Understanding and Managing Process Interaction in IS Development Projects

Bendik Bygstad
Norwegian School of IT
bendik.bygstad@nith.no

Peter Axel Nielsen
Aalborg University
pan@cs.auc.dk

Abstract. Increasingly, information systems must be developed and implemented as a part of business change. This is a challenge for the IS project manager, since business change and information systems development usually are performed as separate processes. Thus, there is a need to understand and manage the relationship between these two kinds of processes. To understand the interaction between information systems development and planned organisational change we introduce the concept of *process interaction*. We draw on a longitudinal case study of an IS development project that used an iterative and incremental development approach. The concept of process interaction enabled us to understand critical events in the case, in particular those that were important for the mutual adaptation between the information system and the organisation. We conclude that process interaction is needed to facilitate socio-technical innovation in a situation where the organisational change process and the IS development process are parallel but incongruent. We also argue that iterative software engineering frameworks are well structured to support process interaction. Finally, we advocate that the IS project manager needs to manage the trade-off between necessary process interaction and the internal IS project schedule.

1. Introduction

This paper is about the necessary interaction between information systems (IS) development and planned organisational change. In 2004 a large American survey reported that the top five concerns of IT executives were (in this order): IT and business alignment, IT strategic planning, security and privacy, attracting, developing, and retaining IT professionals, and measuring the value of IT investments (Luftman and McLean 2004).

The survey indicates that three out of the top five concerns are associated with the relationship between IT and business. We find that this also has important bearings on how information systems should be developed. For IS development project managers it highlights the need to understand and manage the interaction between IS development and organisational change. There is indication that IS project managers are facing several new challenges:

- The speed of change, driven by globalization, demands that IS solutions be delivered in parallel with business change. Often it is no longer an option for the organization to wait while a new system is developed (Stapleton 2003).
- The complexity of the environments is increasing. The new information system should not only adapt to an unstable business environment, but should also adapt to large existing information infrastructures which put insuperable constraints on the IS manager's options (Hanseth and Monteiro 1996).
- The power balance between the organisation and the IS departments has changed. Instead of humble users, the IS project manager meets powerful organisational actors who are well aware of IS failures and are inquisitive of the value of IT investments (Carr 2003).

These challenges amount to a situation of considerable pressure on the IS development project manager. From a project management perspective most of the challenges are outside the direct control of the project manager, forcing him/her to negotiate rather than to plan and control. At the same time, the project manager has to understand and address the organisational change process in a way that ensures that the information system may be successfully used within the organization.

In this paper we are trying to make sense of a large development project where the project managers faced similar challenges. Our main concern is how to organise the interaction between the IS development and planned business change. Thus, we are addressing the following two research questions:

- How can we *understand* process interaction, i.e., interaction between information systems development processes and planned organisational change processes?
- How can a project manager of iterative and incremental IS development processes *manage* process interaction?

Theoretically, our point of departure is the socio-technical IS development tradition, and in section 2 we frame our analysis within IS project management research and iterative software engineering. The research approach behind the case study is longitudinal processes research, which we outline in section 3. That allows us to present the case study in section 4 where we focus on providing context and explanation for a series of critical events.

Throughout this paper we take the practice perspective of the project manager of IS development. The IS project manager has roles and responsibilities that are significantly different from managers of organisational change or general managers of the business organisation. We discuss this further in section 5, where we also discuss the usefulness of the process interaction concept and how it contributes to IS project management research as well as software engineering research. In section 6 we conclude the paper.

2. Process Interaction: A Socio-Technical Perspective

In this section we first discuss the process structures of planned organisational change and IS development. Then we assess to which degree three relevant research streams provide support for process interaction; the socio-technical methodologies, software engineering and IS project management research.

2.1. Process structures for socio-technical innovation

The socio-technical tradition within information systems development arose to deal with the single purpose of creating a fit between an organisation and the social world on the one hand and the technologies and their employment in information systems on the other (Bostrom and Heinen 1977a; Bostrom and Heinen 1977b). An information system design cannot be separated from the organisational design, and it should be developed in an integrated process.

An organisation is seen as a socio-technical work system, consisting of people, organisation, tasks and technology. Ideally, a change process should include both the planned organisational change and the development of new technology. These two processes should be run in an integrated and mutually adaptive way.

The socio-technical methodologies such as ETHICS (Mumford 1985) and Multiview (Avison and Wood-Harper 1990; Avison et al. 1998) took this holistic view into IS development methodologies. Great care was taken to ensure a correct diagnosis of the organisational problem and to establish real business objectives, to analyse the human and technical aspects of the new solution in an integrated way, to ensure real user participation and to design a socio-technical solution.

However, when looking at current organisational change and IS development research and practices, we find that these constitute two quite distinct knowledge communities, with different process structures, different vocabularies and different practices (Giaglis 1999).

Planned organisational change is generally pursued through top-down interventions to improve the problem-solving abilities of an organisation (French and Bell 1998). Some widely used approaches build on Lewin's classical stage model (French and Bell 1998); other use frameworks such as Total Quality Management (Hradesky 1995) or Business Process Reengineering (Hammer and Champy 1993). IS development, on the other hand, is increasingly building on iterative and incremental development methodologies from software engineering, such as Rational Unified Process (Jacobson et al. 1999) and DSDM (Stapleton 2003).

The incongruence between the IS development and business change process models has been noted by several researchers, for example (Christensen et al. 1999; Giaglis 1999), who argued that these differences make it hard to combine the two approaches into a unified process. As Giaglis concluded, the complexity of socio-technical change may be too great for a single, integrated methodology. Thus, the IS project manager must address a situation where two differently structured processes should be coordinated.

2.2. Assessing support for process interaction

An important premise for this discussion is the insight that a socio-technical solution cannot be specified in detail; it is the result of an emergent innovation process, where improvisation and learning play important roles (Leonard-Barton 1988; Orlikowski 1996; Ciborra 1997). This implies that the IS project manager cannot rely on a "socio-technical specification"; he has to prepare for a process of mutual adaptation and learning with the organisational change process. This need has also been documented in recent socio-technical research (Doherty and King 2003).

Which tools are at the disposal for the IS project manager to handle this challenge? We will briefly assess the two socio-technical methodologies ETHICHS and Multiview, software engineering methodologies such as RUP and DSDM, and finally IS Project management research.

Socio-technical methodologies: ETHICS and Multiview

In spite of their holistic approach ETHICS and Multiview are primarily concerned with analysis and design. The ETHICS method consists of 15 steps, but only the two last ones are concerned with implementation, and at a rather high level of detail (Mumford 1985). Multiview has 5 steps, where only the last one is technical design and construction (Avison and Wood-Harper 1990). It is fair to say that the implementation aspects were improved by the introduction of Multiview 2 (Avison et al. 1998), but the overall picture remains; these methodologies have not really addressed the need for process interaction.

Thus, the socio-technical methodologies are quite comprehensive regarding the design of a new socio-technical solution, but have less to say about the actual processes during development and implementation. Further, they are not much used in practice (Fitzgerald 1998).

Software engineering methodologies

Modern software engineering (SE) has addressed the challenge of alignment with the organisation in several ways. In 1988, as a response to the quality problems of software constructions, Boehm proposed a spiral model for software development with an iterative structure allowing for more frequent interaction with users and customers. The iterative approaches took on the challenge of unstable and changing requirements due to complex organisational issues and changing organisations. Further, both object-oriented methodologies like OOA&D (Mathiassen et al. 2000), and later the Rational Unified Process (Jacobson et al. 1999; Larman 2004) and the agile methodologies like Extreme Programming (Beck 2000) and the DSDM (Stapleton 2003) embrace the iterative approaches for these reasons. In the context of process interaction it is relevant that the iterative structure of SE processes allows for a systematic interchange of ideas and solution. The iterations facilitate a process where there is a potential for mutual adaptation between the information system and the organisation (Bygstad and Munkvold 2002).

However, the focus of this research is very much on software requirements. The aim of the development project is seen as a software product. The literature on RUP and agile methods asserts that they purport to address business needs while most of their operational concerns address technical issues. Thus, the dominant software engineering methodologies pay mostly lip service to an integrated approach, but concentrate on producing the software product. The organisation is seen as very important, but mainly as an arena for eliciting the requirements – not as a target for change. This critique may not apply to DSDM which to some degree does support organisational change (Stapleton 2003), but it does not address process interaction.

IS project management

Normative IS project management research has for a long time addressed organisational issues like business alignment, risk management, and stakeholder analysis (McManus and Wood-Harper 2003; Cadle and Yeates 2004). This research tends to be concerned with internal control. Common issues are: managing the IS life cycle, estimation, modelling, quality, scheduling, and cost. It is hardly surprising that control has become a common denominator given the turbulent history of IS project failures. On the other hand, much of the normative

IS project research gives the impression that IS projects are standalone projects starting from scratch.

A recent contribution is the notion of *value management*, which aims to identify and manage business value in addition to cost in IS projects (Boehm 2003; Cadle and Yeates 2004). Value management is based on stakeholder analysis, and thus expands the scope of the project beyond its traditional introvert perspective. Value management is not widely used, and it has so far not been integrated with current software engineering frameworks.

A rather fundamental critique has been raised during the past 15 years against the top-down planning and control approach, that it does not reflect practice. A number of empirical studies of IS development projects find that projects are situated and emergent and require skills like empathy and improvisation rather than managerial control (Orlikowski 1996; Ciborra 1997; Ciborra 2000). This critique has drawn considerable interest from researchers, but the effect on the normative IS project management literature has so far been modest.

While many of the techniques from IS project management research are necessary and useful for project management it also represents a limited view. This stream of research has little to offer regarding the interaction between the IS development process and the organisational change process. It tends to play down what is outside its own scope by treating it as environment and it tends to ignore the IS development project's relationship with the parallel processes of organisational change.

Summing up

We end this section with the following conclusion. The ambitious goal of integrated socio-technical IS development may be difficult to achieve, at least in larger projects. Instead there will often be two different processes, with the corresponding need for interaction between them.

As this short review shows, the need for process interaction is not generally acknowledged. We also found that the reviewed research streams of socio-technical methodologies, iterative software engineering and IS project management research do not provide the IS project manager with tools to manage it. However, running these processes in relative isolation from each other is certainly risky. They need to be coordinated in some way, because they are mutually dependent of each other. In the next sections we will investigate these points through a longitudinal case study.

3. Research Approach

The case study was carried out at an international airline, studying an e-business project over 18 months. The case study was planned and carried out using longitudinal process research (LPR), an intensive research approach that focuses attention on organisational processes as experienced by organisational actors

(Pettigrew 1985; Pettigrew 1990; Ngwenyama 1998). LPR is the study of organisational processes with the intention of developing contextualized theories about them. According to Ngwenyama (1998), the researcher conducts an intensive analysis of the context, temporal order and underlying logic of events in the organisational processes under study. In our case study, we have studied organisational change processes and IS development processes as they were performed over time in and around a complex project.

LPR is based on three criteria for data collection (Ngwenyama 1998):

- Engagement with the research site is required to build any substantive theory of organisational processes.
- Participant observation enables the researcher to contextualize in making sense of practices and situations. It also makes the researcher sensitive to organisational insights encoded into actors' actions and language.
- Validity is ensured through multiple sources of data, systematic data gathering and reliable data recording or transcription.

Data Collection

The data was collected over a period of a year and a half while the IS development project being studied lasted for almost a year. The main data source was semi-structured interviews with different IS development project and business stakeholders. A total of 24 interviews were conducted, each lasting from 1 to 4 hours. In addition, project meetings were observed and the findings were discussed with stakeholders. Interviews with international marketing editors were done by email. A secondary source of data was the huge amount of project documentation comprised of both product and process documentation.

Data collection was done in four phases as summarised in table 1.

Phase/dates	**Activities**	**Stakeholders**	**Documents**
Phase 1 Sept 2001	Initial meeting with management to agree on objectives and procedures in the study.	Line manager Project managers	Project objectives and plans
Phase 2 Nov/Dec 2001	Workshop with project and business stakeholders to get the broad picture, followed by separate interviews	Project manager Project group Business users	Status reports Technical documents
Phase 3 Sept/Oct 2002	Separate interviews with stakeholders to construct full time line in project	Project manager Project group Business users	Status reports Project evaluation report
Phase 4 Dec 2002/ Jan 2003	Last round of interviews. Validation meeting to confirm and discuss findings.	Line manager Project manager Business users	Case description

Table 1 Data collection

Data Analysis

LPR suggests three modes for data analysis to assist the researcher in closing the gaps between the findings and the empirical data (Ngwenyama 1998):

- Comprehensive analysis helps to reveal and surface deeper structures of the organisational processes.
- Temporal analysis helps to contextualize findings by placing events and situations in a narrative structure.
- Member verification ensures that interpretations and case descriptions made by the researcher are meaningful to the organisational actors.

Interview summaries and project documents were registered into an Atlas database and coded. Then a systematic search for patterns was conducted using the Atlas search tool. First, a timeline with significant events and iterative phases was produced. Second, iterations, context, actors, and artefacts were modelled graphically as an emerging socio-technical network. Third, a case description was written gradually over time, in a process of learning and also negotiation between the researcher and the stakeholders.

Member verification was ensured in three steps. First, the documented socio-technical network from the workshop in phase 2 was sent to the participants for comments and corrections. At the end of the case study there was a long validation session with technical and business stakeholders to review the final report. Finally, the papers that were published were also sent to some key stakeholders for comments.

The analysis further builds on the idea of critical events (Pettigrew 1985). We split the temporal analysis into events that were critical to process interaction. We describe each critical event, its preconditions in terms of what led to the event and its consequence in terms of activities following the event. We then illustrate and explain the kind of process interaction taking place. The analysis of critical events is close to the kind of analysis where Newman and Robey sliced a time scale into incidents and episodes (Newman and Robey 1992).

4. The Case

Building on the longitudinal process analysis we identified five events particularly critical to process interaction. The five events occurred in the time order as presented in table 2, and the result of the former event formed the pre-condition for the next. The critical events also coincide with the 5 formal RUP iterations.

Pre-condition	**Critical event**	**Following activity**	***Process interaction***
E-business part of airline tickets expected to grow	The Airline decides to establish a decentralized e-marketing organisation	Two projects started: Organisational change and IS development	Formal agreement between the two processes
Workshops are held to specify solution	Workshops with marketing editors fail	Editors withdraw and project concentrates on technical issues	A breakdown of the interaction between processes
The software project lacks relevant input	The business project manager becomes involved in technical development	Technical solution is developed successfully	Organisational process inactive. IS development process isolated
International editors are recruited	New marketing editors enter	After a course, the editors start testing the system. A lot of change requests and technical problems.	Improvised interaction between the two processes
The technical solution is stabilised	Start-up	The new business organisation starts to use the new solution successfully	The interaction is well structured even into production

Table 2 Critical events in the Airline case

To describe the critical events and the process interaction we make use of a simple illustration (Figures 1-5). These figures illustrate that both the business organisation and its information systems are changed through two parallel processes; the organisational change process and the IS development process. The result is a changed organisation and extended information systems.

Critical Event 1: The Airline decides to establish a decentralized e-marketing organisation

Following the e-business frenzy of the late 1990s, the Airline decided in 2000 to establish a web-based marketing channel in all important markets. Two projects were initiated:

- An organisational project where international editors in the actual markets were recruited, trained and put in charge of the e-business operation, as a part of the marketing division. Part of this project was a group of Scandinavian editors, who represented the Airline in the software project, headed by a business project manager.
- A software development project to develop the new content management and publishing solution to be used. This consisted of an experienced project manager and four developers.

The aim of the two projects was to establish a new organisational process supported by an extension to the Airline's information systems, as illustrated in Figure 1.

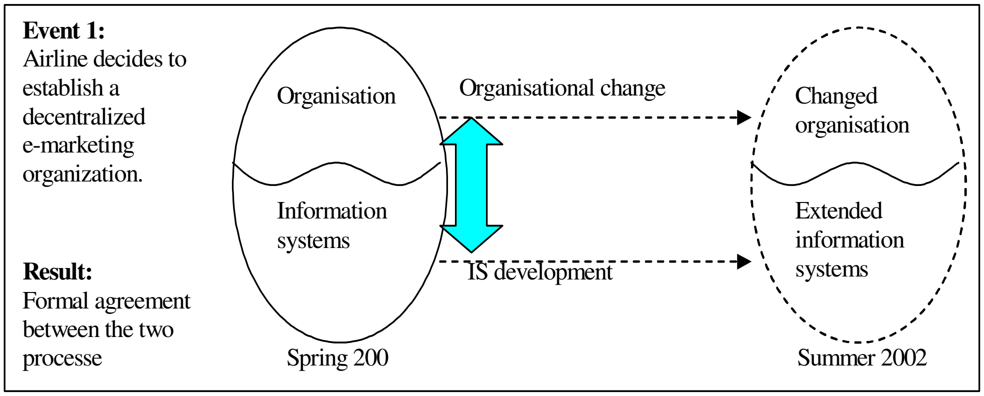
Event 1:
Airline decides to establish a decentralized e-marketing organization.
Organisation
Organisational change
Changed organisation
Information systems
IS development
Extended information systems
Result:
Formal agreement between the two processe
Spring 200
Summer 2002

Figure 1 Critical event 1

The two projects were nicely aligned, and the new organisation and system were planned to go into production in summer 2000. However, the projects were not integrated into a common plan, but were rather run in parallel. The process models were also different: the organisational change project followed a waterfall structure, while the software development project followed an iterative and incremental structure using the Rational Unified Process. The software project was planned with five iterations. Each iteration was set up to follow the workflows in the Rational Unified Process starting with a revision of requirements, proceeding with design, coding and testing, and ending with an increment, a temporary release, to be validated by users.

We have characterized the result of this event as a *formal alignment between the two processes*; they were established to achieve a shared goal and with an intention to interact during the project. This is illustrated by the unbroken arrow between the two processes, which - as with subsequent figures - is used for illustration purposes and not suggested as a formal syntax.

Critical event 2: Workshops with editors fail

In the two first iterations, the two project groups extended the number of use cases into 20 detailed ones. Then, they started working on a graphical prototype trying to translate the use cases visually. The workshops were not very successful. Some of the editors felt alienated from the whole concept:

> "We spoke different languages, and they had no idea how we worked. We were polite and there was no conflict, but that was how we felt. We thought we might get it straight later on in the process. Use cases focused on the new system – not on how things were solved today. Development was system oriented, not on the work process"

Not surprisingly, the results were unsatisfying. Nobody felt that the graphical prototype was useful. Thus, by the end of the elaboration phase (the analysis and design phase in the Rational Unified Process) the two main goals had not been reached: the business users and developers did not have a shared view of the system, and the architecture of the system was not stable.

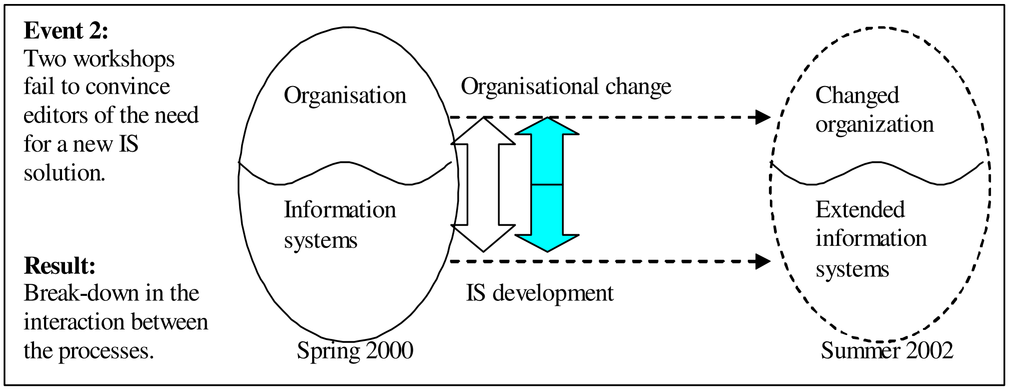
Event 2:
Two workshops fail to convince editors of the need for a new IS solution.
Result:
Break-down in the interaction between the processes.
Organisation
Information systems
Spring 2000
Organisational change
IS development
Changed organization
Extended information systems
Summer 2002

Figure 2. Critical event 2

The result of event 2 was a break-down in the interaction between the two processes. The marketing editor group was unconvinced about the need for the new system, and the software project group lacked both user input and software components. This is illustrated by the discontinuous arrow in Figure 2.

Critical event 3: The Business PM becomes involved in technical development

In the third iteration, the project group got a better grip on the technology and started to work more closely with the Business Project Manager (PM), who was now sitting in the same room. This iteration produced the basic functionality, enabling the users to upload content to the content database.
In the fourth iteration, the first release of the necessary (external) component arrived and the crucial functionality of creating web pages was developed. In a few intense and informal work sessions, a design was developed as the application was prototyped.

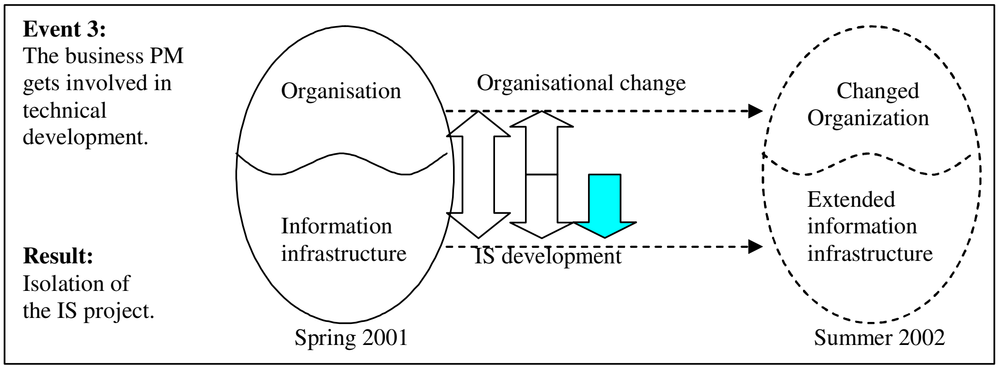
Event 3:
The business PM gets involved in technical development.
Result:
Isolation of the IS project.
Organisation
Information infrastructure
Spring 2001
Organisational change
IS development
Changed Organization
Extended information infrastructure
Summer 2002

Figure 3 Critical event 3

Although the software project spirit and technical results were greatly improved during the third and fourth iterations, the result of Event 3 was that the interaction between the two processes stopped. The Scandinavian editors had withdrawn, and the international editors were not yet recruited while the business PM practically had changed sides. This left the project unintentionally *encapsulated*, concentrating on the (quite challenging) technical issues. In Figure 3, the half arrow, pointing at the software development process, illustrates this.

Critical event 4: New editors enter

In the winter of 2001 the international marketing editors were recruited. After a period of technical problems during testing, a beta version was presented for the international editors. In March 2001, there was a two-day course for all the marketing editors, totalling at that time around 30. Most of them were introduced to the system right away without many preparations. In spite of technical instability problems and long response times, the course was perceived by the market organisation and the software team as rather successful for most of the editors. After the course, the editors went home and started to load materials into a test database that was later set into production. In this period, the SW project worked hard with error corrections and use case change orders. The PM said:

> "Many new features were wanted from editors, both Scandinavian and the others, especially navigation features tightly connected to their work processes, page search and design. We were surprised by the volume of change orders."

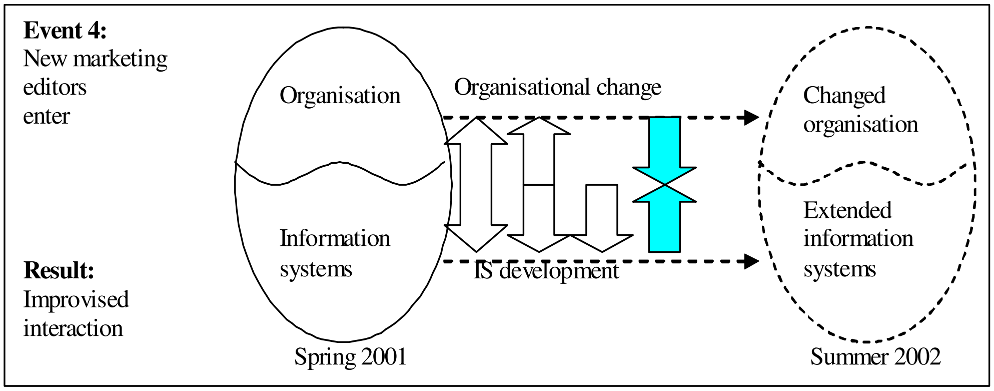

Figure 4 Critical event 4

The result of Critical Event 4 was that the interaction between the two processes was reinitiated. The nature of this interaction, in contrast to the previous iterations of the software project, was not controlled by the RUP iteration. Rather, it was characterised by improvisation and problem solving. This is illustrated in Figure 4 by the two arrows pointing towards each other.

Critical event 5: Into production

The entire solution was set into production during summer 2002. Some technical problems were experienced, but after some minor start-up problems the technical solution was stable and in use in the new international organisation. There were 50-60 users: six marketing editors in Scandinavia, and the rest in Europe, USA and Asia. Most of these were part-time editors with main responsibilities in marketing or sales. Campaigns were started at a central Marketing division level or at a national level. The solution allowed the national editors to tailor their web pages to their local markets. The day-to-day monitoring of the result of the campaigns was done on two parameters: the Marketing department followed the Internet traffic on the web site, while the Revenue Management monitored the actual booking. The running marketing decisions were taken on the basis of this monitoring.

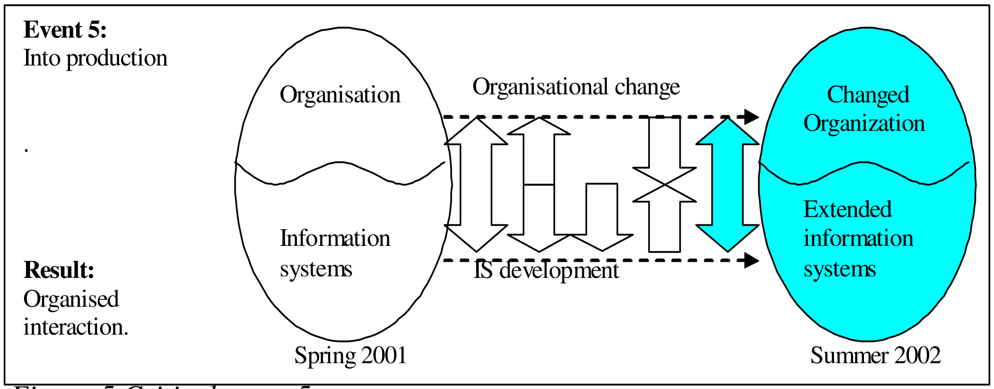

Figure 5 Critical event 5

Setting the system into production was, despite some technical problems, successful. Thus, the result of the 5th critical event was that the two processes interacted as intended. The redesigned organisation (decentralised web marketing) was aligned and integrated with the extended information infrastructure including the publishing solution. This is illustrated in Figure 5 by the final unbroken arrow.

5. Discussion

We will now discuss the case and the analysis of process interaction. First we analyse the process interaction in the Airline case. Then we discuss the research contribution of the concept. For IS research we suggest that the notion of process interaction has two contributions. The first is the need for *coordination* between the IS development process and the organisational change process, which we will discuss in conjunction with IS project management research. The second is that process interaction supports *innovation*, which we will discuss in relation to socio-technical methodologies and iterative software engineering.

6.1. Process interaction in the case

The phenomenon of interest in this paper is process interaction and we have chosen to focus our presentation on the aspects of the case that relate to this perspective. The case shows a perhaps unusually high number of critical events, leading to the following question: Were these events just indications of a poorly managed project?

We think that the evidence clearly indicates that these problems did not emerge because of poor project management, but from a lack of planned process interaction. The two project managers certainly knew they were facing great challenges and they started out by forming two projects with a formal agreement between them (critical event #1). However, between critical event #2 and critical event #4 the two processes lost contact, for reasons partly outside control of the two project managers (waiting for the new organisation to be established, waiting for external software components). The process interaction suffered greatly under this and was not rescued until critical event #4 where new stakeholders entered. The new stakeholders, the marketing editors, allocated time and resources to process interaction by providing detailed feedback to the IS development project through testing of the software. Thus, the projects were not necessarily poorly managed, but the process interaction was not organised or managed.

Could the project managers have managed the Airline case better by paying more attention to process interaction? We believe the answer is yes. First, the two projects could have been designed to interact better. In this planning, the project managers would have seen that the waterfall structure of the organisational change project was incongruent with the iterative structure of the IS development project. The easiest intervention would have been to create more planned interaction at certain intervals, ensuring that the iterative IS development project received the necessary input. Alternatively, and more expensively, the organisational change project could have been designed following an iterative process. Second, when the critical events occurred, the degree and character of process interaction could have been used to assess the situation in much the same way as we have done in the case description. This would provide a better basis for intervening into both the organisational change processes and the IS development processes.

6.2. Contribution to IS Project Management research: A Useful Extension for Inter-Project Coordination

The notion of process interaction is not only useful in analysing the Airline case. We suggest that it will also be useful in project management, planning and tracking. It represents an extension of the normative ISD project management (McManus and Wood-Harper 2003; Cadle and Yeates 2004). An important aspect of the concept is that it acknowledges the emergent nature of socio-technical change, thus incorporating some of the critique raised by (Ciborra 1997) of the somewhat mechanical structure of normative ISD project management.

We have showed in section 2 that normative IS project management research has a strong focus on internal control (McManus and Wood-Harper 2003; Cadle and Yeates 2004). Our concept illustrates the limitations of this perspective. There is a risk that this strong focus on control may constitute a barrier to process interaction. The reason is that both the IS development project manager and the organisational change project manager may prefer to maintain internal project control rather than risk the uncertainties of interaction. As illustrated in the Airline case this will increase project control, but also increase the overall risk of the project. This indicates that a thoughtful IS project manager should aim to understand and manage the trade-off between necessary process interaction and the pressures related to internal project schedule and cost.

Thus, the concept of process interaction could also be used for managing risk in a setting where an IS development project should be coordinated with planned organisational change. As the Airline case illustrates, there is considerable risk associated with process interaction. These risks may be described with the interaction types described in section 4, as summarised in Table 2. The processes may be designed to interact, or the interaction may suffer breakdowns. The consequences of different trajectories may be assessed and different actions may be taken to prevent breakdowns.

Process interaction type	**Description**
Formal agreement on process interaction	There is mutual agreement that the two processes should interact
Breakdown of interaction	The interaction between the two processes is stopped
Process isolation	One of the processes is inactive isolating the other from interaction
Improvised interaction	The two processes interact in an ad-hoc manner
Organised interaction	The interaction is well structured and in operation

Table 3 Process interaction types identified in the Airline case

The list of process interaction types is not exhaustive and other types may well surface in other cases. On the other hand, these types provide an initial vocabulary to describe and understand process interaction. They provide relatively precise descriptions for determining the nature and strength of the interaction, thereby enabling description of process interaction with a finer granularity.

However, we do not claim that process interaction is relevant to all kinds of development projects. Sometimes the organisational change is trivial and needs a less structured process to interact with the IS development process. Conversely, sometimes the technology part is so small that there is no need for a structured IS development process. There also may be cases where the organisational change and the IS development may be executed in one single, fully integrated process. The evidence, however, for this last type of project is not convincing. Large-scale, integrated, socio-technical change is associated with considerable management challenges, being either wholly emergent over time (Orlikowski 1996) or dominated by technological drift (Ciborra 2000). On a smaller scale, though, successfully integrated projects are documented; see for example (Bardram 1996; Quereshi and Vogel 2000).

6.3. Contribution to Software Engineering Research: Supporting Socio-Technical Innovation

Our concept of process interaction builds on the socio-technical IS development tradition. It extends socio-technical research in the sense that it adds a stronger process perspective. We regard this as an extension of both ETHICS (Mumford 1985) and Multiview (Avison and Wood-Harper 1990; Avison et al. 1998). As argued in section 2 the socio-technical methodologies have a too strong focus on analysis and design, compared to development and implementation. With the concept of process interaction we shift attention towards the processes through which results are achieved. Socio-technical innovation is dependent on an iterative process of learning and building where the solution is the result of this process – not of a specification (Leonard-Barton 1988; Ciborra 1997). Our concept is a contribution to a socio-technical vocabulary to describe the interaction of these processes analysed through the critical events.

Building on this we think that our most important contribution is to show that iterative SE frameworks have an interesting potential for a more innovative approach in their interaction with business. As illustrated in the Airline case, the iterative structure of RUP is well suited to support process interaction – provided that the IS project manger takes advantage of this opportunity.

The software engineering frameworks, building on iterative and incremental principles of development processes (Jacobson et al. 1999; Stapleton 2003; Larman 2004) have improved software development considerably over the last years. However, the research on iterative and agile software development has a rather limited view on the interaction with the business organisation, as described in section 2. The Airline case shows that there is much more to interaction than user participation and prototyping. To work effectively in a socio-technical context, the software development process is dependent on interaction with the organisational change process. The key is to acknowledge that socio-technical innovation calls for a mutual adaptation between technology and organisation, and the main point of our argument is that the iterative structure of the SE frameworks is well suited for this. Thus, we agree with former arguments that the iterative approach provides an arena for exploring and learning (Jacobson et al. 1999; Stapleton 2003; Larman 2004). However, we think it is important to acknowledge that this learning arena is not restricted to software construction. The learning in IS development projects should also be fed back to the business organisation, and vice versa. The concept of process interaction supports this mutuality.

At this point we should be careful to add that process interaction also represents considerable challenges, as the Airline case illustrates. The first barrier is that there is often low awareness on the organisational change side of this need since software development projects are frequently seen as merely technical projects. The second barrier is that the structures of the two processes are incongruent. Most organisational change projects follow a waterfall model (French and Bell 1998) making frequent interaction much less desirable. Software engineering projects, on the other hand, are iterative and incremental in ways that would not make sense without frequent interaction.

The concept of process interaction is also an extension of the research on agile software development. We are in concordance with Giaglis (1999) in that the complexity of socio-technical change is too great for a single, integrated methodology. Our suggestion is, rather, that the key is *interaction* – not integration. The processes cannot be fully aligned during the project period but the interaction aspect can be added into the methodologies on both sides.

7. Conclusion

In this paper we introduced the concept of process interaction. Process interaction focuses attention on the meeting between the planned organisational change process and the IS development process. Our two research questions were:

- How can we *understand* process interaction, i.e., interaction between information systems development processes and planned organisational change processes?
- How can a project manager of iterative and incremental IS development processes *manage* process interaction?

Building on the socio-technical IS development tradition we analysed a longitudinal case from the airline carrier business. We found that the concept of process interaction enabled us to understand critical events in the case, in particular those that were important for the mutual adaptation between the information system and the organisation. Thus, to the first research question the suggested answer is that process interaction is needed to facilitate socio-technical innovation in a situation where the organisational change process and the IS development process are parallel but incongruent. We also argue that software engineering frameworks are well structured to support process interaction.

The answer to the second research question is that process interaction should be managed as a part of IS project management, to be integrated in planning and risk management. The need for process interaction is not well described in the normative IS project management literature, and we provide for the IS project manager an initial vocabulary to understand and manage this. An important point is that the IS project manager needs to manage the trade-off between necessary process interaction and the internal project schedule.

Limitations to our conclusions derive from the research approach. Longitudinal Process Research is aimed at developing contextualized theory. This suggests that the area of validity is the context of a large organisation with a concurrent IS development process and a planned organisational change process.

Our findings also imply a need for further research into the applicability of our concept where both existing and new cases could be analysed. A long-term vision would be to extend current software engineering frameworks and IS development methodologies to encompass the process interaction concept.

Acknowledgements

We thank the different stakeholders of the Airline project–for access to the project and for commenting on various versions of the case description. We would also like to acknowledge important comments from Lars Mathiassen and Bjørn Erik Munkvold.

References

Avison, D. and Wood-Harper, A. T. (1990). *Multiview*. London, Blackwell.

Avison, D., Wood-Harper, A. T., Vidgen, R. T. and Wood, J. R. G. (1998). "A Further Exploration in to Information Systems Development: the Evolution of Multiview 2." *Information Technology & People* 11(2): 124-139.

Bardram, J. (1996). "Organisational Prototyping: Adopting CSCW Applications in Organisations." *Scandinavian Journal of Information Systems* 8(1): 69-88.

Beck, K. (2000). *Extreme Programming Explained: Embrace Change*. Boston, Addison-Wesley.

Boehm, B. W. (2003). "Value-Based Software Engineering." *ACM SIGSOFT Software Engimeering Notes* 28(2): 1-12.

Bostrom, R. P. and Heinen, J. S. (1977a). "MIS Problems and Failures: A Socio-Technical Perspective, Part 1: The Causes." *MIS Quarterly* 1(3): 17-32.

Bostrom, R. P. and Heinen, J. S. (1977b). " MIS Problems and Failures: A Socio-Technical Perspective, Part II: The Application of Socio-Technical Theory." *MIS Quarterly* 1(4): 11-28.

Bygstad, B. and Munkvold, B. E. (2002). "Software Engineering and IS implementation research: An assessment of current SE frameworks". In (ed.) Kirikova, M.: *Information Systems Development. Advances in Methodologies, Components and Management*. New York, Kluwer Academic.

Cadle, J. and Yeates, D. (2004). *Project Management for Information Systems*. Harlow, Prentice-Hall.

Carr, N. (2003). "IT Doesn't Matter." *Harvard Business Review* May.

Christensen, G., Grønland, S. E. and Methlie, L. B. (1999). *Informasjonsteknologi*. Oslo, Cappelen.

Ciborra, C. (1997). "De profundis? Deconstructing the concept of strategic alignment." *Scandinavian Journal of Information Systems* 9(1): 67-82.

Ciborra, C. (2000). *From Control to Drift*. Oxford, Oxford University Press.

Doherty, N. F. and King, M. (2003). " From technical change to socio-technical change: towards a proactive approach to the treatment of organisational issues". In (ed.) Clarke, S., Coakes, E., Hunter, G. M. and Wenn, A.: *Socio-technical and human cognition elements of information systems*: 22-40.

Fitzgerald, B. (1998). "An empirical investigation into the adoption of systems development methodologies." *Information & Management* 34: 317-328.

French, W. L. and Bell, C. H. (1998). *Organization Development*. Englewood Cliffs, New Jersey, Prentice-Hall.

Giaglis, G. (1999). "On the integrated design and evaluation of business process and information systems." *Communications of the AIS* 2(1): 1-33.

Hammer, M. and Champy, J. (1993). *Reengineering the Corporation - A Manifesto for Business Revolution.* London., Nicholas Brealey Publishing.

Hanseth, O. and Monteiro, E. (1996). "Inscribing Behaviour in Information Infrastructure Standards." *Accounting, Management and Information Systems* 7:4: 183-211.

Hradesky, J. (1995). *TQM Handbook*, McGraw-Hill.

Jacobson, I., Booch, G. and Rumbaugh, R. (1999). *The Unified Software Development Process*. Reading, Addison Wesley.

Larman, C. (2004). *Agile and Iterative Development: A Manager's Guide*, Addison-Wesley.

Leonard-Barton, D. (1988). " Implementation as Mutual Adaptation of Technology and Organization." *Research Policy* 17:5: 251-267.

Luftman, J. and McLean, E. R. (2004). "Key Issues for IT Executives." *MIS Quarterly Executive* 3(2): 89-104.

Mathiassen, L., Munk-Madsen, A., Nielsen, P. A. and Stage, J. (2000). *Object-Oriented Analysis & Design*. Aalborg, Marko Publishers.

McManus, J. and Wood-Harper, T. (2003). *Information Systems Project Management*. Harlow, Prentice-Hall.

Mumford, E. (1985). *Sociotechnical Systems Design.* Manchester, Manchester Business School.

Newman, M. and Robey, D. (1992). "A social Process Model of User-Analyst Relationships." *MIS Quarterly,* 16(2): 249-266.

Ngwenyama, O. (1998). "Groupware, Social Action and Emergent Organizations: On the Process Dynamics of Computer Mediated Distributed Work."" *Accounting, Management and Information Technologies* 8(2-3): 127-146.

Orlikowski, W. (1996). "Improvising Organizational Transformation Over Time: A Situated Change Perspective." *Information Systems Research* 7(1): 63-92.

Pettigrew, A. M. (1985). "Contextual Research and the Study of Organizational Change Processes". In (ed.) Mumford, E.: *Research Methods in Information Systems*, North-Holland**:** 53-78.

Pettigrew, A. M. (1990). "Longitudinal Field Research on Change Theory and Practice." *Organization Science* 3: 267-292.

Quereshi, S. and Vogel, D. (2000). "Adaptive Processes for Achieving Socio-Technical Fit in Computer Supported Co-operative Work Groups". In (ed.) Coakes, E., Willis, D. and Lloyd-Jones, R.: *The New SocioTech. Graffiti on the Long Wall.* London, Springer.

Stapleton, J. (2003). *DSDM: Business Focused Development*, Addison-Wesley.