Open for Business

Change Management through Open Systems

Edited by
Christopher Jacobs

The X/Open Company Limited
February 1994

Published by X/Open Company Ltd., U.K.

Any comments relating to the material contained in this document may be submitted to X/Open at:

X/Open Company Limited
Apex Plaza, Forbury Road
Reading, Berkshire, RG1 1AX
United Kingdom

or by Electronic Mail to: XoPubs@xopen.co.uk

Contents

Chapters

List of Figures

List of Tables

Acknowledgements

X/Open acknowledges with thanks permission to reproduce *Realising the Value of Open Systems* by Paul A. Strassmann, and the use of material developed for MD Computing magazine by Peter C. Bauer.

The Business Value of Open Information Systems

A fundamental assumption underlies this book — information technology can and should contribute in a major way to the success of business and government. That this has not necessarily been everyone's experience to date is not surprising; that organisations have continued to make major investments in technology that have not contributed to their success, is.

There are many examples of enterprises, private and public, that have been able to implement information systems that contribute massively to their continuing success. There are also many organisations that have found themselves trapped in a rigid structure in which information systems, far from supporting the organisation in a flexible manner, act more like a body cast. In these environments, the flow of information is actually impeded by the very systems intended to promote movement. This has resulted in information hoarding: ''I've got what I need and I'm keeping it''; in lost business opportunities: ''If I had known about that service I'd have told the client about it''; and in frustration at all levels in the organisation: ''Why can't I send this information directly to Dieter in the Berlin office?''.

Openness

Over the last ten years, an increasing number of enterprises have recognised the need to democratise the way they operate, allowing staff in much greater numbers to take on-the-spot decisions that would previously have been referred up the hierarchy. Practical responsibility for key areas, such as customer satisfaction or service and product quality, has been broadened dramatically. While the command economies of Eastern Europe were undergoing fundamental changes, the centralised management style of Western organisations was facing similar challenges. In both cases, new decision-makers need access to all relevant information. Yet, many are working in information environments, manual or automated, that reflect the earlier centralised, hierarchical world. What is being asked, is that people practice open management with closed information systems.

Consequently, there is a corollary to our fundamental assumption — openness is an indivisible characteristic that runs throughout all aspects of the enterprise. Open, or flexible, management structures require open information systems.

What do we mean by ‘‘open systems’’?

While there are many, slightly differing, interpretations of the term, we will define an open system as one which conforms to agreed standards and is available from more than one independent supplier. In this context, a system may be a complete computer or communications system, or a discrete hardware or software component within such a system. The key factors, however, are that the system or component conforms to a set of international or industry standards and that it is available from multiple sources. Both criteria need to be met to fulfill our definition of an open system.

Themes

In developing this book, X/Open called upon a number of leading contributors to the information technology industry to share their views on what makes for successful implementation of information systems. The contributors are a balance of practitioners who work with a range of user and vendor organisations (Norton, Strassmann and Jacobs) and senior information executives currently working within a single user organisation (McCorkell, Bauer and Schmidt). They also represent a balance between public sector experience (McCorkell, Strassmann and Norton) and private sector knowledge, and between supplier backgrounds (Strassmann, Jacobs and Schmidt) and user backgrounds.

None come as evangelists for open systems, but all seek to place such systems in the context of the role of information technology within organisations. If they now support open systems, this support derives from considerable experience within a variety of information systems environments, proprietary and open.

A number of common themes can be discerned in their contributions.

Firstly, introducing information technology has to be seen as one component, albeit a key one, in the process of introducing change into the organisation. Norton defines this succinctly with the equation:

Success = Technology Change + Organisational Change

Strassmann, in his contribution, avers that information technology is there to provide an infrastructure that allows the enterprise to accommodate organisational change in a graceful manner. Jacobs, in both of his pieces, picks up this theme, emphasising the need to ensure the *direct* involvement of all groups in the enterprise in the development of the initial planning if success is to ensue. Continued dialogue between IT management and executive management is also crucial if the implementation of new, open systems-based technologies is to focus on agreed priorities. McCorkell contributes a case study of how political change required organisational changes that in turn forced fundamental technological change in a major governmental department.

The second theme refers to the requirement for an overall architecture or conceptual structure for managing the technology so that it aligns with the needs of the business. *Ad hoc* acquisitions of technology, whether open systems-based or not, do not necessarily contribute to the flexibility which open organisations require. As Jacobs shows in his second contribution, determining the payback from the move to open systems requires that you look wider than a single procurement and evaluate the tangible benefits of organisational flexibility. Both Schmidt and McCorkell demonstrate the evolution of such company-wide architectures over a period of years as the organisation and the technology come into alignment. Strassmann goes further in calling for enterprises to be conscious of making fundamental governance decisions as they consider how to make information available to various parts of the organisation.

The third theme concerns the relationship between supplier and user organisations. Having developed the blueprint for the desired information systems structure and its relationship to the organisation, it is important that the information systems organisation's relationship with the suppliers of its key technologies is established on the correct basis. Schmidt and Bauer clearly lay out a number of areas where suppliers can vitiate many of the organisational gains from open systems, either through lack of understanding of the problems or through unwillingness to commit to their resolution. McCorkell, similarly, is quite explicit in calling for the development of strategic relationships with a small number of suppliers — in return for a commitment by the user organisation to buy from a particular group of vendors, they must commit to work with the user to ensure that the technology works within the user's distributed, multi-vendor environment. Bauer adds into this mix the inclusion of strategic software and service suppliers as well as hardware and system software vendors. Norton argues that, as IT professionals, vendor, user, systems integrator and consulting staff, all have a responsibility to face the issue of accountability for the success of the businesses in which and with which they work.

If there is a fourth theme, it is this issue of accountability for results. As professionals, IT managers in successful organisations must, and do, take clear responsibility for delivering the benefits of information technology in their enterprises.

Benefits

As you will see, there is a pretty clear consensus on what the benefits of open systems are:

- Flexibility to respond rapidly to changes in business needs. This includes the ability to add new links to external organisations, whether suppliers, customers or other business partners, through the implementation of standards-based communication links. It means the rapid deployment and distribution of new applications through the use of open development tools; and it is support for operational consolidation through the ability to move applications from one system to another without major effort.
- Preservation of what Strassmann calls the ‘‘organisation’s permanent information technology assets’’. These are the organisation’s investment in end-user training and experience and in the stored information about how the organisation actually operates.
- Cost reductions through the availability of multiple sources for hardware, software and application development tools. While these cost improvements may not necessarily appear during the initial move to open systems, reductions in subsequent capital expenditure and in ongoing operational costs can be significant. As a number of the contributors point out, lower costs may not be the deciding factor in the move to open systems but they certainly add to the attractions!

We hope that you will gain new insight from the contributions that follow and that you will find things to both challenge and help you.

Executive Expectations of Information Technology

Christopher Jacobs

The shape of open systems is increasingly determined by users of IT rather than by the vendors. But the ''users'' concerned are usually themselves IT professionals, and their view of organisational needs differs from those of their non-IT colleagues. Surveying user views shows that CEOs are more concerned with systems and technologies that support innovation and new development rather than those assisting operational efficiency which is the focus of their technical staff.

Determining Business Requirements

As information technology has moved more directly into the centre of organisational life — a position both supported by, and driving, the development of open systems — the question arises as to who determines what technology will be standardised. How are priorities for work on particular technologies decided and who is involved in that decision?

For most of the fifteen-year history of open systems, the primary force behind efforts to develop standards, both formal and industry-defined, has been the vendor community, with some assistance from government and the military in the formal standards arena. Vendors, for their own competitive reasons, saw the need for, and the value to be gained from, open systems. Consequently, they assigned resources to create standards for the technologies that they considered important. The major role of the user community was to buy the products based upon those standards. But the world is changing.

Organisations that employ IT, rather than those who manufacture and sell IT products, are demanding a much more significant role in the definition of the standards on which those products are based; or, at least, in determining the priorities for the business issues that open systems products are required to solve.

It is clear that users do not want to participate in the way that the supply-side traditionally has; discussing the detailed technical requirements that will be met by a given specification. They want to change the tenor of the discussion and to contribute what they know: the business needs that the specification must address.

This is a fundamental change in the genesis of standards, and has led a number of the key industry standards organisations to change the designation of the groups representing the non-supply side from ‘‘User Council’’ to Requirements Board or Committee. One such organisation is X/Open, whose Open Systems Requirements Board (OSRB) is responsible for collecting and prioritising business requirements for additions to the panoply of available standards. Their work is intended to influence not only X/Open in its technical work but also the wider open systems world.

A new group established early in 1993, a key part of its role is the management of what is becoming an annual survey of user views of key business needs and of the technologies that are required to support the operation of the business. This quantitative survey of IT and executive management throughout the world complements the qualitative work that the OSRB undertakes in developing detailed requirements for IT standardisation.

Xtra ’93 Survey

For the 1993 survey, 9,370 questionnaire were distributed to users around the world. Though designed in English, the questionnaires were translated into Chinese, French, German, Japanese and Spanish in order to obtain as wide a spread of responses as possible. By the September 1993 deadline, slightly more than 8% of the questionnaires were returned for analysis, representing responses from 40 countries on six continents.

As might have been expected, the largest group of respondents were IT directors or IT department staff from medium to large enterprises. In fact, the average size of respondents’ enterprises was $2 billion, with an external IT budget of around $60 million. These people are, in the main, the same ones who represent the business community within open systems forums. Multi-million dollar companies and government agencies can and do maintain IT organisations large enough to be able to participate in industry groups in a meaningful way. Such participation requires a significant commitment of time from technically competent staff, and the availability of travel budget for attendance at meetings in many different locations in the world. Small enterprises often do not have more than one, if they have any, technical staff, and certainly cannot support overseas trips that are not revenue generating. The omission of their views potentially excludes the requirements of the largest number of enterprises in many developed countries, and the enterprises that represent the fastest growing elements in the economy.

However, this survey does give us the opportunity to capture the views of a representative number of small enterprises and to contrast them with the views expressed by their colleagues from the large organisations. In addition, close to 20% of the respondents to the 1993 survey described themselves as being at the most senior executive level in their organisations — Chairman, President, CEO, CFO or equivalent. Clearly, a number of these will be from the smaller enterprises, but not all. In any event, the existence of responses from this group does enable us to evaluate differences in attitudes to the relationship between business needs and technology held by executive management and by IT management.

Key Business Objectives

The first group of questions related to business objectives and the key information systems that enable the enterprise to achieve those objectives. Respondents were asked to identify their organisation's most important business objective over the next three years.

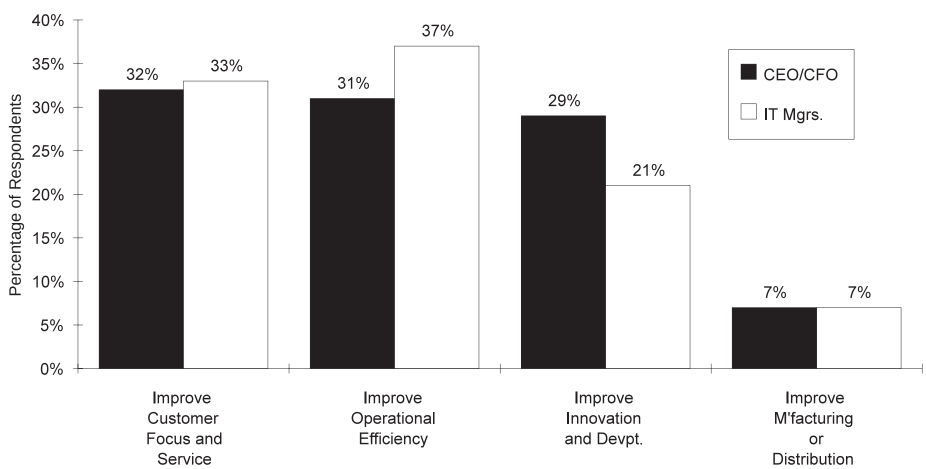

Figure 2-1 Most Important Business Objectives by Position

The views of general executive management and IT managers differ only slightly on the single most important business objective, with Improve Operational Efficiency being the priority for IT managers (37%), while executive management favoured Improve Customer Focus and Service by a small margin (32% to 31%) over Improve Operational Efficiency. However, isolating the responses specifically from those describing themselves as Chairman, President or CEO of their organisation shows a significant number choosing Improve Innovation and Development (41%) and many fewer selecting Improve Operational Efficiency (23%). It appears that CEOs now understand that while customer service is important to the continuing operations of the enterprise, unless the organisation continues to innovate and develop new ideas, products and services, customers will leave anyway. Similar to product and service quality, good customer service may have gone from being a competitive differentiator to the price of entering the game. Success comes from continuous innovation. This would appear to be a message that is not reflected yet in the attitudes of IT and other functional managers.

It is, however, clearly the message given by executives from small enterprises. Figure 2-2 compares their responses to those from medium / large companies.

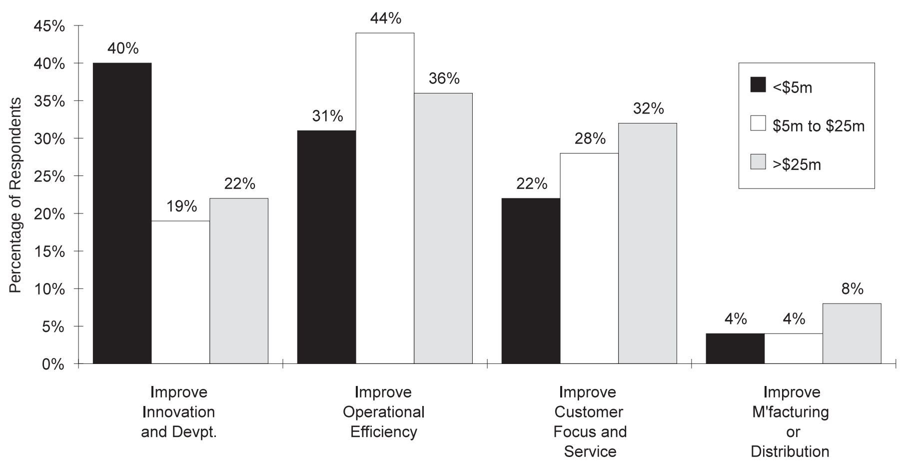

Figure 2-2 Most Important Business Objectives by Enterprise Size

Respondents were next asked to choose the most important business system required to meet the business objective that they had identified in the previous question.

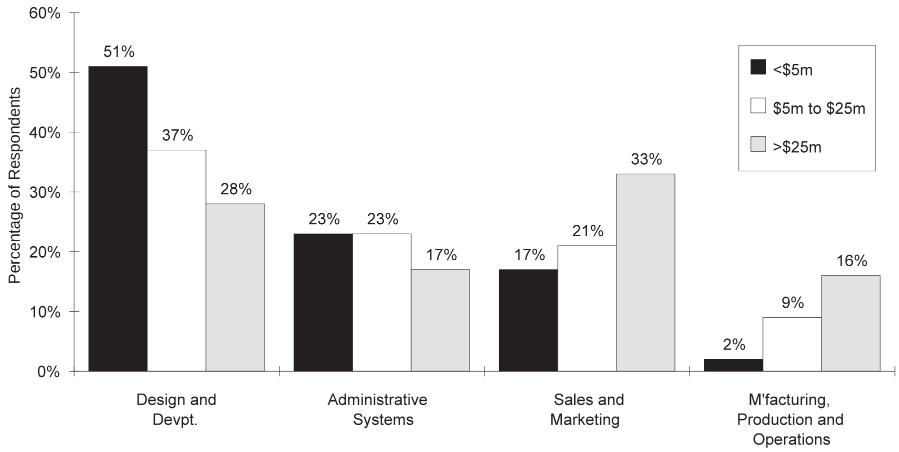

Figure 2-3 Most Important Business Systems by Enterprise Size

Overwhelmingly (51%), both CEOs and those responding from small enterprises identified Design and Development systems as the ones that would best enable them to meet their business objectives. IT management and respondents from larger companies identified Sales and Marketing systems, presumably to support the achievement of their improved customer service objective.

Technologies Required to Meet Business Needs

Having moved from business objectives to the systems that will best contribute to the achievement of those objectives, the survey next looked to link specific areas of technology to the systems identified. Respondents were asked to choose the five most important technologies from a list of 30 possibilities grouped into six clusters:

1. Application Technologies (such as client/server applications, multi-media applications and decision support systems)
2. Software Development Tools (including Computer Aided Software Engineering (CASE) tools and object-oriented program development tools)
3. Systems Management Tools
4. Databases and On-line Transaction Processing

5. Networking Technologies (such as messaging (email) systems)
6. Graphical User Interfaces (GUIs)

Figure 2-4 summarises the top ten choices of executive management compared to those of IT managers. It is clear that all parts of the organisation see the deployment of applications that support the shift towards distributed processing capabilities, offering local manipulation of local and central information, to be key to the achievement of business objectives, whatever they believe the most important business objective to be. This technology, summarised in the term client/server applications, was the first choice for CEOs (67% of whom selected this), CFOs and functional executives (63%) and IT managers (62%). Respondents from small enterprises were less united in their identification of client/server applications as the most important technology with 53% choosing it, though it still was the technology identified most often by these respondents as well.

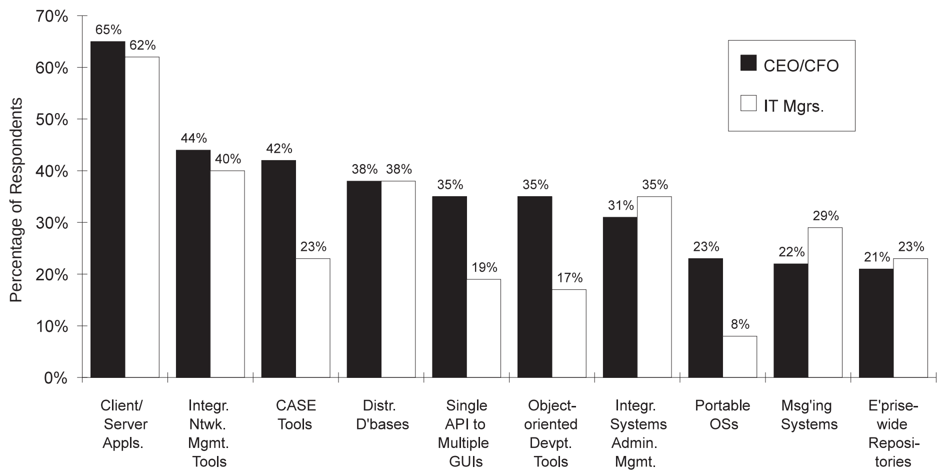

Figure 2-4 Most Important Computer System Technologies by Position

Although all of the respondents were of the same view as to the importance of client/server technology, the remaining choices indicated significant differences depending on function. Technologies associated with applications development and operation — CASE tools, object-oriented development tools and graphical user interfaces — were selected much more frequently by executive management than by IT managers.

This is made clearer in Table 2-1 which summarises the choices — excluding client/server applications — made by CEOs and IT Managers grouped by the technology clusters identified earlier. Thus, what we see are the priorities that each group places on different technology areas, given that client/server applications is offered as the number one priority.

In fact, the responses are in line with their answers to the first two questions. Software Development Tools are crucial technologies for Design and Development systems which the CEOs said that they required to support their primary corporate objective of improving innovation. IT Managers chose technologies required for communication with external partners and to support company-wide information flow, which is also quite in keeping with their emphasis on sales support systems to improve customer service.

	CEOs	**IT Managers**
1.	Software Development Tools	Systems Management Tools
2.	Systems Management Tools	Databases and On-line TP
3.	Applications Technologies	Software Development Tools
4.	Databases and On-line TP	Graphical User Interfaces
5.	Graphical User Interfaces	Networking Technologies

Table 2-1 Most Important Technology Clusters in Priority Sequence

Most Important Technologies for Standardisation

Having identified those technologies that are considered critical to the success of their organisations, the question arises: are these the same technologies that managers and executives believe would be most usefully standardised? Respondents were asked to rate the same thirty technologies from the previous question on a six point scale, where 1 is not valuable at all and 6 is very valuable. Table 2-2 on page 12 lists the technologies that obtained at least a 5 average from each of the major functional management groups that responded to the survey.

	CEOs	Other Executives	IT Managers
1.	Single API to Multiple GUIs	Integrated Network Management Tools	Client / Server Applications
2.	Messaging Systems	Integrated Systems Administration Management Tools	Integrated Network Management Tools
3.	Portable Operating Systems	System and Network Security	Integrated Systems Administration Management Tools
4.	Directory Systems or Services	Client / Server Applications	Messaging Systems
5.	Integrated Network Management Tools	Messaging Systems	Distributed Databases
6.	Distributed Computing Environment Tools		File Transfer Systems or Services
7.	Distributed Database Management Tools		System and Network Security
8.	Distributed Databases		Distributed Computing Environment Tools
9.	CASE Tools		
10.	Client / Server Applications		

Table 2-2 Technology Standardisation in Priority Sequence

The priorities identified by functional managers are, overall, close to those that they identified as being critical to the success of the enterprise. Client/server applications and the Systems Management Tools technology cluster are both well represented in their priority list. The only major difference is in their belief that Networking Technologies, such as messaging and file transfer, are more important for standardisation than Software Development Tools, which are considered more critical to business success. This may reflect expectations of what is technically and politically achievable, in that there is little visible standardisation work in the Software Development Tools arena (though, in fact, some standardisation effort is going on), compared with the highly visible work in creating both industry and formal international standards for messaging systems.

This issue of the perception of what is happening in the standards world may well explain the answers that the CEOs offered. The degree to which they differ both from the responses of their functional colleagues and from their own stated priorities for technologies is very noticeable. However, if we were to remove the first four items and only look at the other six, we would see that these responses are not particularly out of line with their technology priorities, in that the Systems Management Tools and Software Development Tools technology clusters predominate. We can, then, also see that the key differences from the functional manager responses are the CEOs' substitution of distributed database management tools for the functional managers' choice of integrated systems administration management tools, and the CEOs continued inclusion of CASE tools.

Why the removal of the highest four CEO responses and the reference to perception issues? Single API to multiple GUIs, messaging systems, portable operating systems and directory systems or services are all areas on which great emphasis has been placed by formal standards groups and industry consortia over the last five to ten years. Agreed standards of some sort exist for all of them. It is not clear, however, that there are robust, easily available, easily usable products available. The standards groups may have done most of their work, but the supply-side has not followed through with the products that people perceive that they need. Consequently CEOs, removed as they are from the day-to-day world of IT, do not see enough happening and place these fundamental open systems capabilities at the top of their priority list.

Purchasing Plans

The combined annual spend on external IT purchases — hardware, software and services — by the organisations represented in the 1993 survey is close to $40 billion. The expectation of the majority of respondents (52%) is that there will be some growth in the size of IT budgets over the next three years, though a significant minority (32%) expect at best flat budgets. There is consensus that the balance of expenditure between hardware and software will continue to move strongly towards software — 1993 43% hardware 35% software becoming 32% hardware 38% software in 1996 (the balance is purchased services which will increase by 2% over the next three years). This reflects both the downsizing of hardware to much lower cost commodity products and the increasing use of higher priced, more complex, software products.

Underscoring the shift of expenditure away from hardware is the expected rapid move towards open systems computing. Figure 2-5 indicates the proportion of their total information systems budgets the respondents are spending on open systems in 1993 and are planning to spend in 1996.

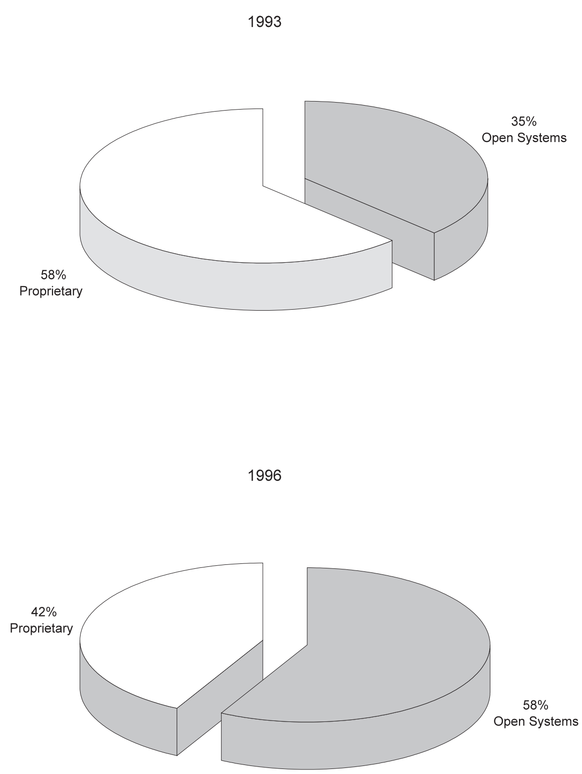

Figure 2-5 IT Budgets for Open Systems

Although there are some small differences in the proportions that are reported as being spent today by different functions (senior managers say

44%, middle managers 33%), the size of the growth through to 1996 is almost exactly the same. If we translate the percentages into dollars spent, assuming no increase in budgets 1993 to 1996, we are looking at an increase in expenditure on open systems from $14 billion in 1993 to $23 billion in 1996 from the group of organisations represented in this survey alone.

If there is to be this massive increase in open systems spending, which technologies will organisations be buying? Figure 2-6 summarises the technology areas that the three groups of management believe their organisations would develop or buy in 1996. Consistent with their view of the most critical technologies in achieving business goals, CEOs identified Software Development Tools and Applications Technologies as the areas where they expected to place most emphasis, while IT managers chose Systems Management Tools together with Applications Technologies.

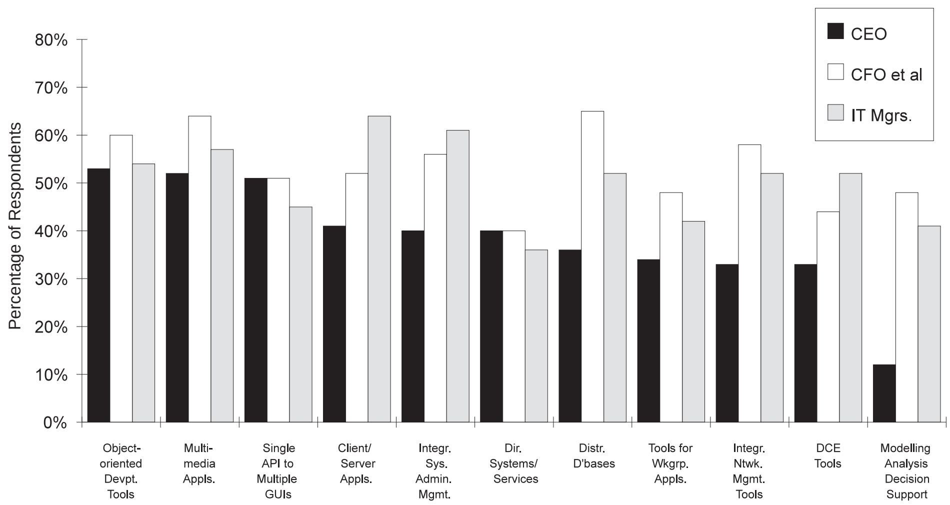

Figure 2-6 Technologies to be Purchased in 1996 by Position

The most surprising single message offered by the answers to this question, however, was provided by all respondents. Multi-media applications, on which expenditure is so low as to be difficult to track in 1993, will exceed every other technology except client/server applications in popularity by 1996. Whether the take up of multi-media applications occurs in 1996 or 1998, the message is clear: executives are looking for the vendor community to provide the tools necessary to link video, sound and text easily, and to begin the much heralded convergence of computing, communications and entertainment.

Conclusions

Open systems standards will continue to be driven by the views, and perceived needs, of large organisations, even if they now represent a better balance between the buyers and the sellers of information technology. The individuals contributing to the definition of the standards will still, in the main, be IT professionals who understand and work with the underlying technologies. Their attitudes and expectations are shaped by the view of the business world that they have, which, as we have seen, is different to that of non-IT executives and of managers from small enterprises.

However, at the end of the day, if the perception by general management of the value of their investments in information technology is to be improved, the expectations and requirements that they place on technology have to be factored into the open systems definition process. Information that is available from surveys like the Xtra Survey, backed by qualitative data from focus groups and other sources, must be taken into account when formal standards groups and IT industry consortia are determining the priorities for their efforts.

As the rate of organisational change increases, what is expected — and required — from information systems is constantly modified. In earlier times, the automation of simple processes sufficed. More recently, operational efficiency was the target for IT systems. As organisations increased their customer focus, support for the sales and marketing functions was the priority. Now, the demand from executive management is for ways to make the organisation more innovative and more able to deliver new products and services to the market quickly.

In 18 to 24 months, the emphasis of the message will have changed. The challenge for the open systems movement is to hear that message clearly and then to be able to respond in a timely way.

Unlocking the Pay-off from IT Investments

David P. Norton

In large measure, investments in information technology have not resulted in significant improvements in productivity for white-collar workers. The implementation of new technology must be seen as one component in an equation which includes organisational and cultural change. IT professionals have a particular responsibility to make these other changes in their own work practices.

Technology and Organisational Change

Economists sometimes use the concept of ‘‘long waves’’ to describe the impact of technology on society. A ‘‘long wave’’ refers to cycles of up to 50 years which sweep through the economy. The cycles are driven by technology, ultimately changing each element of the social fabric and the economic assumptions on which the society was built. The industrial revolution, based upon the steam engine, is such an example. When these technologies come along they create fundamental changes, not just in companies and economics, but in the way people work, learn and live. The fundamental assumptions of the entire society are changed.

Information technology is such a driver. Everything in our society is undergoing change. We can see the beginnings of those changes by looking at the amount of money being spent on this technology. In 1970, the typical organisation in the private sector was spending something less than 1% of its revenues on information technology, maybe $600 per person. (The typical home had more capital equipment to support the productivity of homemakers than the average office had in computer equipment.) In 1980, that had changed to around 1.5% of revenue and the average organisation was spending about $1,200 per person. By 1990, this number had grown to an average of $9,000 per person, with some organisations, such as financial institutions, spending $20,000 per person on technology.

It is easy to take this for granted now. Yet looking back to the 1980s, you find that as much as 50% of the discretionary capital in the U.S. was invested in information technology — in computers, communications equipment and software. What we saw was the transition to an information society, and society invested massive amounts of money in IT.

White-collar Productivity

We then ask the simple question: are we better off today than we were ten years ago? Lester Thurow of the Sloan School of Management has looked at the relationship between the heavy investment in information technology and average productivity growth. He found that in the period 1978 to 1985, productivity in the U.S. grew only about 0.7% a year. However, when he segmented the productivity gains from different classes of workers, and isolated industrial, blue-collar workers he found that productivity gains there were quite impressive — around 2.8% per annum.

The problem was in the service sector. The average white-collar worker showed negative productivity gains. Since white-collar workers represent 70% of the U.S. workforce, it dragged down the great productivity gains in the manufacturing sector and resulted in very insipid overall performance.

While there are major problems in defining the productivity of white-collar workers, the conclusions that Thurow reached have been validated by others. Steven Roach, in an intriguing article in the September/October 1991 Harvard Business Review about productivity in financial service organisations, came to very much the same conclusion. In spite of spending $12,000 per employee, the automation of back offices had done nothing more than rigidify what used to be a flexible industry, creating an influx of old, manufacturing-like structures.

What is particularly distressing about these studies is that, while half the capital in the U.S. was being invested in information technology, primarily targetted at white-collar workers, we observed negative productivity gains. One other study may provide a guide to the way out of this dilemma.

This was undertaken by Harvard professor Ramchandran Jaikumar, an authority on manufacturing, and in particular flexible manufacturing systems. His 1988 study looked at how American companies differed from Japanese companies in the way they used flexible manufacturing technology. He found that although they all used the same basic technology, the same centralised organisation approach and employed the same tools, the average U.S. company produced 10 different parts on a machine, while the average Japanese company produced 93 different parts. In the U.S., the length of a production run once it had been set up was 1,727 units; in Japan it was 250.

The Japanese were doing what flexible manufacturing was intended to do — short, small production lots for many different kinds of products. The Americans were doing what they had always done by employing the old management techniques with the new technology. The result was productivity in Japan about 4.5 times greater than that of the comparable U.S. companies.

Old World of Management

What can we learn from this? New technology came along that had the capability to replace an old approach with one that was potentially 4.5 times more productive. In the U.S., firms used the new technology, but they used the old management paradigm. Not surprisingly, they achieved old world results. An old world result is a ten percent Return On Investment (ROI) and is what you get when you stay with the old operational model. In contrast, the Japanese companies used the new technology, and also the new operational paradigm. Their benefit was a ten times ROI.

This, then, is the basic issue that faces us. Those of us who are driving the new world of technology must deal with that old world of management; a world which is characterised by the industrial model with its hierarchical nature, functional specialisation and its lack of ability to communicate horizontally in an effective manner.

The equation must be:

Success = Technology Change + Organisational Change

Transforming the Organisation

Organisations that have been successful in managing this equation somehow find a way to weave three basic ingredients together, referred to by Tapscott and Caston of the DMR Group in their book *Paradigm Shift: The New Promise of Information Technology* as the ‘‘Three Rs of Transformation’’:

1. Re-engineer the business by looking at the business process itself.
2. Realign the IS function, empowering people by changing the culture of the organisation.
3. Retool the technological infrastructure.

What this argues for, is open systems as one element of an equation that includes open organisations and open people. If we are going to take advantage of open systems, we must deal with these other two pieces of the equation.

Three or four years ago, business restructuring was more of a concept than reality, but now, for the first time in 50 years, organisations are finally looking at the need to transform themselves. For so many years, business strategy was not about integrating manufacturing, engineering and customers, but about finding the right mix of strategic business units, or about portfolio building through mergers and acquisitions. It was never about changing underlying productivity or the assumptions about how enterprises need to organise to satisfy customers and other stakeholders.

It is different today. The move towards organisational restructuring is real. Unfortunately, information technology, which was once touted as an enabler of change, has become a barrier to the redesign of business processes. Current information systems were built to support the old management world. In effect, we have poured cement around our old organisation approaches and cannot change them fast enough to realise the benefits of organisation renewal.

The third part of the equation is the human element, the organisational culture. Business processes can be redesigned in order to enable direct response to customer requests, but if the staff do not understand what it means to be customer-driven, then it is not going to work. What we are finding is the need for an equal emphasis on culture change to enhance awareness of quality and the need to be customer-driven.

Each of these changes is massive: restructuring the organisation, changing the culture and introducing totally new technologies.

Executing the Strategy

There will always be new technologies. The key issue is how to assimilate them successfully and to realise their economic value. The challenge for IT professionals is to take advantage of the opportunities which the Three Rs of Transformation offer to us. The IT profession is having difficulty responding to this challenge. There are problems deriving from the way technology was introduced and the resulting fragmentation that make it difficult to change. Now we have new waves of technology that are fundamentally modifying the ground rules. We have an issue that we have to deal with — execution.

If we look back, by way of analogy, to 1980, the manufacturing profession in the U.S. had a very similar problem. American manufacturing was being criticised. It had poor quality, it took forever to do things, productivity was extremely poor, and it was out of touch with the market. Customers were dissatisfied. Suddenly they had the option of buying Japanese products and they did, in very large numbers.

In the years since 1980, this problem has been turned around, partly because of technology, partly because of business process re-engineering, but mainly because organisations recognised these problems and did something about them.

In 1993, the information technology industry has a lot of the same symptoms and problems. It is incredible how many big and apparently well-managed companies do not know what it costs to do things like make a product or provide a service, or how many things they are doing with the same customer. This is our quality problem — data. What is more, the response of systems people is considered slow and our productivity has not grown. The result is that our customers are dissatisfied. Manufacturing customers had the Japanese. Today, big corporations are looking to out-sourcing as an alternative to dealing in-house with these basic frustrations.

Changing the Culture of the Information Technology Profession

We, therefore, have some work to do as a profession. I would like to suggest that we learn something from the manufacturing people in this country and apply some of their lessons to our own profession.

The first thing they did was to recognise that there was a problem. After recognising there was a problem, there were some key elements that they took to heart:

1. Listen to your customer, and figure out who the customer really is.
2. Deal with the quality issues. In manufacturing, three orders of magnitude of improvement in product quality was the norm (1980 *versus* 1990). Can we do the same thing with data? Can we improve the quality, reliability, timeliness and accessibility of data by three orders of magnitude?
3. Dramatically reduce the time it takes to bring a product to market. In manufacturing, we have seen reductions of 50% to 90%.
4. Eliminate the word ''years'' from our vocabulary and replace it first with ''months'' and then see where we can go from there.
5. Dramatically reduce the cost. Can we, by improving quality and cutting time, also reduce costs? Manufacturing experience indicates that 25% reductions are feasible.

Obviously, these pieces are all related. In manufacturing, it started with a new ethic — an ethic of quality. In information technology I would like to propose that we need a similar change in culture. However, rather than ''quality'' the word becomes ''value''. The only reason that IT exists is to create value in the business; value as judged by stockholders.

It is a simple statement, but if taken to heart, it affects every fabric of the organisation. It will change people's skills, the way they work, the way they are organised, and the way they are evaluated and rewarded. It may be argued that all of these changes are taking place under our feet, but if we can focus on transforming ourselves, then we will have made a step-function move forward in our ability to execute and deliver the potential value of the new technologies.

Organisational Accountability

I believe in a basic organisational precept — fundamental change in organisations cannot be dealt with unless we make accountability a team process. There is no systems manager or CIO who can achieve a result unless he/she is part of a team, and that team takes on a single goal. This is not merely a problem in IT. We have accountability problems in western organisations that go back to the often discussed subject of stove pipe, vertical organisations.

However, we are finally seeing new organisational forms come forth. People use terms like ‘‘clusters’’ or ‘‘networks’’ to describe these *ad hoc* organisations that are multi-disciplinarian in nature and focus on a particular problem. At the heart of this issue of accountability is the issue of performance measurement. Measurement communicates with the greatest honesty what is really meant. If you want to confuse a group or chain it to the past, communicate a vision of the future but measure the group with the systems and the structures of the past.

Balanced Scorecard of Measurements

Three years ago Robert S. Kaplin, Professor of Accounting at the Harvard Business School, and I ran a research programme into alternative approaches to the measurement of organisational performance. The basic premise behind the research was that relying solely on the financial measurement model that preoccupies western organisations is harmful and does not work. Financial measurement is obviously necessary in a capitalist society, but it is a lag indicator. Today’s financial performance is a result of yesterday’s strategy.

The approach that was developed was to create what we called the ‘‘balanced scorecard’’. This includes financial measures that tell the results of actions taken, but complements them with operational measures that are the drivers of future financial performance. The balanced scorecard links:

1. Financial perspective — how does the organisation look to shareholders?
2. Customer perspective — how do you look to your customers?
3. Internal business perspective — what are the business processes that you are going to invest in and to excel at?
4. Innovation and learning perspective — how is the organisation going to continue to improve and create value?

Financial Measurements

For private sector organisations, financial goals typically refer to areas such as return on capital employed, profitability and growth in market share. Other measures are sometimes found as well. For an organisation that is one member of a five company group, measures relating to synergy may be appropriate — the number of opportunities created for cross-selling of sister companies, the degree to which new technologies are shared, and so on. For chemical or process industries in which the value of the company to shareholders can be totally destroyed overnight if there is a major environmental problem, it may be appropriate to develop a measurement that keeps this risk visible.

Customer Measurements

The key concept here is to measure through the eyes of the customer — to get outside the organisation and to measure from the outside in.

One of the companies that participated in the study had a measure of on-time delivery. It defined success as being able to deliver within five days of the window customers needed to fit into their manufacturing schedules. However, it turned out that some of their customers had long windows — nine days was fine — while others had very narrow windows — delivery had to be within three days of the planned date. The supplier had a measure it had devised itself that was pleasing no one. In some cases it was doing better than was necessary, in others its performance was much worse even when it met its internal measurement criterion. What is more, the company found that every one of the customers had their own measurement systems for tracking the supplier. What we learned from this was that rather than go to the expense of building your own measurement system, why not simply ask your customers?

Another facet of looking from the outside in is to use direct customer research. Increasingly, companies are measuring themselves by employing blind surveys where they talk directly to the customer to get their perceptions of what the company is achieving. In particular, questions on issues like: Which company has the people best qualified to support you? Who offers the best overall service? Whose service desk is the most courteous?

A third area in customer measurement that is taking on added importance is that of third party evaluations. For years, we have understood the role of financial auditors to provide objective assessment of financial performance. In recent years, we have begun to see the emergence of a similar function in this part of the performance scoreboard. For example, J.D. Powers provides objective hard measures of quality within the automobile industry. Organisations are using the Powers ratings as their way of communicating to the marketplace that they, in fact, have quality. J.D. Powers is now extending its survey to the airline industry and to computer desktops. Similarly, the U.S. Department of Transportation has had measures of on-time flight arrival and of baggage handling problems that they publish on a regular basis. This has had dramatic impact on the way in which airlines respond and deal with these problems.

Measuring from the outside in creates an honesty and integrity in the measurement system. If your strategy is to get close to your customer, to provide quality, and so on, you have to get out there and understand through the eyes of your customer whether you are achieving your goals.

Internal Measurements

The third area of the scorecard concerns the management of the business. Typically, this is where organisations do most of their measuring. It is important to begin viewing the organisation as a system made up of processes that transcend traditional organisational structures. The key question in developing this measurement system is to identify the processes that have to succeed in order to achieve your business vision.

For example, in one study in which I participated, the most important process concerned new product development. Four measures were developed around that process. The first referred to the product development cycle, the second to milestone effectiveness on projects to bring products to market, the third measured selling effectiveness, and the fourth the cost of new product operations.

Measuring the Ability to Learn

The final piece of the measurement system has to do with the organisation's ability to learn and to innovate. The company may have a great competitive advantage today, with tremendous financial results, but if it cannot support sustained change, then it will not retain its value in the marketplace. It is only through the ability to launch new products and to improve operating efficiencies continually that a company can develop new markets and increase revenues and margins — in fact, add shareholder value.

How do you measure an organisation's ability to learn and grow? The most effective way we have found is to identify the objectives to be achieved through continuous learning and then to develop measurement approaches for those goals.

To go back to the study quoted earlier, their objectives for the learning organisation were:

1. to grow through innovating in the marketplace
2. to develop mechanisms for continuous improvement through empowering staff
3. to develop the necessary skills in staff
4. to create a climate where staff were motivated to use those skills.

The measurement criteria developed to assess these learning objectives were:

1. the percentage of revenue from new products
2. the number of suggestions per employee together with the percentage of them implemented
3. a staff skill profile
4. an employee attitude survey to measure the climate.

Team Accountability and the Balanced Scorecard

The idea of the scorecard needs to be placed in the context of team accountability. To make this approach work it needs to be embedded in the way that the organisation is managed. This is a structure that essentially translates strategy into a scorecard that is available for the team to monitor what it is trying to achieve. It becomes the measure of team accountability.

Measurement is important — not so much the fact of measurement — but for the communication value of being specific. When we talk about value for money, what does that mean? How would it be measured? Measurement forces precision — particularly important in areas like technology investment at a time when half of U.S. investment capital is going into technology.

The scorecard also has value in dealing with the long-term *versus* short-term issue. For example, it helps organisations to maintain perspective when faced with balancing short-term budget issues with long-term questions, such as staff development. It does not eliminate the temptation to take the short-term action, but it makes it more difficult and more visible.

A final case may demonstrate the connection between team accountability and the scorecard. An engineering organisation started by developing a scorecard based on their strategic vision and then went through an objective-setting process. They identified the items to be measured and developed programmes that would be used to achieve the identified goals. In parallel, they embarked upon an internal communication programme that laid out the organisation's vision and strategy and the means by which the implementation of the strategy would be measured. Interestingly, it was called the ‘‘team measurement project’’, and was also tied to the compensation systems. Finally, they built an information system that provided feedback for people to see how they were doing.

From the measurement systems they were able to show how the strategy would help to improve skill levels, product quality and customer acceptance, and would ultimately create $100 million of positive financial impact. They were able to show extremely clearly what the team goals were and they began to create a team measurement process that tied accountability directly to the strategy.

Conclusion

This example brings us back to my initial starting point — information technology is an enabler, but on its own it is not enough. The equation of successful change management has a technology component, an organisational component and a cultural component.

Moreover, we have to recognise that the IT industry will be under continuous pressure as new waves of technology arrive. Given this pressure, we have to really understand that we are under fire and that as IT professionals our approach must undergo fundamental transformation just like everyone else in the organisation. Accountability is one of the keys to making this happen. IT professionals hold the key to much of the future, because without technology the most important structural changes will not take place. Technology more than anything else is going to facilitate the learning organisation. To achieve these benefits, the IT profession must break through a number of barriers. Groups like X/Open are working very hard to eliminate the technical barriers. Ultimately, the major barriers are accountability and the ability to work as teams at the top of the organisation. If we can take a business strategy and find a structure that allows for the alignment of technology investments and other change programmes with that strategy, then make the right set of people accountable for them, we have a way to truly take advantage of these investments.

It is my belief that unless we are able to achieve this single team atmosphere with a single set of goals and a single set of measures, IT investments will continue to be under-realised.

Realising the Value of Open Systems

Paul A. Strassmann

A key underlying issue in information systems is how to preserve strategic IS assets — applications that represent how the enterprise works and data that represents its environment. Open systems have a crucial role to play both in maintaining these assets over time and in indicating a way for people to work within the enterprise, thereby allowing their full value to be realised.

The Fundamental Balancing Act

Increasingly, people in every enterprise are facing information technology budgets that are constrained. Where businesses are expanding, overhead costs have to come down so that there is room for new applications. Where businesses are contracting, technology budgets must be reduced. In the case of an organisation that I know particularly well, the U.S. Department of Defense (DoD), budgets will decline 20 to 30% over the next 6 years.

At the same time that these kinds of budget cuts are being introduced, organisations are discovering that the assets that are in place no longer meet the strategic needs of the enterprise — business processes are being rendered obsolete at such a rapid rate. Consequently, the fundamental issue that has to be faced by IT managements is the balancing of two forces — reductions in total costs while preserving the ability to continue at a high rate of application enhancement and development.

What is clear is that if you introduce new systems that have high maintenance and operating costs, you are going to use a large portion of the cash that should otherwise be allocated to the development budget. If you want to preserve the level of investment in enhancement and development capacity, savings have to come from operating costs.

Consequently, one of the prime parameters in the implementation of new systems is the velocity at which they can be injected into your environment because you need them to generate the operational savings that support further investments. The driving parameter in this kind of model is time. If you make the time, you make the money; and so timing becomes a critical element in the realisation of business value.

However, you need to ensure that the investments that are undertaken not only generate quick returns, but also create a renewable and permanent asset. What you have to do is to replace existing systems with systems that have much longer life.

Reusable Software

At the DoD, we identified that half of the potential cost reductions were achievable through one variable only — reusable software. This is by far the most critical element in generating cash and business value out of information systems software investment.

To be economically advantageous, new systems must have substantially lower maintenance costs than earlier implementations, and they must have a very high degree of code portability. My minimum target for portability is that after 6 years 60% of the code is portable to the next generation of hardware and becomes part of the next generation of applications.

The 6-year cycle is dictated not by hardware technology, which is now on a 2-year cycle, but by the rate of innovation in business. It is crucial to understand that it is the internal organisational structure that generates the kind of information system requirements that lead to investment in new applications. Therefore, the rate of change of internal procedures and the internal organisational structure is what governs the rate of replacement of applications systems — and any corporation that believes that their procedures and organisational structure will be able to survive more than 6 years is working under highly questionable assumptions.

Residual Value

This concept of software usability underlies one of the keys to the evaluation of software investments — that of the residual value of the investment. The residual value is the discounted cash flow of your investment today that still generates utility to the corporation beyond the planning period. In the case of the U.S. Department of Defense and most U.S. corporations, the planning period is 6 years, although in electronics that is now shrinking to 4 years.

	1986	1987	1988	1989	1990	1991
Annual Benefits	0.0	0.0	83.4	72.3	82.2	91.4
Annual Costs	63.3	31.5	14.5	15.0	15.0	17.2

Net present value of cash benefits @ 25% = 123

Net present value of cash costs @ 25% = 94

Benefit:Cost Ratio = (123 – 94)/94 = 31%

Figure 4-1 Project Benefit:Cost Ratio

Figure 4-1 shows what an investment would look like without residual value — the way these analyses are usually done. The number that is important is the cash benefit to cash expenditure ratio of 31%. This allows for the discounting of future gains, but that is not adequately realised even at a high discount rate like the 25% that has been used in this example.

	1986	1987	1988	1989	1990	1991	R.V.
Annual Benefits	0.0	0.0	83.4	72.3	82.2	91.4	340.5
Annual Costs	63.3	31.5	14.5	15.0	15.0	17.2	64.1

Net present value of cash benefits @ 25% with residual value = 195

Net present value of cash costs @ 25% with residual value = 107

Cash + Residual Benefit:Cost Ratio = (195 – 107)/107 = 81%

Figure 4-2 Project Benefit:Cost Ratio with a 12-year Residual Value

Figure 4-2 provides the identical example but with a residual value. The payback has now gone up to 81%. What this demonstrates is that if the useful life of the application can be extended beyond the immediate planning period, and thus the software asset can continue to contribute even in a different organisational environment, then the return on the investment is almost trebled.

This method of determining payback is embodied in a piece of software that is called the Functional Economic Analysis which was developed by the DoD and is generally available as public domain software. It is clearly in the public interest to stimulate a standard approach to determining economic payback using the same kind of reusable routines. Needless to say, the DoD wants to increase the residual value of its software developments.

Maintenance of Permanent Assets

The question is: How do you extend the residual value?

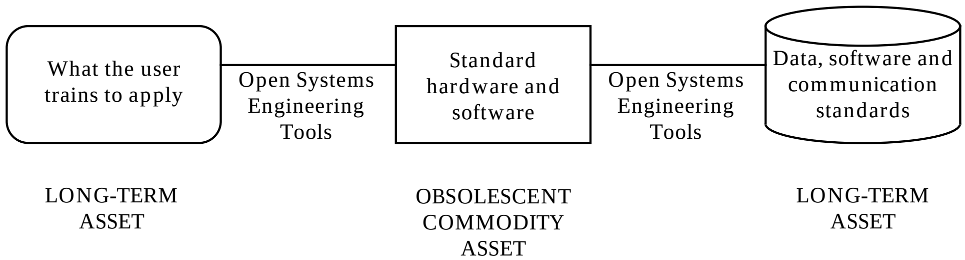

Figure 4-3 Long-term versus Short-term Assets

Figure 4-3 is a highly simplified but powerful view of the information management domain. It suggests that when you look at the total functional cost of an organisation, the most valuable property is user training. Consequently, rather than looking at information technology assets as merely particular pieces of hardware that have to be used until they die, it should be remembered that the fundamental organisational cost is on the user side of the equation. In the defence environment, the user cost is particularly severe because the operators have to use information technology under conditions of extreme stress, and therefore dissimilar protocols, graphic interfaces and commands are not only costly but extremely dangerous.

The other crucial long-term information assets are data and software — software representing the collective intelligence of how the enterprise is put together, and data representing the facts about the environment of the enterprise. Everything in between — equipment and operating software — is commodity products that are being rendered obsolete at a prodigious rate. The current measure of obsolescence of micro-computers, for instance, is the monthly decline in the prices of Intel 486-based micro-computers, now averaging 6.5% per month. Nothing in the history of mankind has ever depreciated or been deflated at the rate of 6.5% per month. What is more, it is considered that over the next decade, the rate at which new technology will arrive will mean that real obsolescence will increase to about 11% per month. At this point, technology will have a half-life of less than 18 months and will clearly be a junkable commodity.

Consequently, the goal of the residual value approach is the preservation of long-term assets, such as the behaviour and training contents of your information system and its supporting intelligence, while ensuring that what is in between — the hardware and operating system software — can easily be removed. In order to protect the long-term assets and to be able to detach obsolescent commodity assets, you have to create systems and engineering interfaces and tools that keep the long-term and short-term assets separated. This is a snap-in, snap-out, disposable type of economy that has to be highly standardised. In fact, one way of looking at the value of open systems is that they make it easier for customers to obsolete the equipment that lies between their permanent assets.

Open Systems Tools

To achieve this snap-in, snap-out world requires an integrated computer-aided system engineering environment, I-CASE. I have come to the conclusion that open systems without open systems tools do not work. You must have a way of bolting onto open systems hardware ‘‘wheels’’, so that the old superstructure can be removed and replaced with a new superstructure with very little pain and a very large amount of reusability of information.

The underlying concept is to have a development environment which has a life of 25 to 30 years. This development environment is not based upon a particular procedural code or target hardware, but upon the functional processes which really define the operational rules and requirements. This development approach employs fourth and fifth-generation machine-independent languages that are specifically targetted at the open systems environment.

Once you have a development and testing environment against which you can continually validate the logic of any extensions, the retention of any particular implementation is not important. The executable code is not maintained in the object environment, but at the requirements level. This means that, for instance, if you wish to move the application to a pocket computer with a special operating system and a unique chip, you can still achieve the portability of the application to that particular machine, provided, of course, that it supports open systems interfaces which are in compliance with the X/Open CAE Specifications.

I want to emphasise the increasing importance of software tools as the key enabler for bolting together long-term assets, such as knowledge about the human interface and user training, with changeable assets of hardware and system software. Open systems toolsets are the key enabler necessary to obtain the economic value of the open systems environment.

Are these tools alone sufficient to preserve the long-term residual value of software investment? I would suggest that in the same way that operating systems standards, interoperability standards and communications standards are necessary but are not sufficient to guarantee the preservation of software investments, tools are necessary but not sufficient. We have to raise our sights to ensure that organisational knowledge is preserved within our organisations.

Software as Collective Organisational Memory

One way of looking at software is to see it as one element in a continuum of an organisational communication process. The software represents the agreements reached, after negotiation and interpretation, on the means users will employ to communicate with programmers, programmers with machines, and machines with programmers and users. In order to increase the residual value of the software assets embedded in databases and computers, there must be a congruence of understanding and ease of communication among the community that participates in this process. It has to be extremely adaptable and fairly rapid. Essentially, programmers, testers, analysts and users must employ open processes and tools in order to achieve lasting cooperation.

This leads to the startling conclusion that what we call software expense is, in fact, purely a way of recording expenses that cover meetings, head scratching and shouting matches. Therefore, if you want real open systems, the human dimensions of software creation, maintenance and development have to be open, transparent, easier and more graceful than has traditionally been the case.

How do you then conserve software assets? You should start to look at software as a way of codifying how the enterprise does its business. It represents the collective experience and the collected, accumulated memory of every person who has ever participated in the conception of that software. You can look at software development as a collective process, a form of recording organisational memory and of encapsulating how members of the organisation have negotiated how they will cooperate. Software should be seen as part of a continual, evolutionary process rather than something that is designed and then thrown away.

Organisations have deep roots. This is seen in the accumulation of what we call organisational culture, that provides a stability which is independent of the individuals making up the organisation at any particular time.

An organisation, particularly for residual value, must be able to carry its culture like a genetic code, with only very small mutations from generation to generation. Software is a form of wealth. In fact, a large number of organisations today have software assets which are worth more than the tangible assets on their balance sheet. Thus, the primary purpose of open systems is to manage that wealth and manage it so that as little as possible is destroyed; so that it accumulates rather than is replaced.

Software as the Organisation's Inheritance

Consequently, so-called legacy systems and legacy software should not be viewed as something you get rid of so you can start anew, but more like an inheritance that is to be built upon through continual rejuvenation and replenishment.

Every new system and enhancement should be conceived of as a way to exploit and increase this legacy value with as little as possible discarded. When you analyse the structure of information systems you discover it is not the elements of logic that change, but the way they are put together. In a typical business application, more than 85% of basic routines deal with information retrieval, information management and information display. That applies to accounting, medical, material or inventory systems. We continually throw out systems even though the fundamental underlying genetic attributes of that system are the same from application to application. We can no longer afford to do this, which is why the implementation of an open systems environment really calls not just for hardware independence, but for a symbiosis between software preservation at a component level and the hardware. By this means, you can move as your environment evolves.

Open Political Structure

The open system is a political dimension which perhaps has not been adequately talked about, but is nevertheless essential because it is the openness of political structures which allows people to work together without hostility and in a spirit of cooperation.

My conclusion is that open systems are there to provide the infrastructure that allows graceful accommodation by the organisation to changing administrative processes and to changing relationships with vendors and customers. Therefore, the underlying reason why you want to have open systems is not because you want to buy cheap hardware, though you may want to do that too, but ultimately because you want to be able gracefully to evolve your business processes on 6-monthly or 4-monthly cycles, rather than the years it takes today. Ultimately, the winning ticket that will show in the residual value of functional economic analysis is business process redesign.

Architectures

However, it takes more than standards for interfacing hardware and software to really build a viable collective organisational memory. There needs to be an overall model or architecture which governs the methods and location of information retention. Unfortunately, the word architecture is perhaps the most abused and misused term in today's business system conversations. It is so overused that it ceases to have meaning. Since I am a student of history I like to look at solutions which have some precedence in history as being survivable solutions. Therefore, rather than the traditional building analogy, I prefer to look at approaches to societal design. Humankind has looked at various forms of organising civilised society — we have had monarchies and dictatorships — but so far our experience suggests that the most survivable institutional framework for the peaceful growth of individuals and the development of social wealth is what is called a layered society. This constitutional structure maintains that there are intermediate points between monarchy and absolutism at one end of the scale, and anarchy at the other. Complex societies must have many layers with the appropriate layer performing certain functions. If a particular function is not preserved for a specific layer, it is automatically delegated down and decentralised. It is really the concept of governance called confederation. Confederation is not absolutist, but nor is it decentralised.

There is sharing.

Given the organisational premise outlined above — that software is a form of organisational memory for complex organisations — I would like to suggest a model that is neither centralised nor fully decentralised as being appropriate. In this model, used in the DoD, the various software assets have to be put into the right level of a federal structure.

Enterprise Level

The enterprise level is where you maintain policy, standards, reference models, data management tools, integrated systems, database configuration and software configuration. In order to be able to change a component of this open systems environment and enable rapid change at the local application level, you maintain very long-term assets at this the highest level in the federal structure. Primary attention at the enterprise level should focus on structures and constructs which are 25 to 50 years out. Once the data definition and data elements have been agreed for the enterprise, they should be pretty well immutable; independent of technology, hardware and applications.

Local Level

At the local level, where you are looking at query languages, customised applications, prototyping, local applications and *ad hoc* simulations, the timescale is measured in days or weeks. Somewhere between 50 years and a day or week there are functions which you assign to the intermediate levels in the organisation. These intermediate levels are where you can locate your open systems assets. In the DoD, for instance, the large databases which are application independent are maintained at the functional level. They are not modified more than maybe once every ten years when major changes in technology are introduced, but even then it should be possible to migrate these databases to new technology in the new environment.

What I am suggesting is that if you really preserve the collective knowledge of the enterprise, there is large residual value from software development. You cannot simply look at interoperability standards, you have to look at what I consider the rules of governance, or the way the organisation is put together. You need to ensure that you put as much of the investment as possible in long-lasting assets like data, the telecommunications structure, and configuration management. However, at the same time it is important that innovation, which is fleeting, local and absolutely essential as a way of preserving a sense of freedom and independence, is not stifled. The beauty of this kind of approach is that when you develop local applications that turn out to have permanent value because they are widely imitated, they can be quickly moved up in the hierarchy. It is extremely important that the whole idea of the dynamics of innovation is preserved in the layered structure.

Preserving Personal Privacy

Within any architecture, personal privacy that is exempted from standards must be preserved. I am a strong believer that you cannot impose standards everywhere. There must be a line at which individuals who have on their desktop the power of a Cray supercomputer should be able to do whatever they want as long as they do not cause damage to the wider organisation. Similarly, no enterprise lives solely within itself, it lives within the society at large, and therefore the architecture must also provide for interoperability with others in the industry, with suppliers, with vendors and with service providers.

Conclusion

The open systems movement should not be seen as dealing with a range of technical devices, but as one of the most critical elements of effective information management. Open systems become the matrix within which we embed information systems that preserve the value of organisations. The pay-offs are enormous not just in preserving hardware and software assets, but most importantly in making organisations viable, competitive and flexible.

Thus, the underlying issue of open systems — the preservation of assets — is bigger than just standards. It really deals with the governance and constitutionality of the structure of the information society. It is concerned with how this is organised within the enterprise and then how enterprises are organised to cooperate as a national and then global society.

Developing a Business Case for Open Systems

Christopher Jacobs

Though you know that open systems are the right answer for your organisation, somehow the justification doesn't seem to work out as you expected. Tools for matching business needs to information technology and for calculating the cost justification for open systems' investments are now available. These include an approach to obtaining enterprise-wide involvement in the justification and X/Open's Open Systems: A Guide to Developing the Business Case.

The Challenge

There is no longer much argument over the need for organisations to move towards open information systems. The debate now is over how this can best be justified to senior management.

Traditional financial measures, like return on investment, seem at best to offer partial answers and at worst to promote the inertia of remaining with existing installed systems. Yet, proposals that talk almost exclusively in qualitative terms about potential benefits that may be derived from the ability to change suppliers or interchange information with customers, seem less than compelling to management. The need is for ways to compare alternative approaches to meeting the organisation's information management needs and assigning quantitative measures to the comparison.

Clearly, such a comparison will not replace measures like return on investment, but will extend and complement them, to provide a full picture of the implications of making choices between alternative, valid means of achieving corporate goals.

If investments in information technology are to realise their potential, there needs to be explicit linking of IT directions and developments with corporate business directions. Any methodology for evaluating IT investment must, therefore, provide mechanisms to ensure their alignment with the organisation's overall strategic goals.

Long-term versus Short-term Considerations

We are not talking about the small, isolated purchases that are made for short-term tactical reasons — a branch wants to add additional access capability to local information for more of its staff or the purchasing department needs to upgrade its links to the material system. These acquisitions are limited enough that they can be justified directly within the operation concerned. Nor are we concerned with the situation in organisations for whom the long-term is six months and everything is undertaken in an *ad hoc* tactical manner.

We are considering the long-term implications of developing an IT framework that supports the business and has the flexibility to change as the needs of the business change. This framework includes the development of an infrastructure to tie together the disparate elements that make up the organisation's information management and delivery capabilities — voice and data telecommunications, personal, divisional and corporate computing systems. It also includes guidelines on how to ensure that new developments and acquisitions fit into this infrastructure, and how to make tactical moves within a strategic context.

While there are examples of single projects for which a financial case can be built for an open systems solution within an otherwise wholly proprietary computing environment, experience shows that it is the longer-term strategic benefits that really ''sell'' open systems. The ability to save money on hardware purchases through the wider competition available is undoubted; however, the real competitive advantage comes in the flexibility to add, modify and extend the use of technology in support of business advances.

Identifying Critical Business and IT Objectives

In developing tools to assist organisations in building a strategic business case for investment in an open systems framework, we have employed a reference model that ties together the various efforts involved.

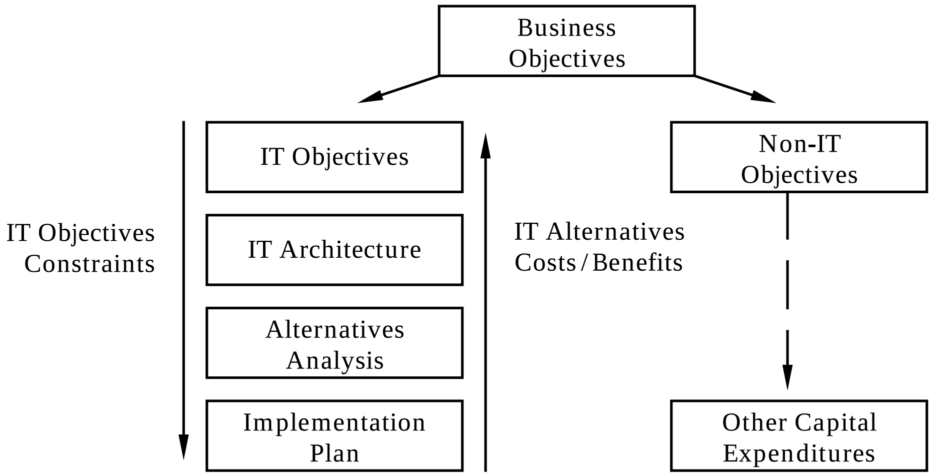

Figure 5-1 Business Value Reference Model

Figure 5-1 shows this in diagrammatic form. This model for identifying the value of open systems and building a business case was developed by X/Open together with Regis McKenna Inc. (RMI) and is based on concepts developed by the Center for the Study of Data Processing, sponsored by Washington University and IBM, and by Oracle Executive Services at Oracle Corporation.

As can be seen, the crucial first steps are the identification of the organisation's key business and technology objectives. It can be argued that this is the most valuable part of the whole exercise — whether or not at the end there is an acceptable case for open systems. Involving the whole organisation as widely as possible in a process that builds agreement on the overall business objectives and on the way that information technology can support those objectives has values that extend well beyond the IS organisation.

AIC Process

While many enterprises have more or less *ad hoc* means for obtaining organisation-wide agreement on key issues, I have found that the employment of a formal methodology aimed at obtaining a wide spread of views but able to move to concrete actions is the most effective approach.

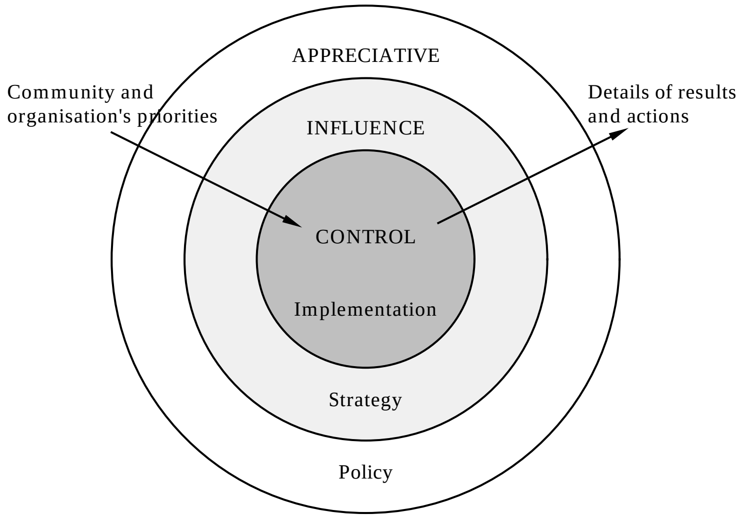

Figure 5-2 The AIC Process

Dr. William Smith of the Organizational Development International Institute in Washington DC has developed an approach which has been used in systems as large as a country — village development needs in Thailand — and as small as an individual plant. In a case study employing Dr. Smith's methodology in an information systems context, we worked with this approach to obtain agreement on the key information technology goals and on the priorities for their achievement at a nuclear power plant in the Czech Republic.

The AIC Process, as it is known, is a multi-phased model for moving from broad-based policies to detailed action plans. It recognises three major groups, or constituencies, that operate within large bodies of people, be they companies, governments or complete countries. Figure 5-2 illustrates the inter-relationships: the Appreciative level who are concerned with overall policy, the Influence level responsible for determining strategies to meet the policy directions, and the Control level charged with implementing the strategies.

The phases of the AIC Process follow the same sequence in moving from wide input on the overall policies, through firming up the strategic options, to definition of implementable plans. This can best be illustrated by the

Czech case study. Here we were working with a newly independent company that was faced for the first time with the challenges of a market economy. Management understood that they had the opportunity to reinvent the organisation and had decided that the introduction of information technology offered them ways to leapfrog several generations of management experience. The questions they needed to answer were:

- What are the business priorities of the company?
- Where can information technology best help in achieving these priorities?
- What technological approach should be implemented?
- What barriers exist to the fulfillment of the priorities?

In deriving answers to these questions, the organisation went through the following phases:

- The *Appreciative Phase* that involved every group concerned with the decision areas and that might be affected by the way the decisions were implemented. This included the Mayor of the nearest township who had an interest in the decisions because of their safety and employment implications, and representatives of every department including personnel, sales, finance and plant operations. Some 36 people were divided into six groups who discussed their individual responses to the questions until a group consensus had formed.
- The *Influence Phase* where the views of all participants that had been aired in the small groups were represented to the whole community through selected spokespersons or influencers. These were six managers from a variety of disciplines (deliberately none were from the IS organisation) who argued for positions and priorities determined by their group in the Appreciative phase. The aim during this phase is to find common ground between groups and to determine the highest priority actions.
- Determination of the management project team to work during the *Control Phase* on translating the detailed objectives and priorities identified into an actionable plan that meets the company's priorities. Moreover, in order to ensure that the whole organisation, as represented by those participating in the Appreciative phase, are kept involved during the implementation, a non-executive group was developed to act

as a two-way communication channel between the management (Control) group and the wider (Appreciative) community.

The whole process, involving the complete middle and senior management of the plant and outside contributors, took only two days to obtain an actionable plan and to agree the membership of the Control and Appreciative groups. This in an organisation that had never met as a complete team and many of whose members had never even met each other before the start of the first day of the meeting.

If we look at this in the light of the Business Value Reference Model, the AIC Process enables you to get basic agreement on the Business and IT Objectives and on the issues to be considered when developing the IT Architecture. It provides you with a group of people who are committed to working with the IT organisation to develop the detailed alternatives and, maybe more importantly, with an enterprise-wide support and information group that you can use when finalising the actual business case.

Tools for Analysing Alternative IT Approaches

The tools included in *Open Systems: A Guide to Developing the Business Case* are based on research sponsored by X/Open and undertaken by Regis McKenna Inc. (RMI) and the DMR Group Inc.

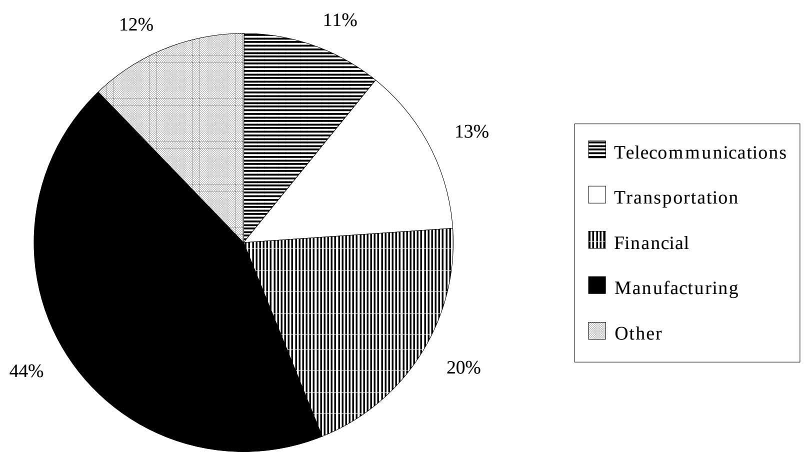

Figure 5-3 Profile of Interviews by Industry

Based on the results of a major DMR survey of companies that had implemented open systems and on RMI research on the adoption of open systems in Fortune 200 companies, RMI conducted in-depth interviews with 40 companies around the United States. These companies ranged from manufacturing giants like Boeing, Motorola and General Electric, through communications leaders like Bell Atlantic, to financial leaders like American Express and Chase Manhattan Bank. Figure 5-3 on page 44 indicates the proportion of interviews conducted in each of the major industry sectors.

Using the information gleaned from the surveys, executives interviewed were offered a list of key business and IT objectives and asked:

- to rate the importance of each item to their business
- to rate the value of open systems in meeting the objective
- to identify an appropriate measurement for the objective
- to compare the success in meeting the objective through open systems or proprietary solutions.

The results of this research are summarised in the *Open Systems Value Guide* and the *Value Comparison Scoresheet* included within *Open Systems: A Guide to Developing the Business Case*.

Value Guides

The *Open Systems Value Guide* is shown in Table 5-1 on page 46. It identifies eight key business and IT objectives, specific areas where meeting these objectives can be measured, suggested metrics for indicating achievement of each objective, and ways in which open systems add value to meeting the objectives.

In addition, specific *Industry Templates* have been developed that tailor the contents of the *Open Systems Value Guide* to the requirements of the financial services, manufacturing, telecommunications and transportation industries, with a separate template for government.

These templates represent the collective experience of managers in each industry in developing major business and IT objectives for companies in that industry, suggested means of measuring each objective, and the ways in which open systems contribute to their achievement. The availability of these templates, and of the full *Open Systems Value Guide* for companies in other industries, means that as you move through the process of expanding the objectives that you have developed, through the AIC Process, for example, you have a complete set of tools to guide you.

Table 5-1 Open Systems Value Guide

1. IMPROVE COMPANY EFFECTIVENESS		
Objective	**Suggested Metrics**	**Open Systems Add Value**
Product Development	• Time to market	• Distributed computing
Manufacturing	• Quality • Cycle time • Manufacturing costs	• Improved vendor communications through common communications standards • Greater system flexibility through reduced time for manufacturing process changes
Service and Support	• Response time • Service quality	• Improved inter-departmental communication via common communications standards
Product Delivery	• Inventory levels • % orders shipped • Delivery time	• Common communications standards promote closer links between manufacturing and distribution

2. IMPROVE OVERALL BUSINESS OPERATIONS		
Objective	**Suggested Metrics**	**Open Systems Add Value**
Better Access to Critical Data	• Data availability	• Greater ease of data exchange through standards adherence
Better Tools for Productivity	• Productivity	• Standard graphical user interfaces improve user productivity
Better Response to User Problems	• Response time	• Standard application development environments speed software enhancements

3. REDUCE IT SYSTEM COSTS		
Objective	**Suggested Metrics**	**Open Systems Add Value**
Multiple Vendors	• Number of vendors bidding	• Open architectures
Lower Prices	• Cost of hardware and software	• Greater price competition through multiple vendors
Lower Investment Risk	• Vendor stability • Useful life of system	• Lower investment required
Lower M'nance and Support Costs	• Cost	• Greater flexibility in deciding maintenance and support strategies

4. IMPROVE COMMUNICATION OF ELECTRONIC INFORMATION		
Objective	**Suggested Metrics**	**Open Systems Add Value**
Company Internal	• % company, customer, suppliers connected • Accessibility of information	• Standards for email and networking
Between Company and Customer	• Time required for communication	
Between Company and Supplier	• % electronic *versus* paper communication • Amount of JIT inventory • Lot sizes	

5. IMPROVE SYSTEMS PERFORMANCE AND FUNCTIONALITY		
Objective	**Suggested Metrics**	**Open Systems Add Value**
Faster, More Responsive Systems	• User productivity • Performance criteria	• More vendor choice, leading to higher performance systems
Better Upgradability / Scalability	• Ease of upgrade • Flexibility in upgrade choices • Cost of upgrade	• Greater degree of vendor competition

6. IMPROVE APPLICATIONS EFFECTIVENESS		
Objective	**Suggested Metrics**	**Open Systems Add Value**
Purchase Costs and Availability	• Cost / availability of application software	• Greater availability of shrink-wrapped software • Superior development environment results in lower application costs
Ease of Training and Use	• Cost / number of hours training required	• Lower training costs because of standard graphical user interfaces • Greater availability of training for standards-based applications
Development Costs	• Cost • Time to market	• Computer Aided Software Engineering tools more prevalent • UNIX provides a technically superior, cost-effective development environment
Portability	• Cost of porting • Time required to port	• Portability almost impossible across proprietary environments • X / Open Common Applications Environment helps ensure portability of applications

Applications Interoperability	• Cost to integrate applications	• Interoperability easier to establish between UNIX / POSIX-based applications

7. IMPROVE DATA MANAGEMENT		
Objective	**Suggested Metrics**	**Open Systems Add Value**
Data Interchangeability	• Ease of data sharing • Cost per conversion • Time required for conversion	• Consistency of data formats

8. IMPROVE SYSTEMS CONNECTIVITY		
Objective	**Suggested Metrics**	**Open Systems Add Value**
Local and Wide Area Networks	• % systems interconnected • Ease of connectivity • Cost of connection	• Open networking standards - common set of open protocols • Greater degree of vendor flexibility in connecting open systems
Systems Interoperability	• Ease of system integration • % direct links without translation • Cost of protocol conversion	• Less data protocol conversion / translation with open systems • Adherence to standards minimises the amount of conversion / translation required
Mgmt of Multiple Systems	• Personnel costs	• Similar system architectures simplifies management of multiple systems

Value Comparison Scoresheet

This is the centrepiece of this approach to developing a business case. It provides the vehicle for determining how effectively alternative information technology strategies meet the organisation's key objectives.

The objectives and priorities identified earlier, as extended with the appropriate measurement criteria, are assigned weightings which balance their relative importance to the organisation. It is now possible to score the various alternative IT approaches on each of the criteria, multiplying each score by the assigned weighting to obtain a quantitative valuation of the alternatives. One of the criteria, clearly, may be a standard ROI calculation that can then be included in the final *Value Comparison Scoresheet* as in the example below. (This example is extracted from X/Open Briefing Set No. 2, Worked Example from *Open Systems: A Guide to Developing the Business Case*.)

Table 5-2 Value Comparison Scoresheet

			Option 1		Option 2	
Measurable Objective	Scoring Criteria	Weight	Rating	Score	Rating	Score
ROI Calculation		40	2	80	6	240
Time to Market						
Share data internally	% connected	5	2	10	10	50
Reduced applications development time	time/cost required	4	3	12	8	32
Appls portability	time/cost to port	4	3	12	9	36
Project management	data availability	3	7	21	5	15
Subtotal				55		133
Customer Service						
Improve problem resolution	information flow & mgmt	5	5	25	6	30
Product updates to customers	documentation flow	4	5	20	6	18
Compatibility with customer system	% key customers	3	4	12	2	6
Subtotal				57		54
User Productivity						
Productivity appls.	number of needed appls.	3	6	18	8	24
System performance	benchmarks	2	4	8	10	20
User training	user evaluation/cost	3	3	9	6	18
Compatibility with current system	compatibility benchmarks	4	10	40	6	24
Subtotal				75		86
TOTAL		80		227		513

Value of the Approach

A crucial determinant in the success of largescale, enterprise-wide technology investments is the degree to which they are perceived to directly support the medium- and long-term objectives of the organisation. Among the significant barriers to the clarification of how technology can help to meet the organisation's goals has been the difficulty in communication that line management experiences when dealing with IS professionals. The approach outlined here addresses this problem directly:

- through involving a wide circle of management and staff early and consistently throughout the process
- through the development of an agreed list of business objectives and priorities which is then linked to the technology attributes that will support them
- through a consensus building approach to the determination of measurement criteria and to the weightings to be attached to each of them
- through the development of a quantified scorecard of the alternative investment approaches.

The resulting business case is couched in terms which are readily understood throughout the organisation and is reviewed by a management team that has been part of the process of its development.

It is rare that the move to open systems can be justified solely on the basis of the payback on an initial investment. The justification comes from the fact that this investment positions you to reap the rewards from future developments. By quantifying the non-financial impacts of the investment and by being able to include them with the financial calculations, the validity of this assertion becomes clear to all concerned.

Convincing the Board

George McCorkell

Making the move to enterprise-wide open systems means getting the Board to commit resources to what is often considered a somewhat risky venture. The U.K. Department of Social Security (DSS) started this process in 1984 and offers a number of lessons in how to keep senior management appropriately informed and involved.

Department of Social Security

The U.K. Department of Social Security (DSS) is a very large organisation responsible for administering welfare payments, known as ‘‘benefits’’, to citizens throughout the U.K; some 39 million customers. These payments total in excess of £120 billion a year and represent close to one third of total U.K. public sector spending. There are over 30 different types of benefit, any of which may be modified or added to at very short notice as changes in government policy dictate. The requirements for flexibility and rapidity of response were an early key to the attractions of open systems for the Department.

Although the DSS is a single government department, it is actually split into four autonomous agencies. Three deal with the primary business of the Department:

- *Contributions Agency* — responsible for taking in National Insurance contributions, the tax that funds benefit payments.
- *Benefits Agency* — responsible for paying out benefits.
- *Child Support Agency* — responsible for child support benefit.

They all deal with the same population and, in effect, have a need for the same basic database. However, although they are part of the same Department, they are allowed to function autonomously, spending money on things they consider priorities. How are common priorities met? Who integrates the pieces? The *Information Technology Services Agency*, the fourth agency, is responsible for technology services to the Department. It operates with a £450 million annual budget (U.S. $700 million), employs 4,500 staff and delivers systems to about 2,000 locations spread right across the U.K. Its network consists of more than 160 mainframe nodes in 4 national computer centres and 4 other major centres, 50,000 terminals, 10,000 personal computers and 3.5 terabytes of data storage. The traffic across the network is of the order of 50 million transactions per week.

The Board

In considering the long path that we have followed in moving to an open systems environment and how we were able to convince the Board to make the necessary moves, I pondered on the nature of the Board with which we are dealing.

The best analogy I can devise is to compare it to an elephant. It is a majestic sort of animal, large and powerful. It looks a little slow, but it is really very intelligent. It is normally very calm and placid, but if you frighten it, it is liable to charge and cause absolute mayhem. It tends to move rather slowly and it is difficult to speed it up. If you want to change its direction, that is even more difficult. An important factor in comparing my Board to an elephant is that, like the elephant, it never ever forgets!

How do you set about convincing your Board of the value of open systems? Well, you recall the old elephant jokes. How do you eat an elephant? You eat it one bite at a time. How do you convince your Board of open systems? You do it one step at a time.

Starting the Process

The first bite of this elephant goes back to 1984. The Department of Social Security started computerisation in the 1960s with lots of magnetic tape-based batch systems. We carried those into the 1970s, and in that decade we did nothing new — we stayed with batch systems. We therefore arrived in the 1980s well behind the technology curve with no on-line access to any of the systems. This created a huge opportunity as we moved to revitalise the Department's use of information technology.

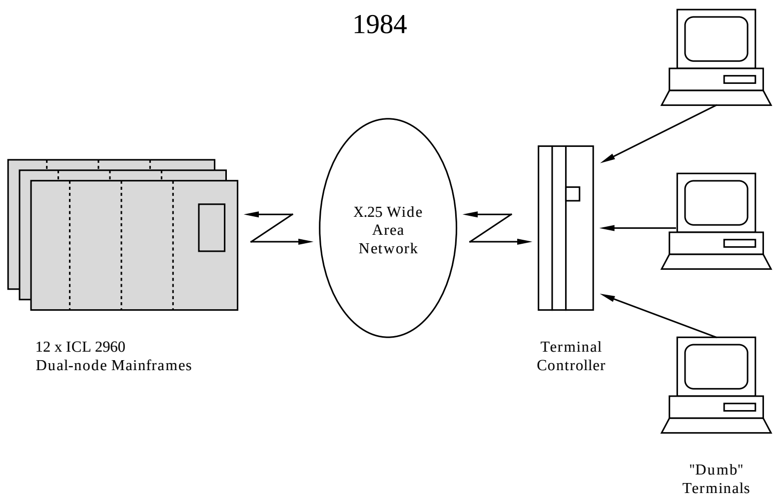

Figure 6-1 Unemployment Benefit System

The Unemployment Benefit System was chosen as the starting point for this modernisation and revitalisation project. The key issue to be resolved concerned the provision of on-line terminal access from 800 Unemployment Benefit Offices widely dispersed across the U.K. to applications running on ICL mainframes located in two data centres — one in Southern England (Reading) and one in Scotland (Livingston).

The resulting solution involved the introduction of 20,000 ''dumb'' terminals on-line to a terminal controller in each of the remote offices where some local processing took place. Communication with the mainframe applications occurred through a remote batch stream over a packet switched wide area network.

One of the advantages of entering the wide area networking field ten years after most other comparable organisations is that we were able to learn from their experiences. In the early 1980s, it had become clear that layered communications protocols provided significant advantages in terms of both flexibility and growth options; and, by 1984 the first practical implementations of the International Organization for Standardisation's Open Systems Interconnection (ISO / OSI) standards were being released. Consequently, we made our first foray into open systems by implementing our new system over an X.25 packet switch network with OSI transport protocols.

How did we convince the Board to allow us to do that?

Very simply. We convinced the Board that there was a large payback from providing on-line access to the central applications, predominantly through staff savings. They were not interested in the means by which the savings were achieved, just that they were achieved.

So this is the first lesson on how to keep the elephant happy — do not upset it by using strange or technical words because it might start charging at you. Talk to it in its own language and it will remain a happy elephant.

The Operational Strategy

In the mid-1980s, not only were there still a large number of batch systems, there were many systems that had not been computerised and many offices with no computer support at all. So, in 1986, we developed what we called the Operational Strategy Benefit Systems. This was, in practice, a set of interlinked projects to computerise a number of benefit systems and to distribute access to the resulting applications into local offices throughout the country. The applications themselves ran on central mainframe systems.

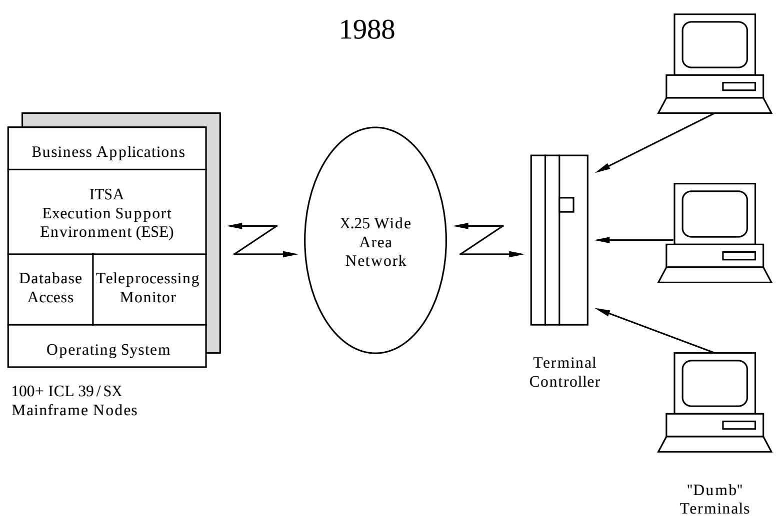

Figure 6-2 Operational Strategy Benefit System

By U.K. government mandate, these new applications were implemented on ICL mainframes. Despite this, we wanted to start to make more moves towards the open systems world. Thus, we quite deliberately developed our own infrastructure to insulate the business applications from the underlying proprietary software base. This infrastructure, which is known as the Execution Support Environment, supplies the critical interfaces that allow interoperability with, and portability to, other execution environments. By this means, we were able to introduce a structured, layered software model that allows for the replacement of individual hardware and software components — a key element in the transition to open systems.

The payback once the systems were fully implemented — $10 million per month — together with the size of the investment involved attracted the attention of the Board. This time they demanded full details of the investments which were planned in order to assess the realisability of this huge payback. In describing and justifying the investment required, we did not use terms like open systems. We justified it on the basis of the real business benefits that derived from implementing an open systems solution:

1. We were able to standardise a number of parallel application developments so that users who had never had IT before could get standard screen presentation and standard methods of operation irrespective of the applications involved. The reduction in training and re-training costs that this enables is a real business benefit to the user community.

2. As the Department had virtually no on-line applications, we were now faced with developing terminal-based business applications with very few people who had any experience in such developments. However, the Execution Support Environment infrastructure had the effect of isolating the technical issues surrounding on-line systems from the developers of the business applications. This benefited the Department both through faster development and through our ability to employ development staff with lower skills and consequently at lower salaries than those commanded by real-time systems specialists.

3. In our moves to standardise the mechanisms used to implement terminal-based systems, we decided to adopt the principle of a generalised ‘‘virtual terminal’’. In 1988, the OSI virtual terminal standards were not mature enough to meet our needs, so we

developed our own approach which we called the Departmental Forms Protocol. This enabled us to deliver the benefits of standardised user presentation and, as important, a dramatic reduction in transmission costs through the reduced network traffic offered by the use of the protocol.

The Board Demands Open Systems

With the approval to initiate the Operational Strategy Benefit systems, we had made our second step, but we were still not really talking about open systems. However, events now conspired to bring forward the circumstances in which we could become explicit about our overall goal — multiple supplier open systems. Two problem areas arose that led the Board to demand that we move to that approach.

Firstly, it became apparent that some of the hardware on which we were implementing the new systems was not of production quality. The stability was just not adequate for a set of users whose whole attitude to the use of information technology in their jobs was going to be shaped by this first experience. The Board instructed us to find an alternate supplier. Fortunately, as a result of the infrastructure that we had developed, we believed that changing hardware vendors would only introduce a six-month delay into the nationwide rollout. Unfortunately, six months represents $60 million of savings; a not inconsiderable sum to any business, but a sum that can attract great attention when it is the public sector. In fact, we overcame these problems without delaying the rollout, but our elephant never forgets. It remembered the threat of losing $60 million.

The other problem with this system became apparent at the end of the three-year rollout programme. As with all Government procurements, we had undertaken this one through open tender. At the time we let the contract for the system — late 1986 — it was very competitive. You could not have bought a terminal system cheaper in the U.K. However, by the time we had finished the rollout in 1991, it had become a very expensive system courtesy of the industry's rapidly improving price/performance curve. The Board, not surprisingly, expressed considerable concern at the high price that we were paying. It then remembered what had happened three years earlier with the threatened loss of $60 million and demanded two things for future IT investments:

1. total security of supply with no reliance on a single vendor who may not deliver product that works
2. a procurement process that allows the Department to take advantage of the continuing reduction in hardware costs.

So now the Board was instructing us to follow an open systems strategy! It told us to go there because it recognised real business benefits to such an approach.

Therefore, in 1991, we gathered together information from open systems guides and standards, such as X/Open's XPG3, and developed an operational requirement statement. This told the IT industry that the DSS wanted totally open platform systems that met a clearly defined set of standards.

We ran a competition, and rather than selecting one supplier, we selected three: IBM, ICL and Siemens Nixdorf Information Systems (SNI). Thus, we now have three suppliers who produce similar capability machines that can act as both communication servers and application servers. They are fully open platforms with all the software between them being totally portable. We can and do mix-and-match as we wish.

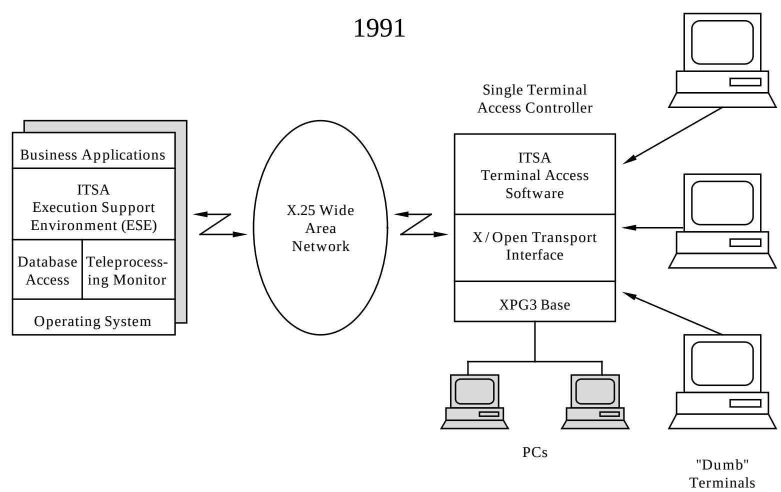

Figure 6-3 Single Terminal Access Controller

So, we have achieved our goal and moved the Department to a fully open systems approach. The benefits are clearly recognised and openly acknowledged by the Board:

- The Information Technology Services Agency's clients, the autonomous operational agencies, now have the flexibility to choose the different approaches that best suit their needs.
- There is ongoing competition between our three strategic suppliers ensuring that we keep up with price / performance trends.
- We share the risks of new projects by undertaking many of our developments through our suppliers.

Collaborative Development

This would seem to be the end of the story. But there are no mature open systems standards yet for a number of areas involved in the operation of a largescale mission-critical information systems environment. This means that there are now a number of activities that have to be undertaken in-house that, in earlier proprietary environments, were primarily handled by the vendor; areas such as systems management and systems-level integration testing.

This has significant budgetary implications — a fact quickly picked up by the Board when they reviewed the 1992 / 93 budget and saw a new line item, ''Integration and Testing - Open Systems''. As was seen earlier, our elephant never forgets and it quickly pointed out that when we had proprietary systems there were no expense items of this type. Unfortunately, in following the Board's instructions to move to open systems, we have effectively unbundled our systems, taking components from a number of different suppliers. There is, therefore, a new task of putting these components together and testing that they work.

However, the Board is not willing to see all the savings from open systems — and there are major savings in terms of procurement — given back in the form of increased expenditure in the test and integration area.

The answer has been for us to change our relationship with suppliers and have them take back the responsibility for the problem. After some discussion on the requirement, our suppliers have agreed that the products they provide in the future will not just conform to the specified standards, they will be tested to work in our technical environment. The suppliers take responsibility for integration testing of the product sets that are needed to meet our operational requirements. This is the type of cooperation that we had expected to develop when we chose a set of strategic suppliers rather than a single remote vendor.

In summary, what this all demonstrates is that you not only need to eat the elephant a bite at a time, but that getting a few friends to help you increases the chances of success.

Rightsizing to Open Need Not Mean Smaller Systems

Peter C. Bauer

The introduction of open systems computing implies to most organisations changing the structure of their applications and moving to a number of smaller networked machines. Marshfield Clinic challenged that assumption by converting their applications to run in a mainframe UNIX environment. The need for openness in their computing environment was successfully balanced with their desire to minimise the impact on end users.

Marshfield Clinic

To most people in information systems, the idea of undertaking a conversion from one environment to another is one of the most frightening thoughts that they can have! Many a data processing manager has had a career change because of a conversion, and a lot of businesses have lost millions of dollars from such conversions.

In May of 1990, the Marshfield Clinic faced it. We had to convert.

Marshfield Clinic is a multi-speciality referral centre in central Wisconsin. We have 400 physicians: 300 in Marshfield and 100 spread over our 17 regional centres, mostly in central and northern Wisconsin. The physicians own the Clinic. We see approximately 3,000 patients each day, with about 900,000 patient encounters a year. We also own and operate our own health maintenance organisation (HMO) which has about 65,000 members.

Our systems were written in-house, all in COBOL, and consist of about 3,000 library routines and 1,100 main programs, adding up to 1.5 million lines of code. The transaction processing monitor and the database management system were written in-house.

From 6.00 a.m. until 6.00 p.m. we support 280 on-line applications, such as soundex, patient demographics and appointments — with over a million patients on-line — medical record tracking, viewing of laboratory results, radiology interpretations, insurance eligibility, charge data entry and claim entry for our wholly-owned HMO subsidiary.

At night, we switch to a reduced on-line system and run batch processing. We rate about 20,000 charges per night. We print laboratory and radiology patient summary reports for our 3,000 out-patients, and also for the in-patients in the 530-bed hospital that we work with. Patient billing is run and third-party claims are filed with Medicare, Blue Cross, and so on.

At the time of our conversion, the database consisted of some 250 files. The one with the largest number of records was laboratory results with 36 million records, the results of laboratory tests done for the hospital and the Clinic since 1986. Another large file was our charges file with 10 million records; 3 years' worth of charges are maintained on-line. Prior to conversion, this file took up about 5 gigabytes of storage, while the whole database was 25 gigabytes.

Our database management system uses interactive development tools that allow programmers to enter file information which is stored in a data dictionary. The file access routines were, and still are, generated automatically in COBOL. In other words, the programmers do not write the COBOL; it is generated. This was important in our conversion. On the new system, we generate COBOL and C for the file access.

This all ran on 3 machines: a Unisys V560 and 2 Unisys V380s, one for test and development, the other for *ad hoc* reporting.

The Decision to Convert

Our computing needs had been growing at the rate of about 25% per year, somewhat higher than the actual growth rate of the Clinic. Until the mid-1980s, the focus of the IS department was primarily financial and administrative systems, such as appointment scheduling. However, in the last few years, virtually all new doctors are computer-literate and attention has moved to medical applications, such as laboratory, radiology, progress notes and discharge summaries, all of which are on-line and available at any PC on the wide area network.

The Clinic had been using Burroughs Medium Systems, the predecessor of the V Series, since the first computer was installed in the 1960s. We had been running on the largest available machine for many years, and since it was proprietary, we were doing everything we could to avoid converting. We struggled in the mid-1980s to run on multiple small machines, but our database is highly integrated. We survived, but it was not easy.

In 1987, Unisys announced that they would have a whole new line: the V500 Series. We would be able to start with a large single processor system, grow to a dual processor, and then to three and four processor systems. We took delivery of one of the first single processor systems in January 1988 and, in January 1990, upgraded to a dual processor system. But, in May 1990, we got word that Unisys were not going to produce larger machines in that architecture.

Faced with a conversion, we decided that we would look at all the options. After a preliminary study, working together with our consultant Achi Racov of MSS International, London, we narrowed the list to just three valid possibilities:

- IBM's MVS environment
- Unisys A Series (formerly Burroughs Large Systems)
- open systems (which we equated with hardware operating under the UNIX operating system).

Our first thoughts were that a Unisys A Series solution would be the easiest and least expensive option; that the move to IBM would be a much more difficult conversion, but would put us in the proprietary mainstream; and that a UNIX-based solution was the least probable — very desirable, but difficult to pull off.

We worked within a major constraint in that we estimated that, with normal growth, the old machine would reach unacceptable response times by the end of 1991. Therefore we set the goal of completing the conversion by 1st December 1991, and basically worked backwards to set all our other deadlines. We also decided that we would just convert, not try to redesign and convert at the same time; that meant we would not look at distributing our applications and databases over multiple machines. This seemed to rule out UNIX, which is typically viewed as running on mid-range machines, not big enough to support us.

Benchmarking

To help make our conversion decision, we designed a set of benchmark programs. It is very difficult to compare Unisys V Series with other machines because it was designed as a COBOL machine and its instruction set matches the COBOL verbs almost one-to-one. Therefore, comparisons of such measurements as instructions per second between a Unisys V Series system and other machines is almost meaningless. For us, therefore, benchmarks were extremely important.

People often believe that they are going to benchmark a new machine by running their production programs. This is normally not practicable, particularly where the systems are highly integrated, there are too many programs to run and too many files need to be present before anything will work. We therefore designed benchmark programs to model the number of inter-program communications, data moves, computations, and so on which exist in our typical applications. We ran these, together with input/output (I/O) benchmarks, on the Unisys and IBM equipment. We felt that it was very important to benchmark the disk I/O as well as the CPU power because many mini-size machines have a lot of processing power but do not do well with high-volume disk access.

By then it was July 1990, and we had done very little with the UNIX option. We had studied some of the smaller machines earlier when we were looking at clinical research databases, and felt that none of the small machines would support the I/O volume that we had to run. We did not believe it would be possible to find a machine big enough to run our systems unchanged under UNIX. However, almost by chance, we discovered that Amdahl had a mature mainframe version of UNIX System V, called UTS, running in native mode on their mainframe systems. This made UNIX a real possibility. Amdahl's machines are capable of far more I/O than we needed — they support large file systems with multiple blocking factors — and the native-mode implementation assured efficient operation. We worked with Amdahl to run the benchmarks on their systems, and they worked well in that environment also.

In September 1990, Carl Christensen, our technical specialist, and I went to the U.K. to visit sites where MSS International, our consulting firm, had successfully converted V Series systems to IBM, to Unisys A Series or to UNIX-based systems. All of these conversions had worked, so we felt that all of our options were viable, and we could have the luxury of picking the best long-term solution.

Proposals

After all of our studies and benchmarks, we worked with each vendor on machine sizing, and received formal proposals from IBM, Unisys and Amdahl:

- The IBM proposal was for a 3090-280J, and a contingency for an additional machine for test and development.
- The Unisys bid was for an A16 for production and an A12 for test and development.
- Amdahl's offer was for a 5995-700A with three ''domains'', or virtual machines, running — one for production, one for development and one for *ad hoc* reporting — which would match the environment we had been in with three machines. It also included 110 gigabytes of disk capacity.

To help make our decision, we made a five-year projection for each option including:

- initial cost of hardware and software
- estimated cost of the conversion
- maintenance costs
- estimated cost of processor and disk upgrades for the five years.

The Amdahl option running UNIX UTS had the lowest cost. The Unisys A Series proposal was only about $100,000 more for the five-year period, and the IBM solution was the highest at about $3 million more than the other options over 5 years. Interestingly, the two lower cost options, the Amdahl and Unisys proposals, were less expensive than our projections for the five-year period for our old systems, assuming planned upgrades occurred.

The Choice

We picked the UNIX/Amdahl combination. Although the low price was an extra bonus, the primary reason was not the initial cost. We feel that there is a huge advantage in going to UNIX. The connectivity, portability, scalability and interoperability of UNIX mean that we will not have to go through another conversion like this again. It opens up the potential for access to software from many vendors. It gives us connectivity undreamed of in our old environment. It gives us the option of running our software on a variety of hardware platforms, including smaller platforms at our satellite clinics if we should decide to go that route. It makes it easy to move to client/server processing, and it means our future hardware vendors have to compete with each other for our business, which should hold prices down.

We picked Amdahl hardware specifically because it is excellent, reliable equipment and because they had the best mainframe UNIX implementation that we could find.

Convincing the Board

The information systems staff had done most of the analysis necessary to make our choice, and now we had to take this information to the owners of the Clinic, the physicians.

First, our recommendations went to the physician computer committee, from there to the executive committee, and lastly to the Board of Directors which is composed of all the physicians who have been at the Clinic more than four years. There are now about 300 doctors on the Board of Directors which meets once per month. Our message was simple: open systems would provide us with a good price/performance ratio, an environment where our programmers could be highly productive, and would provide the Clinic with much-desired vendor independence.

This last point was particularly important to the Clinic. We had been having so much trouble through the tie to one vendor that the Board members were very interested in any solution that would make us more independent. Each committee approved it and, in October 1990, the Board of Directors agreed the proposal with no negative votes.

The Conversion

Approach

MSS International was extremely valuable in helping us to organise the conversion. Though we had complete understanding of our software, we had never undertaken a major conversion and we needed their expertise. They also gave us an outside opinion on our technical design for what we called ‘‘the sanity check’’.

Once you have picked your direction, conversions turn out to be exercises in making sure that you take care of every little detail. We broke everything down into small tasks with deadlines that had to be met. The focus was incredible and the work was intense. The whole staff worked very hard, work-weeks of 70 and 80 hours being not uncommon.

The actual conversion started in October 1990 with the systems programmers developing routines in C Language to support the infrastructure of the converted system. These routines are mainly library routines bound to COBOL programs performing inter-program communication, making calls to the operating system from within COBOL, and performing database I/O operations. The hardware was installed in November 1990.

Application program conversion started in earnest in May 1991, and at this point we froze our systems with only emergency programming changes being allowed. We switched our payroll system over to UNIX on 1st July 1991 as a prototype to test our total design. We also wanted to see if our benchmark and stress tests were giving accurate performance predictions, and payroll is our most isolated system in terms of the number of files affected. This went very well. The last two months were spent testing, testing and testing.

As was mentioned earlier, we did not re-engineer our software. The great advantage of this is that the system should work exactly the same way on the new machine as it did on the old one. We could, and did, capture real on-line transactions on the old machine and then move the corresponding files over onto the new machine, run them again, and compare the results. If the results were not identical, we studied them until we found out why. We compared the outcomes of batch runs on both systems and went through the same process. Another advantage of not re-engineering is that, on the day after the conversion, to the end-user the system looks exactly the same as it did the day before the conversion. They really had very little change at all in their lives.

Once the programs were all converted, the actual logistics of converting the data files was the last problem. We knew how to do it, but all the data had to be converted from the old system before we started running on the new one. We decided we would do this on the Thursday of the Thanksgiving holiday because if all the data files were not converted by Friday morning, we could run with a reduced system on Friday and still be assured of finishing the conversion by Monday morning. The total system was cut over at 2.00 a.m. on Thanksgiving day, 28th November 1991, 3 days before the original target!

Results

The cost of the conversion to the Clinic was about $5.5 million. This includes the hardware less the resale of the old system, some network upgrades, our staff's labour and MSS' consulting fees. It does not include the lost opportunity to develop new applications while we were tied up in the conversion. At various times in the project, we had about 24 application programmers, 8 systems programmers and administrators, and 10 trainers and hardware support people working on the conversion. In fact, there were 70 people in the department at the time and virtually all of them were involved in some way or other.

The day after Thanksgiving is a low-volume day at the Clinic and we were able to shake out some of the miscellaneous problems. On Monday 2nd December, we set a new record for the number of transactions performed on the computer system, and then on the following day we broke that record. Meanwhile, the internal response time went from an average of 0.72 seconds per transaction on the old system to 0.26 seconds on the record setting Tuesday. The response time at the terminal averages about 0.5 seconds.

Although we converted the applications with virtually no design change, the system infrastructure had to be changed because UNIX works quite differently from our former operating system. Fortunately, the UNIX infrastructure design that we originally developed proved to be excellent. Though many details were worked out along the way, the basic direction never changed. We use TCP/IP and IPX to communicate with 20 Novell network servers that currently support about 1,800 devices. All of our users are on networked PCs using terminal emulator and messaging software that we developed in-house.

The database nearly doubled in size to about 45 gigabytes in the process of converting. Data storage was very efficient on the old system and, at first, we were very worried about the data conversion. Having our own database management system and code generator was very useful, however. We were able to automate most of this by generating new database access routines, as well as extraction and conversion programs. We also used standard tools found in UNIX to write our own program filter, so program conversion was mostly automated.

A point to note for anyone considering a conversion. If you do not have a strong technical staff, you may need a large amount of outside help. We have a very strong technical staff, people who have worked for vendors, undertaken hardware design and written compilers, and we were able to get by with relatively little direct assistance from our consultants.

Conclusions

We feel that UNIX does not guarantee open systems, but it is almost impossible to be open without it. We have deliberately done several things to keep our new system open. Wherever possible, we have avoided the so-called ‘‘value added features’’ that vendors like to push. When we do use them, we modularise them so that they can be easily modified for another platform. We decided to do all of our I/O routines in C rather than COBOL because data storage with COBOL on UNIX is not standardised from one COBOL compiler to the next, whereas it is with C.

UNIX still has some weaknesses in a large data processing environment. We anticipated lack of features in system logging, tape handling and volume printing, and we were right. We have had to develop our own solutions in these areas, but it was not a major problem. The weaknesses that do exist are being addressed by the software vendors and I expect that UNIX will soon catch up with the legacy environments.

Portability

The open environment has more than lived up to our expectations. We started our development efforts using SCO UNIX on PCs until the Amdahl system was installed. Importing that code to the mainframe was trivial. We also run UNIX on Data General systems for gateways and print servers, and as an interface to the hospital computer. The portability and connectivity here have also been excellent. The argument that UNIX is not standard and therefore portability does not exist is tremendously exaggerated, though I do believe that the user community would be far better served if some vendors would stop trying to create their own versions of UNIX.

One example of portability involved our only purchased software — our general ledger and accounts payable package from Lawson Associates. They had a UNIX version of their package, but had not ported it to Amdahl's UTS. They brought a tape containing their source code to the Clinic one morning and had it running before the evening.

Pseudo Openness

An example of what I would characterise as ''pseudo'' openness relates to our experience with LU6.2. LU6.2 is supposed to be an example of how certain proprietary systems are really open. We have an interface to the hospital next door which we converted from looking like a dumb 3270 terminal to a peer-to-peer LU6.2 interface. We can interface applications between UNIX boxes in an afternoon. The LU6.2 interface from a Data General UNIX system to an IBM proprietary system took three and a half months of struggle, with both vendors involved. I know this does not always happen, but my point is that just because you are claiming to have published your specifications for your way of doing things, that does not make your system open.

Productivity

The staff have accepted the new environment very well. While some had had UNIX and C experience, the majority had none. We had people on our staff who had been trained on Burroughs equipment in college and had never worked in any other environment. We were worried about that, but they accepted it extremely well. We had about three days of formal UNIX training and two days' instruction on the COBOL compiler. We also conducted an in-house seminar on UNIX internals for those who were interested. The rest of the learning was from manuals and shared experience. We hired no additional staff and no one quit. Later, we ran a class on C for anyone who was interested and 42 people took the class — just about everyone who was eligible to take it.

Productivity has increased. In the process of doing the conversion, the staff learned UNIX very well. Consequently, as soon as we had finished converting they were ready to take advantage of the new environment. They like UNIX better than the old system and I estimate that they are probably 20% more productive than they were in the old one. Productivity is very important to us. The health care industry is under the microscope these days and we are all under great pressure to reduce costs and become more efficient. We have a lot of new development to undertake to help improve operational efficiency within the Clinic, so our own productivity is extremely important.

Reliability

We have found that although mainframes cost more than mini-computers on the basis of millions of instructions per second (MIPS), they do provide added value in terms of reliability and service. We need very high system availability to run our systems 24 hours per day and 7 days per week. The mainframe has provided that. I have also been pleasantly surprised to find that we can buy disk systems for less money on the mainframe than on the minis. We will continue to look at downsizing, but we will look at more than just dollars per MIPS.

Whenever we look at new systems, we will want to preserve low pricing, excellent performance, high productivity, hardware and software independence, good reliability, accessibility and serviceability. I do not think downsizing is inevitable for everyone. I do think in a few more years the terms ‘‘mainframe’’, ‘‘mini’’ and ‘‘micro-computer’’ will be almost meaningless and we will pick the size of the machine that serves us best without regard to the old labels. While hardware gets cheaper and cheaper, software is getting more expensive. If pricing of software is per machine, then distributed processing is going to get very expensive.

In conclusion, I would say that at Marshfield Clinic we are dedicated to open systems. Today, in addition to running our mainframe system on UNIX, we are also using SCO UNIX and Oracle for registries, cardiology and oncology. We have converted our laboratory system to UNIX running on a Data General Aviion 8000. Next, we are bringing up UNIX-based electronic mail for the medical complex.

I believe that mainframes still have a place in open systems. We had originally thought that when we had completed the conversion, the next step would be downsizing. Now I am not so sure. Downsizing is not for everyone, and is not required in order to be open. If you have a large integrated database, as we do, and you have requirements for very high-volume I/O with guaranteed data integrity, then mainframes should be considered. We have also found that the administration of 20 servers and 6 UNIX machines is a lot more work than administering one mainframe.

So, open systems absolutely. But before downsizing to a lot of little boxes, remember 1,000 Chihuahuas can nibble you to death!

Lessons of a Multi-vendor Open Systems Pioneer

Mark Schmidt

The move from proprietary systems to multi-vendor open systems brings a large array of business benefits. However, the full realisation of these benefits is hampered by both process and attitude problems within the information technology community.

Open Systems in Wal*Mart

Wal*Mart is a retailer operating close to 2,300 outlets in 48 states of the U.S., in Puerto Rico, and in Mexico in a joint venture with a local retailer, CIFRA. We open approximately 175 to 200 new outlets every year in the U.S. alone.

Wal*Mart is known throughout the retail industry for the level of technology that is used to support the stores, specifically in areas such as replenishment — restocking shelves in fewer than 10 days — and in customer checkout — speeding customers through the checkout to help make their shopping experience pleasant.

Most of this technology is open systems-based. We have close to 3,900 distributed UNIX systems that operate throughout the enterprise, with more than 3,800 of those operating remotely in the U.S., Puerto Rico and Mexico. In the two years 1991 and 1992, we spent in excess of $100 million on open systems for the most part on downsized technology operating in a distributed client/server environment. That sum exceeds by a significant amount what has been invested by the company in legacy proprietary mainframes, of which Wal*Mart still runs a few.

Homogeneous Open Systems

Wal*Mart moved to open systems to add an application, not by ‘‘moving to open systems’’. We bought a turnkey pharmacy application in 1986 that happened to run on an IBM RT-PC, a machine that operated under IBM’s UNIX variant, AIX. We were comfortable doing this because, while it was

UNIX, it was IBM-supported UNIX and we were still predominantly an IBM shop. At that time, our computing model was based upon a centralised data centre, mainframe environment. We had a trusted relationship with IBM and had long depended on them. We also happened to have about 900 Series/1 systems that ran in our 900 Wal*Mart stores. So, we went into the pharmacy application — and UNIX — with IBM by our side.

As our comfort level increased, we decided to implement our HyperMart system on UNIX and the RT-PC. This was installed in Dallas at Christmas 1987. We liked what we saw with UNIX and developed a receiving application that was implemented in January 1988 on the RT-PC and AIX.

At that point, we had quite a number of UNIX systems between the 900 pharmacies, the distribution receiving centre and our HyperMarts. But what we had was a homogeneous open environment!

This seems a little like an oxymoron and, while it may be possible, one of the things we learned is that it is not really practical.

Multi-vendor Open Systems

In February of 1988, we began to see that we were never going to realise all the benefits of an open environment if we chose to continue this single-vendor approach. At that point, we made a conscious decision to become a multi-vendor open systems user.

We began to rewrite our Series/1 applications and, to ensure that what we were building would give us the benefits of portability and openness, we decided to build them on Hewlett-Packard, NCR and IBM machines. We used standard middleware, compilers and access methods to ensure that the applications were easily portable and in order to minimise any dependencies on proprietary interfaces. In fact, as we were doing the development through 1988 and 1989, we followed the *X/Open Portability Guide.*

The result was a system that is about 99.9% source code compatible between platforms and is now relatively easily ported from one platform to another.

In 1990, after testing through the first half of the year, we began installing all of our new stores — about 15 to 20 stores in an average month — with the new UNIX systems. We waited until 1991 after the holiday season — not wanting to introduce too dramatic a change through such a critical period — to roll out the systems to our other stores, replacing all, by now, 1,600 Series/1s. We did that with HP 9000s beginning in February, and then beginning in June over a rather hectic 12-week period we rolled out 800 NCR 3000 systems, both systems running the same application code and spread more or less equally across our trading territory. 1992 was every bit as brisk as earlier years with the deployment of 500 IBM RS-6000s running AIX Version 3, in effect our fourth variant of UNIX.

It was clear at that point that multi-vendor implementation was not that challenging, that portability of applications was real, and that many of the things that open systems promised to do for us, they were in fact providing.

TCP/IP

In 1992, we were able to implement the TCP/IP protocol stack across our entire enterprise. To achieve that, we had IBM port their OS/2 TCP/IP to run on the 4680 point-of-sale system — a PC-based system that provides price look-up services to electronic cash registers that operate across the front of our stores. We put TCP/IP onto the 4680 to internetwork both on the local area network within the stores and our enterprise-wide network. We operate a satellite network for our primary communications medium so we also worked with Hughes to put TCP/IP stack support into the satellite system.

Largely because we were able to get TCP/IP support into the stores and onto the local networks, we were able to implement a wireless local area network within our stores. We installed 20,000 MS-DOS-based, hand-held, wireless terminals that are connected over the Ethernet LAN via TCP/IP. These terminals weigh about two pounds which means that our store assistants can operate on the floor, as opposed to having to go to the back rooms.

So, today we have two standard communications protocols in our enterprise-wide network, IBM SNA and TCP/IP. New developments take place over TCP/IP.

There are a range of other devices that are all internetworked using TCP/IP to share data and services. These include various graphical workstations — the majority of which (1,500) are MS-DOS Windows, but also several hundred are IBM OS/2 systems, about 100 are Apple Macintosh machines and X-terminals — Novell servers, UNIX Informix database servers, as well as a large NCR Teradata relational database machine.

In addition, because many legacy applications continue to exist in IBM MVS and CICS environments, there are TCP/IP links into both our VM systems and our three MVS ESA mainframes.

Observations

As I review the past six years of our open systems evolution, a number of questions and observations come to mind. The primary question is: Was UNIX the right platform for us to choose for application delivery? The answer is clearly yes. We have been very happy with UNIX and it is clear from our vantage point that we have been able to save tens of millions of dollars through the use of open systems as opposed to proprietary systems of similar capacity.

Open systems have lived up to their billing in terms of portability, interoperability and the freedom to use multiple vendors.

Client/Server Computing

We began implementing client/server systems at about the same time that we entered the open systems world. We are now downsizing and not implementing new applications on our mainframes. We have yet to run into an application that we cannot implement in a client/server architecture.

A simple example will demonstrate the savings that can result from downsizing an application to a client/server approach.

We had a large SNA data communications system that managed most of the communications with our stores. On a typical day, we would move about four gigabytes of data back and forth between stores providing information needed for replenishing stores or giving the store inventory control information. The application ran on an IBM ES/9000 machine, operating under MVS/ESA with CICS and VTAM.

We began to run out of capacity on that particular system. We literally had periods when we would back up data because we could not get it through the system fast enough. We came up with the idea of moving the application to a client/server environment. Converting it over to two HP

827s and twelve IBM RS-6000 320s divided the traffic across two separate subnets with separate routers and hubs. This meant that the new system had no single point of failure. The network is accessible on a wide area basis by a simple socket call from any place on our network in order to queue a message for transmission. We have found the new system can support roughly 50-70% more data throughput than we were able to get with the earlier, mainframe-based implementation. The new application including all the equipment, software and application programming cost no more than $500,000.

Disappointments

At Wal*Mart we absolutely agree with standards. Over the years, we have developed our own standards-based architecture including, where necessary, our own standards that we provide to suppliers of such things as hand-held equipment and store multi-media workstations. However, until recently Wal*Mart has not been visible in the open systems world, neither in terms of promoting open systems nor in participating in standards groups. We had chosen instead to invest our time and energy in our business applications. When we needed support for something, we applied pressure directly on our open systems suppliers to include something that we needed. However, there are areas where we have experienced definite disappointment and where it has become clear that we need to work with other users to address the problems.

The Standards Process

We greatly believe in what X/Open has done and we appreciate users coming together to try to help move the process along.

We still have concerns about the process by which standards are defined and implemented:

1. Precision — Too many standards are too imprecise. They are more like working drafts upon which the system suppliers will deliver extensions or improvisations. In general, we need standards to be more comprehensive and specific.

2. Pace — The standards-setting process in general moves too slowly, leading to real problems for users whose needs move faster then the standards. This has a significant impact on the acceptance of open

systems.

3. Purpose — It appears that sometimes the supplier community acts at cross-purposes to the interests of the open systems movement and of users like ourselves. This is seen in such things as delaying tactics in standards groups, proposing or promoting alternative standards through very small consortia, and in sales people who, perhaps unknowingly, liberally interpret their product's adherence to standards.

Compiler Extensions

A second area where we have experienced some disappointment is that of compilers.

While we have been able to gain many of the promised benefits from developing code that is easily portable, as we deal with some of the more subtle areas and become more concerned with the administration and control of these systems, a number of differences between vendors' implementations of ''standard'' compilers come out.

Suppliers, probably as a favour to the users who implement a single-vendor UNIX environment and possibly not realising the complications they were creating for a user like Wal*Mart, have added their own custom extensions. For users who remain in the ''homogeneous open systems'' world it may be fine to take advantage of those extensions, but for those like us who have a multi-vendor environment these extensions complicate life. These inconsistencies add cost and retard the ability to move quickly to deliver new services to users.

Systems Management

An issue that has proven difficult for us to address on our own is the management and control of distributed UNIX systems. This problem spans so many of the vendors and manifests itself a little differently in most of them.

Our experience of the open systems management area is that it is neither as open nor as standard as open systems proponents would claim. Certainly, if you are working with a single vendor, or if you only employ a small number of systems, it is not much of a problem. However, with our 3,900 systems from multiple vendors, we have found that it costs us a significant amount. This is a particular problem as the majority of these systems run remotely in stores where there are no information systems specialists or operators. This is not to say that it is impossible — we do it every day with great success — but the point is that it is nowhere near as easy as it should be. Basic management services in the open systems environment are weak, especially when compared with the services offered by more mature proprietary systems.

Worse than merely weak, most of these services are shamefully non-standard. Examples that our network and systems administrators run into all the time include shell commands that have different parameters, or similar commands that generate different output, or, even more subtly, commands that generate the same content but do so in different formats. This significantly reduces our ability to gain the full benefits from automating management functions.

Wal*Mart has tolerated this position largely because of the compelling economic advantages of open systems, even though it means that we employ about three dozen UNIX systems engineers and systems administrators who can compensate for some of the differences and adapt to some of the inadequacies. Smaller commercial users may not have such a luxury and may not be able to tolerate the kind of overhead this entails. Unless they intend to implement a single vendor, homogeneous environment, all users will have to deal with some of these same problems.

Specifically, common standards are required to cover the following management and control issues:

- systems administration
- configuration management
- performance measurement
- software maintenance and distribution
- backup, archive and restore functions
- fault isolation and correction
- load balancing and scheduling
- print services

- accounting.

These are all areas where there are weaknesses or differences, areas where standards are either non-existent or imprecise, and generally areas where suppliers feel most compelled to add extensions to compensate for weaknesses.

What would I suggest? I believe that there are actions and responsibilities on both the vendor and purchaser sides.

Supplier Role

For the supplier community, I would request that you show us that you are committed to our success in a multi-vendor environment by acting faster to resolve standards for management and control. As a large user, my first priority would be for a common, standard reference implementation as opposed to any elegance or extension that you might add. Your added value should come from such things as reliability, performance of your implementation, local support or, if you would care to take a page from Wal*Mart's book, your position as a low-cost provider.

If you must extend your implementation in these areas, please document it very clearly. In a Wal*Mart store, if a household chemical spill occurs, our top priority is to take out a little orange pylon and set it down. Sometimes we even station an associate right there at the spill while somebody else goes back to get a mop, and we try to have the floor cleaned up in a matter of minutes. I would suggest that if you feel compelled to offer these vendor extensions, you put the equivalent of a little orange pylon in the documentation so those of us who are wandering through, not really looking for differences, recognise that they exist. If you could also move to the standard version as soon as it is defined, you will minimise the impact on our flexibility and speed of implementation.

User Role

For all the grief we may want to give suppliers, users are probably an even larger part of the problem. We tolerate suppliers when they develop their own extensions and do not complain when they act against the development of standards. It is important that we communicate to our suppliers that standard management services, protocols and objects, for example, are a high priority. Most important: vote with your dollars and buy from suppliers that honestly and sincerely support the development of standards in this particular area.

Conclusion

You have seen that Wal*Mart is committed to open systems and that we have clearly recognised the promise of a multi-vendor, distributed computing model. We have pursued it and seen many of the benefits realised within our business. So we are not turning back.

But, we have also seen many of its shortcomings.

The complexity of our commercial computing environments grows and we are becoming ever more dependent on these environments for crucial business benefits like shortening replenishment cycles and improving customer service. In that environment, these shortcomings become more and more significant. They affect the economics of our business model — our support costs go up, our responsiveness goes down, and ultimately our business risks may begin to increase. But we are pragmatic, and we really cannot afford for these things to become big problems. If they do, I would imagine that even our open systems efforts could slow down. I am quite certain that other users, who do not have the commitment that we do, would probably slow down even faster.

If open systems management fails to keep pace with the needs of multi-vendor users like Wal*Mart, and I honestly do not believe that it is keeping pace today, it is very possible that that momentum will slow. If it does, it will not be in the best interests of users in general, nor in the best interests of Wal*Mart, although, in the short term, it may seem like good news for some proprietary suppliers!

Contributor Biographies

David P. Norton

David P. Norton is President of the Renaissance Strategy Group, an international consulting firm specialising in performance measurement and organisation renewal.

Before Renaissance, Mr. Norton cofounded Nolan, Norton & Company where he spent 17 years as President prior to its acquisition by Peat Marwick. He has authored numerous works, most recently *The Balanced Scorecard - Measures that Drive Performance* with Professor Robert Kaplan (Harvard Business Review, January 1991).

Mr. Norton is a Trustee of Worcester Polytechnic Institute and a former Director of ACME (The Association of Consulting Management Engineers). He has served on numerous client steering committees, most recently receiving a Distinguished Service Award for his support to the U.S. Department of Defense on their approaches to Corporate Information Management.

Mr. Norton earned his Doctorate in Business Administration from Harvard University, an MBA from Florida State University, an MS in Operations Research from Florida Institute of Technology, and a BS in Electrical Engineering from Worcester Polytechnic.

Paul A. Strassmann

Paul A. Strassmann's career includes service as Chief Information Systems Executive (Xerox Corporation, 1956 to 1978 and Department of Defense, 1990 to 1993), Vice President of Strategic Planning for Office Automation, Xerox Corporation (1978 to 1985), and Business Consultant (1986 to 1990 and 1993 to date).

In March 1991, he was appointed to a newly created position of Director of Defense Information and the Principal Deputy, Assistant Secretary of Defense (Command, Control, Communications and Intelligence). He was responsible for organising and managing the corporate information management programme across the U.S. Department of Defense which included a $35 billion cost reduction and re-engineering programme of the defense information infrastructure. He resigned this post when the new Administration took office in January 1993, after receiving the Defense Medal for Distinguished Public Service, the Defense Department's highest civilian award.

Mr. Strassmann is now Visiting Professor of Information Management at the U.S. Military Academy at West Point, visiting lecturer on Information Warfare at the National Defense University, Ft. McNair in Washington, and consultant to the Management Executive committee of the AT&T Corporation. He serves on the Advisory Board of the National Academy for Public Administration, on the Scientific Advisory Board of the Defense Information Systems Agency, and on the Board of Directors of the International School of Information Management.

He earned an engineering degree from the Cooper Union, New York, and a master's degree in industrial management from the Massachusetts Institute of Technology, Cambridge. He is author of over 80 articles on information management and information worker productivity. His 1985 book, *Information Payoff — The Transformation of Work in the Electronic Age* has attracted worldwide attention and is appearing in Japanese, Russian, Italian and Brazilian translations. His most recent book, *The Business Value of Computers*, now translated into Japanese and Italian, shows the results of his research on the relation between information technology and profitability of firms.

Christopher Jacobs

Christopher Jacobs is Principal of Philadelphia-based Jacobs International, a consulting company specialising in international information technology marketing and communications. Mr. Jacobs has over 20 years' management experience in corporate strategy, product marketing, communications, customer services and project management with major corporations in high technology. He has had extended experience in Eastern and Western Europe, Asia and the U.S. managing multi-national, multi-disciplinary groups.

Before establishing the consulting practice in 1989, Mr. Jacobs was with Unisys Corporation where his last position was as Vice President Product Strategy and Integration. Prior to joining Unisys' predecessor company Sperry Corporation, he worked for the U.K.'s largest service bureau, Centre-File Ltd., in a range of computer applications development and systems management roles. His career started in local government in England. He was educated at Wolverhampton University from which he graduated with an Honours Degree in Business Administration.

His earlier book, *Discovering OMNIPoint — A Common Approach to the Integrated Management of Networked Information Systems*, Prentice Hall 1992, has been translated into French and Japanese.

George McCorkell

George McCorkell started his Civil Service career in the U.K. Ministry of Defence (Royal Air Force) in 1967, moving to the Ministry of Finance (Northern Ireland) in 1970. In 1974, he joined the U.K. Department of Social Security where he worked on Computer Procurement, Statistical System and New Methodologies. From 1976 to 1979 he was seconded to the U.K. CCTA as a Departmental Liaison Officer and had a year's full-time study at the London School of Economics in 1982.

Since 1985, he has held senior IT management positions responsible for the development and delivery of largescale systems for the U.K. Department of Social Security. He was appointed to his present post of Customer Director for Shared Systems in June 1993.

Peter Bauer

As Director of Information Systems, Mr. Bauer is responsible for all computing services for Marshfield Clinic and its wholly owned subsidiary health maintenance organisation. With a staff of 80 people, Mr. Bauer handles both administrative and patient care processing. During 1991, Mr. Bauer was responsible for the successful, on-time, on-budget conversion of Marshfield Clinic's computing systems from a proprietary mainframe environment to a UNIX-based system.

Prior to joining Marshfield Clinic in 1984, Mr. Bauer was Assistant Professor of Mathematics and Computer Science at the University of Wisconsin where he worked from 1966.

Mr. Bauer was educated at Northwestern University where he received a Master of Science in Mathematics, and at the University of Wisconsin, where he received a Bachelor of Science in Mathematics.

Mark Schmidt

Mark Schmidt is the Vice President Information Technology and Communications at Wal*Mart Stores, Inc.

Wal*Mart is a $55 billion discount chain which has long been an industry leader in information technology. It pioneered the implementation of UPC scanning for general merchandise, satellite communications, electronic data interchange and open, distributed computing. In 1992, Wal*Mart's Information Systems division was recognised by Computerworld and CIO magazine as the most effective user of information systems in the retail industry. Information Week also recognised Wal*Mart with their 1992 award for excellence in information services.

Mr. Schmidt joined Wal*Mart in 1987 as Director, Technology Development. He managed the implementation of the satellite network in that year. He later led efforts to standardise Wal*Mart's EDI linkages with more than 2,000 suppliers, and served for two years on the Uniform Code Council's EDI Advisory Board.

Since 1988, Mr. Schmidt has directed Wal*Mart's move to UNIX-based open systems for in-store computing, and is still active in the company's continued shift towards open, distributed computing.

Prior to joining Wal*Mart, he worked with International Business Machines. He graduated from the University of Arkansas in Little Rock with a Bachelor's Degree in Computer Science.

X/Open

X/Open is an independent, worldwide, open systems organisation supported by most of the world's largest information systems suppliers, user organisations and software companies. Its mission is to bring to users greater value from computing, through the practical implementation of open systems.

X/Open's strategy for achieving this goal is to combine existing and emerging standards into a comprehensive, integrated, high-value and usable system environment, called the Common Applications Environment (CAE). This environment covers the standards, above the hardware level, that are needed to support open systems. It provides for portability and interoperability of applications, and allows users to move between systems with a minimum of retraining.

The components of the Common Applications Environment are defined in X/Open CAE Specifications. These contain, among other things, an evolving portfolio of practical application programming interfaces (APIs), which significantly enhance portability of application programs at the source code level, and definitions of, and references to, protocols and protocol profiles which significantly enhance the interoperability of applications.

The X/Open CAE Specifications are supported by an extensive set of conformance tests and a distinct X/Open trade mark - the XPG brand - that is licensed by X/Open and may be carried only on products that comply with the X/Open CAE Specifications.

The XPG brand, when associated with a vendor's product, communicates clearly and unambiguously to a purchaser that the software bearing the brand correctly implements the corresponding X/Open CAE Specifications. Users specifying XPG conformance in procurements are therefore certain that the branded products they buy conform to the CAE Specifications.

X/Open is primarily concerned with the selection and adoption of standards. The policy is to use formal approved *de jure* standards, where they exist, and to adopt widely supported *de facto* standards in other cases.

Where formal standards do not exist, it is X/Open policy to work closely with standards development organisations to assist in the creation of formal standards covering the needed functions, and to make its own work freely available to such organisations. Additionally, X/Open has a commitment to align its definitions with formal approved standards.

The Xtra Process

X/Open's Xtra process has become the world's leading open systems user requirements programme.

A continuing process, Xtra identifies and prioritises user requirements for the adoption and implementation of open systems computing. These user requirements are made publicly available through the **Open Systems Directive** (OSD).

The OSD is prepared for users, suppliers, consultants, analysts, corporate planners and others to guide action and planning for open systems development. It reveals current concerns and issues, as well as the direction in which committed users are working.

Selected Bibliography

Listed below is a range of recently published books and articles which complement the material contained within this book:

- Anonymous, *How to Implement Open Systems*, Chain Store Age Executive, November 1992.
- Aryanpur, Sarah, *Make or Break: Migration (Moving to Open Systems in Information Technology)*, Computer Weekly, 6 May, 1993.
- Baggott, Mike, *New Class of ''Open'' Tools Extends Application Life*, Computing Canada, 1 September 1992.
- Bochenski, Barbara, *X/Open Capturing ''Open'' Mind Share*, Software Magazine, Volume 12, Issue 10, July 1992.
- Brodie, Michael, *Openness — It's the New Frontier*, Computing Canada, 12 April, 1993.
- Butler, Janet, *Tool Suppliers Eye Open Systems*, Software Magazine, Volume 13, Issue 6, April 1993.
- Cupp, James G., *A Management Perspective on Open Systems*, Transmission and Distribution, Volume 44, Issue 12, November 1992.
- Dandridge, Jeff, *Five Reasons Why UNIX is Succeeding in the Banking Industry: Defining Open Systems*, Banking Software Review, Volume 17, Issue 2, Summer 1992.
- Dolgicer, Max and DeBoever, Larry R., *Open Systems: Getting Beyond the Hype*, Data Communications, March 1993.
- Dunphy, Ed., *UNIX Industry and Open Systems in Transition: A Guidebook to Managing Change*, Q.E.D. Publishing Group, U.S., July 1993.
- Fensholt, Caro, *Has the EDI Train Left the Supermarket Station? Open Systems Key to EDI*, Supermarket Business, Volume 47, Issue 8, August 1992.

- Fleck E., Ed., *Open Systems in Medicine: The Berlin Approach to Medical Information Processing*, I.O.S. P., Netherlands, June 1993.
- Glass, Brett, *What's Open, and What's Not? (Ways of Assessing the Openness of a System)*, InfoWorld, 21 September, 1992.
- Harrison, Ron, *Business Systems for Building Control; Open Systems for Building Services — A Consultancy Report and Analysis of Benefits and System Alternatives*, D.E.G.W. E.T.L., February 1993.
- Holland, David J., *Yes, But What Does It Mean? (Computer Jargon)*, CMA — The Management Accounting Magazine, Volume 66, June 1992.
- Kaplan, Robert S. and Norton, David P., *The Balanced Scorecard — Measures that Drive Performance*, Harvard Business Review, January / February 1992.
- Leininger, Kevin E., *Open Systems Slam the Door on the Days of Competitive Advantage by IT Alone*, Chief Information Officer, Summer 1992.
- McGarry, Susan, *Open Systems: The User Reality*, Computerworld, 15 February, 1993.
- Network Management Forum (Christopher Jacobs, Ed.), *Discovering OMNIPoint: A Common Approach to the Integrated Management of Networked Information Systems*, Prentice Hall, November 1992.
- Ogilvie, Heather, *When Open Systems Meet Closed Minds*, Chief Information Officer, March / April 1993.
- Perley, Daniel, *Migrating to Open Systems: Taming the Tiger*, McGraw Hill, March 1993.
- Pratchett, L., *Open Systems and Closed Networks*, Leicester Business School Occasional Papers No. 9, De Montfort University, February 1993.
- Robins, Gary, *Open Systems Management*, Stores, April 1993.
- Southcott, John, *The Business Impact of Technology Change (Management Considerations for Firms Moving to Open Computer Systems)*, Computing Canada, 25 May, 1993.
- Trimm, Jerry, *Open Systems May Be Welcomed with Open Arms*, Bobbin, May 1993.

- Tapscott, Donald and Caston, *Paradigm Shift: The New Promise of Information Technology*, McGraw Hill, 1993.
- Valderrama, Ramiro, *Migration Plan to Open Systems*, Inform, November 1992.
- Wakerly, Ralph, *Open Systems Drive in Health Information Networks*, Computers in Healthcare, Volume 14, Issue 3, March 1993.
- X/Open Market Research Report, *Open Systems Directive (Xtra '92)*, ISBN: 1-872630-71-5, O301, March 1993,
- X/Open User Briefings, *Open Systems: A Guide to Developing the Business Case*, ISBN: 1-872630-72-3, B003, November 1992.