Journal of Social Sciences 4 (4): 308-319, 2008
ISSN 1549-3652

Understanding Information System Failures from the Complexity Perspective

Indranil Mukherjee
School of Management, West Bengal University of Technology, BF 142, Sector I, Salt Lake,
Kolkata – 700064, INDIA

Abstract: Problem Statement: Failure of computer based information systems had been a source of major concern in the modern technological era. Information System (IS) researchers have spent a significant amount of their time and effort in understanding the recurring failure of Information Systems. Studies in this regard have ranged from being primarily technical in their approach to those having a much stronger socio-technical bias. The purpose of this paper was to analyze information system failures using the lens of complexity theory. **Approach:** Complexity theory was proposed as an alternative paradigm for understanding and analyzing Information System failures. Other research frameworks within which system failures were studied were also discussed. The core concepts of complexity and the salient features of information systems were elucidated. It was shown that information systems could be interpreted as complex entities both from the structural and functional viewpoints. Pictorial representations were given to corroborate this point. **Results**: It was shown that the complexity framework could be utilized to understand the different types of system failures, viz. process, correspondence and interaction failures in a more meaningful way. The idea of recurrent failures was also examined in the context of complexity theory. It was shown how such failures could be tackled much better by using lessons drawn from complexity. The inadequacy of the systems approach was pointed out that necessiated the introduction of complexity. **Conclusions**: It appeared that adopting certain features of complexity in the analysis, design and management of information systems could help in avoiding certain failures related to information systems. Some of these features were facilitating the process of co-evolution, exploring the space of possibilities and encouraging self-organization and emergent behaviour.

Key words: Complexity, Complex Adaptive Systems, Information System Failures

INTRODUCTION

Human beings have long tried to ease the uncertainty so widely prevalent in their lives by predicting the future. This effort has been bolstered by the epoch-making progress in science and technology over the last century. In fact, it is the way in which hazards are being handled that has distinguished modern ‘man’ from ‘his’ predecessors, as Peter Bernstein[1] argues in his much acclaimed work *Against the Gods*. For modern man, hazards have to be controlled and consequent problems overcome by himself, through systematic application of science, technology and most of all- mathematics. God, in the modern era, has gradually been replaced by equations.

The domain of computer based information systems is one where this systematic/engineering rationale of regulating and/or managing risks has seen its greatest manifestation. But paradoxical as it may sound, the risk involved with and the failure rates of computer based information systems have proved to be of significant concern in this modern technological era.

Major disasters have often hit the world due to the failures of the associated information systems, not to mention the large failure rates of small to medium size software projects. Some of the prominent examples of such failures are: (a) London Ambulance Service Computer Aided Dispatch System in 1992 which cost the city of London approximately 2.3 million dollars and led to the loss of around twenty odd lives[2]; (b) Denver airport baggage system in 1995 which got nearly 24 months late; (c) e-Bay site failure on June 10-11, 1999; (d) hardware flaw undermining the accuracy of Intel Pentium chips in 1994 which eventually forced Intel to take a $475 million write-off to account for replacing defective chips[3]. A very large number of projects are delivered with missing functionality (promised for delivery in later versions). Between 30 to 40% of all software projects are “runaway” projects that far exceed original schedule and budget predictions and fail to perform as originally specified[4]. It is no wonder then that a substantial gamut of Information Systems research over the past three decades has remained primarily focused on the exploration of possible causes of IS failures, ranging from failure to fulfil users’ hopes

Corresponding Author: Indranil Mukherjee, School of Management, West Bengal University of Technology, Kolkata – 700064, India

as well as designers' promises[5] to operational problems and other unanticipated difficulties. This research ranges from mere technological explanation of success and failure to a rich and complex analysis of human organizational systems. Friedman and Cornford[6] in their premier account of the history of the development of computer based information systems, perceive failures as playing a pivotal part in shaping the dynamics of information and communication technologies (ICTs). Characterized as persistent, pervasive and pernicious[7], IS failures have continued to attract the attention of a number of academic scholars and practitioners in the field who have tried to analyze and understand the problem from different perspectives. The multidimensional character of the subject area is reflected in the number of issues related to it – (a) *conceptual*: concerned with the very nature of the subject; (b) *empirical*: attempting to establish a causal relationship between different factors and probing into the effects of those factors; (c) *normative*: designing developed tools and techniques for building and maintaining successful IS applications[8] (Sauer, 1993), which the different approaches continue to address.

There is probably no unified framework to understand failures of information systems. Whatever research - conceptual, empirical or normative - has been carried out in the past and is still in the process of being undertaken, can broadly be seen as representative of different schools of thought, with fundamentally different epistemologies inherent to them. In this paper, an attempt is made to apply the complexity framework to Information System failures. The paper is organized as follows. The theoretical background necessary to discuss about information system failures is first developed. Different research frameworks that are currently being utilized in the study of information system failures are introduced next. Fundamental elements of complexity are then introduced and discussed. The next section takes a first look at information systems through the lens of complexity, emphasizing the structural and functional similarities between information systems on one hand and complex systems on the other. The last section probes information system failures from the complexity perspective, spells out useful lessons for the system designer, discusses the issue of recurrent failures, points out the inadequacy of systems theory in explaining IS failures, and concludes by analyzing the London Ambulance Service case.

Theoretical Background: IS failure is not a well-defined concept. It covers different types of experiences and outcomes. Three types of IS failure were initially characterized by Lyytinen and Hirschheim[9]: process, correspondence and interaction failure. Process failure refers to outcomes of the system development process such as project abandonment, schedule slippages, budget blowouts, or financial crises during the implementation stage. In correspondence failure, IS projects are completed but fail to meet specified objectives such as monetary savings, improved efficiency of resource allocation, greater productivity etc. Interaction failure occurs during or after project completion. It refers to failure to use an IS due to users' non-acceptance because of low level of user involvement or low degree of user satisfaction. Each of these three concepts was criticized as being limited in that one takes no account of the forms of failure defined by the other two. Lyytinen and Hirschheim thus proposed the fourth concept of expectation failure (inability of an information system to meet a specific stakeholder group's expectations). Their work basically stressed on the fact that IS failures occur due to the existence of a gap between an existing situation and a desired situation for a particular stakeholder group in an organization.

Given the rather heterogeneous background in the area of IS failures, research in such failures has been diverse in nature. This variety is clearly manifested as the focal point of the analysis shifts from technological understanding to a systems level approach, considering individuals, stakeholders and organizations as entities of interest and probing into organizational culture, the unique situational contexts and power conflicts germane to the situation. Each of these outlooks stems from different schools of thought, which are at the origins of the different epistemologies used in research into IS failures, and in a more general context, in any sort of IS research.

Currently, research into IS failures is concentrating on the different types of failures outlined above, different varieties of systems, different types of sectors and a number of different organizational change initiatives. The chief problem in formulating feasible theoretical models about IS failures is the high degree of complexity arising out of the intricate combination of technical, human and organizational characteristics of any information system. It is extremely difficult or even impossible to explain, let alone predict, the behaviour of such systems. Another related problem is the fact that previous studies of IS failure generally emphasized single failure common to all failure contexts. This concept of single failure was elucidated through the perspectives of organizational politics, organizational culture, institutional theory and organizational learning[10]. However, failure is not truly a single phenomenon, but rather a diverse set of phenomena including certain failures which are recurrent in nature. New kinds of failure have surfaced in recent years like those related to BPR and BPO and thereby led to more complications[11], [12]. Researchers are trying to grapple with the problem of containing IS failures by looking at both risk containment and risk control strategies, and also by giving up the holistic approach to risk management and rather adopting more specialized techniques.

Researchers are exploring different methodologies in order to analyze and understand IS failures more effectively. In general, a research methodology may be regarded as a philosophical framework or "point of view" within which a set of methods can be systematically applied. There are three broad philosophical perspectives in relation to qualitative research in the domain of Information Systems[13] , viz. positivist, interpretive and critical research. The stance taken by positivist researchers in Information Systems is based upon the assumption that reality is objectively given and that it can be described by reference to measurable properties that are independent of the researcher. Interpretive IS researches, on the other hand, consider that reality can only be accessed through social constructions such as language, consciousness and shared meanings[14]. Critical research relies on the assumption that social reality is historically constituted and that people are constrained in their actions by different forms of cultural and political domination[15]. Efforts are being made to extend the existing research approaches in the area of information systems. Brey[16] proposes that technological change – which is at the heart of all information system projects – can be most easily understood in the context of technological controversies, disagreements and difficulties with which the actors involved in the change are concerned. Brey tries to adopt an approach using some form of social constructivism in which the researcher does not need to evaluate claims made by different groups about any "real" properties of the technology being studied. Brey classifies social constructivist approaches into three groups: strong social constructivism, mild social constructivism and actor-network theory.

As part of the effort to enlarge the scope of research in the area of information systems, the present paper will try to understand the failure of information systems in the framework of complexity theory[17, 18]. Complexity science originated in the natural sciences, but is now rapidly encompassing diverse disciplines like physical sciences, biological sciences, social and organizational studies, economics, etc[19, 20, 21, 22, 23]. Complexity deals with systems composed of many interacting agents. They interact in ways that continually reshape their collective future. While complex systems may be difficult to predict, they can have a good deal of structure and permit improvement by thoughtful intervention. Organizations are now making serious efforts to use the ideas inherent in complexity theory to create conditions for innovation and creativity in order to survive and grow in the bigger world environment. It is the exposition of this approach in relation to the investigation of IS failures that will be one of the major building blocks of this work.

Mitev[24] summarizes this evolution of understanding IS failures along three dimensions, viz. the epistemologies, their originating disciplines and the focus of the different failure models. For example, functionalism as an epistemology has traditional engineering and natural science as its originating disciplines with technology as the focus of the failure model built on it.

A Brief Review of the Different Research Frameworks: Three different frameworks used by IS researchers to study system failures are discussed here, viz. functionalism, interpretivism and the critical school.

Functionalism: Functionalism is based on the philosophy of '*logical positivism*' which maintains that methods of natural science are the only legitimate methods available for use in social sciences. Studies undertaken in this framework look at the importance of presence or absence of specific factors for enhancing the success of information systems. Studies have been undertaken by Rockart[25], who proposed the idea of 'critical success factor*s'* or CSFs, Shank[26] who used it with the MIS planning in his own organization FIAC and Boynton[27]. Flower[28] introduced the notion of the 'critical failure factors' or the CFFs. A related piece of work was done by DeLone and McLean[29]. The functional approach has been criticised for attempting to establish a causal relationship between social events using natural science propositions.

Interpretivism: Interpretive research methods address the problem of dichotomy between the social and technical worlds by considering the knowledge of reality as a socially constructed one by the associated human actors[30]. Examples include Vaughan's account of the Challenger launch decision at NASA[31] and studies like those of Harvey and Myers[32]. Using the interpretive framework, Charles Perrow[33] refuted the functionalist notion of failures in structured systems as the outcome of a set of discrete and independent flaws in their design. He perceived failures as the result of the inherent *i*nteractive complexity and tight coupling of complex elements in the system. The contextual factor centred analysis, due to Sauer, provides a rich understanding of IS failures from the interpretive perspective. This model focused on project organization, the information system and its supporters arranged into a *triangle of dependences.* Sauer also examined the dependence of IS failures on the level of organizational configuration through his Configurational *Theory* model[7] .

The Critical School: The critical IS researchers perceive power and politics as inescapable features of every organisation. They mainly analyse the implementation phase- where user resistance is most frequent, and investigate the causes of such resistance. Using the Lyytinen–Hirschheim terminology for IS failures, we can say that the critical framework is concerned with interaction failures. Initial work in this direction was due to Robert Kling[34] and then by Markus[35]. Warne and Hart studied the impact of organizational politics on the failure of an information system project[36] by using the concept of information wards. It is built on the fundamental assumption of organisations being pluralistic in nature

The interpretive and the critical schools of thought have both been subjected to one common criticism. According to Mitev, "they construe social factors as exogenous from the technology and imply the pre-existence of a technological 'natural' trajectory." [24].

In subsequent sections, we explain how complexity theory can help us better understand information system failures.

METHODS: In this section we introduce complexity theory and see how it can help us to understand information system failures better.

Complexity Theory: There is no single unified theory of complexity, but many different theories arising out of the different disciplines studying complex systems like physical and biological sciences, computer simulation, social sciences, organizational studies and the entire spectrum of management and allied disciplines. The theories of complexity provide a conceptual framework, a way of thinking and a way of seeing the world. Depending on the particular discipline, the specific problems addressed, and the individual researcher's background, the complexity approach has been characterized as "constituting everything from a major paradigm shift which challenges established scientific orthodoxy to an extension and refinement of existing theory" [37]

It is not easy to define complexity in a precise manner, because of its apparently paradoxical and all-embracing framework. Pribram[38] comes to the closest description: not until we come to realize that the concept is paradoxical, can we begin to understand it. The dictionary definition of complexity always hints at the paradox: complex, (1) from the Latin co (together) plectere (weave, braid): entwined; intricate; (2) from the Latin complexus (embracing, surrounding); anything formed by the union of interconnected parts; as assemblage; a system. Note that definition (1) emphasizes the intricacy of internal structure while definition (2) embraces the 'system' as a whole . Complexity is thus shown to encompass two levels or scales- a multiform interior or micro-level and a macro-level whole. It is, therefore, a cross-scale, cross-level concept, and must be measured accordingly. The paradox: an intricate diverse micro-level embraced within a simpler macro-level entity- unity in diversity. Complex systems, both physical and social, require a high level of uncertainty at the local level for stability and development. Complex systems are characterized by global structure and local randomness. The former maintains the strength of the whole. The local randomness creates innovation and resilience.

It is probably easiest to convey the meaning of complexity by enumerating some of the properties of complexity:

Complex systems contain many constituents interacting nonlinearly. The constituents of a complex system are interdependent. The complexity of the system does not arise from complex rules, but rather from the interaction among the large number of entities or subsystems that the entire system is composed of. These entities interact with each other constructing and reconstructing schemata that organize their relations at the local level.

A complex system possesses a structure spanning several scales. Consider the case of a government, which is an example of a complex system. There are different levels (scales) in the government: local (city corporations or municipalities), state and the union government. At every scale there is a structure present.

A complex system is capable of emergent behaviour. Emergence occurs when we switch the focus of our attention from one scale to the coarser scale above it. A certain behaviour, observed at a certain scale, is said to be emergent if it cannot be understood when we study, separately and one by one, every constituent of this scale, each of which may also be a complex system made up of finer scales. Rather, the emerging behaviour is a new phenomenon special to the scale considered, and it arises from the global interactions between the scale's constituents. In other words, it is hard to predict the emergence of the global property from the knowledge of the component parts. This global property – the emergent behaviour – feeds back to influence the behaviour of the individual components that produced it. For example, in industrial societies, the aggregate behaviour of companies, consumers, and financial markets produces the modern capitalist economy. For the brain, billions of neurons interact to yield complex behavioural patterns. Murray Gell Mann, one of the founding fathers of the science of complexity has described this process as "surface complexity arising out of deep simplicity".

The combination of structure and emergence leads to self-organization, which is the outcome when an emerging behaviour has the effect of changing the structure or creating a new structure. Each higher level has its own scale, and each new level has new kinds of relationships and properties. This means that a complex system at one level is made up of lower level complex systems interacting and creating the higher level order. Self-organization is not a strictly nested phenomenon; complex webs of interactions may exist at all levels. Viewed from an organizational perspective, self-organization describes the spontaneous coming together of a group to accomplish a task – the group alone decides what to do, how to do and when to do the job with no direction from outside the group.

There is no single point of control in a complex system. Complex systems show global coordinated behaviour, without the presence of any distinct global controller. A complex system must cultivate variety, but it is an illusion to think that one can direct the variations. One can only disturb the system and observe what is happening. At the same time one cannot be separate from the system, stand outside and influence its direction. For example, neither has the immune system got any "chief lymphocyte" which directs an immune response; nor does the brain have a "chief neuron" in which central control of all behaviour is located.

One way of modeling complex behaviour examines regularity that emerges from the interaction of individuals connected together in a Complex Adaptive System (CAS). These are dynamical systems, which are able to adapt and change within, or as part of, a changing environment. The hallmark of this perspective is the notion that at any level of analysis, order is an emergent property of individual interactions occurring at a lower level of aggregation. A formal definition runs as follows: A Complex Adaptive System (CAS) denotes an open ended system of many heterogeneous agents which interact nonlinearly over time with each other and their environment and which are capable of adapting their behaviour based on experience. Open ended means that there is essentially limitless possibility for variability in agent characteristics and behaviour[39] . There are four major properties of the aggregate dynamics of CAS[40] that distinguish them from other systems: self-organization, evolutionary trajectories, co-evolution and punctuated equilibrium. All these properties are emergent in the sense that complete knowledge of the individual agents is not sufficient to conclude about the details or timings of the aggregate properties.

While we have already discussed about self-organization, the concept of evolutionary trajectories implies that the future of a given system from a given point in time cannot be determined by the complete knowledge of the present state. If the "tape is rerun" many times, every trajectory will most likely to be unique.

In the case of co-evolution, we observe that a large part of each agent's perceived environment consists of interactions with other agents, who are themselves adapting and evolving. Each agent interacts not only with other agents at the same level in the organizational hierarchy (for example, when firms compete in an industry), but also with agents at higher and lower hierarchical levels (for example, a firm's relation with employees). Thus the evolution of one domain or entity becomes partially dependent on the evolution of other related domains or entities[41]. Organizational co-evolution is essentially non-deterministic in nature. In retrospect, it may be possible to follow the evolutionary line, but it is virtually impossible to predict the direction of evolution.

Punctuated equilibrium is the tendency of a CAS to have stable patterns of activity for long periods of time,then a short transition period of very rapid change in patterns, followed by new stable patterns of activity.In open-ended CASs, it is difficult to predict when transitions will occur or what the resulting stable patterns will be. The question depends to a large extent on how the system is viewed. For example, macroeconomists might view the economy of a particular country as steadily growing, with a few spikes here and there and conclude that the country's economy is in an equilibrium state. However, at the level of the firm, thousands of firm might go out of business every year and new ones formed in a way, which can hardly be described as an equilibrium state.

Complexity and Information Systems: A First Look like complexity science, the process of information system development and management is an area that requires contributions from different disciplines[42, 43] . To build an information system, various types of tasks are required to be performed by different kinds of specialists: programmers, system analysts, knowledge engineers, project managers, software engineers, and of course users. While technical experts should develop a clear understanding of where their special skills fit into the overall effort that an organization makes when it is involved in the task of installing a new information system or upgrading an old one, managers need to appreciate the effort and particular expertise that is necessary in information system development. Information systems can be understood from different perspectives, viz. functional, technical, organizational etc. each of which can reveal a rich repository of ideas about the organization in its entirety. A study of the information system development process, ranging from the engineering of a complex technical artifact to the interventions leading to organizational changes, that is, a socio-technical approach, highlights that the process of analysis, design and implementation of an information system is an inherently complex process. A proper understanding of information systems calls for combining knowledge from applied sciences such as computer science or engineering, systems theory, decision theory etc. with that from social sciences such as organizational theory, sociology, psychology, behavioral science etc. The former tends to apply scientific, mathematical or other formal logical methods, and aims at producing precise knowledge. The latter favours interpretative methods and produces more qualitative knowledge, which aims more to explain, rather than to describe the behaviour in a situation under study.

A pictorial way of illustrating the relationship of the field of complex systems to the many other fields of science is indicated in Figure1[44]. It shows the conventional view of science as progressively separating into disparate disciplines in order to gain knowledge about the ever-larger complexity of systems. It also illustrates the view of the field of complex systems, which suggests that all complex systems have universal properties. Because each field develops tools for addressing the complexity of the systems in their domain, many of these tools can be adapted for more general use by recognizing their universal applicability. This leads to the motivation for cross-disciplinary fertilization in the study of complex systems, which in turn appreciates the multidisciplinary nature of IS, rather than developing a mere truncated discourse.

The vast array of subjects that underlie the discipline of information systems reminds us of the complexity paradigm which also evolves from a large number of simple systems belonging to different categories on their own. The analogy becomes more pronounced when we move beyond the structural to the functional area of information systems. It is true that the uneasy mix of qualitative and quantitative methods emerging from the differing perspectives is manifested in both the development and management of information systems. However, transcending the barriers set by the narrow and ill-grounded body of specific knowledge domains, all information systems possess some universal features, just as all complex systems display some universal properties like emergence and self-organization. For example, the basic goal. For any information system development is to endow an organization with a sustainable competitive advantage. The discipline of information systems thus mirrors a complex system both from the structural and functional viewpoints. Therefore, it is appropriate to analyze information systems in the framework of complexity theory.

RESULTS

One of the major problems in IS management is the divergence between the business and the IS strategies of an organization. The focus has been either on technology or management, whereas in reality, both influence each other in a reciprocal manner.

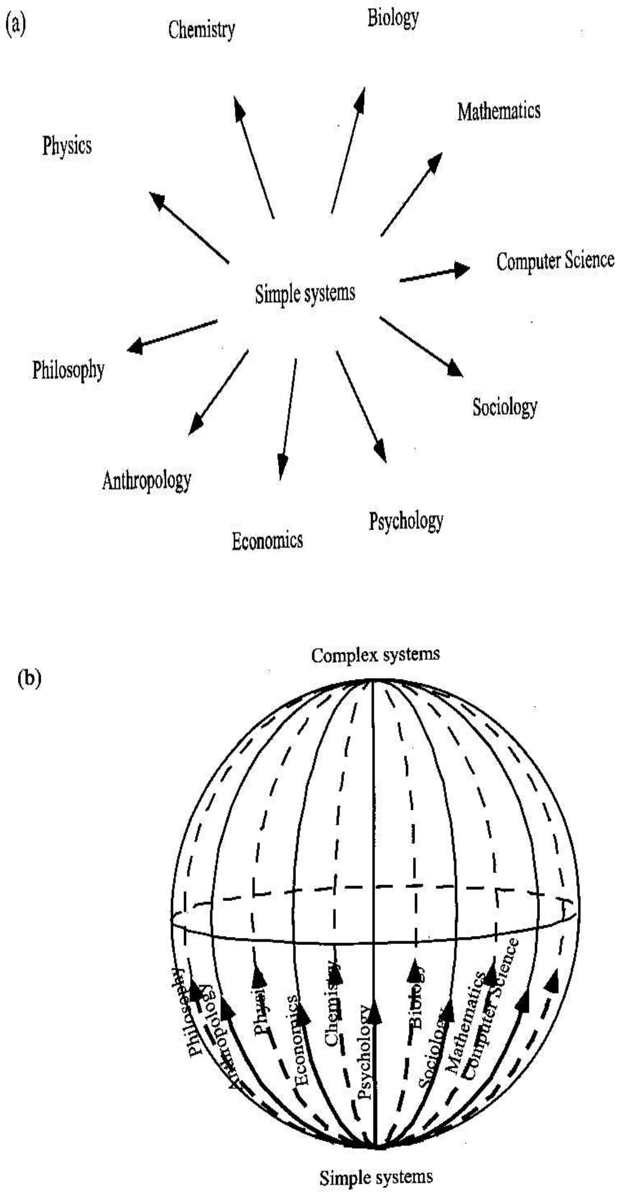

Fig 1: Relationship between Complexity Science and other Natural/Social Sciences

It is necessary to facilitate the interaction between IT development and business management strategies in order to overcome among others, correspondence failure. The objectives of an IS development project are guided by the business goals of the organization, e.g. cost savings, greater productivity or increased efficiency. The IS developers need to manage the project in such a way as to help achieve these objectives. Inability to do so leads to what is termed as correspondence failure. But if it is possible to facilitate the co-evolution of business strategy and IS development then most of the problems intrinsic in this relationship are likely to disappear. Knowledge of the co-evolutionary process is essential in order to create an environment where this co-evolution may be facilitated. At the operational level, this calls for greater transparency in the working of both the business and technology domains, improved communication between the two, frequent interactions and exchange of ideas, leading to the emergence of common goals based on a sense of strong trust and mutual understanding.

A related and very commonly occurring problem in IS development is the inability of the system designers to understand user requirements. Organizations using information systems are normally in a constant state of flux. This ingrained fluidity within organizations makes it difficult to freeze user requirements. The constantly changing working environment within organizations ensures that a gap between information systems and user requirements appears much before implementation is completed. While IS development lead times are shortening, there are no "instant solutions" which could overcome the time related problems by reducing the gap between problem definition and solution realization.The different backgrounds, expertise and goals of the information system developers (typically technical specialists) and the end users can also lead to user resistance towards the adoption of a new information system. Lack of motivation or involvement on the part of the users can easily lead to interaction failure. Again, the way out of this difficulty is to facilitate co-evolution between system development and requirements analysis. This may be accomplished by promoting user involvement in the design and operation of information systems. Users must be given more opportunities to shape the system according to their priorities and business requirements and thus to control the outcome of the change process. External integration tools consist of ways to link the work of the implementation team to users at all organizational levels. Users can become active members of the project team, take on leadership roles, and take charge of installation and training. However, here one must pause to reiterate that the emphasis should be on the co-evolution of the two entities involved and not merely adaptation by one of the agents to a changing environment.

Information systems are developed in order to add value to an organization by addressing a specific problem or a set of problems that the organization needs to overcome. These problems are generally multi-faceted in nature involving human, physical and technical factors. IS developers have traditionally attempted to zero in on an optimal solution to tackle such problems. This attempt has often led to failed IS development projects, the failure occurring due to cost or time overruns, sometimes even resulting in project abandonment examples of process failure. However, a solution is optimal only under a certain set of conditions and when these conditions are altered the solution may not be optimal any more. We have already pointed out the inherent fluidity existing in organizations which try to leverage information systems to solve their problems. Hence, an optimal solution may be infeasible, or even undesirable. Complexity theory argues that instead of choosing a singularly focused optimal solution, IS developers need to explore possible multiple solutions, some of which could even operate in parallel. Depending on the change in its environment, the organization could then utilize one of the alternative solutions, which may become more relevant under the changed circumstances. Generating variety is a central requirement for adaptation and leads to the rich dynamics present in complex systems. In the terminology of complexity, this is the process of exploration-of-the-space-of-possibilities. Uniformity, homogeneity and rigidly following standard procedures, all of which are hallmarks of standard IS development process need to be replaced by a more flexible approach which encourages co-evolution. This is also consistent with the fact that complex systems do not have a global controller, rather they thrive on self-organization.

Recurrent Failures: Earlier in this paper, we pointed out the inadequacy of the single failure concept in relation to research in the field of IS failures. One of the probable reasons for this shortcoming is the complex nature of these systems arising out of their intrinsic nonlinear character. In these systems small disturbances can multiply over time because of the presence of non-linear relationships and feedback effects, making them extremely sensitive to initial conditions. Past events always have a significant but somewhat unpredictable effect on present activity as well as future plans. An example is one where a decision to commit to a particular type of technology had surprisingly large effects at a later time when a change in IT platform was considered and found to be impracticable. In many cases, old mistakes get repeated in new, original ways during the process of system development, leading to recurring failures. IT departments seldom learn from previous successes or failures. What is more remarkable is that even if lessons are learnt from previous IS development experiences, they may not really prove to be useful. Recording of these lessons in terms of standards and best practices may not guarantee future success since the unexpected could occur, new patterns emerge, and intended practices culminate in unintended outcomes. Complexity theory, which tends to focus on exploratory analysis, co-evolution and self-organization, could just be the right candidate for overcoming these kinds of recurring failures. Instead of religiously following the so-called best practices, complexity would explore the space of possibilities and search for the set of optimal solutions under changing scenarios. By facilitating co-evolution between different domains of the business, between management and technology, and between users and developers, complexity would encourage the emergence of a new working order, which is not based on command-control relationship but which emphasizes nimble reactions – expecting the unexpected. Instead of perceiving the fluidity present in an organization as a threat, use of complexity could very well turn it to an advantage in favour of the organization.

Inadequacy of the systems approach: One question that frequently arises while discussing failure of information systems from the complexity viewpoint is whether such failures can be analyzed within the framework of systems theory. The reason for this is not far to seek. Complexity theory shares a common vocabulary with systems theory. Many aspects of complexity also figure in the discussion of systems theory. In particular, terms like connectivity, interdependence, feedback and emergence carry almost the same meaning in both theories. However, complexity extends some of these concepts and also introduces new ones like co-evolution, exploration of the space of possibilities, self-organization etc. We have already seen that each of these new concepts plays a crucial role in enhancing our ideas about IS failures. System theory puts a lot of emphasis on prediction and control[45]. Within the framework of system dynamics, the objective of any analysis is to 'improve' the system. This implies that systems theory favors confirmatory analysis or problem-solving using models based on feedback. On the other hand, complexity with its focus on exploratory analysis is ideal for situations where non-linear effects prevail (as in the case of planning, development and management of information systems), where co-evolution is to be facilitated and emergence of appropriate ideas and work culture is to be encouraged. Complexity is not built on the assumption that one can proactively control what will happen. Rather, the focus is on broadly understanding and coping with the situation as it unfolds in real time.

Absence of control also gets manifested in the following way: in case of an entity moving across a landscape (e.g. an information system getting developed and implemented in an organization), complexity concerns itself both with the movement of the entity and the deformation of the landscape as the entity moves across it. However, systems theory explains this event by simply mapping causal factors that cause the movement of the system from a certain point to another point on the landscape. Systems approach is, therefore, somewhat restricted in nature.

Phelan[45] points out another interesting but subtle difference between complexity and systems approaches. Complexity theory is based on the premise that much of the observed complexity in any system arises from the relatively simple interactions among the agents constituting the system. But from the systems perspective, complexity depends on the number of system components and the amount of interaction between them. According to Phelan, in complexity theory, "there is a reluctance to embrace radical holism, that is, to maintain that the whole can only be understood in its totality and that all interactions are important. In general, systems theory is more sympathetic to radical holism". Let us recall the fact that the holistic approach to understanding IS failures embodied in the singular failure concept is being modified. The focus is now on trying to make smaller but specialized efforts directed towards risk management. In this context, obviously the complexity approach is a better strategy to employ compared to systems theory.

Case of the London Ambulance Service A Complexity perspective: Much has been written and discussed about the case of the London Ambulance Service. We turn to this oft-quoted example of information system failure in order to study how application of the principles of complexity might have helped avoid this major system disaster.

The London Ambulance Service served a population of 6.8 million people, carried over 5000 patients every day, and received up to 2500 calls per day. Its objective was to respond to calls in an average of 14 minutes. A previous system for dispatching ambulances in response to medical emergencies had split up London into three separate zones and used to communicate with the ambulances through a combination of two-way radio, telephones, and computer displays in vehicles. Operators in the dispatching centre received calls about emergencies, co-ordinated with the local ambulance stations to identify the nearest available ambulance and then send it to the place required. A new system was developed which wanted to treat the whole of London as a single zone. It discontinued the practice of radio and telephone calls to stations. Rather, the information system started an automated control of the dispatch of ambulance crews based on the location of the patient and of available ambulances. The new system was pressed into service on October 26, 1992 without completely testing or debugging it. However, on that very day, as the night progressed, calls were missed, several ambulances got dispatched to the same site, and operators in the dispatching centre were inundated with computerized exception messages. Some emergency callers failed to get connected for up to 30 minutes. Between 10 and 20 people probably died because ambulances were late by up to three hours. The situation was described, perhaps aptly, as "a complete nightmare".

A formal enquiry was carried out to ascertain the causes for this catastrophic failure. The general conclusion was that neither the computerized components of the system nor the human participants had been ready for its full implementation. Some of the more tangible factors were found to be the following:

- The software was neither complete nor fully tested
- The system's performance under full load had not been tested
- The dispatching staff and ambulance crew did not have enough confidence in the new information system and had not been properly trained in the use of the system
- The automated dispatching approach required virtually perfect information, but the information it received was imperfect due to incomplete status reporting from the ambulance crews, poor coverage in the radio system, technical inconsistencies between the mobile data terminals and the central computer. Imperfect data in the dispatching system caused inappropriate and duplicated allocations of ambulances to emergencies..

A few things are immediately evident from the above facts. Firstly, the incomplete software points to some kind of process failure while the very low confidence of the users and their lack of motivation about the system indicate interaction failure. Secondly, the fact that the software was pressed into operation without proper testing indicates desperation on part of the top management to thrust the system upon the users and thus act as a kind of "global controller". Complexity tells us that it was imperative to facilitate co-evolution between the new technology and the target user groups in order to secure the trust of the users about the use of the new technology. The dispatching staff and ambulance crew should have been given sufficient time to prepare themselves for use of the new automated system. In the terminology of complexity, enabling conditions should have been created to allow the emergence of a working style which would match the new automated system. Furthermore, there was clearly a great deal of system complexity present which hindered its smooth functioning. The principal reason behind this was the eagerness of the system developers to control all possible outcomes and not allow any self-organization in the system.

DISCUSSION

Some more observations relevant to both information system failures and complexity, which can be generalized from the LAS example, can now be mentioned :

(1) It is simplest to tackle problems relating to IS failures when the failures are independent. The easiest way to manage risk in case of independent failures is to create redundancy within the system. Another useful design feature to deal with local failures is to avoid having any one element of the system be essential to its overall performance. This may be achieved by making the system highly decentralized (without any global controller), like the Internet. In case of the London Ambulance Service disaster the idea of treating entire London as a single zone turned out to be catastrophic. The process of totally eliminating radio and telephone calls to stations and depending solely on computers to dispatch ambulance crews proved to be devastating. The system developers removed redundancy from the system entirely and in the process made the system susceptible to total failure. One of the major advantages of Complex Adaptive Systems over more rigidly centralized organizations is the resilience of the former in the face of local failures. System developers could have avoided the magnitude of the disaster by exploiting this property and designing the system along the lines of CAS.

(2) Design problems get complicated when local failures are not mutually independent. One example is the problem of correlated shocks. These failures occur when the elements of a system begin to break down at the same time due to the same reason. For example, in case of the London Ambulance Service disaster, the use of software not fully debugged led to large-scale breakdown in the operations of the system.The design against correlated failures involves identifying the sources of shocks that are external to the elements and could therefore cause failures in several elements at the same time. After identifying these correlated sources of error, one can achieve redundancy by building new elements into the system that are not susceptible to

these particular shocks. When the sources of external shocks cannot all be identified, the maxim is that the more diverse the elements, the less the chance that they will all be vulnerable to the same kind of external shock. Thus diversity in Complex Adaptive Systems not only allows exploration of new options but also provides safeguard against common shocks. Information systems that rely on widespread use of common hardware or software components can prove to be extremely fragile. We have already pointed out the inadequacy of searching for a single optimal solution, a practice very common among IS developers. A similar method was employed in case of designing the Computer Aided Dispatch system for the London Ambulance Service. The design did not incorporate enough variety or alternatives. Hence failure was quick and irreversible.

(3) A third type of failure possible in information systems is stress propagation failure, which occurs when the elements interact naturally, or are designed to interact. Here failure in one element can cause stress in another leading to the failure of that element as well and eventually to a cascade of failures. Such failures occur in closely coupled systems information systems being one of the examples. It is generally believed that close coupling would increase the efficiency of systems, but so would the chances of large scale system failures (Perrow, 1984). The design for the LAS was clearly a closely coupled one, had virtually no redundancy built into it, with a strong desire of the developers to micro-manage the system and virtually impose it upon the end-users. Developers should note these findings and try to avoid these mistakes so as not to end up with a disaster like the LAS computer aided dispatch system.

CONCLUSION

Information System failures continue to disturb organizations. Both academicians and practitioners have tried to analyze these failures from their respective viewpoints and tried to devise strategies for risk containment. This is a challenge which is extremely difficult; the journey will definitely be long and hazardous. In this paper, we have tried to analyze information system failures in the light of complexity theory. In particular, we have tried to harp on those features of complexity which are likely to improve the situation on the ground. Some of these features are facilitating the process of co-evolution, exploring the space of possibilities instead of focusing on a single "optimal" solution and following the so-called best practices rigidly, encouraging self-organization and emergent behaviour etc. We have also pointed out how this study is positioned in relation to the current literature on system failures by briefly describing the other frameworks to study IS failures which are currently in vogue. We also used the well-known example of London Ambulance Service to discuss how things went wrong with that system and alluded to the improvement which might have been effected by using ideas from the theory of complexity.

We believe that development and management of Information Systems in the complexity framework could help us obtain solutions that are systematic, stable and maximize stakeholder value. More investigation in this direction would be the subject of our future work.

REFERENCES

1. Bernstein, P. (1998) Against the Gods: The Remarkable Story of Risk, John Wiley, New York. ISBN 0471295639, 9780471295631
2. Beynon-Davies, P. (1995) Information Systems 'Failure': The Case of the London Ambulance Service's Computer Aided Dispatch Project, Eur J Inform Syst (4), 171-184. doi:10.1057/ejis.1995.20
3. Markoff, John. (November 24, 1994) Flaw Undermines Accuracy of Pentium Chips, New YorkTimes. http://query.nytimes.com/gst/fullpage.html?res=9C03E3D71630F937A15752C1A962958260
4. Keil, M., Mann, J., and Rai, A. (2000) Why Software Projects Escalate: An Empirical Analysis and Test of Four Theoretical Models. MIS Quart 24 (4), 631-664. Published by – Management Information Systems Research Centre, University of Minnesota. http://www.misq.org/archivist/vol/no24/issue4/keil.html
5. Wood field, R. (2002) Women and Information systems Development: Not Just a Pretty (Inter) face? Information Technology and People, 15(2),119-138. doi : 10.1108/09593840210430561
6. Friedman, A. L., and Cornford, D. S. (1989) Computer Systems Development: History, Organization and Implementation, John Wiley and Sons, Chichester, UK. ISBN:0471923990. http://portal.acm.org/citation.cfm?id=583461
7. Sauer, C., Southon, G., and Dampney, C. N. G. (1997) Fit, Failure and the House of Horrors: Toward a Configurational Theory of IS Project Failure, in Proceedings of the Eighteenth International Conference on Information Systems (K. Kumar and J. I. DeGross Eds.), 349-366, December 14-17, Atlanta, GA, USA. Published by – Association of Information Systems. ISBN:ICIS1997-X. http://portal.acm.org/citation.cfm?id=353141

8. Sauer, C. (1993) Why Information Systems Fail: A Case Study Approach,: Alfred Waller Ltd,Henley-on-Thames,UK; ISBN:1-872474-08-X
9. Lyytinen, K., and Hirschheim, R. (1987) Information Systems Failures: A Survey and Classification of the Empirical Literature, Oxford Surv Inf Tech (4), 257-309; ISBN:0-19-859019-9; http://portal.acm.org/citation.cfm?id=54890.54898
10. Robey, D and Boudreau, M. (1999) Accounting for the contradictory organizational consequences of information technology: Theoretical directions and methodological implications, Inform Syst Res 10(2):167-185. http://portal.acm.org/citation.cfm?id=767721.768526&dl=&dl=GUIDE
11. Clemons, E. K., Thatcher, M. E., and Row, M. (1995) Identifying Sources of Reengineering Failures: A Study of the Behavioral Factors Contributing to Reengineering Risks, J Manage Inform Syst, 12(2),9-36. http://www.jmis-web.org/articles/v12_n2_p9/index.html
12. Earl, M.J. (1996) The Risks of Outsourcing IT, Sloan Management Review, Spring 1996:37(3),26-32. http://sloanreview.mit.edu/smr/issue/1996/spring/2; www.mona.uwi.edu/msb/Student%20Material/download_material.asp?name=The%20risk%20of%20IT%20outsourcing.pdf
13. Orlikowski, W. J., and Baroudi, J. J. (1991) Studying Information Technology in Organizations: Research Approaches and Assumptions, Inform Syst Res 2(1), 1-28. www.mona.uwi.edu/dmcs/staff/gmansingh/Research%20Methods/.../Orlikowski-ISR-2-1.pdf
14. Lee, A. S., Liebenau, J., and DeGross, J. I. (eds.) (1997) Information Systems and Qualitative Research,: Proceedings of the IFIP TC8 WG 8.2 International Conference on Information Systems and Qualitative Research, 31st May-3rd June 1997, Philadelphia, Pennsylvania, USA; Springer 1997; ISBN 0412823608, 9780412823602.
15. Myers, M.D. (1997) Qualitative Research in Information Systems. MIS Quart, 21 (2), 241-242; http://www.misq.org/discovery/MISQD_isworld/
16. Brey, P. (1997) Philosophy of Technology meets Social Constructivism. Society of Philosophy & Technology (2), 3-4. http://scholar.lib.vt.edu/ejournals/SPT/v2_n3n4html/brey.html
17. Gell-Mann Murray, (1994) The Quark and the Jaguar: Adventures in the Simple and the Complex; pp 16-21, Freeman, New York. ISBN:0-7167-2581-9; http://www.santafe.edu/~mgm/mgmquark.html
18. Lewin, R. (2000) Complexity: Life at the Edge of Chaos; pp 10-21, Ist Edition, University of Chicago Press; ISBN-10: 0226476553; ISBN-13: 978-0226476551
19. Gell-Mann Murray. (1995) What is Complexity?Complexity(1),16-19. http:/www.santafe.edu/~mgm/complexity.html
20. Covney, P. and Highfield, R (1995) Frontiers of Complexity; 1st Edition,pp 5-42, Faber and Faber Ltd. London. ISBN-13: 9780571179220 ISBN: 0571179223
21. Anderson, P., Meyer, A., Eisenhard, K., Carley, K., and Pettigrew, A. (1999) Introduction to the Special Issue: Application of Complexity Theory to Organization Science, Organ Sci 10 (3), 233-236. ISSN : 1526-5455; http://portal.acm.org/citation.cfm?id=767727.768578&dl=GUIDE&dl=GUIDE&CFID=11431413&CFTOKEN=78507242
22. Arthur, B. (1995) Complexity in Economics and Financial Markets, Complexity (1), 20-25. www.santafe.edu/arthur/Papers/Pdf_files/Complexity_Jnl.pdf
23. Kelly, E.M. (2003) “Ten Principles of Complexity & Enabling Infrastructures" in Complex Systems & Evolutionary Perspectives of Organizations: The Application of Complexity Theory to Organizations, Elsevier, Chapter 2, 23-50; Ed. Kelly, E.M. Http://www.psych.lse.ac.uk/complexity/Papers/Ch.2final.pdf
24. Mitev, Nathalie N. (2000) Towards Social Constructivist Understandings of IS Success and Failure: Introducing a New Computerised Reservation System, in Proceedings of 21st International Conference on Information Systems. Brisbane, Australia, Dec 10-13, 2000. 84-93; Edited by Orlikowski, W. J.; Weill, P.; Ang, S.; Krcmar, H. C.; DeGross, J. I. Publisher: Association for Information Systems.ISBN:ICIS2000-X. http://portal.acm.org/citation.cfm?id=359640.359725&coll=GUIDE&dl=GUIDE&CFID=11431413&CFTOKEN=78507242
25. Rockart, J. F. (1979) Chief Executives Define Their Own Data Needs, Harvard Bus Rev 57 (2), 81-93. doi : 10.1225/79209
26. Shank, M. E., Boynton, A. C., and Zmud, R. W. (1990) Critical Success Factor Analysis as a Methodology for MIS Planning. Management information systems: readings and cases: a managerial perspective, pp 397-407;ISBN:0-673-38859-X; http://portal.acm.org/citation.cfm?id=103411
27. Boynton, A. C., and Zmud, R. W. (1985) Critical Success Factors: A Case- Based Assessment, Sloan Manage Rev 25 (4). 17-27. portal.acm.org/citation.cfm?id=1189395.1189406

28. Flowers, S. (1996) Software Failures—Management Failures: Amazing Stories and Cautionary Tales, J. Wiley and Sons, Chichester, UK. ISBN-10: 0471951137
ISBN-13: 978-0471951131
29. DeLone, W. H., and McLean, E. R. (1992) Information Systems Success: the Quest For the Dependant Variable, Inform Syst Res (3), 60-95. http://nubs.nju.cn/download.php?.../zcd/Information%20system%20success%20the%20quest%20of%20the%20dependent%20va
30. Walsham, G. (1995) The Emergence of Interpretivism in IS Research, Inform Syst Res 6(4),December,376-394.
http://gkmc.utah.edu/7910F/papers/ISR%20emergence%20of%20interpretivism%20in%..
31. Vaughan, D. (1996) The Challenger Launch Decision: Risky Technology, Culture and Deviance at NASA, University of Chicago Press, Chicago, pp xi-xv, 1-32. ISBN 0226851753, 9780226851754
32. Harvey, L. J., and Myers, M. D (1995). Scholarship and Practice: The Contribution of Ethnographic Research Methods to Bridging the Gap, Information Technology and People, 8(3),13-27. doi : 10.1108/09593849510098244
33. Perrow, C. (1999) Normal Accidents: Living with High-Risk Technology, Princeton University Press, ISBN-10: 0691004129; ISBN-13: 978-0691004129.
34. Kling, R. (1980) Social Analyses of Computing, Theoretical Perspectives in Recent Empirical Research, Computer Survey 12 (1), 61-110..ISSN:0360-0300.
http://doi.acm.org/10.1145/356802.356806
35. Markus, M. L. (1983) Power, Politics and MIS Implementation, Communications of the ACM 26 (6), 430-444. ISSN:0001-0782. http://doi.acm.org/10.1145/358141.358148
36. Warne, L., and Hart, D. (1996) The Impact of Organizational Politics on Information Systems Project Failure- A Case Study, Proceedings of the 29th Hawaii International Conference on System Sciences (HICSS), Jan 03-06, 1996 Hawaii, USA; Vol 4:Organizational Systems and Technology, 191-201.Publisher – IEEE Computer Society, Washington DC, USA. ISBN:0-8186-7333-8; http://portal.acm.org/citation.cfm?id=798582
37. Mathews, K., White, C., and Long, R. (1999). Why Study the Complexity Sciences in the Social Sciences, Hum Relat 52 (4), 439-462. doi: 10.1023/A:1016957424329
38. Pribram, K. H. (1996) Interfacing Complexity At A Boundary Between The Natural And Social Sciences, in Evolution, Order, and Complexity (Khalil, E. L., and Boulding, K. E. Eds), Routledge, London, pp 40-60. ISBN 0415137284, 9780415137287
39. Axelrod, R. M., and Cohen, M. D. (2000) Harnessing Complexity: Organizational Implications of a Scientific Frontier. Free Press, New York, 1-31. ISBN-10: 0684867176; ISBN-13: 978-0684867175
40. Maxfield, R and Lane, D (1996)– Strategy under Complexity: Fostering Generative Relationships – Long Range Plann, Vol 29, 215-231. doi:10.1016/0024-6301(96)00011-8
41. McKelvey, B. (1999) 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. http://www.rotman.utoronto.ca/~baum/variations_toc.html
42. Avgerou, C. and Cornford, T. (1998) Developing Information Systems: Concepts, Issues and Practice, (Second Edition). MacMillan,London. ISBN-10: 0333732316
ISBN-13: 978-0333732311
43. Alter, S. (2004) Information Systems; The Foundation of E-Business.4th Edn., Pearson Education. ISBN-10: 0130617733
ISBN-13: 978-0130617736
44. Bar-Yam Yaneer (1997). Dynamics of Complex Systems. The Advanced Book Program. Addison-Wesley; Reading, Massachusetts, pp 1-15. ISBN 0-201-55748-7.
http://www.necsi.org/publications/dcs/
45. Phelan, Steven E.(1999) A Note on the Correspondence between Complexity and Systems Theory. System Pract Act Res, 12 (3),237-246. www.springerlink.com/index/R5362G375621H218.pdf