Complexity & Information Systems

Eve Mitleton-Kelly
Director
Complexity Research Programme
London School of Economics
&
Visiting Professor
Open University

Frank Land
Emeritus Professor in the Department of Information Systems
London School of Economics

Published in the second edition of the Blackwell Encyclopaedia of Management volume on Management Information Systems.

Abstract

Many of the notions, which underpin the study of Information Systems (IS) and IS Management are based on systems theory. However the sciences of complexity are in many ways complementary to those of systems theory, whilst helping to enrich and deepen our understanding of the arena of discourse. This article outlines the contribution made by complexity to IS. It will define some key principles, explain how they can enrich IS thinking and offer an example to illustrate some of the principles.

Complexity Theory

A full description of ten generic principles of complex evolving systems can be found in Mitleton-Kelly 2003a. The following is a summary of a few of those principles, and their contribution to IS management.

There is no single unified Theory of Complexity, but several theories arising from various natural sciences studying complex systems, such as biology, chemistry, computer simulation and artificial intelligence, evolution, mathematics, physics, economics and the social sciences. Work under the general heading of Complexity Theory is undertaken in many centres in the world.

The research can be summarised as six main areas of study:

(1) Complex adaptive systems studied at the Santa Fe Institute in the USA and Europe;

(2) Work on far-from-equilibrium conditions and dissipative structures by Ilya Prigogine and his co-authors;

(3) Autopoiesis based on the work of Maturana in biology and its application to social systems by Luhman;

(4) Chaos theory;

(5) Increasing returns and path dependence by Brian Arthur and other economists; and

(6) Systems theory, cybernetics, social network theory and other work in social systems and management.

These six areas of research form the background to the ten generic principles of complexity identified by Mitleton-Kelly [2003a], in developing a theory of complex social systems. Since these principles incorporate more than the work on complex adaptive systems (CAS), the term *complex evolving systems* (CES) will be used.

The principles are *generic*, in the sense that they are common to all natural complex systems. However, the nature of the entities (whether genes, molecules, numbers, computer agents or human activity systems) has to be taken into account and the application of the principles in each context has to be made both relevant and appropriate. Human activity systems, for example, do differ from all other complex evolving systems in one important respect, in that the human actors are able to reflect and to make intentional decisions.

Many of the attributes and concepts of complexity theory are well known and have figured in the discussions and explanations of other theoretical frameworks; in particular, the concepts of connectivity, interdependence, feedback and emergence are familiar from Systems Theory Complexity theory builds and extends those concepts and adds new ones such as co-evolution, exploration of the space of possibilities, self-organisation, far-from-equilibrium, historicity and time to provide a coherent description, which enriches the explanatory power of the concepts. The theory also describes the distinctive characteristics of complex evolving systems and thus distinguishes them from machine-type complicated systems.

Connectivity & Interdependence

Complex behaviour arises from the inter-relationship, interaction, and inter-connectivity of elements within a system and between a system and its environment. Murray Gell-Mann (1995/96) traces the meaning to the root of the word. *Plexus* means braided or entwined, from which is derived *complexus* meaning braided together, and the English word "complex" is derived from the Latin. Complex behaviour therefore arises from the *intricate inter-twining or inter-connectivity of elements within a system and between a system and its environment.*

In a human system, connectivity and interdependence means that a decision or action by any individual in a human activity system, (society, group, organisation, or institution), may have an impact on related individuals and systems. That impact will not have equal or uniform effect, and will vary with the 'state' of each related individual and system, at the time. The 'state' of an individual or a system will include its history and its constitution, which in turn will include its organisation and structure. Connectivity applies to the inter-relatedness of individuals *within* a system, as well as to the relatedness *between* human social systems, which include systems of artefacts such as information technology (IT) systems and intellectual systems of ideas. In IS management the man-machine interface is part of that interconnectivity, as well as the requirement to integrate different IT systems, for example, following a merger or acquisition.

Connectivity and interdependence may be part of the purposeful design of the human activity systems. As such it may be explicit and reflected in artefacts such as an organisation chart. But equally it may be informal, non-articulated, tacit and be the result not of purposeful design but serendipitous encounters. Often the explicit depiction of connectivity is no more than a memory of a moment in history. The artefact remains as a legacy of past events.

Complexity theory does not postulate the need for ever-increasing connectivity, for high connectivity implies a high degree of interdependence. This means that the greater the interdependence between related systems or entities the wider the 'ripples' of perturbation or disturbance of a move or action by any one entity on all the other related entities. A high degree of interdependence may not always have beneficial effects throughout the system. When one entity tries to improve its position, this may result in a worsening condition for others. Each 'improvement' in one entity therefore may impose associated 'costs' on other entities, either within the same system or on other related systems. In IT systems non-intended interdependence, in large complicated systems, as every software engineer knows, is often responsible for unexpected consequences when systems are updated or enhanced.

Intense interconnectivity creates multiple and intricate dependencies throughout the system, which cannot be pulled apart. Hence complexity theory suggests that outcomes are often non-deterministic. Interdependence plays an important role in large IT systems, which becomes apparent when one part is changed, as every software engineer knows, and this results in unforeseen and often significant effects in other parts of the system.

Connectivity and interdependence is one aspect of how complex behaviour arises. Another important and closely related aspect is that complex systems are *multidimensional*, and all the dimensions interact and influence each other. In a human context the social, cultural, technical, economic, political and global dimensions may impinge upon and influence each other. Complexity theory distinguishes between complexity in which entities interact often in unpredictable ways and complicated systems. Not all multidimensional systems are complex; machine-type systems for example are merely complicated. A socio-technical system, however, may well be complex and demonstrate all the principles of complex evolving systems.

The defining feature of a CES is that it is able to create a new situation, or in the language of complexity theory, a *new order*, that is, the bringing about of new ways of working, or new structures, or new relationships or even new entities such as a new organisational form. That does not necessarily imply that the new order is an improvement on the old order or that the new order is not in itself destructive (Baskerville and Land, 2004). What is clear is that technology has a significant effect on an organisation and plays an important part in bringing about significant change.

Co-evolution

Connectivity applies not only to elements within a system but also to related systems within an ecosystem. In biology, as entities and organisms interact and adapt within an **ecosystem** they alter "both the fitness and the fitness landscape[1] of the other organisms". (Kauffman 1993: 242) The way each element influences and is in turn influenced by all other related elements in an ecosystem is part of the process of co-evolution, which Kauffman describes as "a process of coupled, deforming landscapes where the adaptive moves of each entity alter the landscapes of its neighbors." (Kauffman & Macready, 1995)

Another way of describing co-evolution is that *the evolution of one domain or entity is partially dependent on the evolution of other related domains or entities* (Ehrlich & Raven 1964, Pianka 1994, Kauffman 1993 & 1995, McKelvey 1999a & b, Koza & Lewin 1998); or *that one domain or entity changes in the context of the other(s).*

In a social activity system co-evolution may be seen as ***reciprocal influence leading to change in the co-evolving entities.*** If influence and change are entirely in one direction then that would be more accurately described as 'adaptation to' a changing environment. However, short-term adaptation may result in long-term co-evolution if the entities in due course influence and change each other. Like genetic evolution organisational or societal co-evolution is largely non-deterministic. We can follow the evolutionary line in retrospect but cannot, in general predict the direction of evolution. Many attempts at business process re-engineering, for example, may in the first instance have had the desired outcome, but in doing so 'deformed the landscape', resulting in unexpected and unwanted adaptation elsewhere in the organisation, sometimes with catastrophic effects on the organisation as a whole.

Nevertheless knowledge of the co-evolutionary process may make it possible to provide an environment in which co-evolution is facilitated. This is relevant in the IS field. The example given at the end of this article will show that when the relationship between IT development and business strategy is seen in terms of facilitating their co-evolution, many of the inherent problems in that relationship may be reduced.

---

[1] Kauffman (1993: 33) borrows the hill-climbing framework with minor modifications, directly from Wright (1931, 1932) who introduced the concept of a space of possible genotypes. Each genotype has a 'fitness', and the distribution of fitness values over the space of genotypes constitutes a *fitness landscape*. Depending upon the distribution of the fitness values, the fitness landscape can be more or less mountainous.

Far-from-equilibrium

Another key concept in complexity is that of 'far-from-equilibrium'. When open systems are pushed 'far-from-equilibrium' they are able to create new structures and order. The original work was done by Ilya Prigogine and his co-authors (Nicolis, Stengers); it applied to physical and chemical systems, but it was of such significance in explaining complex behaviour that the concept has been adopted in other fields. In a social context 'far-from-equilibrium' is taken to mean moving away from established norms, procedures, ways of working and relating.

When an external event disturbs the behaviour of a system significantly, then at a critical point it 'jumps to a new level' and creates new order. The splitting into alternative solutions is called ***bifurcation***, but the term is misleading in that it means a separation into *two* paths, when there may be several possible solutions. Before the system settles into one solution, several alternatives were possible.

An observer could not predict which state will emerge; "only chance will decide, through the dynamics of fluctuations. The system will in effect scan the territory and will make a few attempts, perhaps unsuccessful at first, to stabilize. Then a particular fluctuation will take over. By stabilizing it the system becomes a *historical object* in the sense that its subsequent evolution depends on this critical choice." (Nicolis & Prigogine, 1989: 72)

Innovation takes place at the critical point, when the existing order can no longer be sustained and new order comes into being. Once the decision is made, there is a historical dimension and subsequent evolution may depend on that critical choice; but *before* the decision is finalised, the alternatives are sources of *innovation* and *diversification*, since the opening up of possibilities endows the individual and the system with new solutions. When a social entity (individual, group, organisation, industry, economy, country, etc) is faced with a constraint, it finds new ways of operating, because away-from-equilibrium (established norms or patterns of work and behaviour) systems are forced to experiment and explore their *space of possibilities*, and this exploration helps them discover and create new patterns of relationships and different structures.

Exploration-of-the-space-of- possibilities

Complexity suggests that to survive and thrive an entity needs to explore its space of possibilities and to generate variety. Complexity also suggests that the search for a single 'optimum' solution may be neither possible nor desirable. Any solution can only be optimum under certain conditions, and when those conditions change, the solution may no longer be optimal. If however, a variety of possible solutions exists, then as the environment changes the system is able to draw on these alternatives which may have become more appropriate in the new circumstances. This idea is analogous the Rosenhead's description of what he calls robust design. (Rosenhead, 1989) The idea of an optimal solution is embedded in the IS community and often creates problems such as legacy systems, which were once appropriate but have become 'stuck' in a narrow solution space. The move towards evolutionary development has helped but the idea of allowing different alternatives to be tested in, for example, different sites may prove to be beneficial. Uniformity and homogeneity restrict future development and the drive towards tighter and narrower standardisation is actively inhibiting exploration of possible alternatives, which may become significant when the environment changes.

Self-organisation, Emergence and the Creation of New Order

Kauffman in the 'Origins of Order: Self-Organization and Selection' (1993) brings the importance of self-organisation in the evolutionary process to our attention. He calls Darwinian natural selection a "single singular force" and argues that "It is this single-force view which I believe to be inadequate, for it fails to notice, fails to stress, fails to incorporate the possibility that simple and complex systems exhibit order spontaneously." (Kauffman, 1993: xiii) That *spontaneous order* is *self-organisation* and he argues that natural selection is not the sole source of order in organisms and suggests that both natural selection and self-organisation are necessary for evolution; he then proceeds to expand evolutionary theory to incorporate both evolutionary forces.

Emergent properties, qualities, patterns, or structures, arise from the interaction of individual elements. They are the structures or patterns that appear at the next macro level as a result of interaction at a lower micro level. The relationship between the micro-events and macro-structures is iterative - it is a *co-evolutionary* process whereby the individual entities and the macro-structures they create through their interaction, influence each other in an ongoing iterative process. Emergence is the *process* that creates new order together with self-organisation.

In an organisational context, *self-organisation* may be described as the spontaneous coming together of a group to perform a task (or for some other purpose); the group decides what to do, how and when to do it; and no one outside the group directs those activities. Self-organisation needs to be distinguished from self-management. For example the work system set up within Volvo factories, which designed its own work systems was primarily self-managed as it was deliberately set up by management, although it may have included elements of self-organisation. The following example illustrates self-organisation in an Integrated Product/Project Team (IPT) in the Aerospace industry. The team was brought together to create a new product. The members of the team represented firms, which outside the IPT were competitors, but within the team had to cooperate and to create an environment of trust to ensure that sensitive information, necessary for the creation of the new product, could be freely exchanged. The team had to prepare a six-monthly report for its various stakeholders. This report was on hard copy and was usually several inches thick. Some members within the team decided that they would try an alternative presentation. They found that they had the requisite skills among them and they put in extra time to produce the next report on a CD. The coming together of the sub-team to create the new format for the report illustrates the principle of self-organisation. No one told them to do it or even suggested it. They decided what to do, how and when to do it.

Emergence in a human system tends to create irreversible structures or ideas, relationships and organisational forms, which become part of the history of individuals and institutions and in turn affect the further evolution of those entities: e.g. the generation of knowledge and of innovative ideas when a team is working together could be described as an emergent property in the sense that it arises from the interaction of individuals and is not just the sum of existing ideas, but could well be something quite new and possibly unexpected. Once the ideas are articulated they form part of the history of each individual and part of the shared history of the team - the process is not reversible - and these new ideas and new knowledge can be built upon to generate further new ideas and knowledge.

Complexity principles are *scale-invariant* and apply at all scales from an individual to a team, organisation, industry, economy, etc. Furthermore, to understand complex evolving systems it is important to look at several if not all their characteristics. Focussing on one, such as self-organisation, emergence or co-evolution provides only limited understanding, while complexity theory aims towards a holistic understanding of the system and its relationship to all other systems within a social ecosystem. Each characteristic is intimately related with and influences all others. In IS management the classic mistake has been to concentrate on either the IT system or management, when both affect and influence each other. The following example illustrates how facilitating the interaction between IT development and management strategy can overcome some serious issues, such as legacy systems, and may even enable a project to be completed on time.

An Example: Enabling Co-Evolution between IS Development and Business Strategy

The example (Mitleton-Kelly 2004a) will explore the idea that when co-evolution between business strategy and IS development is enabled, the problems associated with legacy systems may be reduced.

The term ***legacy systems*** is taken to mean those IT systems which no longer fully support the business process and which constrain the development of new products and applications. Most systems described as legacy tend to be 20-30 years old, written in assembly, or an early version of a third generation language (Adolph 1996, Sneed 1995, Chikofsky & Cross 1990). Reengineering, reverse engineering, freeze and encapsulate (Bennett 1995) have been suggested as viable solutions to the legacy systems' problem. They are associated with high maintenance costs (Bennett 1994, Waren 1999) and they have become very difficult and expensive to change. Legacy can also be seen as a *gap* between the organisation's business needs and technical capabilities (Ramage & Munro, 1999). However, legacy has also been recognised as a multifaceted socio-technical situation (Gold 1998).

This article 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 article, 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[2].

In the case under study, the organisation (to be referred to as ABank), admitted to a significant legacy problem. It would have preferred to jettison the old legacy systems, which were perceived as those *systems that no longer supported the current business objectives or were 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.

The UK office acknowledged the complex nature of the problem and by breaking the organisational norms and actively encouraging a sustained dialogue over time, between the IS and business strategy domains, created an enabling infrastructure, which in turn helped them overcome the technical constraints.

The dominant culture of the Bank supported one kind of *order*, that is, a particular way of relating and working, which had inadvertently contributed to a legacy problem. A different way needed to be found and the pressure the UK office was under, led a process of *self-organisation* that 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 individuals to co-evolve in a reciprocal evolutionary manner. 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 from both the IS and the business domains. One of the outcomes of this initiative (which can also be seen as an *exploration of the space of possibilities)*, was an amelioration of the legacy problem.

The Legacy Problem

The legacy system of the UK operation was based on IBM hardware, was at least 10 to 15 years old (with 30 year old elements), was 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

---

[2] 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.

elements added, in an effort to continue to support the business, but without full success. The constant modifications did not provide a system tailored to the changing business requirements, but a *"legacy system* (which) *becomes dysfunctional or it becomes disjoint* (with) *the current business".* Furthermore, full replacement was not regarded as an economically and technically viable option.

A variety of business, organisational and technical elements had combined to produce a complex socio-technical system with a very high degree of interconnectivity and interdependence. These multiple elements also explain why it was 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 CES. If it is seen from that perspective then certain generic characteristics common to complex evolving systems may be identified which are relevant and applicable to this example. The following four interconnected elements contributed to the complexity of the system.

(1) One element arose from increasing interconnectivity and interdependence among the **system components and the applications**. The bank often customised or engineered solutions into its systems for individual customers, and changed coded components. Over time a layered system infrastructure was created and the interconnectivity and interdependence became so intricately intertwined that a point was reached when *"to undo that complexity is almost insurmountable"*. It became very difficult to fully tailor yet another application. Hence the bank *"cuts and pastes"... and you get to a situation where you are suddenly generating subsets for different customers."* With layering and new subsets the systems became increasingly intertwined and gave rise to **emergent** properties, i.e. properties, which were 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) **Organisational restructuring** (a social aspect) **had changed the systems' architecture** (a technical aspect). The main European system was on two hardware bases, using HP and IBM hardware. Originally the IBM system was implemented in six 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"* which run all the local bank's activities. Since then, the bank went through several phases of organisational restructuring, which impacted the systems' architecture, until all the hardware for the six European countries with IBM systems became based in the UK.

(3) A third component was that the bank had 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 communicated with it. They were Windows NT based front-end servers. But they had not succeeded in replacing the full set of legacy software. The part replacements used current technology.

(4) **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. There were so many interdependencies and linkages that isolation of specific modules became extremely difficult. 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 were managed and owned locally in a single country. This example shows how the intricate interrelationship between technological and organisational factors created 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 was that the maintenance and further development of the IT systems had been centralised within the UK group, which controlled 16 systems on both HP and IBM platforms. Thus, as resources for the maintenance and support were held centrally, there was **no local knowledge** of the branch technology of the system. 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, 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 was lost.

Despite the above problems, the UK operation did complete the project on time, by creating an enabling environment, which facilitated the interaction and the subsequent co-evolution between IS development and the business strategy.

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 and allowed the necessary space for self-organisation and autonomy.
- *Stability*: **s**ufficient *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 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 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 could 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 profit and loss. 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. Further, it is not possible to fully predict the way co-evolution will work out, and what may be seen as a minor development of the system can have far reaching consequences. For example, a new capability in the system may make the whole system vulnerable to entry by a virus.

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 example, 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 example 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** which allowed space for (iv) *autonomy* and freedom to *self-organise*; and the realisation that (v) *"a cross-domain process, was a successful way to run business drive requirements".* Above all there needs to be a constant awareness of the effect of the interdependencies whether these are designed or simply evolve without formal explication.

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*.

Bibliography

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

Baskerville, R.L. & Land, F. (2004), Socially Self-Destructing Systems, in Avegerou, C.,, Ciborra, R., and Land, F. (Eds.), *The Social Study of Information Systems and Communication Technologies*, Oxford University Press, pp263-285.

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

Chikofsky, E.J., & Cross, J.H. (1990) "Reverse Engineering and Design Recovery: A Taxonomy." IEEE Software, January 1990. pp 13-17.

Ehrlich, P.R. and Raven P.H. (1964) *Butterflies and Plants: A Study in Co-evolution.* Evolution, 18: 586–608.

Gell-Mann, M. (1994) *The Quark and the Jaguar: Adventures in the Simple and the Complex.* W.H. Freeman.

Gell-Mann, M. (1995) *Complexity Journal.* 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.

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. (2000) *Investigations.* Oxford University Press.

Kauffman, S. & Macready, W. (1995) *Technological Evolution and Adaptive Organizations.* Complexity, Vol 1 No 2 pp.26-43.

Koza, Mitchell, P. & Lewin, A. (1998) *The Co-evolution of Strategic Alliances.* Organization Science, 9: 255–264.

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.

Mitleton-Kelly E. and Papaefthimiou MC. (2000) *'Co-Evolution and an Enabling Infrastructure: A Solution to Legacy?'* in 'Systems Engineering for Business Process Change' Edited by Henderson P., Springer-Verlag, ISBN-1-85233-222-0

Mitleton-Kelly E. (2000) *'Complexity: Partial Support for BPR'* in 'Systems Engineering for Business Process Change' Edited by Henderson P., Springer-Verlag, ISBN-1-85233-222-0

Mitleton-Kelly E. and **Papaefthimiou** MC. (**2001**) *'Co-Evolution of Diverse Elements Interacting within a Social Ecosystem*' in 'Systems Engineering for Business Process Change' Vol. 2, Edited by Henderson P., Springer-Verlag, Oct. 2001, ISBN 1-85233-399-5

Mitleton-Kelly E. (2003) Ed. *Complex Systems & Evolutionary Perspectives of Organisations: The Application of Complexity Theory to Organisations*, Elsevier, ISBN 0-08-043957-8

Mitleton-Kelly E. (2003a) Ten Principles of Complexity & Enabling Infrastructures. In *Complex Systems & Evolutionary Perspectives of Organisations*, Elsevier, ISBN 0-08-043957-8

Mitleton-Kelly E. (2003b) *'Complexity Research - Approaches and Methods:* The LSE Complexity Group Integrated Methodology' in Keskinen A, Aaltonen M, Mitleton-Kelly E *"Organisational Complexity"*. Foreword by Stuart Kauffman. Scientific Papers 1/2003, TUTU Publications, Finland Futures Research Centre, Helsinki, 2003

Garcia-Lorenzo L., Mitleton-Kelly E., Galliers R.D. 2003 Organisational Complexity: Organising Through the Generation and Sharing of Knowledge in The *International Journal of Knowledge, Culture and Change Management,* Vol 3 Article MC03-0023-2003, Ed by Kalantzis M. and Cope B.

Mitleton-Kelly E. 2004 An Integrated Methodology to Facilitate The Emergence of New Ways of Organising, Conference Proceedings of the 3rd European Conference on Research Methodology for Business and Management Studies (ECRM) April 2004.

Mitleton-Kelly E. 2004a IT Legacy Systems: Enabling Environments that Reduce the Legacy Problem: A Complexity Perspective, in *Software Evolution*, edited by N. H. Madhavji, M.M. Lehman, J.F. Ramil and D. Perry, Wiley, 2004

Mitleton-Kelly E., Speh Birkenkrahe M. 2004b Treasury Operations in A Multi-National Oil Company: Dialogue on the Methodology of Using Natural Experiments in Complex Evolving Systems, in *Complexity Theory and Management of Networks*, Ed. Andriani, P & Passiante G, Imperial College Press, ISBN 1-86094-460-4.

Mitleton-Kelly E. 2004c Co-Evolutionary Integration: A Complexity Perspective on Mergers & Acquisitions, 20th EGOS Colloquium July 2004 Ljubljana University, Slovenia

Mitleton-Kelly E. 2004d (forthcoming) The Information Systems Professional as a Hermit: Of Plural Rationalities, Information Rejection and Complexity, in *Innovation: The European Journal of Social Science Research*

Nicolis, G. and Prigogine I. (1989) *Exploring Complexity.* W.H. Freeman.

Pianka, E.R. (1994) *Evolutionary Ecology.* New York: HarperCollins.

Ramage, **M., & Munro, M. (1999)** 'It's not just about old software: a wider view of legacy systems', (Unpublished)

Rosenhead, J (1989) 'Robustness analysis: keeping your options open', in J.Rosenhead, (ed), Rational Analysis for a Problematic World: problem structuring methods for complexity, uncertainty and conflict, Wiley, Chichester, (pp 193-218).

Sneed, **H.M (1995)** 'Planning the Reengineering of Legacy Systems', IEEE SOFTWARE, Jan 1995, pp 24-34.

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