1. Introduction: integration– complexity–risk – the making of information systems out-of-control

Ole Hanseth

1 INTEGRATION, COMPLEXITY AND RISK

ICT has been closely related to risk throughout its history. On the one hand, ICT is essentially a control technology (Beniger, 1986). It is a technology that helps us better predict and control complex processes in nature (develop new drugs, control nuclear power plants, produce weather forecasts, and so on), society (production control systems, project management tools, and so on), and other technologies (intercontinental ballistic missiles, chemical plants, and so on). This also includes various risk prediction, calculation and management tools. But we have also experienced that ICT solutions have their own risks – risks related to ICT projects as well as systems failures. And a whole series of techniques and tools have been developed to help project managers control risks related to the development of ICT solutions as well as making them reliable by minimizing risks for systems failures.

In spite of all the research on ICT risks and the increased sophistication of the tools and techniques developed, ICT risks still prevail. In fact, there are indications that they are increasing rather than diminishing. This apparent paradox is what has motivated this book. Our interest in inquiring into it stems from two sources. The first is an observed growth in ICT complexity which increases and creates new challenges regarding management and control of the development and use of ICT solutions. The second is theoretical developments in other areas regarding risk. The most important source here is Ulrich Beck's (1986) theory of Risk Society which, together with Anthony Giddens, he has later elaborated further into a more general theory of globalization and modernity called Reflexive Modernization (Beck et al., 1994). Through the formulation of this theory Beck and Giddens argue that risk has become a more dominant and characteristic feature of our society at the same time as the nature of risk has

changed. While nature was the dominant source of 'old' risks (like natural catastrophes), 'new' risks have their origin more in our 'created environment', that is our technological and organizational systems. This theoretical development has inspired research on risk-related issues in many disciplines. Within science and technology studies, for instance, it has inspired a great deal of research on science, technology and expertise in relation to various environmental problems like risks related to new bio technologies (see for instance Lash et al., 1996). It has also more recently been adopted by scholars within organization studies (Hutter and Power, 2005) and information systems (Ciborra et al., 2000).

We see these two perspectives as closely related. At the centre of Beck's argument is the assumption that our society is becoming more complex through the development and use of more complex technologies and organizational forms that are both essential to ongoing globalization processes. This increasingly complex world is, still according to Beck (and Giddens), becoming more unpredictable and accordingly increasingly out of control and unmanageable. ICT is increasingly becoming a central part of virtually all technologies. At the same time, ICT in general, and the Internet and applications built on top of it in particular, are central to the ongoing globalization. Consequently, ICT should be central to the creation of the Risk Society. So by adopting Beck's perspective, we might say that ICT has (possibly) changed from a control to a risk technology, that is a technology that generates risks more than it helps us control and manage them.

1.1 Growing ICT Complexity

The British Computer Society (BCS) and Royal Academy of Engineering (RAE) published in April 2004 a report about the state of the art in Software Engineering (SE) and Information Systems Design (ISD) research and practice in the UK, which was called 'The challenge of complex IT projects' (BSC/RAE, 2004). The report, which was the result of a project involving 70 leading UK SE and ISD professors and IT managers, concluded that there was an alarming failure rate in UK IT projects. This failure rate could have been significantly lower, the report concluded, if best practices were adopted. In particular the use of risk management methodologies was lacking. But beyond this, the report also concluded that the high failure rate was a result of the fact that the complexity of ICT solutions has grown rapidly and that existing SE/ISD methodologies do not tackle this adequately. The report further concludes that existing methodologies were developed at a time when IT complexity was at a much lower level, that these methodologies have not scaled regarding complexity (see, for example, Fitzgerald, 2000), and, finally, that new methodologies

addressing the growth in complexity have not been developed. From this they draw the conclusion that IT complexity is *the* issue on which basic research is needed. We need then, according to the report, first, to understand this complexity; then new risk management methodologies that can enable us to control this complexity need to be developed.

It is exactly this risk and complexity issue that is addressed by this book. We share the report's views regarding complexity and risk – except, perhaps, for the optimism regarding the possibilities of controlling the risks that are consequences of this complexity. Our primary aim is to get a better understanding of this new complexity: how it emerges, how it evolves, and, finally, what kind of risks it generates. So far our understanding of this complexity makes us rather pessimistic about possibilities for controlling it.

1.2 Complexity and Integration

The BCS/RAE report sees the growth in complexity as a consequence of the growth in computing power and the development of communication technologies. The growth in computing power has enabled our computers to run more complex software; accordingly we have developed such software. The development of communication technologies has enabled us to integrate the complex pieces of software across computers.

We also consider these two factors as being important causes of the growing complexity. But our view on the sources of growing ICT complexity is slightly different. We see integration as the key source. And we do not look at ICT as purified technology. ICT complexity has grown as the number of components and their integration has increased. But just as important, complexity has also increased because of increased integration between ICT components and the organizations and practices using the technology. We will spell out this argument in a bit more detail.

As the computing power has increased, the number of applications in use in any organization has been growing. These applications, or information systems (ISs), support an increasing number of application areas. This implies that the number of types of ISs and use areas has grown as well. In parallel the platforms or infrastructures these applications are running on have become more sophisticated; the number of components as well as types of components included has grown. All these components are linked in various ways. Applications have become increasingly more integrated with each other. The same has happened within the platforms. And the evolution of computer communication technologies has enabled the development of various kinds of new applications, but first of all they have led to an integration of platforms and applications across organizational and geographical borders. This, again, has led to integration (that is, an

increasing number of connections) between organizations and their routines, and between their ISs. Over time the use of individual ISs grows (partly directly as its number of users grows, partly indirectly through integration with other ISs). This means that the number of links, and types of links, between the technical systems and the social/organizational also grow. And when ISs are integrated across organizational borders, links are established between the routines and practices in one organization and the ISs and platforms in another. And finally, according to assumptions apparently shared by everyone, everything – in particular IT and business organizations – is changing at an increasingly high speed. In sum, this means that it should be quite obvious that the complexity of IS development and use has grown substantially.

1.3 Complexity

Complexity could be defined in a simple and intuitive way as the ‘sum’ of the number of components and connections between them. Schneberger and McLean (2003) have proposed a slightly more sophisticated – but still very simple – definition. They see complexity as dependent on a system’s number of different *types of components*, its number of *types of links*, and its *speed of change*. While the first definition seems to be close to that adopted by the authors of the BCS/RAE report, we have adopted the latter, which accounts well for the complexity created by increased integration as outlined above. The complexity referred to by the BCS report is not really complex according to this definition because the report only talks about systems of technological components – that is systems where the components and the links between them are all technological. We might say that such systems are just complicated (Latour, 1999, p. 304). One might say, though, that the ICT systems as pure technology contain components of many technological kinds such as data base management systems components, security components and communication components. However, the second definition says that ‘real’ complexity emerges when components of different kinds are integrated. This is the kind of complexity that emerges when platforms, applications and organizational routines, practices and structures are integrated. This is also illustrated by John Law and Annemarie Mol’s (2002, p. 1) (somewhat obscure, perhaps) definition of complexity: ‘There is complexity if things relate but don’t add up, if events occur but not within the process of linear time, and if phenomena share a space but cannot be mapped in terms of a single set of three-dimensional coordinates.’ We interpret this definition as underscoring the same point. A system is not complex if it consists of (or can be reduced to) components of one kind – for instance technological. A system is really complex only

when it contains components of different kinds – for instance technical and organizational – and when the problems or issues related to one cannot be reduced to the other, that is we cannot understand the problem by addressing only technological or organizational (or social) issues, we need to understand both and their interactions and interdependencies.

1.4 Complexity and Risk

While the first part of our argument is that increased integration leads to increased complexity, the second is that increased complexity leads to increased risks. The fact that increased complexity leads to increased risks may be intuitive. But from the perspective of integration efforts it is paradoxical, since the motivation behind integration efforts is primarily to increase control (Ciborra et al., 2000). In spite of this, the result of the integration efforts is increased risks, that is less control. The basic argument for this goes as follows: when we are confronted with a complex system, our knowledge and understanding of how different components work and interact, and accordingly how the system as a whole works, will always be incomplete. The components may act and interact in ways we cannot fully predict. When one component acts, or interacts with another component, in an unpredicted way, this may produce outcomes that make other components act or interact in unpredicted ways, and so on. Such unpredictable behaviour may cause the complex system as a whole to sometimes behave in totally unpredictable ways.

The fact that we have only partial knowledge about complex systems has, of course, implications for the consequences of human interventions into complex systems. When we have only partial knowledge about a system, the outcome of our interventions will be partly unpredictable. This brings the concept of *side-effects* to the centre stage. When we change a component about which we have only partial knowledge, after our intervention the component will behave in ways we did not predict or intend. A side-effect of our intervention will be some unpredicted behaviour that will interfere with other components in unpredictable ways, which again will produce new unintended effects, and so on.

The more complex a system is, the more incomplete our knowledge will be, and the more unintended effects our interventions will produce. We can say that the more complex a system is, the more its overall behaviours will be caused by propagation of side-effects rather than intended effects. All this means that the more complex a system is, the more unpredictable the outcome of our interventions – that is the higher risks for negative outcomes of interventions. So: through integration we increase complexity which again leads to increased risks.

Integration is the ‘holy grail of MIS’ (Kumar and van Hillegersberg, 2000). That means that the way in which integration produces risks is a very fundamental aspect of information technologies. Axelrod and Cohen (1999) make a very similar point:

If complexity is often rooted in patterns of interaction among agents, then we might expect systems to exhibit increasingly complex dynamics when changes occur that *intensify* interaction among their elements. This, of course, is exactly what the Information Revolution is doing: reducing the barriers to interaction among processes that were previously isolated from each other in time and space . . . Information Revolution is therefore likely to beget a complexity revolution. . . . It is ironic that exploiting the promise of short-run possibilities for better prediction and control can create longer-run difficulties (ibid., p. 26).

2 OUTLINE OF THE BOOK

The argument, or rather, the hypothesis, presented above will be explored in the nine chapters following this. Chapters 1 to 3 form a theoretical section which is followed by an empirical section that includes five chapters – one case study presented in each.

2.1 Theory Section

In the first chapter of the theory section, and the second in the book, Claudio Ciborra reviews the research literature on risks related to ICT and locates this research within the broader field of risk management.

With time and experience, managerial attention to ICT applications such as the study of organizational impacts, management of ICT strategy and redesign of business processes has crossed the boundaries of quantitative risk management and widened its scope. As a result, an organizational/managerial literature about information systems risks and their management in a business-wide perspective has emerged. This ranges from the more micro concerns about how to get the user requirements right (an issue that still overlaps with SE risk management) up to the strategic choice in selecting an application portfolio or an ICT infrastructure, where questions need to be asked about the risks of large ICT investments for a business as a whole.

Ciborra argues that the cultural and professional milieu in which these approaches (or methods and techniques) had been originally tried out led to their emergence as strictly quantitative approaches based on a positivist, probabilistic definition of risk: probability of occurrence of a problem multiplied by the value of its impact. He further argues that this naïve view of

risk has dominated the SE discipline and practice since the 1980s. It is supported by a myriad of sub-techniques for identifying risks, measuring impacts and assessing probabilities. To be sure, the methods and techniques are accompanied by words of caution from senior software professionals who suggest that they should be applied with a grain of salt and situational common sense. Here, as often happens in the professional field, formal models seem to get gradually substituted by rules of thumb, prescriptions and war stories disguised as articulated experience.

Ciborra also points to the limitations of existing risk management approaches regarding complexity as it is presented in the theories the research reported in this book draws upon – and as it emerges in the cases analysed. Central to these complex systems is the fact that we have only partial knowledge about how they work, accordingly which risks we may encounter is largely unknown, so we cannot calculate their probability nor imagine their impact. These issues are discussed more extensively in the next chapter where Jannis Kallinikos explores theoretically the relationships between risk and ICT. His focus is on recent developments in the study of risk in relation to the ‘essence’ of technology in general and large-scale ICT solutions in particular. Traditionally risk has been the concern of specific risk and risk management theories or methodologies (which Ciborra presented and discussed). Kallinikos says that the concept of risk has experienced a remarkable renaissance in the last two decades; now the theories are not primarily about risk per se, but as a central concept in the wider field of social theory. Relevant work includes the works of Beck and Giddens already mentioned, but also Mary Douglas’s cultural theory of risk (Douglas and Wildavsky, 1980) and Nicolas Luhmann’s (1993) monumental theory. The latter is central to the perspective on risk and complexity as well as technology. On this basis Kallinikos points to three key issues. The first is our relation to our world and the paradoxical fact that the more (scientific) knowledge we have about the world we inhabit and the more advanced technology we develop to help us control our world, the more it seems to move beyond our control, that is the more risks. The best illustrations of this are global warming and the possible implications of climate change, or even worse – the risks for nuclear war that may destroy the whole planet. Kallinikos attributes this paradox to the fact that however much knowledge we produce, our knowledge about our world will be incomplete. In addition, our world and the knowledge about it are not independent. When we are producing more knowledge about our world, we are at the same time constructing a more complex world.

The second issue that Kallinikos discusses relates to time and what he calls our ‘future orientation’. At the core of this argument is the fact that our world is inherently contingent and unpredictable. We cannot escape

this fact in spite of all advance calculative devices that are supposed to help us predict our future – or at least calculate probabilities of various scenarios. In addition, the contingent nature of our future is a precondition for many activities. The world is constituted as open and plural, amenable to initiatives and interventions that could possibly render it different from the present. Without risk there would be no opportunities and therefore no profit, and without profit, no economic initiatives either.

The third issue that Kallinikos discusses is the relation between technology and risk. Technology is traditionally seen as a tool for control and accordingly risk management or reduction. But there are important limitations to how technology can address the issues mentioned here. Drawing on Luhmann, Kallinikos portrays technology as a structural form whose main characteristics are functional closure and simplification. That means that we can deal with reality by simplifying this into a closed domain and specify how the technology can deal with each element in this domain and its states. Technology, then, has important limitations when it comes to uncertainty – in Kallinikos' words: 'technology deals with the unexpected by excluding it'.

After this general discussion of risk and technology Kallinikos turns towards what is specific for large-scale information systems highlighting three aspects: self-referentiality, interconnectivity and de-sequencing of time. Self-referentiality denotes the fact that information is not just data referring to some reality. The more information we gather and process, the more we will also produce information referring to information. This may bring us towards a sort of autonomous complex system of information and a 'de-anchorage from the immediate reality'. The other two aspects of large-scale information systems relate more directly to the issues regarding integration, complexity and risk discussed above. Through increased interconnection of information systems we create more complex systems which are more challenging to control. And such increased interconnection of systems contributes to what Kallinikos denotes de-sequencing of time, that is processes will increasingly be linked, not only sequentially, but also running in parallel. This kind of link between parallel processes creates what Perrow (1984) defines as interactive complexity (as opposed to linear interactions) and which he sees as a main source of unpredictability and unreliability of technological systems.

The last theory chapter, written by Ole Hanseth (Chapter 4), presents the theories that have inspired the research reported in this book and which form the basis of the hypothesis presented above. This is first of all Ulrich Beck's theory of Risk Society/Reflexive Modernization (Beck, 1986; Beck et al., 1994). Beck's general argument is that the world is increasingly dominated by risks so that it can best be characterized as a Risk Society. Beck

does not discuss technology as such – he hardly mentions the term, but implicitly technology is central to his theory. His theory is a social theory, but it is a social theory of our technological society. Central aspects of the theory are risks created by advanced technological systems like pollution and global warming, risks related to nuclear weapons and power plants, bio-technology, and how modern transport, communication and production technologies are rapidly transforming working as well as family life.

Complexity is hardly mentioned in Beck's writings. However, his theory can very well be seen as a complexity theory because what is making risks more central in our world is exactly its growing complexity – a complexity which is created through scientific research and the development of increasingly sophisticated technological solutions. And the core dynamic of this complex world is the propagation of side-effects. Beck highlights one particular pattern of how side-effects often propagate. They are not only creating a chain of side-effects (domino effects), but increasingly often they form boomerang effects – they are reflected back to where the original act triggering the whole changing was taking place and causing the result to be more or less the opposite of what the actor intended. This is what Beck calls reflexivity. And it is exactly the extent to which such reflexive, that is self-destructive, processes are created through the integration of ICT solutions that is at the very centre of this book.

In Chapter 4 Beck's theory is supplemented with and compared to a few others. This includes theories from the Complexity Sciences. Side-effects are also at the centre of these theories. However, their originators and proponents have focused on phenomena different from those of Beck – in particular within the natural sciences and economics (standardization), and they have highlighted other patterns created through the propagation of side-effects. Rather than self-destructive processes, they have focused on self-reinforcing ones. However, the reflexive processes that will be presented in this book are at the same time self-reinforcing. This happens because the processes aiming at controlling complexity achieve the opposite because control is tried to be achieved in ways that increases complexity. Complexity bootstraps and becomes self-reinforcing.

This chapter also presents a third theory of complexity which the research draws upon: Actor-Network Theory (ANT). ANT has traditionally not been labelled a complexity theory, but one might very well say that complexity has been the issue addressed in ANT research. Not complexity in all its varieties, but the complexity created by the dense web of relations between the scientific and the non-scientific (that is social, organizational, political, and so on) within science studies and similarly between the technological and non-technological in technology studies. In line with this we draw upon ANT to disclose and analyse one aspect of the

definition of complexity presented above: how elements of different kinds are related.

More recently, complexity has been addressed explicitly within ANT (Law and Mol, 2002). One issue that is highlighted is what is sometimes labelled the ‘orders’ dis-orders’, which means that when we try to make order within a domain or ‘world’ (by creating a new standard, for instance), unfortunately the domain or ‘world’ we try to order is connected to other domains or ‘worlds’ beyond what we are able to see and cope with. Accordingly, when we are trying to make order in one domain, as a *side-effect* we are creating dis-orders in others (Berg and Timmermanns, 2000).

Chapter 4 also presents the most relevant research on complex technologies from the Organization Studies field. This includes first of all Charles Perrow’s Normal Accident Theory (1984). He argues that complex technological systems that are tightly coupled and characterized by interactive complexity (as opposed to sequential interactions) will never be completely safe – accidents are normal. This is because we will never have complete knowledge about how they will operate in all circumstances. Accordingly we cannot make sure they will be free of errors or unpredictable behaviours. Risks are inherent to their existence. In this way Perrow’s theory is well aligned with the theories presented above and supports the hypothesis we will explore in our research.

However, some scholars are sceptical to Perrow’s theory. This includes in particular Todd La Porte who was the original creator of the theory of High Reliability Organizations (HRO) (La Porte, 1988; 1996; Weick and Sutcliffe, 2001). HRO proponents have also studied complex technological systems finding that there are some that prove to be characterized by very high reliability. The typical example is air traffic control. The main aspects of HROs are their focus on risks and learning from experience. Based on the theory of HROs we could extend the research hypothesis presented above with its complementary opposite by asking if or under what conditions the development and use of the emerging complex ICT solutions could be established as HROs. We have not explored this hypothesis explicitly, but it will be briefly discussed below in light of our empirical material.

2.2 Cases

The five cases will present and discuss a number and variety of complex ICT solutions. They will illustrate

- how complexity is the result of various forms of *integration* efforts,
- what kind of *side-effects* these kinds of integration trigger, and
- what kind of *risks* they create.

The three first cases present project risks (that is risks for project failure), the fourth systems risks (that is risks caused by a system in operation), while the fifth case illustrates a case where loose integration is dominating and where the ICT solutions have been managed rather successfully. Two of the cases illustrate large-scale, ambitious integration efforts where a huge number of tasks within a large organization are supported by one complex application where a huge number of software components are integrated around a shared data base. Some of the software components are providing general and basic functions and are used by most members of the organization while others are supporting highly specialized functions and are used by small user groups. This integration strategy has become very popular during the last decade, and its typical manifestation is the implementation of ERP systems like SAP R/3. The two first cases presented here are not about organizational implementation of a commercial product like an ERP system. The first one, presented by Knut Rolland and Eric Monteiro, is about in-house development of a system by and for a 'global' ship classification company with offices in more than a hundred countries. We follow the system from its conception until it is implemented in most of its offices and even integrated with customers' systems. This is a process where more and more computer (software and hardware) components are integrated with each other, where these components are integrated with organizational routines and practices, and where these routines and practices are integrated with each other. The main elements of the process are

- the implementation of the first version of the system in a few pilot offices and the integration of the new system with the existing data base and legacy system;
- the implementation of the system throughout the organization and the integration with the underlying infrastructures; and, finally,
- the integration of the system with customers' systems and work practices.

Each step in this process triggers major unforeseen side-effects. Many of them are serious and challenging. But, overall, the project has been successful – although its cost and duration were far beyond what was planned and the benefits less significant.

The authors discuss if the side-effects and the risks they represent could be managed by traditional approaches. They conclude that that is not the case. The risks encountered emerge as the scope of the integrated system is expanded, and the risks are unique for each level to which the scope is expanded. So the risks could not be identified and managed by traditional techniques like prototyping. Also, since the risks are different at each level,

when encountered at one level, they can neither be predicted based on experience at previous levels, nor be managed based on such experience.

The second case, presented by Ole Hanseth, Edoardo Jacucci, Miria Grisot and Margunn Aanestad, concerns the development and organizational implementation of an Electronic Patient Record (EPR) system. The project started out as collaboration between the five largest and most advanced hospitals in Norway and the Norwegian branch, here called Alpha Norway, of a big international company. The strategy and aim of this project were both very much the same as for the ship classification support system, the difference being application area (hospitals), the role of the software development organization, and the fact that the aim was the development of a system satisfying the need of all Norwegian hospitals.

The overall user requirements for the system were that it should make real the idea of one integrated record for all patients that was shared by all medical personnel in contact with them. This implied that the EPR had to include all information in the paper record and support all work tasks where the paper record was used. This means that the EPR system would be integrated – in one way or another – with all work practices where the patient record is involved, and, implicitly, these work practices would be integrated with each other through the EPR system. In addition, because the system was to be used by five hospitals, the system would also indirectly integrate the work practices of all these hospitals.

Central to the evolution of this project was its relation to other products, activities and strategies inside the software product owner Alpha. A side-effect of choosing a big international company like Alpha as development organization was that the project also became integrated with this organization. This fact first revealed itself when some members of the project team at one of the hospitals were informed at a conference that Alpha was involved in a similar project in the UK. They concluded that sooner or later higher-level management would discover this and that they would decide that Alpha should not develop several systems. Further, they assumed that Alpha would consider Norway the least significant market and close down the Norwegian project. Accordingly, they decided to approach Alpha and propose collaboration between these projects, assuming that if they took the initiative they would be more influential in the development of such a collaboration than if the initiative came from Alpha's headquarters. And so it happened. But that also meant that more user and development organizations became involved and that the system had to support a wider range of work practices. This was, in a way, the starting point of an escalating, self-reinforcing spiral. The increased complexity of the version of the system the hospitals and Alpha now aimed at required more resources, which again implied that they had to look for a larger market to make the

investments profitable. A larger market implied involving more hospitals with their own work practices and requirements, which again implied a more complex system and project organization – and more investments, which again required a larger market, and so on. Important steps in this process were Alpha's acquisitions of software companies with competence and products for the health care sector in India and the US.

As the project escalated, so did its strategic status inside Alpha. This meant that it became a more and more central product within its medical division (which included a wide range of instruments like X-ray and other imaging equipment, all with their complex systems to store and manipulate images and other patient-related information), and accordingly it also became more and more important to align and integrate the EPR system with the other products and strategies.

The general pattern of this project was: at each stage, the complexity of the system and the project triggered risk that threatened the whole project. The risks were interpreted as threats caused by lack of alignment with parts of the outside world. Accordingly, the strategy chosen to cope with the risk was to align, that is integrate, the system and project with the unaligned part of the outside world creating the threat. This again increased the overall complexity, which again created new risks, and so on. After eight years, at Rikshospitalet in Oslo, the EPR included only 20–30 per cent of the information in the patient records, the volume of paper had increased dramatically, the portfolio of information systems with patient record functionality had increased dramatically, and so also had the overall fragmentation of the patient records. The aims and strategies concerning integration had produced a more and more complex socio-technical system where the side-effects were taking over the project. In the end, these side-effects caused the end result to become the exact opposite of what had been intended. The risks identified, largely caused by complexity, were managed in a way that created increased complexity and accordingly more serious risks.

The next two cases focus less on the integration process creating a complex system, but rather on aspects of complex systems in operation. The third case, presented by Jennifer Blechar and Ole Hanseth, is a project aiming at replacing the billing system of a telecom operator with a new one. The existing system had been developed in-house. It had been extended and modified for about two decades – since the company started its mobile phone business. The plan was to replace it with a commercial off-the-shelf product as part of the launch of 3G services. This was initially assumed to be a plain and simple process requiring just a couple of months. But as the members of the project that was set up started to work on this task, a more and more complex system was revealed. They discovered that the billing

system was very complex in itself. But more important, over the years it had been tightly integrated with an increasing number of components including mobile phone infrastructures (NMT400, NMT 800, GSM, GPRS), a huge number of support or front-end systems (for instance so that subscribers could change their subscription over the Internet), the variety of services and 'call plans' had been growing, and so on. The overall complexity also reflected the rapid speed of change in the mobile phone business in general and within this company in particular. One important implication of this is the fact that new, that is, young and well educated but inexperienced, people were hired all the time while the experienced ones moved upwards into managerial positions or abroad as the company engaged in start-up operations in other countries. All this means that the complexity was determined by the number, and number of types, of components and links between them, but probably first of all by the lack of knowledge about it in the organization due to the rapid growth and change of the systems and the personnel moving around in the organization.

As the complexity revealed itself throughout the project, the actions taken to deal with it in this case also contributed to the increase in the overall complexity, that is the project created new risks and challenges just as much as it helped cope with the ones discovered already. In the beginning more consultants were brought in. This created, first, communication problems, which again triggered the establishment of a more complex project organization. Then, as new complexities revealed themselves, more consultants were brought in. They came from different consultancies to avoid too strong a dependence on just one. This again implied several different methodologies and documentation practices – again: increased communication problems and overall complexities.

Project members became well aware of the presence of risks from an early stage, and risk management methodologies were adopted – but with limited success. There were just too many risks involved. Rather than being resolved, the risks were just shuffled around or transformed into new ones.

The fourth case, presented by Daniel Osei-Joehene and Claudio Ciborra, illustrates risks of integrated systems in relation to the operations they support. The system involved is an integrated email infrastructure in a 'global' bank. This bank had over the years expanded geographically from its home base in Canada to the US and later on to Europe and South-East Asia. Its product portfolio had also expanded from retail banking to virtually any kind of financial services. The overall pattern of this expansion process had been one where new units established in new markets were first experimenting and exploring possibilities, then, as they were growing, their operations were streamlining and more tightly integrating into the rest of the organization. The integration of the email infrastructure took place in

a period with focus on streamlining and integration – of both operations and technology. At that time a number of different email systems were in use throughout the organizations. They were delivered by different vendors and operated locally. In addition, they were integrated through a gateway at the bank's headquarters in Toronto. The motivation for the integration was primarily cost containment: replacing the different systems with just one and centralizing its operations would cut licence costs and make the operations of the overall infrastructure much simpler and more efficient. At least, that was assumed to be the case. And it was to some extent true.

The integration of the infrastructure was seen to be a technical issue. But in reality, working practices were in some areas within the bank tightly integrated with the email systems. So the implementation of a shared and integrated, that means also standardized, email infrastructure also implied a sort of integration and standardization of work practices. And these work practices were very different: in some business areas, retail banking for instance, email was used to send simple messages around; after the message was read it was largely irrelevant. In other areas, derivatives trading for instance, email was used extensively, and emails contained important information. The folders with old emails were a kind of business-critical information system. This had significant implications: the first group did not need much storage space to keep old emails, while the latter group really did. And when the integrated infrastructure was established, a space limit was set for how much email each user was allowed to keep. This was done without much knowledge about its possible implications, and the rule was not implemented in a strict sense either. For some users, old email was deleted automatically. But because old emails were so important for some users, this regime was not implemented universally.

Central in this case is one incident – an incident triggered by the interaction of several events, and interaction between side-effects of how the events were responded to. It all started when the London office closed down all equipment during the Easter bank holiday to carry out the regular annual check of the electricity back-up system. After the check, some of the email servers didn't boot. It turned out that the reason was the amount of email stored as the space used for this was far, far beyond the limits set. After some time it was discovered that it would take a considerable time to recover from this failure, and in the meantime most email users in the bank would be affected. Because of the number of users affected, it was decided to go for what was believed to be the fastest, but more risky, recovery procedure. This procedure quickly generated some unforeseen situations, and the way these situations were dealt with triggered new unforeseen side-effects, and so on. The whole email infrastructure was up and running again after one week. The incident did not seem to have any significant long-term

negative effects, but during the recovery process, the bank was vulnerable and under unfortunate circumstances, and the crisis could possibly have escalated further to a level that could have been disastrous for the bank. For instance, for a period all users were given full access rights to the whole folder system, which contained a large amount of sensitive information for the bank and its customers. If this fact had become widely known and important sensitive business information about the bank's customers had been spread widely, the reputation of the bank could have been seriously damaged.

The last case, presented by Claudio Ciborra, is a bit different. It includes different technologies, different use areas, and different integration strategies. It is about the evolution of a substantial part of the IS portfolio in a pharmaceutical company over almost two decades. At the centre of this chapter are the various technologies used to support communication within and across the various business areas and national subsidiaries, how these technologies have evolved and been used, and the strategy and managerial practices behind their evolution. The main focus is on the development and use of Intranets over the last few years. Central to the overall picture presented is the very high number and enormous variety of systems, projects, use areas, rapid organization change, and so on – that is, complexity – we find in an organization like this one. From an early stage this complexity has moved beyond what can be managed from a central point and in a coherent manner. This is also true if one looks just at the Intranet. The Intranet – even within one division – is made up of a huge number of more or less independent sub-units and without any central or strict coordination of the sub-units and the activities through which they are developed and enhanced. These sub-units are integrated in various ways and to various degrees. But overall, they are loosely coupled (Perrow, 1984), and not tightly coupled (*ibid.*) around a shared data base as the systems in the previous chapters are. But what this chapter reports is a case where the Intranet development and use has more or less worked very well.

3 CONCLUDING REMARKS: COMPLEX ICT SOLUTIONS – HIGH RELIABILITY ORGANIZATIONS OR RISK SOCIETY?

The cases presented here should demonstrate quite clearly that ongoing efforts at making large scale integrated ICT solutions indeed create complex socio-technical systems that also embed numerous risks. The cases illustrate a wide range of different kinds of risks that are produced by side-effects of a wide range of different activities and components. Questions to

be discussed related to the cases are: were the risks caused by poor practices? Could more of the risks have been predicted? Would the adoption of 'best practices' within software engineering and risk management have made a difference? As far as we can judge, 'best practices' were followed in the cases reported. Or at least, we cannot find that any decisions were made or actions taken that were against what is considered best practices. And the risks that were not predicted, like those related to the scaling up of the IS in the ship classifications case, for instance, could not easily have been predicted. Similarly, the fact that the establishment of the integrated email infrastructure in the bank case should involve the kind of risks that actually emerged also seems quite hard to predict to us – at least within the framework of established software engineering practices. However, among the cases there is at least one that needs some discussion before the conclusion that 'best practices' would not help may be considered valid. This is the mobile telecom operator case. In this case, a major challenge was that the project members could not find people in the organization that knew how the existing billing system was working and how it was integrated within its context. One might easily argue that any organization needs to make sure that they have competent people knowing how its systems work. And this is what any textbook would say – unless it is considered so obvious that it is unnecessary to say it. So in this way this organization did *not* act in alignment with best practices. However, following this 'best practice' is certainly easier said than done and it would have huge implications. We cannot see how this mobile telecom operator could follow this best practice without slowing down its growth rate dramatically. When growing, they had to put their most experienced people into the most important positions; slowing down the growth rate would imply that they were making less money. It would also imply that their market share would decrease and they would possibly be a loser in the market and in the end disappear. If we look at the specific project reported, we see that there were so many risks involved that they were beyond what could be managed. The solution to this problem, again, would also be to slow down planned progress speed of the project and reduce its ambitions. However, this solution would also have potentially high costs. The operator embarked on the project in order to be able to bill the new 3G services. They believed, like everybody in the business, that these services would be offered soon and consumed in huge amounts. Not being able to bill these services would have huge negative impacts on the market position and profits of the company. It turned out that this prediction was false. However, when the project started, not being able to bill 3G services in due time (that is, fast) was considered a more serious risk than the risks related to the project. So we cannot see that 'best practices' have much to offer these projects. This leads us to the next

question: can better risk and project management techniques be developed so that the risks reported can be managed? Or alternatively, can the development and use of complex ICT solutions be turned into High Reliability Organizations? Or are risks inherent to these solutions so that we are in fact producing a Risk Society where accidents will be normal?

Most interesting to discuss here is the HRO hypothesis, in other words to what extent stronger focus on risks and learning from experience would make a difference. We have not collected our empirical material in order to test this hypothesis, so no firm conclusions can be drawn, but we do believe it would make a difference in some of the cases. A stronger focus on risks and learning from experience would first of all imply that organizations embark on less ambitious projects and plan to proceed at a slower speed, that is that they, in principle, take only one step at a time and a step no longer than previous experience would be valid for. In one case, however, this medicine does not work – the mobile telecom case. In this case the company had to move ahead fast to stay in business, or, at least, the possible gains of moving ahead fast were so big that the risks were worth taking. But in other cases slower project progress and lower ambitions would make sense. This was certainly the case for the hospitals involved in the ERP project. Slower progress makes sense for them if risks are substantially reduced. The same could perhaps have been said about the bank and ship classification cases. However, of these we think the mobile telecom case is the most relevant and interesting – at least if we take most of the business literature seriously. This literature is telling us that the business environment is continuously becoming more competitive – due in particular to globalization processes heavily fuelled by integrated ICT solutions, the fact that the future is becoming more unpredictable and that companies have to change faster and faster. So this case illustrates the central aspects of the Risk Society. Risks cannot be managed – they can only be transformed and distributed, that is when trying to managing risks, we just transform them into new ones which may pop up somewhere else (and then also possibly become someone else's risk).

The last case reported, the development of Intranets in a pharmaceutical company, sheds some new light on this issue. This case was a more successful one. Drawing upon Perrow's (1984) Normal Accidents Theory, we describe this as a case where the overall integration strategy was different from the other cases: the different Intranets (as well as other solutions mentioned) were more loosely coupled than the others reported in this book. And because they were only loosely coupled, the totality they created was less complex than if they were more tightly coupled. Accordingly each component (Intranet) could be managed and modified more independently without triggering cascades of side-effects. This is also in line with Perrow's

recommendations for developing complex systems: do not build them! And, in fact, that is the only strategy we can recommend based on the research reported in this book. (And the assumptions about how risks can be reduced by reducing speed of change and growth also means, avoid making complex systems according to the definition of complexity presented at the beginning of this chapter.) But, in fact, we do not see this as a very realistic strategy. We do believe that, among other processes, globalization is making the world more complex, unpredictable and more rapidly changing. This will generate huge needs (or 'needs') for more integrated information systems – which still is the mantra of IS. Accordingly, we do believe that integrated ICT solutions will bring more fuel to the processes making the Risk Society and Reflexive Modernization more real and High Reliability Organizations more rare. Unfortunately.

REFERENCES

- Arthur, W.B. (1994), *Increasing Returns and Path Dependence in the Economy*, Ann Arbor, MI: University of Michigan Press.
- Axelrod, R.M. and M.D. Cohen (1999), *Harnessing Complexity: Organizational Implications of a Scientific Frontier*, New York: Free Press.
- BCS/RAE (2004), 'The challenges of complex IT projects', British Computer Society and Royal Academy on Engineering Report, www.bcs.org/NR/rdonlyres/3B36137E-C5FE-487B-A18B-4D/281D88EF7/0/complexity.pdf, accessed 31 August 2005.
- Beck, U. (1986), *Risk Society: Towards Another Modernity*, London: Routledge.
- Beck, U., A. Giddens and S. Lash (1994), *Reflexive Modernization*, Cambridge: Polity Press.
- Beniger, J. (1986), *The Control Revolution: Technological and Economic Origins of the Information Society*, Cambridge, Mass: Harvard University Press.
- Berg, M. and S. Timmermanns (2000), 'Orders and their others: on the constitution of universalities in medical work', *Configurations*, 8(1), 31–61.
- Ciborra, C., K. Braa, A. Cordella, B. Dahlbom, A. Failla, O. Hanseth, V. Hepsø, J. Ljungberg, E. Monteiro, K.A. Simon (2000), *From Control to Drift: The Dynamics of Corporate Information Infrastructures*, Oxford: Oxford University Press.
- Douglas, M. and A. Wildavsky (1980), *Risk and Culture: An Essay on the Selection of Technical and Environmental Dangers*, Berkeley, CA: University of California Press.
- Fitzgerald, B. (2000), 'System development methodologies: the problem of tenses', *Information, Technology & People*, 13(3), 17–185.
- Hutter, B. and M. Power (2005), *Organizational Encounters with Risk*, New York: Cambridge University Press.
- Kumar K. and J. van Hillegersberg (2000), 'ERP experiences and evolution', *Communications of the ACM*, 43(4) 23–6.
- La Porte, T.R. (1988), 'The United States air traffic system: increasing reliability in the midst of rapid growth', in Renate Mayntz and Thomas P. Hughes (eds),

- The Development of Large Technical Systems*, Boulder, CO: Westview Press, pp. 215–44.
- La Porte, T.R. (1996), ‘High reliability organizations: unlikely, demanding, and at risk’, *Journal of Contingencies and Crisis Management*, **4**: 63.
- Lash, S., B. Szerszynski and B. Wynne (1996), *Risk, Environment and Modernity. Towards a New Ecology*, London, UK: Sage Publications.
- Latour, B. (1999), *Pandora’s Hope: Essays on the Reality of Science Studies*, Cambridge, MA: Harvard University Press.
- Law, J. and A. Mol (2002), *Complexities: Social Studies of Knowledge Practices*, Durham, NC: Duke University Press.
- Luhmann, N. (1993), *The Sociology of Risk*, Berlin: de Gruyter.
- Perrow, C. ([1984]1999), *Normal Accidents: Living With High-Risk Technologies*, New Jersey, USA: Princeton University Press.
- Schneberger, S.L. and E.R. McLean (2003), ‘The complexity cross: implications for practice’, *Communications of the ACM*, **46**(9), September, pp. 216–25.
- Urry, J. (2003), *Global Complexities*, Cambridge, UK: Polity Press.
- Weick, K.E. and K.M. Sutcliffe (2001), *Managing the Unexpected. Assuring High Performance in an Age of Complexity*, San Francisco, CA: Jossey-Bass.