

Exploring reliability in information systems programmes

DAVID DENYER

PROFESSOR IN ORGANIZATION STUDIES,
DIRECTOR OF MSc IN LEADING LEARNING AND CHANGE,
SCHOOL OF MANAGEMENT
CRANFIELD
BEDFORDSHIRE
MK43 0AL

ELMAR KUTSCH

LECTURER IN PROJECT AND PROGRAMME MANAGEMENT
SCHOOL OF MANAGEMENT
CRANFIELD UNIVERSITY
CRANFIELD
BEDFORDSHIRE
MK43 0AL

UNITED KINGDOM

TEL.: +44 (0) 1234 750111

E-MAIL: ELMAR.KUTSCH@CRANFIELD.AC.UK

ELIZABETH (LIZ) LEE-KELLEY

SENIOR LECTURER
SCHOOL OF MANAGEMENT
CRANFIELD UNIVERSITY
CRANFIELD
BEDFORDSHIRE
MK4 0AL

MARK HALL

SENIOR LECTURER
UNIVERSITY OF BRISTOL
DEPARTMENT OF MANAGEMENT
8 WOODLAND ROAD
BRISTOL
BS8 1TN

Exploring reliability in information systems programmes

ABSTRACT

The recent epidemic of information systems (IS) programme failures worldwide suggests that the effective management of programmes to cope with uncertainty and achieve mission in the medium term remains a key challenge. Research into high reliability organisations (HRO) has shown that it is possible to avoid, trap and mitigate the risks inherent in complex socio-technical systems. Studies of HROs originally focused on the operation of high risk technologies but have begun to explore other situations that present a similar need for reliable performance. One such environment is the IS programme. By comparing and contrasting the salient features of programme environments and HROs and presenting an in-depth case study with two embedded units of analysis (two troubled IS programmes), we aim to contribute to the ongoing debate about IS programme failure and to the theoretical development of programme reliability.

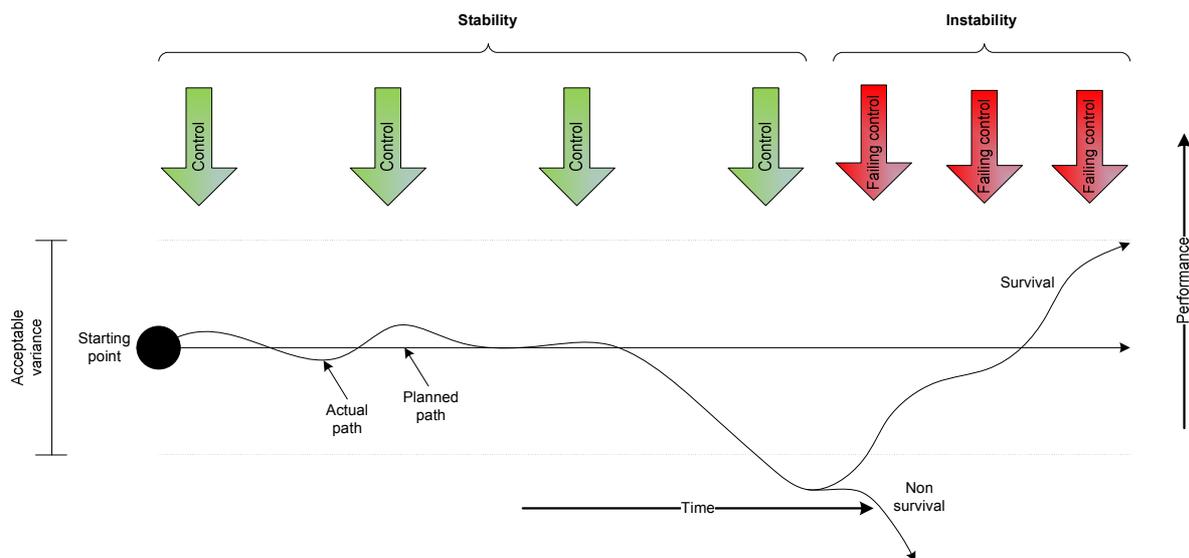
Keywords: Programme Management, Project Management, Information Systems, Resilience, High Reliability Organisation

INTRODUCTION

The failure of large scale IS programmes has attracted considerable attention in the media. It appears that every year billions of pounds are wasted on new IS programmes such as the US Advanced Automation System project (Nelson, 2007), or the UK National Offender Management System implementation (National Audit Office, 2009). In 2004, only 29% of all IS programmes succeeded in meeting their time, budget and specification objectives (Johnson, 2006). The reasons for IS programme failure are manifold. Commentators have argued that the contributory factors include ineffective stakeholder management (Nelson, 2007, Cerpa and Verner, 2009) and lack of commitment from business leaders and lack of cross-functional communication (Shehu and Akintoye, 2010). Increasingly, a concern has emerged that sole emphasis on achieving more from less, whilst initially laudable in itself, can also contribute to mission failure in the medium term.

Reliable performance, like safety, shows itself only by the events that do not happen! (Hollnagel, 2006). Stability from this standpoint can be regarded as a ‘dynamic non-event’ (Weick and Sutcliffe, 2001, p. 69), which is not involuntarily obtained, but has to be accomplished every day. Failure from this perspective, does not result from a singular failed component or barrier, but occurs as a result of an inability to respond to unpredictable changes in the context (Woods and Hollnagel, 2006, p. 14). It is the “*intrinsic ability of an organisation (system) to maintain or regain a dynamically stable state, which allows it to continue operations after a major mishap and/or in the presence of a continuous stress*” (Woods and Hollnagel, 2006, p. 14). The primary purpose of this study is to explore how programme reliability can be achieved by keeping performance within a zone of acceptable variance (Cleden, 2009) (see Figure 1).

Figure 1: Programme Reliability



The established strategies for achieving reliability in volatile programme environments are formal structures, hierarchical decision making and adherence to plans, procedures and processes and the implementation of standardised risk management methodologies promoted by institutions such as the Project Management Institute and the Association of Project Management (de Bakker et al., 2010). However, recent high profile failures of IS programmes suggest that there was no shortage of formal rules nor prior examples from which to learn. We find that an over-reliance on the structures and processes intended to control programme risk and ensure stability, can generate outcomes that are unanticipated and suboptimal. Likewise, allowing members of the programme too much latitude for developing and maintaining a repertoire of spontaneous and improvised responses to unpredictable or uncontrollable programme volatility may also lead to the escalation of crises.

In order to provide a theoretical framework for our study we draw on the literatures on high reliability organisations, organisational resilience, and crisis management to help explore how failures occur in complex systems and how some organisations have developed the ability to avoid and mitigate them. Studies of HROs originally focused on the management of high risk technologies but have begun to explore other situations where reliability is paramount (Roberts, 1990b, Rochlin, 1993). It has been argued elsewhere (Ivory and Alderman, 2005) that research into HROs offers a compelling alternative and/or expansion to the more

traditional, linear, deterministic approach to managing programmes. However, thus far, programme reliability has not been the subject of empirical inquiry.

Adopting a processual perspective (Langley, 1999), we investigate the core characteristics of high reliability in two troubled IS programmes (embedded units of analysis) involving the same client organisation, a Computer Service Provider (CSP) and two software development organisations (case study setting). All of the companies are well established in the IT industry.

In what follows, we first consider the research on IS programme environments and consider the extent to which there is a similar need for reliable performance to HROs. We then provide a review of the literature on the core characteristics of HROs and derive our four research questions. We next introduce the setting for our research and the methods of data collection and analysis. We then offer a thin description (Snook, 2000) of what happened across the event sequence before using theory to help explain why events unfolded as they did. In so doing we aim to contribute to the ongoing debate about IS programme failure and to the theoretical development of programme reliability.

THE CHANGING NATURE OF IS PROGRAMMES

Organisations are increasingly relying on information systems (IS) that are significantly more complex, dynamic and distributed than earlier technologies. The design, development and implementation of IS within a defined scope, timeframe and cost has become ever more challenging. As dependence on these technologies has grown, the nature of new IS implementation has changed with the emphasis moving from single projects to the coordination of multiple projects aligned to business objectives (Maylor et al., 2006, Evaristo and Van Fenema, 1999).

Programmes are a temporary form of organisation (e.g. Lundin and Soederholm, 1995, Lundin and Steinthorsson, 2003), rather than just a scale-up of a project (Artto et al., 2009, Lycett et al., 2004). Whereas a project can be defined as a predefined scope of work delivered using existing capabilities to achieve agreed outputs in accordance with an authorised case, a programme is a dynamic collection of related projects and activities that, in combination, achieve agreed *organisational objectives* and *emergent outcomes*, including the creation of capabilities (Lycett et al., 2004). Programmes are said to be adaptable to changes in the external environment (Thiry, 2002) but less capable of taking into account internal factors and conditions such as power dynamics and relationships (Lycett et al., 2004). With the focus

on strategic goals, programmes are a vehicle of organisational change, renewal and capability development. Examples range from information systems implementations (e.g. National Audit Office, 2009) to construction initiatives (National Audit Office, 2001).

ENVIRONMENTAL CONDITIONS THAT REQUIRE HIGH, RELIABLE PERFORMANCE

Whilst there are clear differences across organisations and sectors, IS programmes and reliability seeking organisations (Koch, 1993), share three analogous challenges posed by their environments: the potential for significant loss, interactive complexity and competing goals and interests.

The potential for significant loss

Over the past twenty five years, research has been conducted in a large number of high reliability organisations that operate in high hazard environments but have far fewer ‘errors than one would expect, given the nature of its inherent hazards’ (Rochlin, 1996) and they almost never experience an operating failure of grievous consequences (LaPorte and Consolini, 1991). Studies originally included US Navy carrier aviation (e.g. Rochlin et al., 1987), the Federal Aviation Administration’s air traffic control operations (Schulman, 1993b), commercial nuclear power plants (LaPorte and Lasher, 1988) and offshore platforms (Bea, 2002). As Weick and Sutcliffe (2001, p. 18) note, these HROs

‘have a big incentive to contain the unexpected because when they fail to do so, the results can be catastrophic. Lives can be lost, but so can assets, careers, reputations, legitimacy, credibility, support, trust and goodwill’.

Any failure of these hazardous technologies is perceived by operators and the public to have such potentially grave consequences as to warrant their absolute avoidance.

More recently, researchers have explored HRO theory in other ‘ordinary’, low hazard environments (Zohar and Luria, 2003) that also present a need for reliable performance such as police (Roberts et al., 2008), healthcare (Vogus and Sutcliffe, 2007), train operations (Jeffcott et al., 2006) and railways (Busby, 2006), electricity provision (Roe and Schulman, 2008), software firms (Vogus and Welbourne, 2003), banking (Roberts and Libuser, 1993), microcomputer firms (Eisenhardt, 1989b), and schools (Reynolds et al., 2006).

These environments differ from the contexts of the original HRO studies in two important ways. Firstly, in some of these contexts cost of failure is grave, but not necessarily life

threatening. Secondly, managers often do not have command and control over the organisation's technical core (Roberts, 2009). However, these environments also suffer from adverse events that are "physical, cultural, and emotional event[s] incurring social loss" (Vaughan, 1996, p. 292). The failure of IS programmes has the potential for significant disruption, data loss, damage to reputation and may even jeopardise long-term business survival. Failures of IS programme events that have caught public and media attention have been widespread and have occurred in the private, public and voluntary sectors and in many industries. While research has focused on major incidents, IS programme failures also occur frequently beyond the public gaze.

Interactive complexity

Reliability seeking organisations tend to operate in environments that are interactively complex (Perrow, 1984, Zohar and Luria, 2003). Interactive complexity increases when outcomes are unknown and potentially unexpected, socio-technical systems are multifaceted with incompatible functions, and where information flow is indirect and ambiguous (Roberts, 1990a). Interactive complexity is often associated with but does not necessarily co-exist with tight coupling (Schulman, 1993a). Tight coupling includes 'time dependent processes', 'invariant sequences of operations', 'the only way to reach the goal', and 'little slack' (Roberts, 1990b). According to the Perrow (1984), failure is an 'inherent property' of interactively complex and tightly coupled systems because they will inevitably experience accidents that cannot be foreseen or prevented. Perrow (1984) called these 'normal accidents'. Environments such as extended construction or IS programmes requiring significant technology and infrastructure management have high levels of social and technical interactivity. In these circumstances, people, complex technologies and physical assets become crucially interdependent. This tight coupling means that error in any particular process or activity can quickly cascade into more significant events and potentially lead to destabilisation or failure of the whole, an effect which is especially relevant in non-repetitive processes such as programme design and management.

Competing goals and interests

A focus on reliable performance is also essential when there are clear tensions between competing organisational goals such as production targets, and risk and safety goals (Leveson et al., 2009) or a potential trade-off between thoroughness and efficiency (Hollnagel, 2006). People balancing multiple goals will tend to take greater risks (Woods and Hollnagel, 2006)

and rarely have the courage to sacrifice production and efficiency goals when faced with ‘warning signs’ of impending problems (Woods, 2006). Thus,

“When organizations focus on today’s profits without consideration of tomorrow’s problems, the likelihood of accidents increases” (Roberts and Bea, 2001, p. 74).

For example, Vaughan (1996) reports that NASA was under heavy political pressure and in danger of having its budget cut at the time of the ill-fated Challenger launch.

As programmes are temporary in nature, with people and organisations entering and leaving the programme environment, achieving goal congruence between the parties involved in the programme, is particularly difficult to achieve. In programmes which often have elongated timescales, an overemphasis on reducing costs in the design and inception phases can threaten the quality of service provided and the viability of the programme in the long-term. Programmes delivered over protracted timeframes create conditions where traditional lean models may prove too ‘fragile’ to be effective. Removing slack (dubbed waste) also limits flexibility in terms of ‘space to experiment’ and ‘time to think’ (Lawson, 2001, Lamming, 1996) and this impedes organisational learning and performance improvement.

Some organisations have developed the ability to operate effectively in volatile environments. They have become known as high reliability organisations.

THE CORE CHARACTERISTICS OF HIGH RELIABILITY ORGANISATIONS

Leveson, Durac and Marais (2009) argue that high reliability organisations exhibit four essential organisational characteristics.

First, HROs *prioritize both reliability and performance and consensus about the goals across the organization* (Leveson et al., 2009). HROs find an appropriate balance between both productivity and safety goals (LaPorte and Consolini, 1991) and “consensus about these goals is unequivocal” (Leveson et al., 2009, p. 239). HROs are said to have a *preoccupation with failure* (Weick and Sutcliffe, 2001), that enables them to remain sensitive to the possibility of failure (Hollnagel, 2006). For Woods (2006, p. 29),

“effective balance seems to arise when organisations shift from seeing safety as one of a set of goals to be measured (is it going up or down?) to considering safety as a basic value”.

This leads to our first research question:

RQ1: To what extent do programmes find an appropriate balance and consensus between reliability and other performance goals (e.g. scope, timeframe and cost)?

Second, HROs promote *a culture of reliability in simultaneously decentralized and centralized operations* (Leveson et al., 2009). HROs find a way for centralisation based at the collective level to coexist with decentralisation at the individual level by exhibiting an adaptive, flexible or ‘organic’ nature (Weick et al., 1999).

Our second research question is as follows:

RQ2: To what extent do programmes reconcile the tension between the need for centralisation (formal structures, hierarchical decision making and adherence to plans, procedures and processes) and the need for decentralisation (anticipation of problems followed by rapid, improvised and mindful responses)?

Third, HROs *maximise the learning from accidents, incidents and near misses* (Leveson et al., 2009). HROs believe that errors, incidents and near misses provide a potential for understanding. They tend to treat errors as windows that reveal the status and health of the system. People in HROs are *reluctant to simplify* explanations (Weick and Sutcliffe, 2001), and are wary of interpreting information out of context. HROs are conscious of the labels, clichés (Weick and Sutcliffe, 2001) and categories (Langer, 1989) that can stop them from exploring events deeply. Even that the costs of failure are so high, learning from trial and error is impracticable; HROs compensate by using “imagination, vicarious experiences, stories, simulations, and other symbolic representations of technology and its effects” as substitutes (Weick, 1987).

Our third research question is:

RQ3: To what extent is learning, particularly from errors, incidents and near misses, achieved in programmes?

Fourth, HROs *make extensive use of redundancy* (Leveson et al., 2009). Reserve capacity allows systems to cope with unexpected circumstances (Rochlin et al., 1987). Redundancy is developed by duplicating technologies (e.g., backup computers) or people (e.g., more than one person is capable of performing a critical task) (LaPorte and Consolini, 1991). Time is also regarded as an important resource and slack is added to the decision making process, enabling actors to assess the effects of their decisions first, without affecting the overall system (Lawson, 2001). In this way, the potential consequences of faulty decisions may be understood before they actually escalate into major failure (Hollnagel et al., 2006).

Our final research question is

RQ4: To what extent is redundancy (e.g. technological, human and time) created, fostered and used in programmes?

This study aims to address these four questions by means of a case study with two embedded units of analysis (troubled IS programmes).

SETTING AND METHODS

The research methods comprise a single in-depth case study approach (Stake, 1995) with two embedded and linked units of analysis (IS programmes), underpinned by a processual perspective (Eisenhardt, 1989a, Pettigrew, 1990, Langley, 1999), paying attention to the programme context and sequence of critical events (Buchanan and Dawson, 2007). Process studies of single case study organisations have become increasingly prominent in management and organisation studies (Langley, 1999). For example, three Academy of Management prize-winning papers were process studies based on archival data and retrospective interviewing (Isabella, 1990, Dutton and Dukerich, 1991, Plowman et al., 2007). With a single case, or a small number of cases, the aim is not to generalise from sample to population, but from experience and observation to theory through a process of analytical refinement (Tsoukas, 2009) (also called analytical generalisation). As we are focusing on specific IS programmes, we will seek themes that emerge from this case and then compare them to the findings from the literature on programme management and high reliability organisations, as discussed above, in order to make more general propositions about programme reliability in general.

Data elicitation

The two programmes chosen for this study (which, for purposes of anonymity, we have referred to as *Alpha* and *Beta*) were both part of a major IT transformation for a Defence client, whom we have called Def. Ltd, and had both experienced serious pressures. At the time the study was carried out, the programmes were considerably over budget and delayed. The data collected were both real time and retrospective. The study was initiated at the critical period of ‘Rescue and Salvage (see Figure 3) which was worthy of particular attention because this phase provided insights into the possible discontinuation of both programmes. Both programmes involved were chosen for their strategic importance, i.e. the implementation of a major information system which, if it failed, could even threaten the viability of the client.

The client for both programmes, is a major player in the Defence industry. As a main contractor they used a Computer Service Provider (CSP) whom we have named All Inc. Solution. Typical services provided by this company include “planning, operation, implementation and use of computer hardware, computer software and computer personnel” (Howard, 2001, p. 2). Examples of programmes include “Roll Outs” and the implementation of “User Help Desk” structures or “Outsourcing” programmes. In 2007 in the UK, such services alone represented £22.3 billion in turnover, having increased by 5.6% on 2006, for the stand-alone CSPs of which approximately 50% of this service volume was delivered through programme work (Howard, 2001, p. 8). The strategic importance and costs involved in developing IT systems have raised the stakes associated with the programme outcomes (Keil, 1995).

The task of the CSP All Inc. Solution was to integrate a software solution on a hardware platform. In the programmes *Alpha* and *Beta*, the development of the software solution was done by two other CSPs subcontracted to All Inc. Solution; an organisation we have called Dellsys in the case of *Alpha*. All companies are well established in the IT industry and have a track record of IS successes. All companies, including the client Def. Ltd are active worldwide and yet the results show that the integration between software and hardware may be more challenging than expected.

Langley (Langley, 2009) argues that the data sources for the examples of process research typically comprise some or all of the “big three” of qualitative research: observation (participant or non-participant), interviewing (retrospective or real-time; individual or group) and archival documents (internal or external; public or private). In total, 25 semi-structured interviews were carried out with a 20 key stakeholders for each project. Most interviews took place during the critical phase of Salvage and Rescue and some were repeated every 3 - 4 months. They varied in duration between 1 and 3 hours depending on the amount of events and closeness to them.

The selection of the respondents was purposeful (see Table 1). First, individuals who were singled out had the greatest in-depth knowledge about the two programmes; however, we made sure that we questioned a variety of managers at different hierarchical levels who were specialists or generalists in their field. Second, the retrospective examination of the phases before Salvage and Rescue required interviewing, for example, programme managers that are now not actively involved any more in *Alpha* and *Beta*. It is interesting to note that most

changes to the hierarchy throughout the phases have been done at a programme management and project management level.

Table 1: List of respondents in programmes Alpha and Beta

Organisation	Alpha		Beta
Def Ltd.		1 x Account executive (interviewed twice)	
	1 x Relationship manager		1 x Relationship manager
	1 x Programme manager (interviewed twice)		1 x Programme manager (interviewed twice)
	1 x Programme Management Office director		
	2 x Project manager	2 x Project manager (interviewed four times)	1 x Project manager
	1 x Solution specialist	1 x Solution specialist	2 x Solutions specialist
All Inc Solutions			1 x Programme manager (interviewed twice)
	1 x Project manager		1 x Project manager
	1 x Solution specialist		
All Inc-Subco			1 x Programme manager

The questions that were asked revolved around a key ‘incidents’ or ‘events’ or ‘what happened?’ and followed questions on the impact, explanations of causes and whether anything was learnt from the event. The ladder of questions was extended to understand the trigger that made the event possible and how programme resilience was affected.

Documentation such as risk logs, schedules and status reports were analysed and the results discussed with the participants. Information derived from document analysis was compared

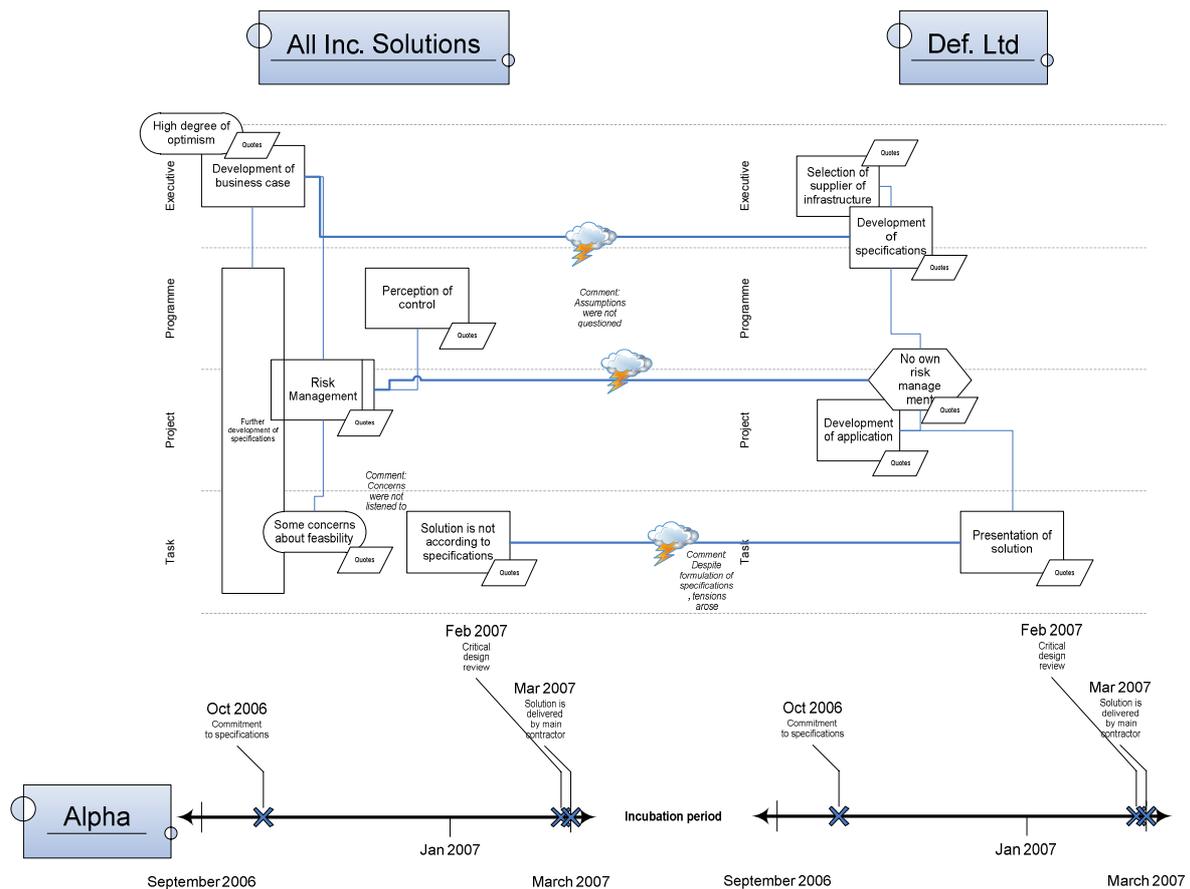
with the findings from the interviews to gain insight into the phases, manner and content of management in programmes.

Data derivation

Langley (1999) proposes seven strategies for making sense of process data: narrative; quantification; alternate templates; grounded theory; visual mapping; temporal bracketing; and synthetic strategy. Our approach was to adopt a combination of three of these approaches. We developed a detailed story from the raw data (narrative strategy), produced several timelines and graphical displays (visual mapping) and decomposed the overall programme timelines into six successive periods (temporal bracketing) related to the phases of the Disaster Incubation Model (Turner, 1976): (1) starting point, (2) incubation period, (3) precipitating event, (4) onset, (5) rescue and salvage, (6) full cultural readjustment (see Figure 1). The case studies were produced using a multi-methods approach, primarily qualitative, relying on a combination of semi-structured interviews and local documentation.

The result of the data elicitation phase was a mass of data that needed to be organised. Firstly, the temporal decomposition into phases enabled an explicit view into the logical sequence of events. Such decomposition of data enables the explicit examination of how action or inaction of one phase affects subsequent phases. Events, actions, feelings and other salient indicators were mapped at different hierarchical levels. For, example, the simplified extract in Figure 2 highlights not only the issue of lack of interaction during the incubation phase between *All Inc. Solution* and *Def. Ltd* but also within *All Inc. Solution* related simplification of concerns raised at an operational level.

Figure 2: Simplified extract from process flowchart



These visual forms of interpretation have been shared with the participants in the study for the purpose of validation. Predominantly defined as a data-driven approach, this method enabled us to derive findings that are helpful to examine programme reliability in a new light. The findings are split into two parts. We first provide a brief thin description (Snook, 2000) of what happened across the six time periods before; secondly, we analyse the two programmes through the lens of HRO theory to help explain why events unfolded as they did, why decisions were taken, why changes have or have not occurred, and the consequences.

FINDINGS

The *starting point* with Programme *Alpha* (see Figure 3) started in October 2006. The involved parties agreed on the systems requirements. All parties set out with the misguided assumption that the programme involved the straightforward implementation of an “off the shelf” product with the subcontractor *Dellsys* being in charge of the application and *All Inc. Solution* being responsible for the infrastructural platform. *Def. Ltd* defined their requirements, while *Dellsys* evaluated the ‘fitness’ of the existing application. As it stood, the system chosen for the programme *Alpha* was perceived not to require any further customisation.

Shortly after the contract was signed, *Dellsys* requested some customisation. In October 2006, *Dellsys* determined that all the required customisation could be accommodated within two releases. However, the acceptance of the system was refused in May 2007. *Dellsys* called in an audit team to investigate the refusal by *Def. Ltd.* From that stage, the period of *incubation* was marked by an iterative cycle of patching, testing, raising issues and fixing them.

In October 2007, awareness set in that management had lost sight of the development of *Alpha*. Cost and time overruns were the result. Another audit, this time a joint audit by *Def. Ltd/Dellsys* was carried out. *Dellsys* claimed not to be able to keep pace with the changing requirements. As a result of the discrepancy, testing times were increased,

“So that gives you an indication of what the domino effect was, that as soon as one slipped, it put more pressure on the next release, without actually changing any of the dates because politically, people had to hit the dates for the benefits case and you're not allowed to fail in [*Def. Ltd*]. You're not allowed to say no.” (Interview data)

During the *precipitating event*, in March/April 2008, ten months later than scheduled, a pilot and acceptance testing was carried out. In January 2009, the system went ‘live’. Nevertheless, during *onset*, stability issues arose. In a period of *rescue and salvage*, 10 weeks of ‘fire fighting’ were necessary to find a resolution to the stability problems. Despite the delay and budget concerns, during *full cultural readjustment* members of the senior management reframed the programme as a success:

“Have we delivered what we said we'd deliver? Yes. So, having got to the end of it, we can say it's a success... Despite all the things that have caused us to be 35% over budget and 50% over time-wise, we've delivered.” (Interview data)

As a result of this reframing, at the end the data collection, relationships between organisations remained awkward, informants suggested that issues lay dormant, bugs and issues were left unresolved, and neither party had effectively made the full *cultural readjustments* required to prevent recurrence.

Programme *Beta* followed a similar pattern (see Figure 3). In this case, the *starting point* involved the parties *Def. Ltd, All Inc. Solution* defining and committing themselves to software and hardware specifications. Workshops to define the requirements of the programme in more detail were carried out in October 2006.

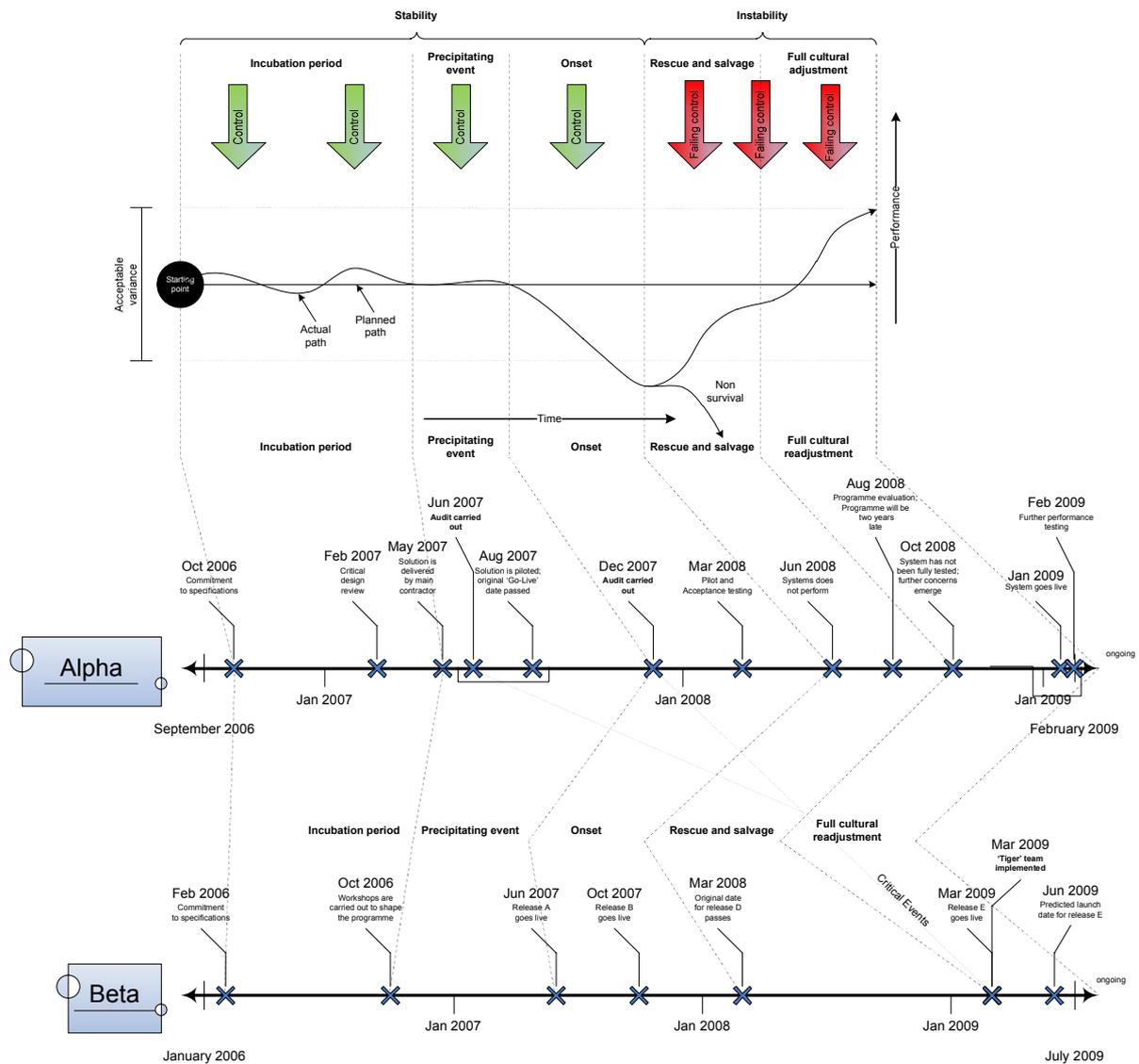
June 2007 saw the first release of the system *Beta*. Already at that stage, problems were occurring during *incubation* but signals that the programme was in trouble were missed, masked and de-emphasised and, left unattended, these faults, errors and problems quickly escalated. The system did not run in a stable manner and functionalities did not materialise as expected. Despite the initial setbacks, due to growing pressures to meet deadlines, the planned times for launching further releases were not moved back – this was the *precipitating event*.

In March 2009, during *onset*, a further release was implemented and went live. However, the problems incurred by the previous releases added to the complexity of the entire non-performing system and the programme team lost sight of how to overcome the increasing problems with *Beta*. In March 2000 *rescue and salvage* was achieved by an external team of specialists who were set up to carry out a high level audit on *Beta* and devise a suitable solution to break the cycle of testing, patches and errors.

In 2009, *Beta* was considerably delayed and over budget. Once again, no *cultural readjustment* occurred and the programme outcome was again described as a success. The programme manager argued:

“The [*Beta*] programme Release D in itself, in my opinion, was a huge success in what we did because we're one of the only companies who's managed to do a four tier PLM architecture deployment on a global scale. Not many other companies, I believe, have managed to do it successfully.” (Interview data)

Figure 3: Timeline of programme Alpha and Beta



To help explain the case, we now draw on the four primary aspects of High Reliability Theory outlined above. These are: identification and prioritisation of shared goals, organisational learning, simultaneous decentralised and centralised operations, and the use of redundancy.

Identification and prioritisation of shared goals

In both the programmes *Alpha* and *Beta*, the gateway review led to a definition of a business case, in which requirements were defined and agreed upon. It was clear that all parties involved were over-confident and possibly complacent from the *starting point*:

“And indeed, they gave a commitment there that whatever customisation they did, they would include it in their [off the shelf] product within two releases. So, on that basis, we went forward with it.” (Interview data)

The degree of faith and confidence in the guaranteed success of the programme was reinforced by many of the programme and account managers.

“... we’ve had a long period of pretty successful SAP implementations and deliveries, and we have generally been quite successful on timescale and cost over a ten year period.”

Determining the priority of goals in both programmes also revealed a confusing picture. Senior management at the client *Def. Ltd* emphasised that the primary goal of both programmes was to provide the end-user with an information system that was perceived to be useful and easy to use. Yet, this perception was not shared:

“There’s pressure from [*Def. Ltd*] to keep the cost down at the start of the programme, and there’s pressure from [*Def. Ltd*] to, you know, to deliver to a certain date.” (Interview data)

The result of that pressure was the curtailing of testing times and the questioning of suppliers’ estimates of work packages by *Def. Ltd* (see redundancy). In some cases, the estimated resource requirements were downgraded to a level only to be accepted by *Def. Ltd*.

There was also significant pressure to hit deadlines. Due to previous delays in launching ‘releases’ of the information system, the importance of meeting original time commitments became more prevalent:

“Loads of additional work came in, but the date couldn't move. It had been communicated to the business. So we were up against... And the guys, the technical guys upstairs, just worked every weekend for, like, six weeks. They worked stupid hours. And we just said we'll focus on the build and get it in and the programme will have to support it while we write up all the documentation afterwards.” (Interview data)

Despite the increasing amount of problems that emerged, baselines were not moved. Consequently, first the workload increased. Second, greater risk taking that involved more and more issues (such as stability issues) were waived and “no go decision was never really a no go.” (Interview data) and supposedly error-free IS releases were launched:

“Because they have political dates to hit. We must hit it or we'll go into the change freeze, we'll lose the benefits case. I mean, as I say, I actually wrote five or six paragraphs on it, a saga, and the Chief of [*Def. Ltd*] just ignored it, as did my counterpart, the programme manager that I faced off to. He just [said] ‘thanks very much, we're buying off the risk’, their decision.” (Interview data)

Third, not surprisingly, greater tensions arose between *Def. Ltd* and their suppliers:

“I had a huge fall out with the Chief of [*Def. Ltd*] in that Christmas and basically just said you're not listening to what the delivery team are telling you. You're still driving us to March, it's impossible. The plan we look at is talking about August/September. We don't have the resources, we don't have the skills, we don't have the time and we don't have any testing in there. We have like two weeks testing, okay. So you're gonna go live and this thing underpins all your business, and you're going to risk the business with two weeks' testing.” (Interview data)

In HROs there is a preoccupation with failure (Weick and Sutcliffe, 2001). However, respondents were not aware of the importance of the programmes that they were working on and the possible costs of failure:

“I don't know what the quantified impact is, to be honest. I would need to find that out.” (Interview data)

“...I'm not 100% on what the business impact is but there will be one. This programme going live is a dependency for other programmes that I'm aware of, so they will also be impacted.” (Interview data)

It is not surprising that balancing and achieving consensus on goals was difficult at first – key stakeholders in these programmes were not aware of the impact of failure. Second, the goal of a resilient information system was incongruent with efficiency measures. The erosion of the goal of IS resilience was caused by past failure to meet deadlines and increasing pressure to reign in costs that were escalating due to the unexpected problems that emerged in a tightly coupled system that those close to the programmes claimed to be a ‘simple off the shelf’ implementation.

Simultaneous decentralised and centralised operations

In both *Alpha* and *Beta*, a strong top-down approach could be observed. Those who were considered close to the ‘coal face’ had little or no discretion to act. The capability of

localised decision making was limited. Required changes often needed to be escalated to higher authorities. Centralisation was also evidenced by the unchallenged compliance with the rules and procedures imposed by *Def. Ltd*:

“So, you know, for example, if we need to instal a new server into an existing server room, connect it to the [*Def. Ltd*] network, there’s no way that we would be able to bypass any of the processes that are replaced to protect that environment from cowboys like us coming along. So, you know, we must put in our request to make a change to the production environment, you know, at least one week in advance of doing it. We won’t get that request granted unless we’ve passed all the necessary programme phase quality gates before we get to that point. We must fill in our questionnaires that go round to the 25 different stakeholders within the [*All Inc. Solution*] leverage team organisations, to get approval for our programme to proceed beyond a certain point.... not even your Theme Lead or your Programme Director in the organisation has the authority to say, ‘No, this is a royal train, we’ll bypass all of this,’ it just doesn’t happen.”

The ultimate decision to launch a release is taken by chief programming engineers. The decision is influenced by systems engineers:

“I’m just trying to enforce the point that the final decision would rest with them but it’s almost to the level where we’re both making a decision because we’ve given that recommendation which, not forces, but strongly says, that this is the view to be taken.” (Interview data)

Neither information systems specialists nor end-users alike have a ‘veto’ to prevent the launch of a release from happening,

“No, I don’t think there’s [any point] trying to get too hung up on that point. We would always give a strong recommendation, and it will never be a weak recommendation. The point was just that, you know, we would try to promote one option which we feel is best.” (Interview data)

The launch of releases in programmes *Alpha* and *Beta* were mainly driven by *Def. Ltd*, at times against the concerns raised by *All Inc. Solution*. Overall, the programmes appeared to be top-down with lead times that did not do justice to the constant changes in the IS development. The ‘launch’ decision was highly centralised and hierarchical.

Organisational learning

Weak signals of failure were evident throughout the period of *incubation* and programme risk management procedures were in place to encourage learning about the unknown. However, management of risk only appeared to confirm that the desired solution was ‘simple’:

“I assumed that *Dellsys* knew what they were doing.” (Interview data)

Further, there was limited scanning and sensemaking (Weick, 2009) of the changing context by the programme team. Rather than critical reflection, deviations from the plan were passively downgraded, ignored or accepted:

“I think the way it was, was when we did the new plan to get the civil instance into production we raised a load of planning assumptions and risks around that schedule and they were bought off by [*Def. Ltd*]. Because the blind optimism of the [*Def. Ltd*] programme was that this will work, that the fact that they knew it was gonna be issues to them I was saying, ‘Well, if you’re going for a date operational readiness should be what it is, you are ready to go live.’ It was not ready to go live and it was lucky it only picked up five reds because it could have picked up six or seven if there’d have been a less political sensitivity because it’s a joint gate between [*Def. Ltd*] and [*All Inc. Solution*] so it’s [*Def. Ltd*] telling [*Def. Ltd*] to slow down and put all this testing in place.” (Interview data)

Overall, a blind faith in the competence of the providers prevailed. At no point did any senior members of the team question the core assumptions and premises of the programmes. The performance of risk management exercises appears to have reinforced situational blindness and the status quo.

Provision of redundancy

In the programmes *Alpha* and *Beta*, the ability to act locally if necessary is further constrained by the availability of specialist resources. In the case where capacity is required to adapt to changes in the programme, lead times hamper the ability for immediate action (interview data). For efficiency gain, a leveraged model was applied in *Alpha* and *Beta*. The aim was to fully utilise resources, in particular those that were critical to the success of the programmes:

“What we are doing is planning and resourcing and we want a plan that we think is gonna be deliverable because we need to give the resource groups a heads up on when

they need to supply their leverage staff, we need to do the resource levelling, we need to work out who we really need and if we haven't got enough then we've got enough lead time in the plan to get those resources in from somewhere else; never happened.” (Interview data)

The availability and planned access to resources was further limited by unwillingness to share the burden of slack resources:

“And then, it's a case of but we're driven by being on time and materials contract and I can't invest.... I wanted to bring a quality manager in, okay? Last year, I said to my manager, can I bring a quality manager in to start, you know, being the policeman and make sure we do the processes, that we do the right thing? Only if [*Def. Ltd*] fund it. Well can't that be our investment? Only if [*Def. Ltd*] fund it. [*Def. Ltd*], will you pay for a quality manager?” (Interview data)

The drive for lean programme management and the attention of efficiency measures left programmes *Alpha* and *Beta* in vulnerable positions. First, and in particular, specialist software engineers were not available to form slack resources. Second, the volatility of the programmes seems not to have allowed a form of planning certainty that is necessary for a leveraged model to work. Third, the ambiguous distribution of the burden of using slack resources led to highlighting the competitive and blame led culture.

TOWARDS A FRAMEWORK FOR PROGRAMME RELIABILITY

Our case study has shown that the challenges presented by the high potential for loss, interactive complexity, and competing goals and interests, were amplified in a rapidly changing programme environment. In programmes, it is impossible to plan with certainty (Packendorff, 1995) as they are likely to unfold in unique, unpredictable and unexpected ways (Cleden, 2009). Due to the interactive complexity of the programmes there were “unfamiliar, unplanned, or unexpected sequences of activities that were not visible or immediately comprehensible” (Roe and Schulman, 2008, p. 53). The programmes moved from a state of relative stability to high volatility. High volatility is characterised by uncontrollable changes or unpredictable conditions (Roe and Schulman, 2008). This volatility can be produced by factors, conditions and mechanisms that are internal or external to the programme. Our data also show that managers attempted to use a variety of strategies to achieve programme reliability.

Roe and Schulman argue that reliability is achieved by, “developing and maintaining a repertoire of responses and options in the face of unpredictable or controllable system volatility” (Roe and Schulman, 2008, p. 41). By building and further extending their framework, we argue that there are four approaches to achieving programme reliability (see Figure 4). According to Roe and Schulman (Roe and Schulman, 2008), reliability is also achieved by dynamically matching the contextual conditions with appropriate mechanisms and approaches. Problems occur when there is a mismatch between the strategy and the circumstances or when the organisation has a preoccupation with one approach.

Figure 4: Four approaches for achieving programme reliability

		Programme Volatility	
		Low	High
Options Variety	High	Containing: Accomplishing Reliability	Adjusting: Maintaining Reliability
	Low	Deactivating: Enforcing Reliability	Fire fighting: Restoring Reliability

Containing: accomplishing reliability

When programme managers have multiple options at their disposal and the volatility inside and outside the programme is low, reliability can be achieved by means of control. Risk management processes allow programme managers to identify and put in place strategies to ameliorate potential risks. Slack is built into the programme in the form of technical backups, additional actors, or space and time to consider options. During periods of relatively low volatility, decision making can be centralised. Changes to specifications, however small, can be ‘signed off’ by authority, and decision-making processes can be slow and thorough. Coordination is achieved through standardisation of a wide range of established policies, procedures, processes and practices. Little communication is required across the members of the organisations involved in the programme. In periods of low volatility, it is important to guard against overconfidence and complacency. Thus vigilant watchfulness (Roe and

Schulman, 2008) is required to anticipate, surface and act upon early warning signs of changing conditions.

Our study shows that slack resource was hard to justify in programme environments. Both *Alpha* and *Beta* drew on a small pool of skilled people and embraced a fixed set of beliefs relating to the need for 'lean' (meaning 'waste free') programme delivery. The result was stretched resources leading to a brittle programme environment. Through a process of practical drift (Snook, 2000) local action often departs from written procedure, creating loopholes in the system's defences. Our data showed that members of the programme also developed a false impression of low system volatility from the outset, believing that the programme was a simple implementation of an off-the-shelf product. They also adhered to a reductionist model and approaches that oversimplified explanations, hindering the detection of a deteriorating performance. Weick and Sutcliffe (2001, p. 74) note that: 'With every problem, someone somewhere sees it coming. But those people tend to be low rank, invisible, unauthorized, reluctant to speak up. In our study, weak signals of impending problems were detected by people in the programmes but this information was not acted upon.'

Adjusting: maintaining reliability

When volatility increases due to changes in programme or the external environment, rapid and flexible responses are required (Roe and Schulman, 2008). Reliability in this mode is achieved when timely action is taken and changes made before unforeseen incidents and events become too costly. Cleden (2009) highlights the critical role of the programme manager and members of the programme in achieving reliable performance by applying 'dynamic controls'. Those with operational expertise are empowered to make quick decisions and take corrective action when unexpected situations arise (Weick and Sutcliffe, 2006). Actors create new practices by inventing solutions with the resources available through bricolage (de Certeau, 1984). Coordination is achieved through people talking in real time (mutual adjustment) (Mintzberg, 1983).

Our data show that reliability breaks down in this mode when repeated successful operations lead to the misconception that the application of 'fine-tuning' (Starbuck and Milliken, 1988) is working, thereby creating and reinforcing an 'illusion of stability', an erroneous belief that programme are under control (Butler and Gray, 2006, Cleden, 2009). This mode of reliability requires communication of rich, real-time information about the health of the system and any anomalies or incidents (Weick, 1987). Achieving reliability in conditions of high volatility by

means of adjustment was surprisingly absent in our study. Our data did reveal that cross organisational and cross functional communication in both *Alpha* and *Beta* was inaccurate, insufficient and ambiguous.

Fire fighting: restoring reliability

In fire fighting mode, unpredictability and uncontrollability is high but the number of options available to programme managers is low. Under these conditions “even small deviations in the elements of the market, technology, or other factors in the system can ramify throughout the system” (Roe and Schulman, 2008, p. 46). In such settings it is crucial for everyone to take responsibility – the buck stops everywhere – it is everyone’s duty to intervene if they have a concern, to stop errors escalating and prevent failure (Roberts, 1990a). Such situations create high levels of anxiety and pressure. Weick and Sutcliffe (2001) demonstrate how, in times of elevated stress or high error potential, expertise and experience is more highly valued than rank in decision making. Under conditions of intense stress or high hazard operations, HROs exhibit elevated levels of collaboration and collegiality. Groups exhibit ‘heedful interrelating’ and ‘collective mindfulness’ (Weick and Roberts, 1993). Weick (1993) demonstrates that developing situational awareness and understanding are critical. Rather than the linear strategic thinking evident in most programmes sensemaking requires,

“contextual rationality. It is built out of vague questions, muddy answers, and negotiated agreements that attempt to reduce confusion.” (Weick, 1993, p. 636)

Our study shows that reliability can break down in this state when individuals are confused and unclear on the limits of autonomy and discretion. Heedfulness is also impeded when there is a lack of trust, respect and honesty. In these situations people are inclined to diffuse responsibility for action to others (Latané and Darley, 1970). Rather than technical risk management systems, the key to reliability in situations of high volatility and low options is the provision of support mechanisms to “help people cope with complexity under pressure to achieve success” (Hollnagel et al., 2006). Our data show that these support mechanisms were not evident in *Alpha* and *Beta*. The fire fighting, therefore, had consequences. Respondents reported high levels of personal stress, relationships between members of the programme team were fraught and problems left unresolved.

Deactivating: enforcing reliability

In the final state, high programme volatility is reduced by removing components, restricting or shutting down operations. Studies have shown that in high reliability organisations employees are empowered to abort operations regardless of rank and are commended for their decision even if it turns out to be a false alarm. In extreme cases, reliability can also be enforced by an outside body such as a regulator. During this mode, predetermined crisis management and business continuity processes are imposed to limit damage. Management in this mode tends to be command and control. Extensive dialogue, negotiation and patience are required to overcome the potential tensions between the stakeholders to share the burden (costs) of enforcing reliability. Forms of relational contracting as practised in, for example, Heathrow's Terminal 5 programme (Gil, 2009), may be seen as a means of balancing and achieving agreement on seemingly incompatible interests in volatile environments.

In our study, neither *Beta* nor *Alpha* required the enforcement of reliability. However, it is interesting to note that, like the engineers at Morton Thiokol in the ill-fated Challenger disaster, technical experts in our study could only make a 'recommendation' (albeit a 'strong' recommendation) to abort operations but had no power to enforce deactivation.

CONCLUSIONS

By comparing and contrasting the salient features of programme environments and HROs and presenting an in-depth case study with two embedded units of analysis (two troubled IS programmes), we have demonstrated that programme reliability is currently challenging to accomplish, sustain, restore and enforce in contexts where there is high potential for loss, interactive complexity and competing goals and interests. To achieve programme reliability, four key changes are required from all stakeholders: fostering programme reliability as a basic value and a willingness to sacrifice short term efficiency and productivity goals to ensure medium term mission delivery; enabling a decentralised system to coexist in a centralised world; undertaking a deep and critical examination of weak signals and discrepant events; and, finally, nurturing a readiness to invest in redundancy, even when these seemingly 'slack' resources are rarely utilised.

These general recommendations mirror existing literature on high reliability organisations, which tends to offer a singular set of organisational characteristics that are seemingly applicable to a broad range of contexts. Our study provides a more nuanced, contingency framework, suggesting that the strategies used to accomplish, sustain, restore or enforce

programme reliability need to match the volatility in the programme environment. Volatility is the degree to which the programme faces uncontrollable changes or unpredictable conditions (Roe and Schulman, 2008). A mismatch between the strategies adopted and environmental conditions is likely to intensify rather than attenuate levels of adversity in programmes. Further, not only have we identified four different approaches to programme reliability but have also highlighted that, to be effective, each mode requires a different set of assumptions, practices, mechanisms and support structures.

Given the spate of IS programme failure, creating programme reliability arguably should constitute a new and critical mission for the programme management community. Further, developing the concept of programme reliability might help programme managers not only to better manage the unexpected but also use adversity as a springboard for high performance in the future.

REFERENCES

- ARTTO, K., MARTINSUO, M., GEMÜNDEN, H. G. & MURTOARO, J. 2009. Foundations of program management: A bibliometric view. *International Journal of Project Management*, 27, 1-18.
- BEA, R. 2002. Human and Organizational Factors in Risk Analysis and Management of Offshore Structures. *Risk Analysis*, 22, 29-45.
- BUCHANAN, D. A. & DAWSON, P. 2007. Discourse and audience: organizational change as multi-story process. *Journal of Management Studies*, 44, 669-686.
- BUSBY, J. S. 2006. Failure to Mobilize in Reliability-Seeking Organizations: Two Cases from the UK Railway. *Journal of Management Studies*, 43, 1375-1393.
- BUTLER, B. S. & GRAY, P. H. 2006. Reliability, Mindfulness, and Information Systems. *MIS Quarterly*, 30, 211-224.
- CERPA, N. & VERNER, J. M. 2009. Why Did Your Project Fail? *Communications of the ACM*, 52, 130-134.
- CLEDEN, D. 2009. *Managing project uncertainty*, Farnham, Gower.
- DE BAKKER, K., BOONSTRA, A. & WORTMANN, H. 2010. Does risk management contribute to IT project success? A meta-analysis of empirical evidence. *International Journal of Project Management*, 28, 493-503.
- DE CERTEAU, M. 1984. *The Practice of Everyday Life*, Berkeley, University of California Press.
- DUTTON, J. E. & DUKERICH, J. M. 1991. Keeping an eye on the mirror: Image and identity in organizational adaptation. *Academy of Management Journal*, 34, 517-554.
- EISENHARDT, K. M. 1989a. Building theories from case study research. *Academy of Management Review*, 14, 532-550.
- EISENHARDT, K. M. 1989b. Making fast strategic decisions in high-velocity environments. *Academy of Management Journal*, 32, 543-576.
- EVARISTO, J. R. & VAN FENEMA, P. C. 1999. A typology of project management: emergence and evolution of new forms. *International Journal of Project Management*, 17, 275-281.
- GIL, N. 2009. Developing Cooperative Project Client-Supplier Relationships: How much to expect from relational contracts? *California Management Review*, 51, 144-169.
- HOLLNAGEL, E. 2006. Resilience – the Challenge of the Unstable. In: HOLLNAGEL, E., WOODS, D. D. & LEVESON, N. (eds.) *Resilience engineering: Concepts and precepts*. Aldershot: Ashgate.
- HOLLNAGEL, E., WOODS, D. D. & LEVESON, N. (eds.) 2006. *Resilience engineering: Concepts and precepts*, Aldershot: Ashgate.
- HOWARD, J. 2001. Computer Services: 2001 Market Report. Key Note Ltd.

- ISABELLA, L. A. 1990. Evolving interpretations as change unfolds: How managers construe key organizational events. *Academy of Management Journal*, 33, 7-41.
- IVORY, C. & ALDERMAN, N. 2005. Can project management learn anything from studies of failure in complex systems? *Project Management Journal*, 36, 5-16.
- JEFFCOTT, S., PIDGEON, N., WEYMAN, A. & WALLS, J. 2006. Risk, Trust, and Safety Culture in U.K. Train Operating Companies. *Risk Analysis*, 26, 1105-1121.
- JOHNSON, J. 2006. *My life is failure*, West Yarmouth, The Standish Group International.
- KEIL, M. 1995. Pulling the plug: Software project management and the problem of project escalation. *MIS Quarterly*, December, 421-447.
- KOCH, B. A. 1993. Differentiating reliability seeking organizations from other organizations: development and validation of an assessment device. In: ROBERTS, K. (ed.) *New challenges to understanding organizations*. New York: Macmillan Publishing.
- LAMMING, R. 1996. Squaring lean supply with supply chain management. *International Journal of Operations & Production Management*, 16, 183-196.
- LANGER, E. J. 1989. *Mindfulness*, Camb. MA. , Perseus Publishing. .
- LANGLEY, A. 1999. Strategies for theorizing from process data. *Academy of Management Review*, 24, 691-710.
- LANGLEY, A. 2009. Studying Processes in and around Organizations. In: BUCHANAN, D. & BRYMAN, A. (eds.) *Sage Handbook of Organizational Research Methods*. New York: Sage.
- LAPORTE, T. R. & CONSOLINI, P. M. 1991. Working in Practice but Not in Theory: Theoretical Challenges of High Reliability Organizations. *Journal of Public Administration Research and Theory*, 1, 19-47.
- LAPORTE, T. R. & LASHER, T. 1988. Cold Turkeys and Task Forces: Pursuing High Reliability in California's Central Valley. *Working Paper 88-25*. Berkeley, CA: Institute of Governmental Studies, University of California.
- LATANÉ, B. & DARLEY, J. M. 1970. *The unresponsive bystander: Why doesn't he help?*, New York Appleton-Century Croft.
- LAWSON, M. B. 2001. In praise of slack: Time is of the essence. *Academy of Management Executive*, 15, 125-135.
- LEVESON, N., DULAC, N., MARAIS, K. & CARROLL, J. 2009. Moving Beyond Normal Accidents and High Reliability Organizations: A Systems Approach to Safety in Complex Systems. *Organization Studies* 30, 227-249.
- LUNDIN, R. A. & SOEDERHOLM, A. 1995. A theory of the temporary organisation. *Scandinavian Journal of Management*, 11, 437-455.
- LUNDIN, R. A. & STEINTHORSSON, R. S. 2003. Studying organizations as temporary. *Scandinavian Journal of Management*, 19, 233-250.

- LYCETT, M., RASSAU, A. & DANSON, J. 2004. Programme management: a critical review. *International Journal of Project Management*, 22, 289-299.
- MAYLOR, H., BRADY, T., COOKE-DAVIES, T. & HODGSON, D. 2006. From projectification to programmification. *International Journal of Project Management*, 24, 663-674.
- MINTZBERG, H. 1983. *Structure in Fives: Designing Effective Organizations*, New Jersey, Prentice-Hall.
- NATIONAL AUDIT OFFICE 2001. Modernising construction. *Report by the comptroller and auditor general* London: The Stationary Office.
- NATIONAL AUDIT OFFICE 2009. The National Offender Management Information System. London: The Stationary Office.
- NELSON, R. R. 2007. IT Project Management: Infamous failures, classic mistakes, and best practices. *MIS Quarterly Executive*, 6, 67-78.
- PACKENDORFF, J. 1995. Inquiring into the temporary organisation: New directions for project management research. *International Journal of Project Management*, 11, 319-333.
- PERROW, C. 1984. *Normal accidents*, New York, Basic Books.
- PETTIGREW, A. M. 1990. Longitudinal field research on change: theory and practice. *Organization Science*, 1, 267-292.
- PLOWMAN, D. A., BAKER, L. T., BECK, T. E., KULKARNI, M., SOLANSKY, S. T. & TRAVIS, D. V. 2007. Radical change accidentally: The emergence and amplification of small change. *Academy of Management Journal*, 50, 515-543.
- REYNOLDS, D., STRINGFIELD, S. & SCHAFFER, E. 2006. The High Reliability Schools Project. Some Preliminary Results and Analyses. In: CRISPEELS, J. & HARRIS, A. (eds.) *School Improvement: International Perspectives*. London: Routledge.
- ROBERTS, K. H. 1990a. Managing High Reliability Organizations. *California Management Review*, 32, 101-113.
- ROBERTS, K. H. 1990b. Some characteristics of one type of high reliability organization. *Organization Science*, 1, 160-176.
- ROBERTS, K. H. 2009. Managing the Unexpected: Six Years of HRO-Literature Reviewed. *Journal of Contingencies and Crisis Management*, 17, 50-54.
- ROBERTS, K. H. & BEA, R. 2001. Must accidents happen? Lessons from high-reliability organizations. *Academy of Management Executive*, 15, 70-78.
- ROBERTS, K. H. & LIBUSER, C. 1993. From Bhopal to banking: Organizational design can mitigate risk. *Organizational dynamics*, 21, 2-17.
- ROBERTS, K. H., YU, K. F., DESAI, V. & MADSEN, P. 2008. Employing Adaptive Structuring as a Cognitive Decision Aid in High Reliability Organizations. In: HODGKINSON, G. P. & STARBUCK, W. H. (eds.) *The Oxford Handbook of Organizational Decision Making*. New York: Oxford.

- ROCHLIN, G. I. (ed.) 1993. *Defining "High Reliability" Organizations in Practice: A Taxonomic Prologue*, New York: Macmillan.
- ROCHLIN, G. I. 1996. Reliable Organizations: Present Research and Future Directions. *Journal of Contingencies and Crisis Management*, 4, 55-59.
- ROCHLIN, G. I., LAPORTE, T. R. & ROBERTS, K. H. 1987. The Self Designing High-Reliability Organization: Aircraft Carrier Flight Operations at Sea. *Naval War College Review*, 40, 76-90.
- ROE, E. & SCHULMAN, P. R. 2008. *High Reliability Management: Operating on the Edge*, Stanford, CA, Stanford University Press.
- SCHULMAN, P. 1993a. The Analysis of High Reliability Organizations: A Comparative Framework. In: ROBERTS, K. H. (ed.) *New Challenges to Understanding Organizations*. New York: Macmillan.
- SCHULMAN, P. R. 1993b. The negotiated order of organizational reliability. *Administration and Society*, 25, 353-372.
- SHEHU, Z. & AKINTOYE, A. 2010. Major challenges to the successful implementation and practice of programme management in the construction environment: A critical analysis. *International Journal of Project Management*, 28, 26-39.
- SNOOK, S. A. 2000. *Friendly fire: The accidental shootdown of U.S. Black Hawks over Northern Iraq*, Oxford, Princeton University Press.
- STAKE, R. E. 1995. *The art of case study research*, London, SAGE Publications Ltd.
- STARBUCK, W. H. & MILLIKEN, F. J. 1988. Challenger: Fine-Tuning the Odds Until Something Breaks. *Journal of Management Studies*, 25, 319-340.
- THIRY, M. 2002. Combining value and programme management into an effective programme management model. *International Journal of Project Management*, 20, 221-227.
- TSOUKAS, H. 2009. Craving for generality and small-n studies: a Wittgensteinian approach towards the epistemology of the particular in organization and management studies. In: BUCHANAN, D. & BRYMAN, A. (eds.) *The Sage Handbook of Organizational Research Methods*. London: Sage Publications.
- TURNER, B. A. 1976. The Organizational and Interorganizational Development of Disasters. *Administrative Science Quarterly*, 21, 318-397.
- VAUGHAN, D. 1996. *The Challenger Launch Decision: risk technology, culture, and deviance at NASA*, Chicago University of Chicago Press.
- VOGUS, T. J. & SUTCLIFFE, K. M. 2007. The Impact of Safety Organizing, Trusted Leadership and Care Pathways on Reported Medication Errors in Hospital Nursing Unites. *Medical Care*, 45, 1-6.

- VOGUS, T. J. & WELBOURNE, T. M. 2003. Structuring for high reliability: HR practices and mindful processes in reliability-seeking organizations. *Journal of Organizational Behavior*, 24, 877-903.
- WEICK, K. & SUTCLIFFE, K. 2001. *Managing the unexpected: Assuring high performance in an age of complexity*, San Francisco, Jossey Bass.
- WEICK, K. E. 1987. Organizational Culture as a Source of High Reliability. *California Management Review*, 29, 112-127.
- WEICK, K. E. 1993. The Collapse of Sensemaking in Organizations: The Mann Gulch Disaster. *Administrative Science Quarterly*, 38, 628-652.
- WEICK, K. E. 2009. *Making sense of the Organization: The impermanent organization*, Chichester, John Wiley & Sons.
- WEICK, K. E. & ROBERTS, K. H. 1993. Collective Mind in Organizations: Heedful Interrelating on Flight Decks. *Administrative Science Quarterly*, 38, 357-381.
- WEICK, K. E. & SUTCLIFFE, K. M. 2006. Mindfulness and the Quality of Organizational Attention. *Organization Science*, 17, 514-524.
- WEICK, K. E., SUTCLIFFE, K. M. & OBSTFELD, D. 1999. Organising for high reliability: Processes of collective mindfulness. *Research in Organizational Behavior*, 21, 81.
- WOODS, D. D. 2006. Essential Characteristics of Resilience. In: HOLLNAGEL, E., WOODS, D. D., AND LEVESON (ed.) *Resilience engineering: Concepts and precepts*. Aldershot: Ashgate.
- WOODS, D. D. & HOLLNAGEL, E. 2006. Prologue: Resilience Engineering Concepts. In: HOLLNAGEL, E., WOODS, D. D. & LEVESON, N. (eds.) *Resilience engineering: Concepts and precepts*. Aldershot: Ashgate.
- ZOHAR, D. & LURIA, G. 2003. Organizational meta-scripts as a source of high reliability: The case of an army armored brigade. *Journal of Organizational Behavior*, 24, 837-859.