THIS PAPER IS PUBLISHED AS CHAPTER 14 IN 'SYSTEMS ENGINEERING FOR BUSINESS PROCESS CHANGE', EDITED BY PROFESSOR PETER HENDERSON, PUBLISHED BY SPRINGER-VERLAG, ISBN-1-85233-222-0

CO-EVOLUTION & AN ENABLING INFRASTRUCTURE: A SOLUTION TO LEGACY?

Eve Mitleton-Kelly and Maria-Christiana Papaefthimiou

Complexity Research Programme, **London School of Economics.**

ABSTRACT

A major international financial institution has a substantial IT legacy systems problem. Its European operation based in the UK has resolved many of the issues and this chapter will explore the conditions, which have enabled this development to take place. It will argue that 'legacy' is not solely a technical issue, and the 'solution' in this case would not have been possible if many other organisational and cultural conditions were not in place. It will argue that the enabling organisational infrastructure was such, that it enabled the *emergence* of new ways of working, not typical between the business and IS domains and that it encouraged *co-evolution between* the two domains. By providing support and direction it was able to allow *self-organisation* in the development of new relationships. By exploring its *space of possibilities* the project also brought together the necessary technical skills, tools and approaches which facilitated the technical solutions. The exogenous pressure of the Euro created the need for internal unification, but that pressure alone would not have been sufficient to address the multiple problem space and to overcome deep cultural differences. Despite its success, the existing management approach of moving managers every 2-3 years, is threatening the stability and sustainability of the project in the future. Furthermore, the head-office in the USA was unaware of the European development. The case will be used by the USA office to learn about a *natural experiment* and the special conditions it created to enable co-evolution and to resolve the multi-faceted problem of legacy.

Key words: legacy, banking, enabling infrastructure, emergence, co-evolution, self-organisation, space of possibilities, Euro, sustainability, natural experiment.

INTRODUCTION

The chapter reports work-in-progress on a research project looking at the *co-evolution* of the business process and information systems (IS) development. The hypothesis being tested is that if co-evolution between the business and IS domains is enabled, then the problems associated with legacy systems will be reduced. Co-evolution is [1]taken to mean that *the evolution of one domain is partially dependent on the*

"Section 2 *The Legacy Problem* was written by Maria-Christiana Papaefthimiou. The rest of the paper was written by Eve Mitleton-Kelly."

evolution of the other [Ehrlich & Raven 1964, Pianka 1994, Kauffman 1993 & 1995, McKelvey 1999a & b, Koza & Lewin 1998], *or that one domain changes in the context of the other.* The notion of co-evolution places the emphasis on the *evolution of interactions* and on *reciprocal evolution* [Futuyma 1979], which is to be distinguished from the simultaneously changing, *mutual causal process* posited in systems theory [Maruyama 1963]. In human systems, co-evolution in the sense of the *evolution of interactions* places emphasis on the relationship between the co-evolving entities. The project is therefore studying the *relationship between* the business and IS domains, and the hypothesis being tested is based on the assumption that the degree, intensity and density of *interaction* between the two entities affect the rate of co-evolution between the two domains.

This paper will argue that legacy is not merely a technical issue, but that legacy arises from a multiplicity of intricately inter-related and inter-dependent socio-technical factors. It will also argue that the degree of connectivity between the IS and business domains may improve the *fitness* of each domain or it may result in *complexity catastrophe* due to the increased constraints brought about through increasing dependencies. Increased fitness will, in the context of the paper, be interpreted as the emergence of a new organisational form, which has helped to reduce the problem of legacy. Complexity catastrophe will be interpreted as an extreme state of dependencies between the IT systems and the business applications which give rise to an almost intractable problem of legacy.

Since co-evolution is defined as the reciprocal interactions among agents at all levels of analysis [McKelvey 1999b], the study deals with both the ***micro*** level of interaction between individuals and between individuals and artefacts (IT systems) and the ***macro*** level of interaction between the business and IS domains as well as between the organisation and its environment. [Endnote1]

The paper is in 4 sections. Section 1 will explain the theoretical background to the study. Section 2 will summarise the various factors contributing to the problem of legacy in one case study and identified through a series of semi-structured interviews conducted at the USA and UK operations of an international financial institution. Section 3 will describe the situation found in the UK office of the same organisation and the final section will identify the *enabling infrastructure* or the conditions necessary to enable co-evolution and to create a new organisational form. An enabling infrastructure is taken to mean the cultural, organisational and technical elements, which support particular ways of working and relating.

1. THEORETICAL BACKGROUND

The current IT Legacy systems project builds on work done earlier, on the relationship between strategists in the business and IS domains [Mitleton-Kelly (U)] in 85 organisations in both the UK and the USA. The research was based on semi-structured interviews with over 350 strategists and others over a four-year period between 1988 and 1992. The findings showed restricted interaction and inadequate communication between the two domains. As a consequence there was a significant limitation in understanding of the each other's sphere of operations which was a

cumulative outcome of lack of information, interest, and opportunity to become acquainted with the intricacies of the other domain. Furthermore, these articulated factors were strongly associated with deeper cultural and personality traits. The cultural issues constrained and reinforced the attitude that the other domain 'did not understand' the issues of one's own field of interest, and the corollary that they did not 'need to know' about future plans and strategies.

Consequently, the strategists in both fields did not inform the other at an early enough stage, of likely plans and strategies which inevitably led to separate and distinct evolutionary paths, until they were forced to acknowledge the resulting gap when the information system failed to support the business activities. In other words, there was a low rate of micro-agent co-evolution or evolution in the context of the other. McKelvey [1999b] defines the rate of micro-agent co-evolution as *"a function of the rate of the knowledge accumulation of agents and the level of interconnection among them"*. In the context of the study, McKelvey's definition needs to include an awareness of the possible future strategies being considered by the other domain. Co-evolution in human systems also presupposes interaction or dialogue between the co-evolving agents. However, the degree and intensity of the interaction or the *level of interconnection* between agents in the IS and business domains, is one of the elements under study in the current project. [Endnote 2]

In the earlier LSE study, the cultural environment (behaviours and ways of thinking, relating and working) was both a reflection and a product of the personality profiles of the IS and business strategists which were attracted to the different cultural domains. One way to look at these differences is through Cultural Theory, which was developed by the anthropologists Mary Douglas [1982, 1992] [Endnote 3] and Michael Thompson [Thompson, Ellis and Wildavsky 1990] [Thompson 1995]. According to the Theory, there are five different archetypes. The four main archetypes are the hierarchist, the individualist, the egalitarian and the fatalist. The fifth archetype of the *hermit* is often neglected. Mitleton-Kelly [U] however has shown that this archetype is of relevance when considering the differences in worldview [Checkland 1981] or cosmology [Douglas 1982, 1992] between the business and IS strategists as there is a strong affinity between the IS professional and the hermit archetype.

Comparing the IS professional in terms of the hermit archetype and the business strategist in terms of the hierarchist archetype, it becomes a little clearer why there is such a lack of understanding between the two domains. Not only is the way each sees and interprets the world quite different [Douglas 1982, 1992], but the way that each uses and rejects information [Thompson et al 1990] is also quite distinct.

One of the outcomes was that communication between the business users and the IS developers was restricted and often confined to formal procedures and exchanges. As a result, business requirements either for new applications or whole systems were transacted in a particularly constrained way. There was rarely any real *dialogue* to elicit and understand either the business requirements or the limitations of the available technology. Assumptions were made on the basis of incomplete information and applications and systems were built based on partial and often erroneous understanding of the present and future requirements of the business. The current project builds on that work and is based on the assumption that the gap between the

two domains is a significant contributing factor to legacy systems. While if co-evolution between the two domains is enabled then, it is argued, the problem may be reduced. It is further argued that the legacy issue is not primarily or solely a technology issue but is the outcome of a combination of both social and technology elements.

The case study described in this paper is used to illustrate the multiple socio-technical elements which interact to create the *problem-situation* [Checkland 1981] seen as the *IT legacy systems problem*. The case study shows two distinct approaches within the same organisation, in two different geographical locations. The case is that of an international financial institution (to be referred to as 'ABank'), and its European operations based in the UK. The USA operation is still struggling, whereas the UK office has partially resolved its legacy problem.

The contrast arose from a difference in focus. While in the US legacy was primarily seen as a technical issue, the UK office acknowledged the multifaceted nature of the problem and the associated organisational aspects. By breaking the organisational norms and actively encouraging a sustained dialogue over time, between the two domains, they created an enabling infrastructure, which in turn helped them overcome the technical constraints.

The UK office in effect participated in a *natural experiment* in the sense that it found a different way of working and relating based on a different way of thinking about the relationship between the business and IS domains. *Experiment* in this context does not mean the designed testing of a hypothesis or of pre-established conditions. It means exploration of new possibilities. No one could predict the outcome. The dominant culture of the Bank supported one kind of *order*, that is a particular way of relating and working which had inadvertedly contributed to a legacy problem. A different way needed to be found and the UK office *self-organised* itself and created a new order. Although certain individuals took particular actions, no one was deliberately *orchestrating* the process. Certain conditions were introduced which encouraged and supported a different type of interaction and enabled the individual agents to co-evolve in a reciprocal evolutionary context. In other words, certain individuals in the UK office, initiated the conditions which helped to create a new enabling infrastructure, which in turn allowed a new organisational form to emerge through the interaction of a group of *agents* from both the IS and the business domains. One of the outcomes of this natural experiment or *exploration of the space of possibilities*, was an increase in fitness in terms of the amelioration of the legacy problem.

The theoretical framework used to examine the case is derived from the ***theories of complexity***. Complexity in this context means the intricate inter-relationships that arise from the interaction of agents, which are able to adapt *in* and evolve *with* a changing environment. The theoretical framework being developed is based on work in the natural sciences (in physics, chemistry, biology, mathematics, and computer simulation) studying complex adaptive systems (CAS). [Arthur 90 & 95, Gell-Mann 94, Goodwin 95 & 97, Hodgson 93, Holland 95 & 98, Kauffman 93 & 95, Nicolis & Prigogine 89, Prigogine & Stengers 85, Stacey 95 & 96, Waldrop 92, Webster & Goodwin 96] The work at the LSE is focusing on complex *social* systems using the generic characteristics of CAS as a starting point, but without direct mapping between the disciplines. [Mitleton-Kelly 97 & 98a] In other words, organisations are studied as

complex social systems in their own right, not as metaphors or analogies of physical, chemical or biological CASs.

In an organisational context, complexity provides an *explanatory framework* of *how* organisations behave. How individuals and organisations interact, relate and evolve within a larger social ecosystem. Complexity also explains *why* interventions may have un-anticipated consequences. The intricate inter-relationships of elements within a complex system give rise to multiple chains of dependencies. Change happens in the context of this intricate intertwining at all scales. We become aware of change only when a different pattern becomes discernible. But before change at a macro level can be *seen*, it is taking place at many micro-levels simultaneously. Hence *micro-agent change leads to macro system evolution.*

2. THE LEGACY PROBLEM

The definition of a legacy system has evolved over the years according to the different perspectives taken to study them. The relevant literature has shown a progression from addressing only the purely technical perspective to encompassing the wider socio-technical issues of legacy. Within the software engineering literature the problems of legacy systems are looked at mainly from their technical perspective (Adolph 1996, Sneed 1995, Chikofsky & Cross 1990). They are often described as old and large, are written in assembly, or an early version of a third generation language. They have been developed 20-30 years ago without anticipating that they would be still running decades later. Reengineering (Adolph 1996, Sneed 1995), reverse engineering, freeze and encapsulate (Bennett 1995) are suggested as viable solutions to the legacy systems' problem.

As the architectures and technology used to build the systems were relatively inflexible, they had not been designed to accommodate change. Lehman's laws (Waren 1999) warned that software should be regarded as evolving and indeed, the software systems have changed extensively over time to accommodate changes in their environment albeit in an ad hoc manner. Many years of changing and adding-on has resulted in improved functionality but also in mass complexity and poor system understanding. Moreover, they are associated with high maintenance costs (Bennett 1994, Waren 1999) and they have become very difficult and expensive to change to further support the business objectives. When the balance between the technical and business dimension is lost, legacy can be seen as a *gap* between the business needs and the technical capabilities (Ramage & Munro, 1999). Furthermore, legacy has started been recognised as a multifaceted socio-technical situation (Gold 1998).

In the case presented in this chapter, the organisation admitted to a significant legacy problem. It would prefer to jettison the old legacy systems, which are perceived as those *systems that no longer support the current business objectives or are inhibiting future developments* (e.g. the creation of new financial products). They are typically large, the cost of maintaining them is very high and they constrain the business from responding fast enough to desired changes in the business domain. Legacy systems are not sufficiently flexible to allow significant modifications, and cannot meet current and future architectural standards. However, the applications supported by the legacy systems are typically large, complex and vital to the business.

This section will summarise the three types of factors contributing to the problem of legacy in the case study. They were identified through a series of semi-structured interviews with an international financial institution to be referred to as ABank.

An initial finding is that *legacy* is not solely a technical issue but has multiple aspects. The research has identified three categories of contributing factors: business, organisational and technological factors. The distinction is primarily conceptual in the sense that it offers a framework for understanding the interdependence and interrelationships between the three categories. It is also of practical relevance in recognising and in creating the complex socio-technical conditions, which enable co-evolution between the business and IS domains. By using complexity as the theoretical framework to analyse the case study, this project enriches the legacy debate.

Part of the problem is that the bank's efforts in the USA, are focused on the technical issues, while the business and organisational factors and their relationship to the technical issues are not given equal value or emphasis. In contrast, the UK operation of the same organisation has partly resolved the technical problem by acknowledging the importance of the social factors and by creating the conditions whereby all three factors are closely related and seen to be so.

Business factors

Due to the changing environment in which the business operates, changes in its processes, products and services are necessary in order to sustain its competitiveness. These business changes have an impact on the bank's technological infrastructure and potentially contribute to the legacy problem. Examples include new business development in other geographical areas and changing business objectives. These demand new system development or amendments to the existing ones to accommodate local market characteristics (legal and customer). Financial pressures to cut costs affect the technology budget and the allocation of funds to build or rebuild an application to satisfy the business requirements are often insufficient. As a consequence new applications are often built on old technology or incremental functionality is added onto the existing system, which in turn contribute to the problem of legacy.

Another way of looking at this aspect is that co-evolution needs to take place at all levels: from the macro level between the organisation and its environment or social ecosystem (which includes all related businesses, customers, competitors and suppliers as well as the economic and cultural environment) to various micro levels within the organisation. Furthermore, changes at the macro level affect the various inter-related micro levels within the organisation, such as the IT systems. Co-evolution is the reciprocal influence between the environment and the organisation. In other words it is NOT a case of *adaptation to* a changing environment but of *co-evolution with* the social ecosystem. [Mitleton-Kelly 1997 & 1998a]

Organisational factors

Some of the legacy issues were found to be closely linked to the bank's human and organisational context. Problems with the delivery of the applications and the

relationship between users and developers emphasise the non-technical nature of the problem. Moreover, the organisational culture and politics have a role to play in the perceived *gap* between the business and the supporting information systems.

The **applications' development process** faces a number of difficulties. Due to the great demand for new applications and the difficulty that users have in prioritising projects, the time to develop and deliver the requests seems to be problematic. Time pressures mean that the developers might take short cuts to development, making compromises on the systems capabilities and deliver only a part of the original specification. Due to the initial compromises the systems have to be enhanced to fulfil the business requirements and that inevitably leads to complex applications and problems with their maintenance.

A **communication gap** between the developer and user communities is evident within the organisation. It seems that the two communities have different views and use different languages that are not fully understood by each other. The users' perception of the developers is that they are generally people whose job is mainly to deal with the technology and do not seem to appreciate fully the business needs. On the other hand, the developers' perception of the users is that they cannot communicate the system requirements clearly and that they are not consistent in the priorities put forward from the business decision-makers. Moreover, the business side does not seem to appreciate the work and time involved in developing or changing an information system. As a consequence of this poor appreciation of each other's domain, developers do not deliver according to users' expectations.

The **lack of skills** to maintain the legacy systems is another consideration. This is a three-fold problem as it reflects the age of the employees, the type of system they have been working on and the attitude of the organisation to training needs.

It is difficult and expensive to recruit people who have knowledge of the old systems since current training is focused on the current rather that the older technologies. On the other hand, as far as systems development is concerned, resistance to change and lack of awareness of the latest technological developments might prevent people from moving away from old technology and given the opportunity they will avoid using new methods and technologies to implement requests for applications. Some individuals even refuse to be trained in the new technologies and prefer to stay with what they know. Consequently, some new applications do not benefit from the state-of-the-art technologies and the legacy problem is perpetuated.

Age of employees as well as exposure to new technology can contribute to an individual's attitude towards any kind of change, but the attitude of the organisation is also important. The bank's lack of supporting change was indicated in that **training** and **education** provision was often inadequate and did not really help employees to embrace change. Older employees are affected more than their younger counterparts, in that when the latter's personal aspirations are not dealt with they look for employment elsewhere.

Personal **career agendas** are sometimes in conflict with business considerations as far as systems' development is concerned. As technology changes and becomes more friendly and pervasive, new and younger employees are keen to use the latest

technological tools. They refuse to work with the old systems and use different platforms and programming skills to enhance them. In addition, a great deal of time and resources are committed to the development of a new system and old systems seem to be ignored. *"Obsession with the new technology"* and personal choices in moving on with ones' career have priority over the considerations of the business.

Furthermore, **management discontinuity** seems to exacerbate these issues. The managers responsible for new initiatives do not as a rule stay in their job long enough to complete a project and to make any real impact. *"I am the fifth manager in the job in four years… but it takes about 3 years to identify, fix and stabilise a large banking operation."* On the other hand every manager is under pressure "*to show progress in six months.*" As a result some system development projects are not completed which contributes to the legacy.

Technological factors

Technological aspects of the legacy problem have been covered elsewhere in this volume. However, in relation with the points outlined above, some issues must be highlighted. Firstly, the rapid technological change and the need to keep up with current technology exert a constant pressure on management, which must be offset against the cost of the investment. Secondly, the existing technological infrastructure, in combination with the increasing obsolescence of technology fails to meet emerging expectations and to keep up with new business requirements. And thirdly, issues of alignment between existing and new incoming technology (in terms of new platforms, new hardware, new software and processes) introduce multifarious problems contributing to institutional friction.

The above section outlined some of the generic problems identified in ABank. Section 3 will articulate in some detail the case of the UK office where some of the legacy issues were partially resolved.

3. THE UK BASED OPERATION OF ABANK

This section is based on interviews conducted at the UK office in April and June 1999 and will describe some of the conditions that have contributed to the complexity of the legacy issue. The conditions were both technical and social. It was recognition of the problem space, involving multiple, interrelated social and technical aspects, which led to a partial resolution of the legacy issue. The conditions that enabled co-evolution will be explored in section 4.

The legacy system of the UK operation is based on IBM hardware, is at least 10 to 15 years old (with 30 year old elements), is written in Cobol with assembler language components, and *"needs resources that now are in their 50s or even 60s"*. With greater insight it was also described as *"what is left behind by the previous organisation – the system that was built for a different organisation than the one we are to-day"*. This observation points to part of the problem. The systems were

designed and built to support a different business environment. As the business environment changed, the IT systems were modified, enhanced, partially replaced and new elements added, in an effort to continue to support the business, but without full success. The constant modifications have not provided a system tailored to the changing business requirements, but a *"legacy system* (which) *becomes dysfunctional or it becomes disjoint* (with) *the current business"*. Full replacement, however, does not seem to be an option. In ABank in the UK they have *"tried on a number of occasions to replace components of the legacy environment but we've never got to a full replacement because of the complexity."*

What is meant by complexity in this context is the *operational complexity* of the technical system, which has arisen from the intricate intertwining of multiple socio-technical elements. A variety of business, organisational and technical elements have combined to produce a complex socio-technical system with a very high degree of interconnectivity and interdependence. These multiple elements will be outlined below. They are relevant in setting the context and in explaining why it is so difficult, if not impossible to jettison the entire legacy system and to start afresh. The description may also be read as that of a socio-technical CAS. If it is seen from that perspective then certain generic characteristics common to complex adaptive systems may be identified which are relevant and applicable to the socio-technical system under study.

(1) One element arises from increasing interconnectivity and interdependence among the **system components and the applications**. ABank services *"very high value global corporate clients"*. The basis of that service is that it will provide those customers with the technology infrastructure to support their business. This means that the bank will often customise or engineer solutions into its systems, and change their coded components. Over time a layered system infrastructure is created, which is tailored to service many different customers. The interconnectivity and interdependence become so intricately intertwined that a point is reached when *"to undo that complexity is almost insurmountable without going back to the business perspective and understanding where those customers are going and whether they are willing to accept a change in the way that we're working with them which allows us to undo some of the legacy and therefore some of the complexity."* An important point to note is the emphasis placed on the **relationship** between the business and the IT developers, based on a reciprocal understanding of business direction (related to the future needs of customers) and IT constraints. This kind of relationship, leading to an understanding of each other's domain, helps create the conditions that enable co-evolution.

On the technical aspect, once a many-layered stage is reached, it becomes very difficult to fully tailor yet another application. Hence the bank *"cuts and pastes"... "rather than reinventing the wheel you take something that looks as though it's a fairly good fit, you will change it slightly but you will not try and create an architectural component that will serve as a common component, you will generate another subset. And you get to a situation where you are suddenly generating subsets for different customers."*

This kind of constant modification increases the connectivity and the interdependence between components and makes renewal of parts of the system very difficult, as they

cannot be isolated and replaced without a great deal of arduous work and some risk. The risk arises from the consequences of interconnectivity and interdependence as well as the lack of full documentation of every modification. As the applications are layered, as new subsets are created and as the systems become increasingly intertwined they also give rise to **emergent** properties. That is, properties which are unexpected and unpredictable. If these properties also happen to be unnoticed, then changes in one part of the system could have significant consequences in other parts.

(2) Another element contributing to the operational complexity of the socio-technical system has been that **organisational restructuring** (a social aspect) **has changed the systems' architecture** (a technical aspect). The main European system is on two hardware bases. Eleven European countries, with smaller branches, using HP hardware, are serviced from the USA. While the larger branches, with IBM systems are now run from the UK. Originally the IBM system was implemented in seven different countries and it started in the late 70s, early 80s, as a branch or country-centric system, referred to as *"a bank in a box"*. That is, *"it ran on a very small computer in the branch, did all of the branch activity from running the books to having the capability to service the front end product such as funds transfer, money market, loans etc."*

Since then, the bank has gone through several phases of restructuring. The first set of changes in the mid-80s was to regionalise the infrastructure, that is the hardware and the software were brought into central service centres and the branches were run remotely. The branch users run their terminals connected over leased lines into a centre. It was first set up in Belgium and was subsequently moved to the UK. To day, all the processing and the infrastructure, that is, all the hardware for the six European countries with IBM systems is based in the UK.

The first phase involved moving the technology. The next phase was to move the branch back-office processing, that is the funds transfer processing, the foreign exchange processing, etc. In the late 80s and early 90s the back-office processing was pulled out of the branches and put into a regional centre first in the UK and subsequently in Ireland.

The outcome is that the legacy system sits on a hardware box in the UK or in the USA, but the day to day banking operations for each European country are based in the Irish service centre (running the back-office operations) together with branch based specific activities, such as customer enquiries, undertaken in the country branches. However, the branch banking activity on the country system has diminished considerably. This gave rise to an ownership issue, which became significant when major changes needed to be made to the legacy system to implement the procedures necessary for the Euro.

(3) A third component is that the bank has made a conscious effort to try and isolate elements of the legacy 'bank in a box' system and to **create stand alone components**, which still communicate with it. They are Windows NT based front-end servers. But they haven't yet succeeded in replacing the full set of legacy software. The part replacements use current technology.

(4) The problem of increasing and multifaceted operational complexity was recognised and appreciated by the UK operation. In contrast to the US office, it was seen as both a social and a technical issue. It was appreciated that the business needed to recognise the difficulties involved and to make certain decisions. For example, one of the difficulties was identification of ownership of the part of the system that was being upgraded. There were so many interdependencies and linkages that isolation of specific modules became extremely difficult. The business therefore needed to decide. *"We cannot take a technology view because it's only by looking at it from the business perspective that you can get the business to accept that yes they want it changed because they can't get what they want in time, but … we have to have a dialogue with them to work out how we're going to move those customers as well as the new customers that they want to get into a new architecture."* Until ownership was established the work could not be *"signed off"*. This became particularly taxing with core common components. It was easier with specific sub-systems such as the funds transfer subsystem, when a particular part of the organisation would recognise the benefits of upgrading or replacement, and were prepared to take ownership and pay for the work.

The identification of ownership of common components and of the need for upgrading, was much more difficult as multiple owners had to be identified and to be persuaded of the benefits, before they would sign off. The technical problems impacted the organisational issue of ownership and the geographically dispersed organisational structure added to the problem. The multi-ownership issue did not arise with systems that are managed and owned locally in a single country. This example shows how the intricate interrelationship between technological and organisational factors creates the complex problem space of legacy: a technical problem impacted an organisational issue while organisational changes exacerbated the technical concerns.

(5) Another element contributing to the complexity of legacy is that the maintenance and further development of the IT systems have been centralised within the UK group, which now controls 16 systems on both HP and IBM platforms. Thus, as resources for the maintenance and support are held centrally, there is now **no local knowledge** of the branch technology of the system.

Hence a consequence of a regional structure, is that the local knowledge base is lost: *"what happens over time is that the people or the resource base that knew that legacy system inside out, they knew all the vagaries of the system and how they would bypass some of the inefficiencies or deficiencies, you then find as you transition and you move to regional structures, that you lose that knowledge base."* To overcome the loss of local knowledge, written formalised procedures were established to enable the day-to-day running of the system. However, when a relatively unusual request came, for example when a customer wished to repeat a transaction in a similar way that had last been done two years earlier, then *"nobody knew how to use that part of the system anymore"*. Thus in formalising the procedures, the informal ways and means used to bypass certain problems, which were difficult to articulate, were not captured and that knowledge is now lost.

Despite the above problems, the UK operation has completed the first phase of adaptation of the systems to support the Euro. *"That was for the 4th of January 1999.*

That was done very successfully in terms of our legacy base. As well as changes to all of our more modern systems that work with the legacy systems."

One of the main drivers, in the success of the first phase, was the exogenous pressure of legal and regulatory requirements for the Euro. However, although the exogenous pressure was a necessary condition, it was not sufficient for success. Furthermore, to complete the operation over the next two and a half years, it will be necessary to understand which conditions enabled the first phase and to build them into a *socio-technical enabling infrastructure.*

One set of conditions is to do with the technology. The current state of the IT system is a mixture of new components interfacing with the core legacy system. There are two points to note: (a) the multiple interfaces are a constraint to the future development of the system, as they have added to the complexity of interdependencies; and (b) the work has forced the developers to clean up and understand parts of the system that had become obscure through constant and undocumented modifications.

With improved understanding they were able to provide their customers with the service they needed. But this does not guarantee future success. Unless the business is able to identify its direction in a re-regulated environment, and communicate its future needs and requirements, the legacy issue will re-establish itself. This indicates the importance of continuous co-evolution between the business and IT development.

But the current success also indicates that the apparently inherent problem of lack of communication between the business and the information systems developers has been partly overridden.

The exogenous pressure imposed by the need to implement the legal requirements of the Euro, encouraged internal unification, in the same way that the external threat of a war often creates a degree of internal unity. *"We had such a unified programme to deliver the Euro between both the business and the IT, all being driven by senior management that in fact what we came out with was a much closer rapport"*. The outcome was that the business people (i.e. those that drive the requirements of IT), the IT specialists and the processing people, all gained an understanding of both the issues and the systems, *"everybody got a tremendous amount of both interpersonal worth out of that and knowledge of our environment out of it, that everybody came out of it understanding a lot more about what we were really doing. Rather than being as they were two years ago in their individual silos, just throwing grenades."*

But how was this understanding achieved and what overcame the inherent lack of communication?

4. THE ENABLING CONDITIONS

The conditions in the UK operation which enabled a closer working together between the business and information systems professionals, can be summarised as:

- New procedures introducing regular monthly meetings, which enabled *good networking* and *trust*, as well as a *common language* leading to mutual *understanding*.
- *Autonomy*: the project manager was left alone to introduce the new procedures.
- A *senior manager supported* the changes.
- *Stability*: sufficient *continuity* to see the project through.
- An *interpreter* mediated the dialogue between the domains. This ensured understanding on both sides but also protected the technologists from constant minor changes in requirements.

The process started sometime ago, when a new Business Product Manager moved to the UK office and found a substantial disconnect between the business requirements and technical support in the *cash management business*. She then brought in a Project Manager to help bridge the gap. The procedures, which the Project Manager introduced for the cash management business, provided the necessary background and set the conditions, which in turn enabled the work on the Euro to succeed. The outcome of these two projects was a significant improvement of the legacy issue.

When the Project Manager came in, he had to define his role and that of his group. A number of initiatives were taken. He created the conditions for the three environments of technology, business and operations to talk together, but in doing so they went against accepted established ways of working. Initially, he acted as the *translator* between the business and the technology groups and he used "*control of the purse"* to initiate the dialogue. But that was only to get the process started. The important initiative was instigating regular monthly meetings, supported by weekly information updates.

Senior managers from the business, the technology and operations were invited to attend the monthly sessions. Every month they would go through each one of the projects reporting on the latest status, where the project was, what happened in the last month, what was planned for the future month, what the issues were. But something else was also taking place. The people involved in the different projects began to identify **cross-dependencies in terms of the business project relationships,** which led to new insights, and new ways of working. *"They're business related dependencies. And ... people suddenly open up and realise that maybe there's a better way of doing something. Maybe there is another view to take on this and in fact these sessions proved to be very useful."* Once the conditions were provided the individuals involved were able to make the necessary decisions and take the appropriate actions. This illustrates micro-agent interaction, which is neither managed nor controlled from the top. Once the inhibitors are removed and the enablers put in place new behaviours and ways of working can emerge.

The monthly sessions improved communication between the different domains by improving understanding, but they also allowed for the ***emergence of new ways of working***, and in the process helped the business become fitter or more competitive.

"That monthly session was fairly well attended and as time went on, I think it proved itself out in terms of its value because we had good understanding between all of the project managers looking at the projects from a business perspective. We had ownership in that the business could see what we were doing so they were interacting

with us. But also they were almost inadvertently interacting with technology because they were both in the same forum. And so I think what this did it broke down any sort of barriers and we got ***common understanding*** *and in fact we delivered projects that the business wanted to see. And that the business has since found to be key. And since the Bank is number one bank in cash management. We must have done some good."*

The key was **simplicity, regular communication** and a **common language**. The reporting was content based with the emphasis on "*a simplicity of explanation on a regular basis of where a given project was"*. The technology person reporting and the operations person could both relate to the project. The business project manager, who owned the project, was also expected to present every month in a standard format to his or her colleagues who were all running the business projects for the cash management business. The business project managers, were usually ex-bankers with 10-15 years experience of the bank. They therefore knew both how to network in the bank and how to understand the rudiments of what the technologists were trying to articulate.

The meetings took half a day each month, with a continual rotation of presentations, which was a significant amount of time, yet people did attend regularly, and that regularity was a further key element of success. In a constantly changing environment it provided a necessary degree of stability and continuity *"the way the culture of the bank works, it changes so rapidly, people change so quickly from role to role*".

Another important element was the articulation of business requirements as an iterative process with regular face-to-face meetings between the technology specialists and the business project manager who owned the project and who "*solicit(ed) well-articulated business requirements in writing from the business product people"*. These meetings were at a senior management level with (a) a vice president who would own the product and be responsible for the P&L. They would determine what they required. They would meet with (b) a senior and experienced business project manager who was a seasoned banker, with a good knowledge of the bank, and (c) a senior technology project manager who would have to define the IS platform(s) and the technical development of the project. This constant dialogue created a willingness to **communicate** and a level of **trust** which were essential enablers of co-evolution. *"If you're willing to communicate and get down to a base level of discussion with techno-phobic individuals, then what you have is a willingness to participate and listen and over time you get a certain rapport and confidence level built up"*.

What was achieved at the UK ABank took a particular individual, supported by his senior manager, to create the conditions that enabled dialogue, understanding and a good articulation of requirements. He created the initial conditions, to improve the relationship between the domains, but he could not foresee how the process would work or whether it would work. As it happened, it did work and a substantial *network rapport* was established between the domains based on trust, a common language and mutual understanding. They worked well together, because the conditions were right and they were prepared to *self organise* and work in a different way. The new relationships were not designed or even intended. They happened spontaneously in the sense that they were enabled but not stipulated.

The achievement however, could be a one-off. Unless the new procedures and ways of working become **embedded** in the culture of the organisation, they are likely to dissipate over time. Once the initiator is no longer in place, and there is no new energy to sustain the process, the danger of dissipation or reversion to the dominant mode of working will assert itself.

In this case there has been some embedding and some continuity, but the process is fragile. A new set of organisational changes could destroy it. Part of the embedding is the networking rapport that has been established. The business project managers know whom they have to talk to in the cash business, whom they have to talk to in operations and whom they have to talk to in technology. That network is established. It is part of the social capital of the organisation, but it is implicit and informal.

Because the network rapport is implicit and informal it is under threat if there are too many and too frequent changes and the Bank's culture is one of constant change in management positions. *"Every two years someone else is in the post so that there is that lack of continuity."* If the rate and degree of change is too great then the network will become invalid.

There is a fine balance between stability and change. A degree of stability, a sense of continuity is necessary. It strengthens the network of relationships, thus **increasing the organisation's social capital**. While a degree of change, ensures a constant exploration of the space of possibilities. The two must be held in balance, in tension. If one predominates then the fitness of the organisation will decrease. In the case under study, a person who acted as *interpreter* and who helped to mediate the dialogue between the business and IT domains provided a necessary element of stability. This does not contradict what has been said above. The direct dialogue between the domains takes place face to face. The interpreter simply *"protects"* the technologists from constant minor changes in requirements. There is a distinction between (a) clarification and a good understanding of requirements and (b) the constant minor changes that the business people want to introduce. By providing a degree of needed stability, he gives the technologists space in which to work and to meet the agreed requirements.

CONCLUSION

In summary, encouraging co-evolution (as opposed to the pursuit of separate evolutionary paths) between the domains requires an enabling infrastructure, which provides the conditions for self-organisation, emergence and exploration of the space of possibilities. In human systems, co-evolution in the sense of the *evolution of interactions* places emphasis on the relationship between the co-evolving entities. The study therefore focused on the *relationship between* the business and IS domains, and explored the assumption that the degree, intensity and density of *interaction* between the two entities affect the rate of co-evolution between the two domains. In this case, the enabling conditions were: (i) enhanced **communication** between the domains, based on **trust** and mutual **understanding**; (ii) sufficient **stability** and **continuity**; (iii) **senior management support** and the realisation that *"a cross-domain process, was a successful way to run business drive requirements"*; (iv) *autonomy* and freedom to *self-organise*.

One important outcome from this process was the emergence of a new *organisational form*, or a new way of working and relating, which helped to reduce the problem of legacy and thus increased the organisation's *fitness*.

This chapter has reported work-in-progress on a single case, ten months into a three-year project. It highlighted certain aspects and emphasised the importance of interaction and relationships, while the technology details were only mentioned in passing. The way the technology problem was resolved will be reported in a subsequent paper, as will many others aspects, which were only glossed over in the present chapter.

Acknowledgements

The paper is based on research enabled by two EPSRC awards under the SEBPC Programme: IT & Computer Science Programme (GR/MO2590). The first one-year preliminary study of the two-phase project has been completed and the second 3-year project started in May 1998. The title for both projects is *"The Implications of the Theories of Complexity for the Co-evolution of the Business Process and Information Systems Development"*.

Both projects explore the findings from the sciences of complexity and examine the implications of generic characteristics of complex systems for organisations. The focus of the second phase will be to develop tools, models and approaches, which will aid the co-evolution of the business process with, IS development.

ENDNOTES

[1] The term IS or information system is used to denote the entire socio-technical system of information exchange using a variety of artefacts, while the term IT or information technology, refers primarily to a computer-based system. The term 'IS domain', however, is used to refer to the professionals working in the IT department of an organisation. They may be involved in the development of brand new systems, in the development of applications or in maintaining the existing system.

[2] The term *evolution* is often used interchangeably with *change*. It is also used rather loosely and not as a direct analogue of biological evolution. One of the areas under study in the LSE Complexity Programme, which includes the IT Legacy project, is an exploration of the concept of evolution within a social and organisational context. It is however worth noting that evolution is not synonymous with incremental change. Evolution can also occur by mutation or step change.

[3] The cultural theory framework was introduced by Professor Mary Douglas in the paper *"Cultural Bias"* in her collection of papers under the title *"In the Active Voice"*, Routledge & Kegan Paul, 1982. The Royal Anthropological Institute first published Prof. Douglas's paper as Occasional Paper 35 in 1978.

BIBLIOGRAPHY

ADOLPH, W.S., 1996: 'Cash Cow in the Tar Pit: Reengineering a Legacy System', *IEEE Software*, May 1996, p.41.

ACKOFF, R. L. 1970: 'A Concept of Corporate Planning' New York (Wiley, 1970)

ALLEN, P.M. 1997: 'Cities & Regions As Self-Organizing Systems : Model of Complexity' Environmental Problems & Social Dynamics Series, Vol 1, Gordon & Breach Science Pub.

ARTHUR, B.W. 1990: 'Positive Feedbacks in the Economy' Scientific American, Feb.

ARTHUR, B.W. 1995: 'Increasing Returns and Path Dependence in the Economy', Michigan

ASHBY, W.R.: 'Self-regulation and Requisite Variety' in Systems Thinking, 1956, Ed by F E Emery, Penguin 1969

BENNETT, K., 1995: 'Legacy Systems: Coping with Success', IEEE Software (12:1), Jan 1995, p.19-24

CHECKLAND, P. 1981: 'Systems Thinking, Systems Practice', John Wiley

CHECKLAND, P. & **SCHOLES,** J. 1990: 'Soft Systems Methodology in Action', John Wiley

CHIKOFSKY, E.J., & **CROSS**, J.H. 1990: "Reverse Engineering and Design Recovery: A Taxonomy." IEEE Software, January 1990. pp 13-17.

DOUGLAS, M.1982**:** 'Cultural Bias" in the collection "In the Active Voice', Routledge & Kegan Paul

DOUGLAS, M. 1992: 'How Institutions Think' a presentation at the LSE Strategy Seminar, 11 March 1992

EHRLICH, P.R. and **RAVEN** P.H**.** 1964: 'Butterflies and plants: A study in coevolution.' Evolution, 18: 586–608.

FUTUYMA, D. J. 1979: 'Evolutionary Biology', Sunderland, MA, Sinauer Associates.

GELL-MANN, M. 1994: 'The Quark and the Jaguar: Adventures in the Simple and the Complex', WH Freeman

GELL-MANN, M. 1995/96: *Complexity J.* Vol. 1, No.5

GOLD N. 1998: "The Meaning of "Legacy Systems"", Technical Report 7/98, Dept of Computer Science, University of Durham, UK, http://www.dur.ac.uk/CSM/SABA/meaning.html.

GOODWIN, B. 1995: 'How the Leopard Changed Its Spots', Phoenix

GOODWIN, B. 1997: LSE Strategy & Complexity Seminar, on 23/4/97, report on http://www.lse.ac.uk/lse/complex

HODGSON, G.M. 1993: 'Economics and Evolution: Bringing Life Back Into Economics', Polity Press

HOLLAND, J. 1995: 'Hidden Order: How Adaptation Builds Complexity', Addison Wesley

HOLLAND, J. 1998: 'Emergence: From Chaos to Order', Addison Wesley

KAUFFMAN, S. 1993: 'The Origins of Order: Self-Organisation and Selection in Evolution', Oxford University Press

KAUFFMAN, S. 1995:'At Home in the Universe', Viking

KAUFFMAN, S. & **MACREADY**, W. 1995: 'Technological Evolution and Adaptive Organizations', *Complexity J.*, Vol 1 No 2 pp.26-43

KOZA, MITCHELL P. and **Lewin A.** 1998: 'The co-evolution of strategic alliances' Organization Science, 9: 255–264.

KUHN T. S. 1970: 'The Structure of Scientific Revolutions', The University of Chicago Press, Second Edition 1970, (Originally published 1962)

LANE, D.A. & **MAXFIELD, R**. 1997: **'**Foresight, Complexity and Strategy', in 'The Economy As an Evolving Complex System II : Proceedings' (Santa Fe Institute Studies in the Sciences of Complexity, Vol 27) by ARTHUR B.W. (Editor), DURLAUF, S. (Editor), LANE, D.A. (Editor)

MARUYAMA, M. 1963: 'The second cybernetics: Deviation-amplifying mutual causal processes.' American Scientist, 51: 164–79

McKELVEY, B. 1999a: 'Self-organization, complexity catastrophe, and microstate models at the edge of chaos.' In J. A. C. Baum and B. McKelvey (eds.), Variations in Organization Science: In Honor of Donald T. Campbell: 279–307. Thousand Oaks, CA: Sage

McKELVEY, B. 1999b: 'Visionary Leadership vs Distributed Intelligence: Strategy, Microcoevolution, Complexity' in Proceedings of EIASM Workshop, Brussels, June 1999

MITLETON-KELLY, E. [U] Unpublished thesis on 'Strategy Process, Plural Rationalities and Complexity in the Strategy Domains of Business and Information Systems'

MITLETON-KELLY, E. 1997: 'Organisation as Co-Evolving Complex Adaptive Systems', British Academy of Management Conference, London 8-10 September 1997

MITLETON-KELLY, E.1998: 'Organisations as Complex Evolving System', Warwick and LSE Conference on Organisations as Complex Evolving Systems, Warwick 4-5 December 1998.

MITLETON-KELLY, E.1998: 'Complexity: Partial Support for BPR?' IEE Conference, London.

NICOLIS, G. and **PRIGOGINE** I. 1989: 'Exploring Complexity', WH Freeman

NICOLIS, G.: 'Physics of far-from-equilibrium systems and self-organisation' Chapter 11 in 'The New Physics' ed. by **DAVIES**, P. (ambridge University Press, 1989 (reprinted 1994)

PARKER, D. & **STACEY**, R.D. 1994: 'Chaos, Management and Economics: the Implications of Non-Linear Thinking' Hobart Paper 125, Institute of Economic Affairs

PARKER, D. 1995: 'Nonlinear Dynamics and Organisational Adaptability' Research Paper, April

PIANKA, E.R. 1994 : Evolutionary Ecology, 5th ed. New York: HarperCollins.

PRIGOGINE, I. & **STENGERS,** I. 1985: 'Order Out of Chaos', Flamingo

PRIGOGINE, I.: 'Time and the Problem of the two Cultures', First International Dialogue on the transition to Global society, Landegg Academy, September 3-9 1990

RAMAGE, M., & **MUNRO**, M. 1999: 'It's not just about old software: a wider view of legacy systems', (Unpublished as @ 10/05/1999).

SNEED, H.M.: 'Planning the Reengineering of Legacy Systems', IEEE SOFTWARE, Jan 1995, pp 24-34.

STACEY, R.D.: **'**The Science of Complexity: An Alternative Perspective for Strategic Change Processes' in Strategic Management Journal, Sept 1995, Vol 16, No 6, pp. 477-495

STACEY, R.D. 1996: 'Complexity and Creativity in Organizations', Berrett-Koehler

THOMPSON M., **ELLIS** R., **WILDAVSKY** A., 1990: 'Cultural Theory' Westview Press

THOMPSON M. 1995: "Managing the Unmanageable" presentation at the LSE Complexity Seminar January 1995

VARELA, F. & **MATURANA,** H. 1992: 'The Tree of Knowledge', Shambhala

WALDROP, M.M. 1992: 'Complexity: The Emerging Science at the Edge of Order and Chaos', Penguin

WARREN, I. 1999 'The Renaissance of Legacy Systems: Method Support for Software-System Evolution'. Practitioner Series. Springer Verlag

WEBSTER, G. & **GOODWIN**, B. 1996: 'Form and Transformation: Generative and Relational Principles in Biology', Cambridge University Press