to be retrained when they move to another.

Many existing technical infrastructures inhibit systems interoperability

The result is that, in many organisations, technical infrastructures have developed in an unplanned way and are now obstacles to the development of the systems required to support the business. In particular, the lack of interoperability between systems and applications within organisations has proved a major problem. Departmental systems and personal computers from a variety of vendors have been unable to communicate with one another or with 'corporate' systems, leading to inefficiencies and lost opportunities. Data has been rekeyed into one system from a printout produced by another, analyses have been based on incomplete or outdated data, and the use of electronic mail has been inhibited. These incompatibilities are also increasingly important between organisations because, as we demonstrated in Reports 59 (*Electronic Data Interchange*) and 77 (*Electronic Marketplaces*), access by trading partners, and even competitors, is becoming more and more important in all industry sectors.

There is a growing requirement for connectivity and integration

In his recent book, *Shaping the future: business design through information technology*, Peter Keen, director of the International Center for Information Technology, writes about the extent to which the cash flow of US companies is based on electronic transactions. More than half the revenue of banks in most major money centres now derives from ATM (automatic teller machine) transactions, foreign exchange trading and electronic funds transfers. He postulates that this will reach 90 per cent or more by the end of the decade. He argues that technical infrastructure is therefore of fundamental importance to most major organisations and he has derived a set of policy-level requirements (summarised in Figure 2.1, overleaf) that, taken together, could be used as an agenda for defining a technical architecture.

The technical infrastructure is of fundamental importance to most major organisations

The ability to integrate information from different areas of the business, and to enable people in different parts of the organisation to work together using the same information, is also of increasing importance. In our research on competitive-edge applications, we found that many such applications arose from new ways of using

Figure 2.1 Policy-level requirements define the agenda for creating a technical architecture

1. *Practicality:* Our IT base will never block a practical and important business initiative.
2. *Competitive lockout:* If our competition uses IT as the base for an effective business initiative, we will not automatically be locked out of countering or imitating it.
3. *Electronic alliances:* We will match the competition in being able to make alliances, create value-added partnerships, or enter consortia in intercompany or intra- and inter-industry electronic operations such as point of sale, electronic data interchange and customer/supplier linkages.
4. *Re-organisation and acquisitions:* If we re-organise, make acquisitions or divestments, or relocate operations, our core operations, information systems, communications and processing will be able to adapt to the changes quickly and simply.
5. *Third-party intrusions:* No firm in our industry, or third parties outside it, will be able to intrude on our areas of strength or into the mainstream of our marketplace because their IT base gives them advantages that can be turned into a competitive differentiator that we cannot match.
6. *Vendor staying power:* We will not be dependent on IT 'brochureware' – capabilities vendors claim but do not have – nor on vendors with doubtful financial, technical, R&D or managerial resources and staying power. We will choose only vendors with the ability to move towards integration at the same pace as the rest of the IT industry and to adapt proven innovations to its core products, and *vice versa.*
7. *Comparable international capability:* The above requirements will be applicable in an international context.

(Source: Keen, P G W. *Shaping the future: business design through information technology.* Boston, MA: Harvard Business School Press, 1991.)

information that was already stored in existing databases. In most organisations, there is an enormous amount of existing computer-based information that could be used to advantage by the business. However, the opportunities are not realised, partly because few people are aware of what is available and partly because of the technical difficulties involved in assembling and integrating the information. A coherent technical infrastructure helps to solve both these problems.

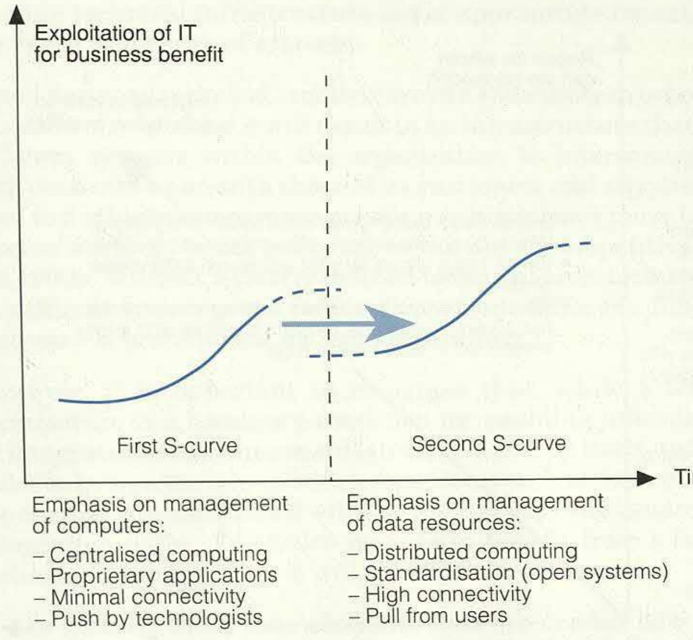
A coherent technical architecture facilitates the re-use of existing information

Robert Kaplan, now Arthur Lowes Dickenson Professor of Accounting at the Harvard Business School, has argued that the demand for connectivity derives directly from the general shift in the late 1980s away from computer-focused management to data-resource management (see Figure 2.2). He suggests that such a shift requires a greater degree of connectivity between applications (and their associated databases). In turn, this promotes a move to two-tier (mainframes and minicomputers) and three-tier (mainframe, minicomputers and workstations) structures that distribute the processing and the data across the organisation.

Many of what Kaplan calls 'traditional' users are stuck on the first S-curve shown in Figure 2.2 and are unable to move to the second. Traditional users argue against making the shift to the second S-curve because:

- They see little need for the new technologies involved with distributed computing and open systems, and believe that they can use their existing infrastructures to support any requirement for growth.

Figure 2.2 The emphasis of IT management has to change



(Adapted from: Kaplan, R. Trading in tired technology. *Datamation*, vol. 36, no. 16, 15 August 1990, p.88-91.)

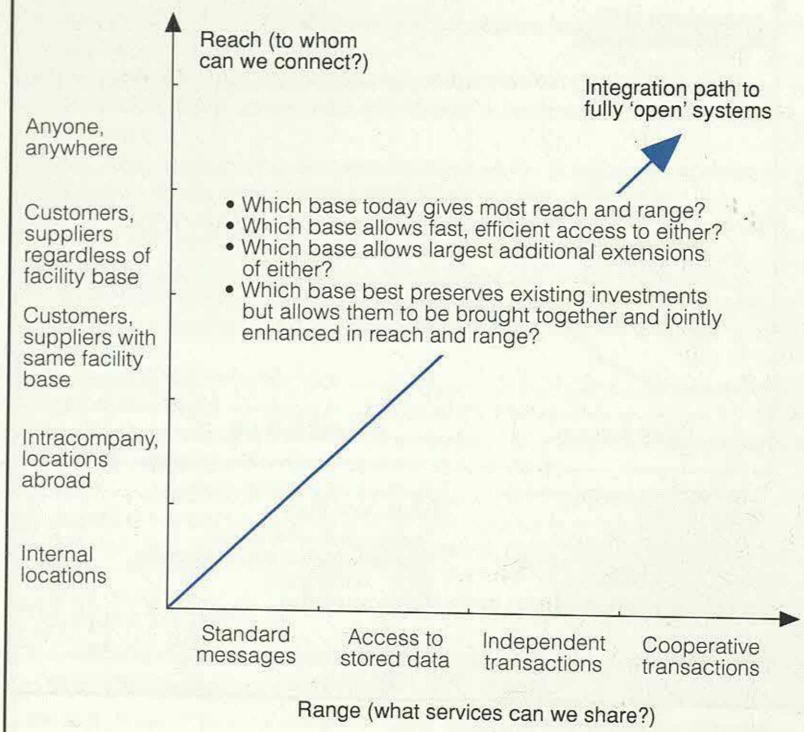
- They are under pressure to contain or reduce costs, and moving to a new infrastructure could not be justified.
- They have suffered in the past from the results of being 'locked in' to one architecture and are not about to try again.

The difficulty for the traditionalists is that the tide is running against them. Much of the requirement for management information cuts across the old functional divisions of the business and the systems that support them. For example, factory production-control systems and financial systems for the overall business have traditionally been developed separately by the manufacturing and financial departments. It is difficult, if not impossible, to integrate these if senior management wants to know the full picture or wants to introduce techniques like 'activity-based costing'. In a similar vein, end-to-end management (or 'total logistics') of purchasing, production, stocks and warehouses, distribution and customer order processing requires much greater levels of connectivity between the applications. Increasing interest in business process redesign has made the need for such connectivity even greater.

Connectivity and interoperability must also extend to the computer systems of trading partners

Connectivity and interoperability are not restricted to the organisation's own computer systems; they also extend to those of its trading partners. Peter Keen has provided a framework for discussing connectivity and interoperability in terms of concepts he calls 'reach' and 'range' (see Figure 2.3, overleaf). Although he does not use these words in their ordinary sense, his usage is widely accepted in this context and we shall use it here.

Figure 2.3 Keen's 'range' and 'reach' concepts address the question of interoperability



(Source: Keen, P G W. *Shaping the future: business design through information technology*. Boston, MA: Harvard Business School Press, 1991.)

Reach refers to the locations and organisations with which systems can, or need to, interwork. In a manufacturing context, for example, 'reach' might be restricted to internal departments like manufacturing sites and sales and head-office administration, or it might extend as far as customers, suppliers, distributors, agents and transport contractors (all external to the organisation).

Range describes the nature of the interaction that is available or needed. The degrees of range typically include unstructured messages, structured messages, access to stored data, independent transactions and cooperative transactions. In a manufacturing situation, at one extreme, these might take the form of low-range, informal, unstructured messages ('cannot make today's meeting', or 'no model XYZ in stock until Friday'). Such messages might be delivered by telex or facsimile, which can provide a very long reach. At the other extreme, payments might be automatically initiated upon receipt of goods. This would require a high degree of integration and cooperation between the supplier's and the buyer's systems – in Peter Keen's terms, 'intermediate reach, high range'.

It can be seen from this description that the issues of integration and connectivity are central to Keen's concepts. Decisions about the required reach and range, both within an organisation and to other organisations, are essentially business decisions, although business managers will require guidance from the systems department about feasibility, costs and implications. In general, however, the reach and range of systems demanded by the business are growing. The difficulty is that, in many organisations, the existing technical infrastructure is not appropriate for extending the reach and range of systems.

A well designed technical architecture can extend the reach and range of an organisation's systems

A well designed technical architecture can contribute to overcoming this difficulty because it will result in an infrastructure that allows different systems within the organisation to interconnect both with each other and with those of its customers and suppliers. The need to do this becomes more pressing as businesses move towards process working, which many see as the key to competitiveness in the 1990s. Without a clearly defined technical architecture, it will be difficult to integrate information from different functional systems – a prerequisite for process working.

However, it is important to recognise that, while a technical architecture is a necessary condition for enabling information to be integrated across the organisation, it is not, by itself, sufficient. Rules to govern the way in which data items such as customer codes are created and maintained will also be required to safeguard future integration paths. To ensure maximum benefit from a coherent technical infrastructure, it will therefore be necessary:

- To identify those data elements that are needed now or that may be needed in the future to provide the links for integration.
- To persuade the organisation to agree to standards for coding them.
- To design and implement a corporate data model.
- To provide adequate data validation and database quality checks.
- To develop and impose standards for data interchange within the organisation.
- To negotiate, agree on and implement standards for data interchange with third parties.

There is a growing need for flexibility

Not only will a well designed technical architecture provide integration and connectivity, but it will also provide the flexibility to respond quickly to developments in technology and to changing business situations and opportunities.

Flexibility to support new technologies

Systems managers are confronted with a constant stream of new developments from the IT industry. Many of these are of limited significance and are short-lived. Some, however, are of great importance, and the technical architecture must take account of the more important trends in the industry, anticipating how they might be used to meet the organisation's needs. To provide the required flexibility, a technical architecture must be able to support new and innovative products, both hardware and software, that may become available after the architecture is established and the resulting infrastructure has been put in place.

The technical architecture must support new and innovative products

In the past, it has often been possible to apply new technologies, such as computer-aided design or document-image processing, in areas that made little or no use of existing computer systems. This will be increasingly difficult in future as workstation penetration grows and as integration between existing and new technologies becomes critical. A technical architecture should therefore include some flexibility, especially in the choice of workstation (the focus for most innovative IT developments today) and the range of software that can be supported. In Report 80, *Workstation Networks*, we also emphasised the growing importance of client-server systems. Client-server systems can provide greater flexibility and higher total performance, greater functionality and better price/performance than centralised systems in many cases, as well as allowing existing workstations to be integrated into corporate systems.

Flexibility should be provided in the choice of workstation

The technical architecture should also be defined to take account of the general move towards open systems. For many organisations, this will mean a move towards Unix, although there are still great uncertainties about the suitability of Unix for commercial applications with large databases and large numbers of concurrent users.

The technical architecture should take account of the trend to open systems

Other organisations will prefer to obtain the benefits of openness by adopting 'universal' software products that sit above the operating systems level and are independent of hardware. Such products have been developed by software vendors like Oracle, Ingres, SAS Software and Computer Associates. The SAS products have been rewritten in C and the same version will be available for IBM mainframes, for Digital's VMS environment, and for Unix hardware from Sun, Bull, Hewlett-Packard and Data General. Computer Associates has introduced the CA90s architecture to enable its software packages to be used across a range of hardware, operating systems, networks and graphical user interfaces.

Flexibility to support changing business requirements

The technical architecture must also be flexible enough to cope with changing business requirements. One of the unforeseen effects of the introduction of computer systems during the past 30 years has been to make it more difficult to introduce organisational change. The automation of clerical procedures has made it costly and difficult to change them; in effect, the organisation has been 'set in electronic concrete'. The definition of a technical architecture is a starting point for resolving these problems. The architecture will provide a basis for the establishment of a technical infrastructure that will facilitate future business flexibility.

The architecture results in a technical infrastructure that will facilitate business flexibility

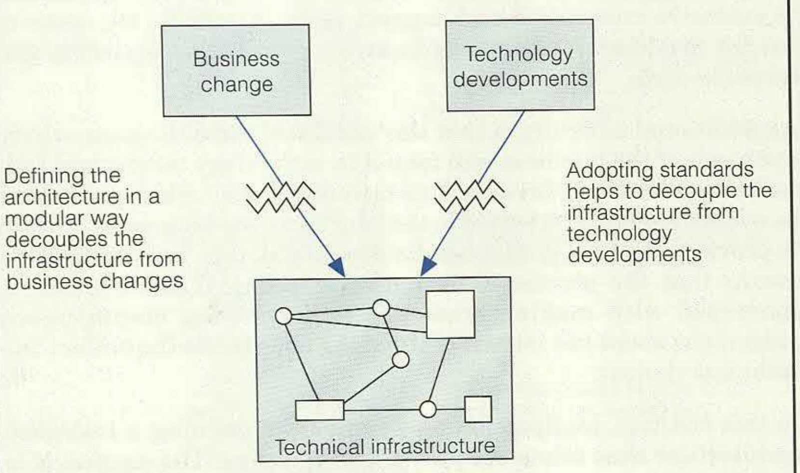
A coherent technical architecture could, for example, enhance the organisation's ability to acquire new businesses or sell off existing ones. A technical architecture that cannot easily be expanded may mean that the systems of an acquired business cannot be integrated with those of its parent. Similarly, the ability to sell off a business unit can be hindered if its systems are not easily separable.

A technical architecture should therefore take account of short-term and long-term business objectives, leading to an infrastructure that supports current and foreseen applications, and that is economical, manageable and robust. In particular, the infrastructure and the architecture on which it is based should be flexible. It should provide wide connectivity for internal systems and for the systems of trading partners (but it should also be secure). It should enable machines, applications and users to be relocated. Ideally, the components of the infrastructure should be available from several suppliers. This is hard to achieve because, often, the most appropriate technology is proprietary, is available only from small companies, or is hard to manage. The architecture should also enable new technologies to be incorporated in the infrastructure or to make use of the infrastructure as they become available.

The architecture enables the infrastructure to be decoupled from business changes and technology developments

The architecture therefore needs to be defined so that the technical infrastructure is, as far as possible, decoupled both from business changes and from technology developments. As Figure 2.4 illustrates, this can be achieved by defining the architecture in a modular way and by adopting standards for the interconnection of the modules.

Figure 2.4 A flexible architecture should be defined in a modular way, with clearly defined standards



Modularity provides the flexibility needed to respond to changing user requirements. If the architecture can be defined as a set of distinct parts, rather than as a monolithic structure, changes may be restricted to one or a few parts. The architecture could be partitioned by business function, but this would make radical business redesign difficult. However, current trends in technology make it more and more possible to consider the technical components, such as workstations, generic applications and networks, separately, and we believe that this is a better approach.

Modularity and standards are critical concepts for defining an architecture

Architectural modules will communicate through appropriate standards. Carefully defined standards reinforce the independence of the modules and provide future-proofing. Standards, however, evolve rapidly in areas where technology is advancing quickly, and care is needed to ensure that changes in standards do not undermine the basic objective of creating a flexible technical architecture. Nevertheless, modularity and standards are critical concepts for defining an architecture. Our suggested method for defining a technical architecture is set out in the next chapter.

Chapter 3

Defining a technical architecture

The primary purpose of a technical architecture is to provide a framework for selecting the IT platforms upon which the organisation's applications will be built and run. Today, no single platform can, either practically or economically, support every application in a large organisation. A technical architecture therefore needs to include several types of platform. The 'architect' – if, indeed, there is such an individual – must decide on the number of different platforms, the number of times that a particular platform will be implemented as separate machines, and the location of those machines. Too few platforms may mean that valuable applications cannot be implemented; too many may result in excessive integration and support costs. Similarly, too many or too few machines in the wrong locations can inhibit flexibility and increase costs.

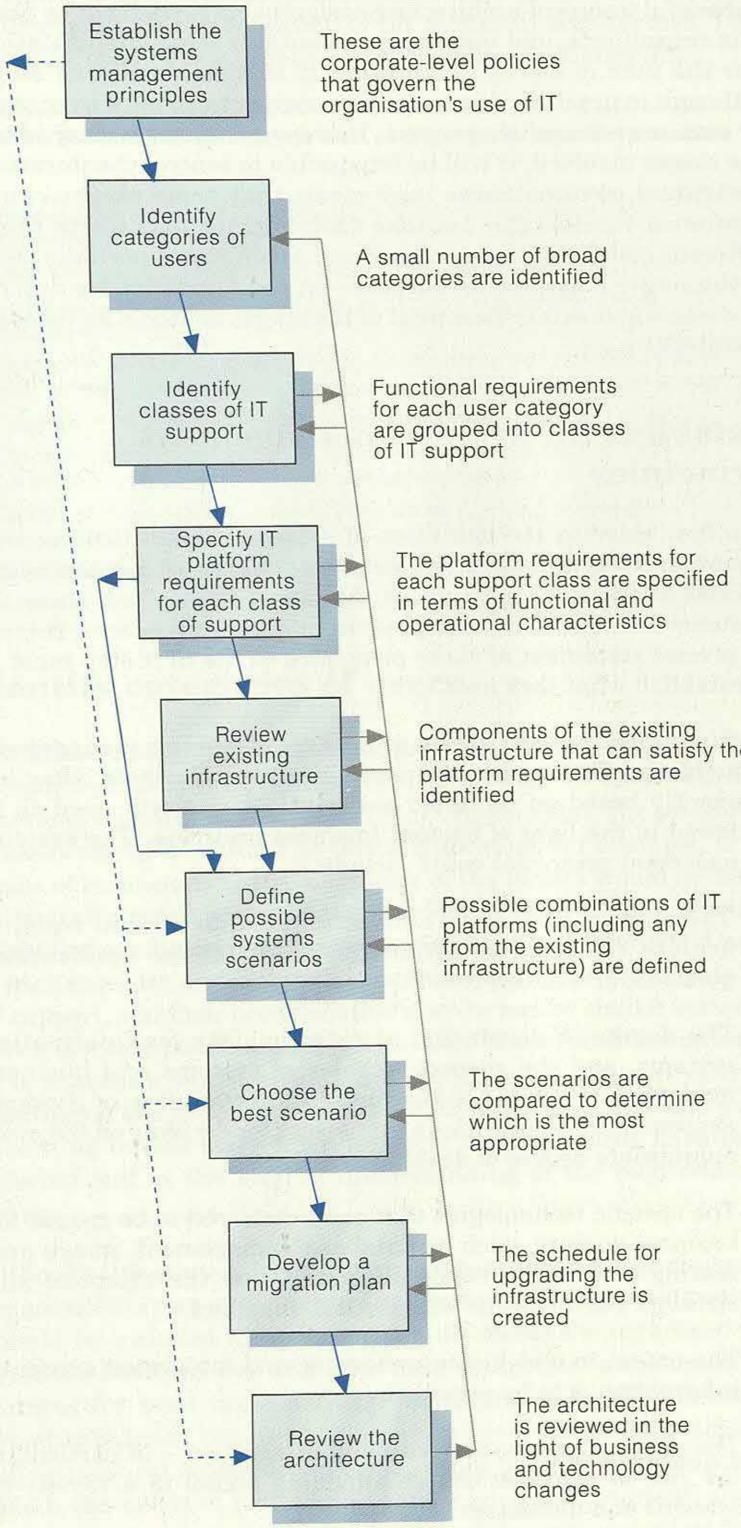
A technical architecture needs to include several types of IT platform

An additional difficulty is that the 'architect' can only guess where the needs of the business will take it in technology terms, and how technology itself will develop. This means that the architecture must be wide in scope and oriented to the long term, but be specific enough to provide practical guidelines for day-to-day use. Inevitably, this means that the process of designing a technical architecture is concerned with making trade-offs and applying management judgement about the impact of the risks and trends that affect the technical choices.

In this chapter, we describe an approach to defining a technical architecture that takes account of these issues. The approach is one of successive refinement, from generalised functional requirements for broad categories of users, through classes of IT support, platform requirements and systems scenarios, to the particular grouping of IT platforms that will form the architecture. Finally, a migration plan is developed to upgrade the existing infrastructure so that it conforms with the new architecture.

The approach, which is summarised in Figure 3.1, starts with the systems management principles – the corporate-level policies governing the use of IT. It then works from the users' needs for IT support, expressed in general functional terms for a small number of distinct user categories, to identify the classes of IT support that are required. The next stage is to specify the requirements of the IT platform for each class of support. The existing infrastructure is then reviewed both to determine if any of it can be used to meet the IT platform requirements and to identify any elements that have to be carried forward to the new infrastructure. Other platforms that could satisfy individual platform requirements are also identified, and possible combinations of IT platforms (including any from the existing infrastructure) that provide all the classes of support are grouped as system scenarios. Each scenario is

Figure 3.1 A nine-stage process can be used to define a technical architecture



compared with the others to choose the one that is most appropriate to form the basis of the architecture.

This approach is not the only way to develop a technical architecture. This particular breakdown has been developed from successful technical-architecture assignments carried out by Butler Cox consultants, and inevitably, it contains some simplifications. For the sake of clarity, we present it as a series of nine stages, although in practice, it is usually necessary to go back over stages or even over the whole process. However, without a clear idea of the stages involved, it will be impossible to control the iterations. Individual circumstances may mean that some stages can be shortened considerably because each organisation starts from a different installed infrastructure and applications portfolio. Some of the stages may even be omitted – in the Appendix, we describe how one organisation used most of the stages to define its technical architecture.

We recommend an iterative and flexible approach

Establish the systems management principles

The first stage in the definition of a technical architecture is to ensure that the 'architects' are aware of the current corporate-level policies and assumptions about the use of IT. We call these the 'systems management principles'. In many organisations, there is no precise statement of these principles so the first step must be to establish what they are.

Each organisation will have a unique set of systems management principles, reflecting its business objectives. However, they are frequently based on outdated assumptions and will need to be reviewed in the light of current business priorities. The systems management principles could include:

Each organisation's business priorities give rise to a unique set of systems management principles

- The main objective of IT investments – to contain costs, to achieve competitive advantage, to renew the applications portfolio, or whatever.
- The degree of devolution of responsibility for information systems, and the respective roles of systems and business managers. (Report 81, *Managing the Devolution of Systems Responsibilities*, provides a framework for deciding on the most appropriate degree of devolution.)
- The specific technologies that are considered to be crucial for business success, such as database management, image processing or satellite communications, and the extent of any strategic partnerships with critical suppliers.
- The extent to which comprehensive and integrated access to information is to be provided.
- The approach to connectivity and integration – in particular, the role of workstations in providing access to a variety of systems and databases.
- The approach to applications development. This might dictate the use of packages wherever possible or the complete

replacement of existing applications in one major development project. Alternatively, a 'fast track' development method might be recommended to meet urgent business needs.

The systems management principles are the 'givens' that are the starting point for the definition of the technical architecture. The principles used by one organisation are listed in Figure 3.2.

Figure 3.2 Thorn EMI Rentals has a set of systems management principles

- Develop software so that, if necessary, it can be run in a distributed environment.
- Keep software applications independent of the locations and hardware.
- Select systems that will support all sizes of group organisation – small, medium and large.
- The technical architecture will provide the foundation for future requirements such as distributed databases, office automation and end-user data access.
- The core of each application will be the same in each country.
- The technical architecture can include some level of risk if this is offset by future benefits in areas of flexibility and future-proofing.
- Rapid application-development facilities will be provided, but not at the expense of quality and standards.
- The information systems will be owned by the national businesses.
- Above all, keep the architecture as simple as possible.

Identify categories of users

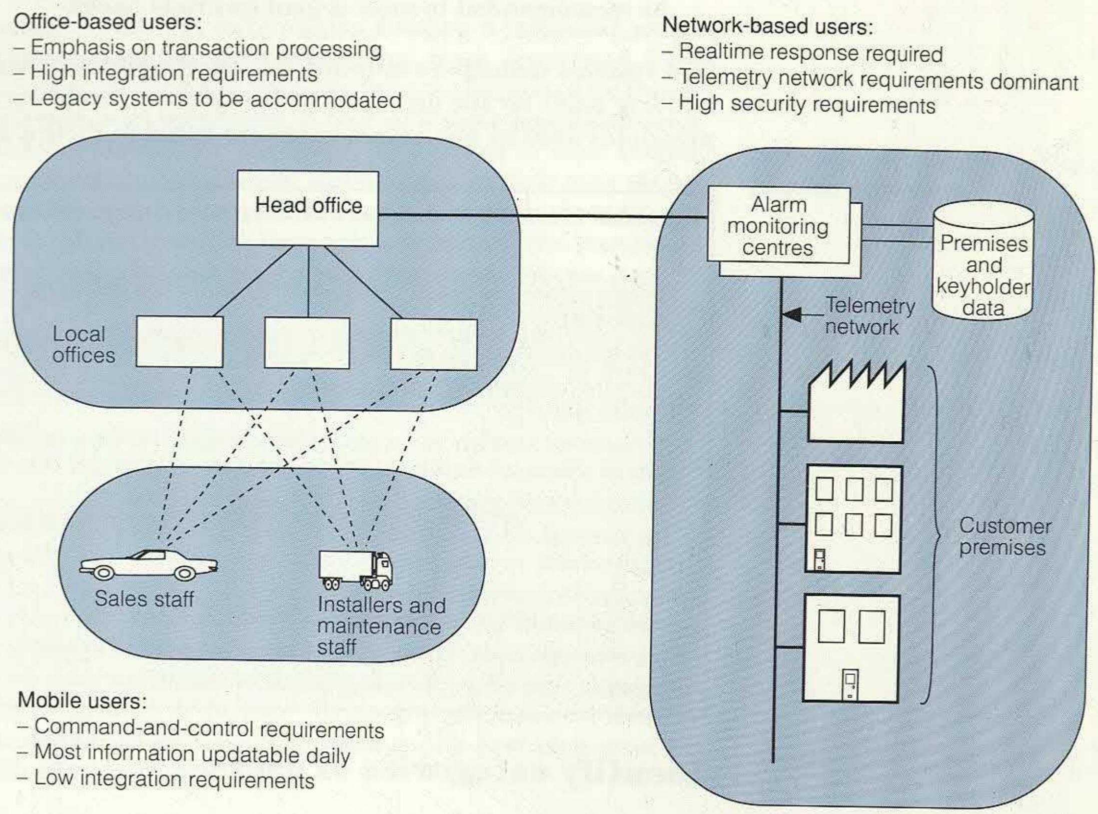
The actual and potential users of information systems should be categorised according to the characteristics of their work – determined mainly by their roles in the organisation. The categories of users might be defined in terms of location, for example, or in terms of business activity. Examples of the former would be 'head office staff', 'retail premises staff' and 'field sales staff'. The latter might include 'business management' and 'customer service'. Users in each category are likely to have similar overall requirements for IT support, and their needs can therefore be met by similar systems. Often, it will be possible to categorise users in several different ways. It is therefore worth spending a considerable amount of time identifying the most satisfactory set of user categories and even returning to this stage after products and standards have been selected and as the level of understanding of the requirements grows.

It is worth spending considerable time defining the most suitable set of user categories

Users from outside the organisation should be included

Although this stage will be concerned mainly with categorising the organisation's own staff, the staff and systems of other organisations should be included if they are likely to access the organisation's systems. Between five and nine user categories is the optimum number for most organisations. Identifying a small number of broad categories means that organisations will be able achieve infrastructural economies of scale, while providing systems that match the needs of individual users. Figure 3.3, overleaf, shows how a security-alarm company categorised its users under just three broad headings – mobile staff, office-based staff and those connected to its alarm network.

Figure 3.3 A security company identified three categories of users



Identify classes of IT support

The aim of this stage is *not* to identify specific application requirements. Rather, it is to identify the classes of IT support that are required. The way to do this is to identify the general IT-support requirements for each of the categories of users and then to group these into classes of support.

Establishing the IT-support requirements for each user category

We believe that it is important to identify IT-support requirements by categories of users, rather than by applications, for two main reasons:

- Because the support requirements of user categories remain relatively stable over time, an architecture based on these attributes will be relatively independent of day-to-day changes in the organisation and therefore more robust.
- It is increasingly important to provide IT support to help people do their jobs better, rather than to change the way they work in order to fit in with computer applications.

The IT-support requirements for each user category should be described in functional terms, rather than in terms of specific applications – for example, access to the divisional sales database,

Support requirements for each user category should be described in functional terms

use of the messaging system, use of the engineers' scheduling system and so on. The levels of responsiveness and serviceability, the kind of user interface and the degree of service integration required should also be described. Figure 3.4 shows how the IT-support requirements for each user category might be documented.

Figure 3.4 The IT support requirements for each user category should be specified in functional terms

User category	User interface requirement	Functional requirements: Requirement	Level of security	Level of serviceability
Sales engineers	WYSIWYG preferred, but Unix interface acceptable; integration of services not critical	Aid to calculating quotations	High	High
		Validation of engineering designs	Moderate	Moderate
		Word processing	Low	Low
		Business and CAD graphics	Moderate	Moderate
		Order entry	Moderate	Moderate
		Checking of manufacturing schedules	Moderate	Very high
		Messaging	High	Moderate

Grouping the support requirements into classes

The functional support requirements for each user category are then grouped into IT-support classes, as illustrated in Figure 3.5. These classes describe the need for online transaction-processing support, batch-processing support, office-systems support, personal-productivity support, application-development support and so on. Some organisations will also have more specialised support requirements, such as computer-aided design, image processing or support for the company's dealers. An oil company, for example,

Figure 3.5 Functional support requirements should be grouped into classes of IT support

IT-support class	Functional requirement
Online transaction processing	Order entry Checking of manufacturing schedules
Batch processing	Validation of engineering designs
Office systems	Messaging Word processing Business graphics Aid to calculating quotations

identified the need for four classes of IT support: industrial automation, online transaction processing, professional automation (including IT support for business and computer professionals) and office automation.

It is not possible to give definitive rules for deciding how to group functional requirements into classes of IT support, but the following characteristics should be considered: database size, need to store and process text, structured data or multimedia information, the balance between processing and data management, special processing requirements, and the levels of serviceability and security.

Specify the IT platform requirements for each class of support

The next stage is to specify the IT platform requirements for each class of support. To achieve this requires creative technical expertise, because those specifying the platform requirements need to understand the inherent nature of each class (online transaction processing, for example) and the way it is evolving. The platform requirements should be specified in terms of functional characteristics and operational characteristics:

Platform requirements should be specified as functional and operational characteristics

- Functional characteristics for online transaction-processing support, for example, might include requirements for data-management functions supporting SQL, for transaction-management facilities and for an SNA interface. For decision support, the functional characteristics might include requirements for access to structured data and for easy-to-use data-manipulation tools.

- Operational characteristics will include sizing information about peak and average transaction rates, disc accesses per transaction, database size, number of users and so forth. If the platform is likely to be replicated at several sites, the sizing information should be specified as a range, since this will have a bearing on the platforms that can be considered to meet these requirements (some platforms can be scaled up or down more easily than others). In addition, operational-performance characteristics for each class of support should be specified in terms of levels of security, reliability, availability, response times and so forth.

Often, the platform requirements for a class of support can be satisfied either by a centralised or a decentralised platform. High-serviceability online transaction-processing services with moderate-sized databases, for example, could be supported by a single mainframe, a small number of minicomputers or a larger number of local area network servers – or all of these. In determining the most appropriate type of platform, we believe it is useful to focus on the 'natural scope' of key data stores. Doing this will help to identify whether a particular class of support requires several databases (one for each country, for example) or a single 'global' database. In turn, this will determine the sizing information for each platform, and the machines that comprise it. In addition, databases are less easy to disperse than processing facilities, and in many cases, their locations and the scope of their use determine

Considering the 'natural scope' of databases can help to determine the most appropriate type of platform how many processing facilities are needed and where they should be located.

Often, one type of platform will be a natural fit with each type of database . . .

Often, the functional support requirements for a particular category of user are to access data stores, either databases or documents, whose locations are determined by components of the existing infrastructure that cannot easily be changed. Common cases include access via a central mainframe to operational databases and access to documents stored on existing local file servers. In other cases, it is helpful to determine the organisational or geographical scope of the information – that is, the part of the organisation or geographical territory to which the information relates. When viewed in this way, there is often one type of IT platform that is a natural fit with each database or type of database. Thus:

– For information that relates to the whole organisation (corporate accounts are an example), central mainframes are usually the obvious choice.

– For decision-support databases, a central database machine may well be suitable.

– For information that relates specifically to the business of one department, minicomputers installed in that department are often appropriate.

– For local information that relates to one workgroup, local-area-network-based database servers located with the workgroup may well be appropriate.

. . . but the obvious platform may not always be the best choice

This does not necessarily mean that information should be stored on the most 'obvious' type of IT platform. Remote-access requirements and economies of scale may make a more centralised solution appropriate, for example, or performance considerations and network costs may dictate a more distributed solution. However, considering the natural scope of data stores provides a useful starting point for determining the most appropriate type of platform.

Italimprese Finanziaria SpA (an Italian multinational group) determined its IT platform requirements by considering the natural scope of its databases. (This was done as part of a downsizing project aimed at giving more autonomy to the companies in the group and at distributing applications and databases in line with the systems management principles.) As a consequence, it has decided that most applications and databases will run on decentralised minicomputers (IBM AS400s were the chosen platform). No central database will be implemented except for a financial and cost-control system.

Once the IT platform requirements for all classes of support have been specified, it is necessary to consider the 'stakes in the ground' – those components of the existing infrastructure that can either be used to meet some of the platform requirements or that cannot be changed, at least in the medium term.

Review the existing infrastructure

The purpose of reviewing the existing infrastructure is to assess its suitability for meeting the IT platform requirements identified in the previous stage. In many cases, the need to define a technical architecture will have been prompted by the perception that the existing infrastructure is inadequate. The review will determine whether the organisation should persevere with its current infrastructure, and if not, will identify the nature and size of the barriers that have to be overcome in migrating to a new architecture.

In our experience, existing applications (sometimes known as legacy systems) are the most critical constraint on the development of a new technical architecture. Major applications can have a life cycle of up to 20 years, which means that applications developed to conform with the new architecture will often need to interface with existing systems.

Existing platforms may be able to satisfy the specified requirements

The reviewers should consider each of the IT-support classes identified earlier, and for each of the platforms currently used to provide that class of support, ask questions under the following headings:

- Is the existing platform satisfactory?
- Does the platform provide an adequate basis for the future?
- How dependent are the organisation's systems on the platform?

Figure 3.6 gives examples of the specific questions that need to be asked.

Figure 3.6 Each of the existing platforms should be reviewed to assess its suitability

What are the current levels of satisfaction with: – Support from the platform supplier? – Serviceability of the platform? – Security levels? – Usability of the applications running on the platform? – Ongoing support costs?
Can the unsatisfactory aspects of the platform be remedied by the supplier or are they due to fundamental shortcomings of the technology employed?
To what extent is the organisation 'locked in' to the platform? For example: – How large is the investment in systems that cannot readily be transferred to another platform? – Do the applications use platform-specific report writers, screen painters and so on? – What proportion of the applications code is specific to the platform? – Do the applications use proprietary applications programming interfaces?
Will the platform continue to meet the organisation's needs for the future? For example, what about: – Its fit with modern CASE tools? – Its ability to provide graphical user interfaces? – Its ability to support multimedia (image, voice, graphics)?
Is the platform tied into a proprietary standard with limited supplier investment?

Define possible systems scenarios

The next stage is to define the possible combinations of IT platforms (the systems scenarios) that will meet the platform requirements

Systems staff should resist the temptation to start with this stage

identified earlier and take account of existing infrastructure investments. Systems staff are often tempted to start with this stage because it enables them to consider the technically interesting questions associated with choosing suppliers, selecting products and setting standards. In our view, this is a grave mistake. Unless the earlier stages described above have been gone through, there is no guarantee that the chosen platforms will be able to support the required classes of service. It will also be easier to gain commitment to the new architecture because business managers and users will have been involved in the early stages of defining it.

There is a complex relationship between platform requirements and possible platforms

Again, this stage requires considerable technical expertise, because of the complex relationships between the IT platform requirements and the platforms (expressed as specific combinations of hardware and systems software) that can meet those requirements. There will, for example, be many platforms that can meet the requirements for one class of support (online transaction processing could be provided on an MVS platform, on a Vax platform, on a Tandem platform or on a variety of other platforms). It may also be possible for one platform to provide several classes of support – as with a minicomputer that is used to provide office systems and batch processing. It might also be necessary to provide a single class of support by two or more platforms – where different types of transaction processing are provided on separate platforms (financial systems on one platform, and manufacturing systems on another, for example). Other support classes, such as computer-aided design and electronic point-of-sale systems (EPOS) may also require separate platforms.

In addition, some parts of the existing infrastructure will have to be carried forward into the new architecture. The planners should therefore consider whether each class of support can be provided by one of the existing platforms. If it can, it will usually be preferable to retain the existing platform rather than introduce a new one (unless the benefits of a new platform are particularly great).

Possible scenarios are defined based on different combinations of products

Because of these complications, it is usually impractical to proceed in a phased, deductive manner to identify the optimum set of platforms that will meet all the platform requirements. Instead, several possible systems scenarios should be created, any one of which could meet all the requirements. Different scenarios will be based on different combinations of products, and can be used to test the feasibility of the solutions and to estimate the costs involved.

Ideally, between three and six scenarios should be defined. Fewer than three is unlikely to represent the range of possible solutions, and more than six is likely to confuse those who have to decide which platforms are to be included in the architecture. Limiting the number of scenarios to about six also keeps the amount of work required to a manageable level. Increasing the number above six is likely to result in scenarios that are only slightly different from those already under consideration.

The possible scenarios will often be suggested by the location and use of the data stores. However, each scenario should be distinctive, and together, they should cover the range of technical options. Thus, in most cases, the scenarios should include both minicomputer and client-server options for local processing, and both centralised and distributed options for major applications.

Each scenario should be described in terms both of the number and distribution of the machines and of the data stores and the software environments supported. IT support common to each scenario may either be omitted or described in a generic way. If, for instance, a PC word processing package is needed but there are no special requirements, all scenarios might describe this class of support as "WP package (for example, Microsoft Word)". The feasibility of migrating to each scenario should also be considered at this stage.

The three possible systems scenarios identified by one organisation are shown in Figure 3.7. The figure also shows that some of the classes of IT support will be satisfied by existing 'stakes in the ground'.

Figure 3.7 The possible systems scenarios represent the principal options for IT platforms

IT-support class	Scenarios: 1 Vax scenario	2 Unix scenario	3 PC LAN scenario	
Batch database	Bull	Bull	Bull	'Stakes in the ground'
High-volume transaction processing	Bull	Bull	Bull	
Telemetry	MicroVax + private network	MicroVax + private network	MicroVax + private network	
Office systems – Computer – User environment – User interface	Digital ALL-IN-1 Windows 3.0	Unix X-Desktop Motif	PC NetWare Windows 3.0	
EDI and external messaging	ALL-IN-1	Uniplex	Novell MHS	
Personal computing – Workstation operating system – Applications	MS-DOS Word, Excel	X-Windows + MS-DOS Word, Excel	MS-DOS Word, Excel	

Choose the best scenario

Once the possible systems scenarios have been identified, they should be compared to decide which should form the basis of the technical architecture. As well as being assessed on its ability to provide the required classes of support, each scenario should be evaluated against the identified requirements for availability, security, response times and transaction volumes (both average and peak). Its ability to survive disasters should also be evaluated. The requirements of applications developers and system operators, as well as users, should be included in the evaluation. It will also be necessary to consider how integration across the platforms is to be achieved.

The main evaluation criteria are relative cost, functionality and risk

The main evaluation criteria will be relative cost, functionality provided beyond that required to meet the essential requirements, and the degree of risk involved. Care must be taken to ensure consistency in evaluating the scenarios according to these criteria.

If, for instance, the additional cost of providing duplicated files has been included, the additional serviceability thus obtained, not that of the basic configuration, should be taken into account elsewhere in the assessment. Checklists of the factors to include in the evaluations are shown in Figure 3.8.

Figure 3.8 Systems scenarios should be evaluated on relative cost, functionality and risk

Assessing costs

Include everything, even if only approximately.

Be 'roughly right' rather than 'precisely wrong'.

Ignore small cost differences – the timeframes involved introduce too many uncertainties for the costs to be precise.

Include all life-cycle costs, including acquisition and ongoing support and upgrades.

Cost categories include staff, hardware and software procurement, network installation and operations, systems installation and testing, user training and support, and physical facilities (buildings, air conditioning, cooling and so on).

Assessing functionality

Will the end-user facilities be easy to use, thus reducing the need for user support and speeding up the learning process for users?

Are there systems management facilities readily available for the set of platforms comprising the scenario?

Will the features of the application-support functions (for example, a database management system with data-integrity conditions) reduce the development workload or help speed up the development schedule?

Are there development facilities in place, such as code libraries, data dictionaries and code generators?

Assessing risk

Is there a comparable IT infrastructure elsewhere that combines this particular set of platforms?

If not, does the organisation understand the risks involved in taking on an untried combination of platforms and is it willing to make the trade-offs involved?

What happens if the supplier does not meet the promised delivery dates for further releases of the platform with essential features needed for (say) compatibility with existing platforms?

Is there a possibility that the supplier will discontinue support for the chosen platform during the planned lifetime of the architecture?

Is there a development path for the chosen platforms?

Are the platforms capable of coping with future business changes?

Will the supplier be able to scale up the platform to meet future expansion requirements?

What if the standard underpinning the platform fails to achieve widespread acceptance in the future and support becomes difficult and expensive to obtain (for example, OS/2)?

No scenario will be ideal; it will be necessary to make trade-offs

No scenario will be best in all these aspects so it will be necessary to make trade-offs between the scenarios. Conventional comparative methods based on scores and weightings may, of course, be used, but we find these to be more helpful in clarifying the issues than in making the decision. Any trade-offs are, inevitably, a matter of managerial judgement and it is better to aim to make this explicit than to attempt to construct an evaluation scheme that avoids the need for such judgement. Such attempts invariably fail.

It can be helpful to refer back to the systems management principles at this stage. For example, one organisation with a large number of retail outlets applied its systems management principles in selecting the architecture for new systems, but as Figure 3.9 shows, it was not able to comply with them all.

Figure 3.9 The systems management principles can be used to guide the choice of technical architecture

A group of UK leisure companies (part of a multinational group of companies) commissioned a study of their complete applications portfolio, which in turn led to the development of a technical architecture. The group includes hotel and restaurant chains, leisure clubs, and retail outlets for wines and spirits. In total, there are more than 2,000 locations.

The systems management principles determined most of the choices, but the group agonised over whether to choose open or proprietary standards (open standards are the accepted direction for all the companies in the multinational organisation) before settling on proprietary standards for pragmatic reasons. The main systems management principles in the technical architecture and the outcome are summarised below.

Systems management principles	Outcome
Business activities will be controlled locally. Systems will provide corporate management with a means of assessing the performance of the retail business. Systems will provide central control of marketing and property management.	The accepted solution was a two-layer architecture, with electronic point-of-sale (EPOS) applications at the retail outlets and a mainframe at the centre. The local EPOS systems are supplemented by PC networks, where appropriate, for improved local management control. Portable PCs are used for mobile staff (field sales, order takers and so on). This solution provides corporate management with the necessary information and control, and local management with control of business transactions.
The new core system will be delivered quickly.	A fourth-generation applications-building tool with industry 'templates' was selected. The alternative – a series of packaged solutions for the core applications – was not practical because of the incompatible database management systems used by the range of packages needed.
A certain level of risk is acceptable in the development of the new applications but the level of risk in the infrastructure will be minimised.	An established supplier architecture was selected, with several reference sites of a similar scale.
Open systems will be selected wherever possible.	For reasons of risk management and the availability of a suitable applications-building tool, a proprietary platform was chosen.

The number of different platforms in the architecture should be considered carefully

In comparing the scenarios, the number of different platforms in each scenario, and ultimately in the architecture, should also be considered. This is often both a politically sensitive and a genuinely difficult task.

Minimising the number of platforms in the architecture will typically have the following advantages:

- Lower purchasing costs, arising from volume discounts.
- Lower training costs for both users and systems staff.

- Lower support costs because specialist skills are spread over a larger number of systems and users.
- Lower development costs for those applications that have to be provided on all platforms.
- Easier integration between applications, and greater flexibility.

Of course, the disadvantage of reducing the number of platforms must be set against these advantages: some applications will inevitably be provided on less suitable platforms, leading to lower benefits, higher costs, or both.

In practice, the optimum number of platforms will be determined by going over the possible choices several times, looking for overlaps in the platform requirements, or in the scenarios, that will make it possible to reduce the number of platforms. For some organisations, consideration of the number of platforms is bound up with the number of data centres they wish to operate. Many have recently consolidated several small mainframe data centres into fewer large ones. In doing so, they have achieved economies of scale in hardware costs and software licences, and have been able to reduce the number of operations staff.

Once the systems department has chosen the most appropriate systems scenarios, it is then time to start planning how to migrate to an infrastructure that conforms with the chosen architecture.

Develop a migration plan

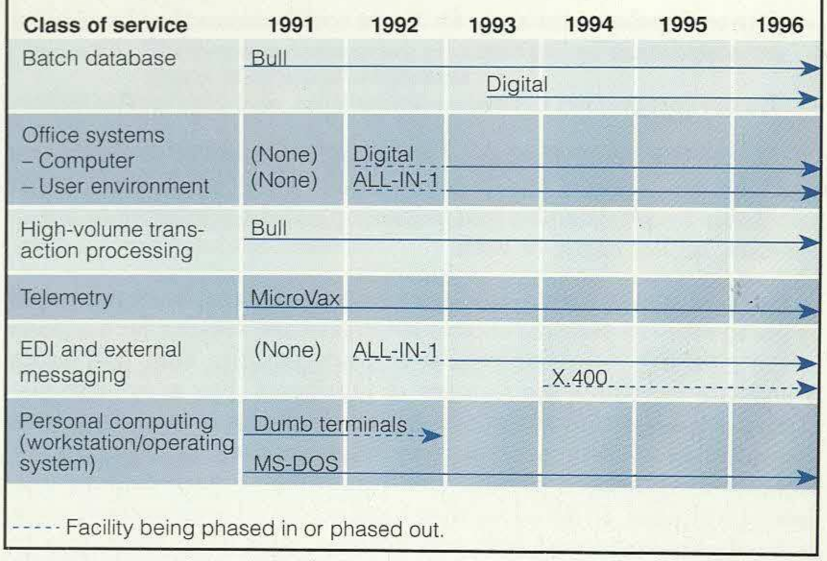
The migration schedule includes the facilities required to build the infrastructure

The technical architecture comprises a set of standards and policies – the actual benefits to the business come from its practical realisation as a working infrastructure. This means that it will be necessary to develop an implementation schedule, with commitments from all those responsible, to ensure that the migration to the new infrastructure happens sooner rather than later. The schedule will include the facilities (such as wide-area network gateways, EDI services and CASE tools) needed to build the infrastructure, plus training, conversion work and organisational changes. The installation plan, with the main development phases mapped out, will include the following elements:

- The main hardware, software and telecommunications components to be installed.
- The dates when the new platforms and networks will be in place.
- The staff numbers and skills required to develop, test, install and maintain the technical infrastructure.
- Any organisational changes required in the systems function.
- The contribution required from external suppliers of materials and services.
- The systems and user staff to be dedicated to training and support before, during and after implementation of the infrastructure.

An outline of such a plan, based on a public utility's migration schedule, is shown overleaf, in Figure 3.10.

Figure 3.10 The installation plan shows how an organisation will migrate to its new technical infrastructure



Review the architecture at regular intervals

The process of defining a technical architecture set out in this chapter is essentially a planning process. It is unrealistic, however, to view this planning process as a rational, analytical, once-off activity that results in a definitive 'blueprint' for the technical infrastructure. While a blueprint may be a useful device for selling the idea of the architecture to business managers, it can, at best, present only an approximate view of the future situation and should not be viewed as a definitive goal. Translating the blueprint into a technical infrastructure requires assumptions to be made about the future shape of the business and about future developments in technology. Inevitably, some of these assumptions will be wrong.

The technical architecture should therefore be reviewed at regular intervals (usually annually) in the light of changes in technology, standards, markets, regulation, international relations and other factors beyond the organisation's control. The review should also take account of any changes in management's objectives for the organisation and in the systems management principles. The general shape of the technical architecture should not change radically as a result of these regular reviews. After all, the whole purpose of a technical architecture is to provide the flexibility to cope with such changes. A radical rethink of the architecture should not be necessary for several years.

The review should take account of changes in management's objectives and the systems management principles

In this chapter, we have set out an approach to defining and reviewing a technical architecture and to creating a migration plan for translating it into a technical infrastructure. To achieve this, it will be necessary to justify the investment required to create the infrastructure and to set up procedures to ensure that the infrastructure continues to conform with the architecture.

Chapter 4

Justifying the investment and ensuring conformance

By itself, a technical architecture is nothing more than a set of documents containing rules, standards and plans. It becomes valuable only if it is adopted and investments are made in building a technical infrastructure. The two biggest obstacles to achieving the benefits of a technical architecture are justifying consequent investments in infrastructure and ensuring that the architectural rules are adhered to.

Justifying infrastructure investment requires business judgement

In most large organisations, an increasing proportion of the investment in IT is being spent on the technical infrastructure. Justifying this investment often creates a major problem, for two main reasons. First, the value of infrastructure investment derives from its role in facilitating the development and successful operation of subsequent applications. The benefits of the investment stem from these applications, rather than from the infrastructure itself, and while the size of the investment may be estimated with accuracy, the value of the application benefits may be very speculative. Second, the infrastructure is rarely limited to a single department or to one manager's area of responsibility. The ultimate benefits of the infrastructure investment accrue to the organisation as a whole, and are not always readily visible as improvements in the short-term performance of individual business units. Managers may therefore find it difficult to take an organisation-wide view of the value of the infrastructure, and so may be reluctant to sponsor the investment.

The benefits of infrastructure investments stem from applications, not the infrastructure itself

As we said in Report 75, *Getting Value from Information Technology*, corporate management must make a judgement on whether the expected benefits of the proposed infrastructure justify the investment. Admittedly, this is often difficult, but so is justifying expenditure on an advertising campaign, or on research, or on a new warehouse.

Business managers should avoid two extremes when considering IT investment proposals

We believe that managers should avoid two extremes when considering IT infrastructure-investment proposals. One is to accept the proposal without question, because technical experts say that the investment is essential. Clearly, this is an abdication of managerial responsibility. The other extreme is to insist that the whole cost of the infrastructure must be loaded onto the first application, since this will frequently destroy the case for that application. It will also open the door for unjustified applications designed to take advantage of 'free' use of the infrastructure.

The right approach is to make a management judgement based on the known cost of developing the infrastructure, compared with an estimate of the likely benefits of the applications portfolio that the infrastructure is required to support. Ideally, these benefits should be expressed in financial terms. At the very least, they should be quantified. If the investment is required to improve the performance of the existing infrastructure, rather than to support new applications, the judgement should be based on the strategic or operational value of the business benefits that will arise from that improvement.

In preparing the formal business case for investing in the technical infrastructure, the emphasis should therefore be on the future capabilities that the new infrastructure will provide. Similar difficulties exist in justifying factory-automation investments in the automobile industry. Writing in the January 1991 edition of *Communications of the ACM*, Eric K Clemons (Associate Professor of Decision Sciences and of Management at the Wharton School of the University of Pennsylvania) says, "The problems of evaluating investments in factory automation are illustrative [of the problems of evaluating IT investments]. This should not be viewed as new technology for making *today's* automobiles, but rather as new technology for making *tomorrow's* automobiles. This requires assumptions about [the future directions of the business]."

The business case for investing should be based on the future capabilities of the infrastructure

He suggests that similar arguments should be deployed in making the business case for investing in a technical infrastructure that conforms with the technical architecture. While there will be long-term benefits, such an infrastructure usually costs more in the short to medium term than one planned in an *ad hoc* way, because it requires that individual investments in the infrastructure be made in the light of a long-term vision and that their impact on other parts of the infrastructure be recognised. The technical architecture may also require 'enabling' investments to be made, with no direct payback. These might include interim solutions to link systems together (for example, wide-area network gateways or EDI services) or development processors and specialised CASE tools.

Figure 4.1 suggests the factors that should be considered when preparing the formal business case for investing in IT infrastructure.

Various techniques can be used to ensure conformance

The process of defining a technical architecture must not only produce a plan for a technical infrastructure. It must also build commitment to the architecture among the systems and business managers who will subsequently have to abide by it. Without this commitment, it will be impossible to ensure that the infrastructure is constructed and maintained so that it conforms with the architecture. The key to gaining this commitment is to involve users at all stages in the definition and implementation of the architecture.

Without commitment, it will be impossible to build the infrastructure so that it conforms with the architecture

Involving users

The benefits of a new technical architecture will be obtained only when the infrastructure is in place, together with applications that

Figure 4.1 Considering these factors when preparing the business case will make IT infrastructure investment proposals compare favourably with other IT-related investments

Uncertainty and risk

- Compare the infrastructure proposals with the consequences of doing nothing.
- Compare the options not just on cost and practicability, but also on the level of risk involved.
- Consider the risks under the headings of financial (Can we afford it?), technical (Can it be done?), project (Can the organisation do it?), functionality (Will it work in our environment? Is the environment changing too quickly?), and pragmatism (Will the external environment — users, customers, owners, regulatory bodies — accept the result?).

Finance

- Review the responsibilities for budgets and the sources of finance.
- Consider start-up funding or central funding for enabling sub-projects.

Tangible benefits

- Set out a series of business-related questions based on the expected benefits of building a technical infrastructure that conforms with the architecture (for example, improved flexibility, globalisation, open systems conformance). Include questions relating to the types of rewards that the business will receive from the expected benefits of the infrastructure. For example, will the marketplace reward greater flexibility with increased market share?
- Identify the technical benefits arising from a technical architecture. These include reduced technical complexity, increased technical flexibility, improved communications between sites, the ability to operate common, or even global, systems, easier interoperability, and the ability to incorporate new technologies.

exploit its strengths. This requires active cooperation from many systems staff and user managers. The architecture should not, therefore, be devised by systems staff working in isolation and then imposed on a reluctant user community. Instead, users and user managers should be involved at several stages in the process of defining the architecture. By involving all the interested parties, it will be possible to create shared understandings and gain general support for the architecture. User managers should be involved in the requirements-definition stage so that they can confirm the business objectives and critical IT-support requirements. They should also be involved in testing the acceptability of products, particularly workstations and presentation software.

Users and their managers should be involved at several stages in defining the architecture

A German manufacturer of household products, for example, adopted a consensus-oriented approach to the definition of its technical architecture, focusing on the need for change. The requirements for moving towards a coordinated technical architecture across Europe were prepared and agreed in a series of workshops. User managers participated in the workshops, thus ensuring that full account was taken of their business needs, and that a consensus of the priorities was arrived at. The need for the new architecture was therefore seen to be directly linked to supporting the business needs, and the ground was laid for ensuring future compliance.

In many organisations, however, the development of a consensus is impeded by vested interests and entrenched positions. In these situations, it is often valuable to involve an independent consultancy that can provide briefings on industry trends, and information about products and architectures, and that can help to structure the debate and present critical information.

A cross-section of users should be involved in any testing of products, particularly for those related to workstations and presentation software. Figure 4.2 explains how this approach was adopted by a public utility whose systems department has a reputation for paying great attention to its users at all levels. It has experienced few problems in gaining compliance with the new technical architecture, which is focused on two main technologies – information presentation and networking.

Users must be involved in choosing workstations and presentation software

Figure 4.2 By involving its users in product testing, a public utility has experienced few difficulties in gaining conformance to its technical architecture

The business aim was to move away from functional demarcation at district level, and to enable one staff member to respond effectively to customers' calls regardless of the function (engineering, sales, service and so on) involved. Thus, data relating to all functions had to be transparently available at the workstation, with operations simple and fast enough to be used while also using the telephone. The systems department involved the users in the districts in the design of the user-interface systems at workstation level. This was an essential factor in gaining acceptance of the change to the new architecture. It became clear in debating the results with users that there were two areas of risk in pursuing this as a long-term technical strategy:

- Although VT 200-style terminals (the standard dumb terminal for the Vax) would be suitable for a Vax-based approach, they would be inappropriate for a Digital open environment. How inappropriate would depend on the speed of migration to Unix.
- User satisfaction with a central-processor-based system was likely to be limited – potential issues concerned response times, the possible user benefits from using PC software with its increased facilities, and the disadvantages of a mixed PC/dumb-terminal environment.

Having decided upon a PC-based workstation policy for all its end users, this utility followed these steps to involve its users in the implementation:

- A user survey provided an understanding of current usage of technology and expectations of the system.
- An education programme was put in place to explain to users what would be provided and on what timescale.
- The existing base of hardware and software was established in detail and the new requirements spelled out.
- Software requirements and test plans (to demonstrate the integrity of each installation) were drawn up.
- Central administration procedures were set up for hardware and software (for example, maintenance, filing conventions, security and so on).
- Once the system was delivered, tested and installed to standards, a training programme was set up.

During our research, we identified a variety of other methods used by different organisations to gain conformance to their technical architectures. We give some examples below.

Establishing a high-level review body

Some organisations have established a high-level body that monitors IT strategic plans and capital expenditure, so that independent developments that threaten the integrity of the technical architecture will come to light at an early stage. One UK local-government authority, for example, successfully uses this method to maintain control of its technical architecture in an environment where much of the responsibility for systems is devolved to the 13 departments:

- There is a corporate IT steering group that has declared the mission and objectives.
- Each department has its own systems function, and the departmental IT manager develops the IT strategy.
- This strategy cannot be implemented until it has been approved by the corporate IT steering group.
- The corporate IT steering group meets four times a year, and is chaired by the deputy county treasurer. High-level managers from all 13 departments are involved, and the corporate IT manager (the county computer officer) is adviser to the group.
- The group is primarily a steering committee, but it has strong influence over investments in IT, as the capital-programme working group requires it to comment on, and endorse, the capital plans and budgets for IT before approving them.

A high-level review body is particularly important in a devolved systems organisation

The role for a high-level review body is particularly important in a devolved systems organisation. In Report 81, *Managing the Devolution of Systems Responsibilities*, we spelt out the need for a group coordinating committee that agrees to and mandates groupwide IT policies. These policies include compatibility requirements, which define the requirements for the technical architecture required to ensure that the organisation does not disintegrate into incompatible 'islands of automation'.

Providing senior management with IT education

Some Foundation members have established senior management education programmes with the aim of communicating the rationale for conforming with the technical architecture and the need for the business to make changes to comply with it. One of the US subsidiaries of a major European chemicals manufacturer, for example, paid considerable attention to senior management education when its systems department decided to rationalise under one architecture the disparate computer systems inherited from a series of acquisitions. The project team considered it essential that top management thoroughly understood the need for a corporate technical architecture. Team members set about explaining to senior managers the implications of the changes proposed, so that the team could count on high-level support if it encountered resistance from the businesses. They also needed to convince top management that there were serious implications for the business if the *status quo* were maintained. Much time and effort was therefore spent communicating the rationale for change to the executive vice-president and the chief executive officer, and a programme of seminars for senior managers was introduced to explain the business case.

This approach has also been very advantageous in an Italian manufacturer of automation products. This company is moving from a centralised mainframe architecture (plus some standalone minicomputers) to an integrated distributed architecture. All users will be directly affected by this change, which will give them access to more facilities and move data and applications nearer to them. The company has long taken special care to provide IT training to its users, and as a result, the user population is particularly well informed. Because of this, no problems are being experienced in obtaining user approval for the change to the new architecture.

Demonstrating support for business objectives

Another way of encouraging conformance to the technical architecture is to ensure that the architecture is seen by users to be designed directly to support business objectives that have been agreed at a high level and published within the company. These objectives frequently require a major change in a business process, or in a major common application, for which the particular technical architecture is a prerequisite.

Often, a particular technical architecture is a prerequisite for a major change in a business process

A chemical company we spoke to is currently developing global systems for its total order chain and for global product codes. Such systems obviously require a unified technical architecture. The company achieves compliance with the architecture as a by-product of implementing major corporate projects that are sponsored by business management in corporate headquarters. Line managers in the various European companies are informed of a central decision to implement a major global project, and are therefore receptive to exhortations from local systems staff to install the required hardware and software in conformance with the technical architecture.

For example, a current project will enable product, customer and other codes to be issued from a single source and distributed to various systems around the world. With such an application in the pipeline, line managers appreciate the need to comply with the architecture. If they do not comply, they will not be able to operate their businesses.

Providing support for cooperative working

User demand for conformity with the technical architecture can sometimes be stimulated by promoting the concepts of cooperative working and the advantages of exchanging text, graphics and spreadsheets.

A pharmaceutical company, for example, has found this to be a more effective approach than direct mandating of standards by the systems department. This company was finding it difficult to standardise its office automation products throughout its European offices. The central systems department had been exerting pressure for conformity by refusing to support non-standard products, but this had been only partially successful. Now, the business is starting to press for conformity because users are changing their ways of working towards cooperative methods that involve the exchange of text, graphics, spreadsheets and financial models. To achieve compliance, systems staff are explaining how the technical architecture can enable the exchanges to take place, and changes in working practices are being introduced that require cooperation between users and hence raise their awareness of a need for a unified technical infrastructure.

Anticipating users' needs

Having viable alternatives available is an effective way of gaining conformance to the architecture

Another effective way of gaining conformance to the technical architecture is to anticipate users' moves to install non-standard hardware or software, and have viable alternatives that comply with the architecture ready to offer. This approach is strongly advocated by a UK local-government authority, which maintains its technical research and product-selection activities at a high level in the organisation. This ensures that, if a departmental initiative is rejected, an alternative product conforming with the architecture is immediately available.

An international airline has a similar philosophy, and takes pride in anticipating new requirements before the users have formulated them. As an example, this airline was one of the first organisations to standardise on the use of FOCUS for accessing central databases.

Demonstrating the applications-delivery advantages

Business managers will be much more disposed to comply with the technical architecture if it is presented to them in terms that clearly demonstrate that it will enable critical applications to be delivered within the desired timescales, at reasonable cost and with low risk. In a UK charter airline, for example, this was the main factor that ensured compliance. The systems director had no difficulty 'selling' the need for the technical architecture to his fellow directors, because they were all struggling with information problems in and across their divisions, and understood that a unified architecture was a necessary element in improving the situation.

However, the new architecture would not only enable improved information to be available, with its not-always-tangible benefits, but would reduce clerical costs in the divisions. This would be achieved by eliminating the effort, and hence the headcount, currently involved in re-organising data for input to technically incompatible systems. Reduced headcount is a powerful argument in a company operating in a highly competitive or resource-limited environment.

Limiting the support provided

Offering responsive, high-quality support to all authorised products, and no support to products outside the technical architecture (unless there is no viable authorised alternative) can be an effective way of gaining conformance with the architecture. One of Australia's major retail banks believes that this approach will persuade users to migrate from their well established systems to ones that conform with the new architecture. It has introduced OS/2 LAN Manager as the standard on which the new branch-automation systems will be based. However, there are many current users on the previous standard (Novell NetWare) and, in due course, they will have to convert. Until they do, access to mainframe data will be made progressively more difficult and more expensive. The systems department is expecting to overcome resistance by offering users consistent levels of support and high-quality training, documentation and assistance. The danger of this type of policy is that users will seek support for non-standard products from outside the organisation – thus increasing their isolation.

The danger of limiting support is that users will seek external help

Approving purchases

Another effective way of ensuring compliance with the technical architecture is to establish mandatory approval by the systems department for all purchases of IT products, including equipment and software packages, and ensure that the best discounts are obtained. For example, an Australian oil company currently has a distributed processing architecture, but is planning to change to a client-server structure accessing central databases. In either case, the systems department carries out all purchasing centrally, recovering the charges from users. This central buying power is used to ensure that products conform to the standards. The systems director's dramatic comment – "If users play around, they are zapped" – may prove to be more than mere words: he is currently contemplating cutting off 150 users where a network bridge has been installed without approval.

Central purchasing ensures the best discounts and conformance to standards

In our experience, such autocratic authority is unusual and does not often succeed. It is usually more effective to ensure that users understand the reasons for corporate standards and their overall benefits, and to sugar the pill for users by using corporate buying power to negotiate attractive discounts on hardware and software.

Publicising corporate data

The benefits of a technical architecture can be promoted by publicising the availability of corporate data, and the advantages of being able to access it via standard software. This has been a critical factor for a major chemical company in achieving compliance with its technical architecture. The systems department explains to users that the hardware, software and applications will enable them to benefit from the sharing of resources. The new architecture will enable them to access data and to share applications. This is important for this company, because the regional groups need access to the global cross-functional-processes databases. The company's level of integration and cross-functional working requires compatible systems.

Providing responsive development services

In some organisations, conformance with the technical architecture can be encouraged by emphasising that it will be able to support centralised systems development and maintenance, in either one or several centres of excellence, and forbid local development (apart from end-user computing). This policy has been adopted by an Italian conglomerate, which is introducing a new technical architecture to support downsizing. It foresees a major problem in persuading users to adhere to standards, and in avoiding autonomous development of new applications. The systems department has therefore decided to adopt the following policies:

Centralised development and maintenance encourages conformance with the technical architecture

- Every division will have a small systems staff, but to manage only current operations. No development staff will be moved out to the divisions.

- All development work will be carried out by headquarters' staff; they will develop a common set of applications and database structures that will be distributed to the divisions, and every request for modifications, or for new functions specific to a division, will be handled by the central development staff.

– The installation of any kind of development tool on the operating companies' machines will be forbidden. If any operating company violates this rule, no support whatsoever will be given to local developments.

Identifying non-compliance

Systems audits are an obvious way of ensuring compliance

The obvious way of ensuring compliance with the technical architecture is to have an effective internal systems-audit function, carried out by an audit group separate from the systems department. This group will report on non-compliance and put independent pressure on the business managers to conform. This has proved to be a powerful method of ensuring conformance with the technical architecture in an Australian bank. Systems audits are combined with physical inspections. If the auditors issue an adverse report, the line manager is required to take action.

In general, however, it is easier to get compliance to technical architectures in organisations where the culture encourages it on all fronts. This may come from a strongly centralised management structure, or an authoritarian management style, or from a subtler system of rewards for conformist behaviour. In one company that has little difficulty gaining compliance to the technical architecture, we were told that the general creed in the organisation was summed up by the Japanese proverb, "the nail that sticks out gets hammered".

Education and involvement are the keys to achieving the full benefits from the technical architecture

The common thread running through the above examples is the need to gain understanding throughout the business that conformance to the architecture will be of long-term benefit to the organisation as a whole. In our experience, most business-unit managers and staff are prepared to make compromises, provided they have a full understanding of the potential benefits. This can be achieved by education and involvement. We believe that these are more effective than prohibition and policing.

Appendix

Technical architecture in practice – the experience of a multinational plastics business

Wavin is a Dutch multinational group with 35 factories in 12 countries and a turnover of Fl1.8 billion ($900 million). It operates in the plastics sector, specialising in extruded plastic pipe for construction, agriculture and utilities. It also manufactures plastic film and packaging, U-PVC extrusions, chemical piping, and production machinery for the plastics sector. The companies in the group are located in several countries, their operations vary in scale and the basic product-development, manufacturing and sales cycles of the various product lines can be quite different. This poses a challenge for those defining the technical architecture. They cannot simply enforce similar solutions across the companies but must try to meet individual company needs, while accommodating future growth both through acquisition in new markets in southern and eastern Europe, and through the organic growth of the existing businesses.

Identifying user categories and their requirements

Analysis of the companies in the group indicated that user requirements should be categorised by business type and scale. For example, the extruded pipe businesses are largely make-for-stock operations, while the others concentrate on make-to-order. This means that the customer-servicing requirements are quite different, as are the timings of the production cycles. The sizes of the companies also vary. In terms of turnover, they are categorised as 'large' and 'small', although small businesses might grow into large ones. The possibility of acquisitions has to be taken into account as well.

The differences in size have implications for the complexity of the applications solutions and also for the level of integration needed. For example, smaller companies typically operate in smaller and more volatile markets. It is more critical for them to have access to current stock data, for example, than (say) complex integrated planning methods.

In practice, five user categories were identified for the purposes of the technical architecture (see Figure A.1). Specific user requirements were defined for such aspects as:

- Level of integration of functions, such as production planning, financial accounting, stock control, sales order processing and so on.
- User interfaces.
- Transaction volumes.

Figure A.1 Users at Wavin fall into five categories

Size of business			
Large	Users with a need for improved customer links and high transaction rates	Need for high levels of integration across business functions	
Small	Less complex needs; need for scalability and simple, cost-effective IT platforms	Need for high levels of integration across business functions	Self-contained businesses, with a need for flexibility
	Make-for-stock businesses	Low-margin make-to-order businesses	High-margin make-to-order businesses
		Type of business	

– Cross-business data aggregation. (Using Peter Keen's 'reach' and 'range' concepts, the level of integration (reach) needed between businesses is moderate, but the range needs are significant. The implication is that EDI-style links will suffice to meet the needs of the strategic business units.)

Defining IT platform requirements

There was a broad set of management requirements, equivalent to our systems management principles, driving the IT platform requirements. Specifically, these were concerned with:

– Providing appropriately scaled hardware, by company.

– Providing fast delivery of systems for specific business problems.

– Delivering simple, low-cost systems for smaller businesses.

– Establishing standards for data to ensure efficient communications and consistent business management.

– Containing systems costs by optimising hardware configurations and making the best use of alternative sources of supply.

In the case of Wavin, these kinds of management requirements led to serious consideration of how packaged solutions could be deployed as the core of the systems strategy, given that they appeared to provide a path to rapid implementation for the smaller businesses. This question was complicated by the existing infrastructure, however, which was based on five IBM mainframe computers located in five countries and linked by a private network – not a hardware environment conducive to packages for smaller businesses. On the other hand, access by users to the systems was satisfactory, supported by the network and a combination of PCs and terminals throughout the group.

In short, the platform decision was dominated by the installed mainframe investment and the need to evolve to a more diverse architecture that would provide the ability to exploit packages, support common solutions within categories of businesses, and provide scalability in support of organic growth and acquisition.

Choosing the preferred solution

The cost-containment and scalability requirements suggested that, as a first step, there should be a shift towards minicomputers and some consolidation of the mainframe operations (computer processing was the main cost driver for the group's systems expenditure). The scenarios were developed on this principle, and represented a range varying from mainframe only to minicomputer only. The intermediate options were a mixture of the two. Specific products considered at this level of analysis were the IBM range of mainframes and mid-range computers, and the operating systems choices associated with these. Unix was considered as a cost-effective approach for the smaller companies.

Migrating to the new architecture

The decision was made to progress rapidly towards a minicomputer-based architecture relying on Unix. Cost and scalability were the primary motives. New development is largely concentrated on this platform. Migration plans were drawn up showing the directions and timings of the changes, and working parties were set up to consider the detailed implications for the organisation and the existing data centres and to work on the data standards. Although consolidation of the mainframe centres was initially considered, it proved possible to reduce considerably the costs of all the existing installations – in one case, by purchasing the mainframe for a nominal price, in another by replacing it with the latest air-cooled model. The mainframes will remain in place for some time.

The Butler Cox Foundation

The Butler Cox Foundation is a service for senior managers responsible for information management in major enterprises. It provides insight and guidance to help them to manage information systems and technology more effectively for the benefit of their organisations.

The Foundation carries out a programme of syndicated research that focuses on the business implications of information systems, and on the management of the information systems function, rather than on the technology itself. It distributes a range of publications to its members that includes research reports, management summaries, directors' briefings and position papers. It also arranges events at which members can meet and exchange views, such as conferences, management briefings, research reviews, study tours and specialist forums.

The Butler Cox Foundation is one of the services provided by CSC Index. CSC Index is an international consulting group specialising in information technology, organisational development and business reengineering. Its services include management consulting, applied research and education.

Membership of the Foundation

The Foundation is the world's leading programme of its type. The majority of subscribers are large organisations seeking to exploit to the full the most recent developments in information technology. The membership is international, with more than 450 organisations from over 20 countries, drawn from all sectors of commerce, industry and government. This gives the Foundation a unique capability to identify and communicate 'best practice' between industry sectors, between countries, and between information technology suppliers and users.

Benefits of membership

The list of members establishes the Foundation as the largest and most prestigious 'club' for systems managers anywhere in the world. Members have commented on the following benefits:

— The publications are terse, thought-provoking, informative and easy to read. They deliver a lot of messages in a minimum of precious reading time.

— The events combine access to the world's leading thinkers and practitioners with the opportunity to meet and exchange views with professional counterparts from different industries and countries.

— The Foundation represents a network of systems practitioners, with the power to connect individuals with common concerns.

Combined with the manager's own creativity and business knowledge, membership of the Foundation contributes to managerial success.

Recent research reports

61 Competitive-Edge Applications: Myths and Reality
62 Communications Infrastructure for Buildings
63 The Future of the Personal Workstation
64 Managing the Evolution of Corporate Databases
65 Network Management
66 Marketing the Systems Department
67 Computer-Aided Software Engineering (CASE)
68 Mobile Communications
69 Software Strategy
70 Electronic Document Management
71 Staffing the Systems Function
72 Managing Multivendor Environments
73 Emerging Technologies: Annual Review for Managers
74 The Future of System Development Tools
75 Getting Value from Information Technology
76 Systems Security
77 Electronic Marketplaces
78 New Telecommunications Services
79 The Role of Information Technology in Transforming the Business
80 Workstation Networks: A Technology Review for Managers
81 Managing the Devolution of Systems Responsibilities
82 The Future of Electronic Mail
83 Managing Technical Architecture

Recent position papers and directors' briefings

The Changing Information Industry: An Investment Banker's View
A Progress Report on New Technologies
Hypertext
1992: An Avoidable Crisis
Managing Information Systems in a Decentralised Business
Pan-European Communications: Threats and Opportunities
Information Centres in the 1990s
Open Systems
Computer Support for Cooperative Work
Outsourcing Information Systems Services
IT in a Cold Climate
Object Orientation

Forthcoming research reports

Downsizing — An Escape from Yesterday's Systems
Visual Information Technology
Strategic Alignment

Butler Cox plc
Butler Cox House, 12 Bloomsbury Square,
London WC1A 2LL, England
☎ (071) 831 0101, Telex 8813717 BUTCOX G
Fax (071) 831 6250

Belgium and the Netherlands
Butler Cox Benelux bv
Prins Hendriklaan 52
1075 BE Amsterdam, The Netherlands
☎ (020) 6 75 51 11, Fax (020) 6 75 53 31

France
Butler Cox SARL
Tour Akzo, 164 Rue Ambroise Croizat,
93204 St Denis-Cédex 1, France
☎ (1) 48.20.61.64, Télécopieur (1) 48.20.72.58

Germany, Austria and Switzerland
Butler Cox GmbH
Richard-Wagner-Str. 13, 8000 München 2, Germany
☎ (089) 5 23 40 01, Fax (089) 5 23 35 15

Australia, New Zealand and South-east Asia
Mr J Cooper
Butler Cox Foundation
Level 10, 70 Pitt Street, Sydney, NSW 2000, Australia
☎ (02) 223 6922, Fax (02) 223 6997

Finland
TT-Innovation Oy
Sinikalliontie 5, 02630 Espoo, Finland
☎ (90) 358 0502 731, Fax (90) 358 05022 682

Ireland
SD Consulting
8 Clanwilliam Square, Dublin 2, Ireland
☎ (01) 764701, Fax (01) 767945

Italy
RSO SpA
Via Leopardi 1, 20123 Milano, Italy
☎ (02) 720 00 583, Fax (02) 86 45 07 20

Scandinavia
Butler Cox Foundation Scandinavia AB
Jungfrudansen 21, Box 4040, 171 04 Solna, Sweden
☎ (08) 705 83 60, Fax (08) 730 15 67

Spain and Portugal
T Network SA
Núñez Morgado 3-6°b, 28036 Madrid, Spain
☎ (91) 733 9910, Fax (91) 314 3198