

2008

# Embedded System Development Process: A Quest from Innovation to Adaption

Wolfgang Molnar

*University of Warwick, w.a.molnar@warwick.ac.uk*

Joe Nandhakumar

*University of Warwick, joe.nandhakumar@wbs.ac.uk*

Follow this and additional works at: <http://aisel.aisnet.org/ecis2008>

## Recommended Citation

Molnar, Wolfgang and Nandhakumar, Joe, "Embedded System Development Process: A Quest from Innovation to Adaption" (2008). *ECIS 2008 Proceedings*. 193.

<http://aisel.aisnet.org/ecis2008/193>

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2008 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

# EMBEDDED SYSTEM DEVELOPMENT PROCESS: A QUEST FROM INNOVATION TO ADAPTION

Wolfgang Molnar, Warwick Business School, University of Warwick, Coventry CV4 7AL,  
United Kingdom, W.A.Molnar@warwick.ac.uk

Joe Nandhakumar, Warwick Business School, University of Warwick, Coventry CV4 7AL,  
United Kingdom, Joe.Nandhakumar@wbs.ac.uk

## Abstract

*This paper describes the findings of a case study that explores the improvisation and bricolage in a structured process management environment. The research focuses on software developers in an embedded system development project. This investigation adopts an interpretive approach, which involves the collection and analysis of qualitative data. The research offers insights into a range of dimensions, levels, and paradoxes of improvisation and bricolage that can be observed during a system development in a structured process environment. Furthermore, the paper addresses the question of how the structured process environment influences these dimensions, levels, and paradoxes. We observed various degrees of bricolage and improvisation with the movement of highly innovative practices towards a development phase of adaptive practices. This paper points out the value of structured processes as a scaffolding for improvisational and bricolage techniques and practices. The paper claims that within a turbulent environment, improvised and bricolage practices need pieces of structured processes reciprocity.*

*Keywords: Improvisation, Bricolage, Structured Process Management, System Development*

# 1 INTRODUCTION

Information system development activities mostly involve bricolage (Ciborra, 1996), which is defined as making do with the items or resources at hand (Levi-Strauss, 1966). Improvisational actions may often involve bricolage, but bricolage also happens when planning precedes execution (Baker, Miner and Eesley, 2006; Cunha, 2005; Moorman and Miner, 1998). Boccardelli and Magnusson (2006) state that these dynamic capabilities expose themselves as bricolage. In addition, many researchers portray patching as a strategic process to secure organizational reliability (Amabile, 1988; Cunha, 2005). Such studies have also contributed to this philosophical shift of organizational view in the complex system development environment (Ciborra, 2002; Orlikowski, 1995). Innes and Booher (1999) describe bricolage as a nonlinear, holistic approach to dealing with difficulty that results in some practical product. Improvisation and bricolage may be an applicable practice in turbulent environments (Cunha, 2005; Weick, 1998).

Improvisation is understood as the convergence of design and execution (Baker et al., 2006; Moorman and Miner, 1998). Weick claims that improvisation is about processes and designs that are continuously rebuilt (Weick, 1993). Improvisation in general is positively associated with accelerated product development in turbulent industries (Akgün and Lynn, 2002). Furthermore, improvisation seems to be an effective behavioural strategy for dealing with change, particularly under dynamic conditions (Hmieleski and Corbett, 2006). Brown and Eisenhardt (1997) examined continuously changing processes within technology oriented firms and found that those organizations which are most successful at change tend to have a greater capacity for improvisation. These organizations provide enough flexibility for their organizational members to influence their improvisation competencies, but are not so unstructured as to allow their operations to become unmanageable (Brown and Eisenhardt, 1997). Augier and Vendelø (1999) claim that improvisation can be considered as a rapid problem-solving technique that is used by organizations to cope with surprises. Furthermore improvisation is often stimulated by time-pressures (Augier and Vendelø, 1999).

Improvisation and bricolage are in contrast to the structured management practices and techniques, which are influenced by international standards like such as ISO 9000 (Benner and Tushman, 2002; Guler, Guillén and MacPherson, 2002). A growing number of organizations adapt these process management techniques. Rahman (2001) states that the implementation of ISO 9000 was found to improve customer satisfaction, helped to gain a competitive advantage as well as increased profitability and product and service quality. Furthermore, it is argued that a greater extent of process management activities in a firm results in a larger number of exploitative innovations (Benner and Tushman, 2002). Further, Benner and Tushman (2002) claimed that the greater the extent of process management in a firm, the larger the share of innovations that are highly exploitative. However, there are also some conflicting viewpoints concerning the benefits of structured process management. For instance, a common misconception is that ISO would entail higher levels of product quality (Motwani, Kumar and Cheng, 1996). Studies by Dick (2000) show, that the motive for seeking certification is an important factor for later performance of the implemented practices and techniques. Structured process management may discourage creative and critical thinking in an organization, because employees are forced to work according to well-described procedures and rules. Also, the attempt to gain the ISO 9000 certificate could incur extra costs, which would make this a “hollow achievement” (Singels, Ruël and van de Water, 2001). In addition, the physically engaged world of embedded software is different from the abstract way of thinking found in the non-embedded software literature (Lee, 2000). Therefore, design of embedded software has benefited less from the well-developed abstractions of software development processes and structured process management practices and techniques.

Our study investigated the problem of innovative development within the structured process management context of an ISO 9001 certified organization. To date limited studies have dealt with this issue. This paper aims to address this and focuses on the improvised and bricolage actions during an embedded system development. We develop a better understanding of various dimensions, levels, and

paradoxes of improvisation and bricolage that were observed during an embedded system development in a structured process environment. How does a structured process environment influence improvisation and bricolage in an embedded software development context? What are the implications for the structured process environment in such a context? This paper addresses these questions by investigating the reciprocal process of improvisation and bricolage on one side, and structured processes on the other side. We present our theoretical foundation in the next section. The third section will clarify our research methodology and will be followed by a case description and our analysis. This paper concludes with outlining some implications for theory and practice.

## 2 THEORETICAL FOUNDATIONS

While the core aim of this research is to derive a theoretical interpretation from the empirical data (Currie, 2004; Orlikowski, 1993), we also draw on other theories as sensitising devices (Walsham, 1993) to develop our theoretical interpretation. Furthermore, our theoretical foundation focuses on the differences of improvisation and bricolage in the respective time domain in which they occur. This helps us to avoid any misunderstanding of this phenomenon. While improvisation and bricolage are defined differently, nevertheless they are similar in character. Many scholars state that the convergence of design and execution is defined as improvisation (Moorman and Miner, 1998). As a result, bricolage may frequently occur during improvisation (Weick, 1998). However, bricolage could also occur during the implementation of pre-existing plans (Baker et al., 2006). Figure 1 depicts the overlap between the nature of bricolage and improvisation, which is the focus of this paper.

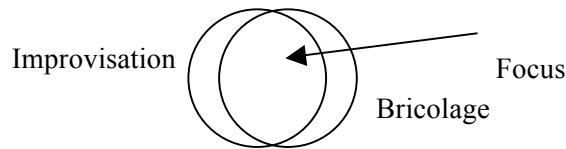


Figure 1. *Understanding of Improvisation and Bricolage*

Three dimensions of bricolage have been observed by Ciborra (1996). The first refers to tinkering or fiddling. It relates to the observation that organizational members need to fiddle or tinker in order to survive in an unpredictable environment. The second dimension highlights the importance of trial and error in an organizational context. Bricolage means that actions are not established patterns, but are results of implicit or explicit trial and error behaviour. However, this behaviour can be justified by experience. These thoughts give rise to the third dimension of bricolage – creation of an entity. An entity is a creative collage of resources, in ways they were not originally intended to be used. Ciborra emphasizes that all dimensions of bricolage are needed to gain a competitive advantage. Two distinct levels of bricolage were studied by Butor and Guynn (1994). First, the bricoleur uses tools at hand to tinker. These tools might be available via the usual channels and inhabit a vulgar aspect of bricolage. The other level is the gathering of objects for later use. This recycling is the recovery of what has been eliminated. The following improvisational paradoxes were examined by Mirvis (1998): Rehearsed spontaneity, anxious confidence, collective individuality, and planned serendipity. These paradox pairs seem to genuinely emerge through the creative interplay of contrary forces.

Weick (1998) claims that spontaneity and intuition are two important dimensions of improvisation. The spontaneous facet of improvisation was studied by Vera and Crossan (2005). In addition, Weick (1998) describes various instances of improvisation. Activities that alter, revise, create, and discover are purer instances of improvisation, because they are manifestations of the full improvisational spectrum. These kinds of changes occur smoothly. In contrast, activities that shift, switch, or add are weaker instances of improvisation, as a result of solitary improvisation. Rather, change occurs suddenly. In addition to this classification scheme, Weick (1998) claims that improvisation is affected by one's organizational members, previous experiences, current settings, and the kernel that provides the pretext for assembling these elements in the first place. These pretexts are not neutral and they support some lines of development and exclude other ones (Weick, 1998). In addition to these

findings, the full spectrum of the continuum was examined. Orlikowski (1995) describes the everyday improvisations and slippages, the incremental improvisation, and the radical improvisation within this continuum. Furthermore, Dybå (2000) distinguishes between explorative and exploitative improvisational actions. The explorative improvisation is more a search for new knowledge, either through imitation or innovation, whereas the exploitative improvisation is the adoption and use of existing knowledge and experience. Finally, the researchers therefore describe various tiers of bricolage and improvisation. Table 1 lists the various instances of bricolage and improvisation. We particularly focus on the tier concept bricolage and Mirvis' improvisation paradoxes in this paper. The reasons for doing so are the similarities between the frameworks of improvisation and the integrity of Mirvis' theoretical work.

Theme	Aspect	Author
Bricolage	Tinker or fiddle	(Ciborra, 1996) and (Butor and Guynn, 1994)
	Trial and error	(Ciborra, 1996)
	Entity	
	Recycle	(Butor and Guynn, 1994)
Improvisation	Rehearsed spontaneity	(Mirvis, 1998)
	Anxious confidence	
	Collective individuality	
	Planned serendipity	

*Table 1. Tiers of improvisation and bricolage*

Orlikowski (1995) implied that the focus on modification as situated offers an approach of seeing that change may not always be as structured, planned, or predictable as people tend to imagine. Rather, it is often understood through the fragmentary variations which emerge in the bricolage, drifts, and improvisations of everyday activity (Ciborra, 2000; Orlikowski, 1995). Drawing on Giddens (1986) we consider organizational members as reflexively monitoring their everyday actions, that of others and the contexts of social activity. This reflexive monitoring, rationalisation, and motivation of action are processes which occur inside the organizational members' head and are maintained, enacted and replicated endlessly (until change appears) by the organizational members (Giddens, 1986). With this understanding we derive practical implications for the structured process management of organizations and induce our discussion.

### **3 RESEARCH APPROACH**

This research was carried out in an Engineering Company (EC – a pseudonym) in Germany. The research approach adopted in this study is an interpretive case study (Orlikowski and Baroudi, 1991; Yin, 1994). This involved a collection of detailed, qualitative data concerning the context and processes of software developers' actions and practices (Avison and Nandhakumar, 1995; Currie, 2004; Walsham, 1993). EC is an ISO certified organization and the study was conducted during the process of an embedded system development. We conducted 21 interviews (1 structured interview, 2 group interviews, and 18 semi-structured interviews) and carried out observation of the software development team (description in section 4.1) for six months. This was followed by several informal visits and discussions between February 2005 and January 2006. The goal was to try to understand how the development team handled their system development process. In order to encourage participation the interviewees were assured that their statements would remain anonymous and that they would not be explicitly linked to their company or product. Detailed notes were taken of the interviews and the observations of work practices that were made during each visit. All interviews were held in a relaxed climate that was representative of the informal way of life within EC.

In addition, company documents were examined to gain more insights into the processes and practices observed and to verify interview notes. These field notes from observations, interview transcripts and

company documents were read and reread to familiarise with the data and identify the significant events and incidents. This analysis approach is based on Miles and Huberman (1994) and is named three tier coding. First, descriptive coding is necessary to get an understanding of the data load. The first action was to deposit data with some meaning and hereby to expose the various events and incidents. Interpretive coding is the second phase of the three tier coding approach. It is the attempt to attach meaning to the data and to put it into context to clarify the significance. Furthermore, it is possible to realize social structures and aspects of structured process management that permit, hamper and describe the participants' actions. Third, pattern coding is the final step to understand the various meanings in context by applying patterns. This helped us to sort data on the basis of their underlying concept rather than by the source of information. This helped us to understand the phenomena better (various instances of improvisation and bricolage; influence of structured process management techniques and practices toward improvisation and bricolage – vice versa) through the meanings that team members attached to their context and processes. As a result, we were able to enforce a bottom up conceptualization of the collected and analyzed data whilst using the theoretical concepts as sensitising devices.

## **4 CASE DESCRIPTION**

### **4.1 Research Site Context**

The Engineering Company (EC – a pseudonym) is within the automobile sector in Germany. The automobile sector is an important pillar of the German economy. In addition to the large auto manufacturers, there is a supplier industry that is also of great international importance. Suppliers' innovations are shown at international automobile events and exhibitions. Its reputation critically relies on creative high quality innovations. The EC is comprised of several departments, whereas the service supplier and the development department are the main departments. The service supplier organizational members' work is embedded within the customers of the automobile sector. The organizational members of the development department work at the EC offices. Its customers demand high-tech products of high quality. Therefore, quality is an important characteristic for the seven-year-old EC. The organization has seen significant growth over the past couple of years. Starting with only four people, the size of the current organisation is around 170 people. Due to this tremendous growth the entire environment both within and outside the organization has changed. Internal communication paths have lengthened in that decisions need more time, and more people are involved in certain decisions. The way to communicate with customers has changed as well. As their market power and experience strengthened, more and more customers noticed the EC. One effort to bring more structure into the organization was to establish the ISO 9001 framework. The EC has been an ISO 9001 certified organization for 4 years. This certificate shows that EC established a working process management framework for the entire organization. The certificate itself is provided by independent organizations which are able to measure and evaluate process management frameworks of organizations. The EC has a management hierarchy with four layers in the development department: senior management, middle-management, project leaders and associates. This research largely focused on the work practices of the middle-manager, project manager and the associates within the development department. These were the major organizational performers involved in this new product development.

Michael, senior manager and co-founder of the EC, was the director of the hardware and software development. Gabriel (middle-manager – software) was responsible for software development within EC. This organization operates at two locations within Germany. Martin was the project manager of this new product development team. During the initial phases of development he spent half of his time as a service supplier with the customer and the other half with the organizational members. With this split of responsibility the EC was able to have first-hand contact with its customers. Raphael's role in this project was of critical importance. The EC organizational members portrayed him as a 'fire-

fighter', which is a person who comes to help and solve problems. There were seven other developers, each with specific responsibilities. David's role was to establish a structured process management framework within the organization.

#### 4.2 From origin of the concept to a final product

This study focused on the processes and context during the development of a new embedded system at EC. This product was an electronic device (based on Linux) that used software to connect to a computer for programming and data readout (Windows Client-Software). The product could be called an Event Data Recorder for Automotive-Infotainment Development (EDR-AID). This device could be described as a new innovation. It caused changes to the immediate business environment in technology as well as marketing.

We observed a transformation of the internal processes at the beginning of the system development. Martin, the project leader of this system development worked previously full time as a service supplier. Similar problems constrained his work and the work of his colleagues. As a result, the conceptual idea of the EDR-AID was born. However, this conceptual idea needed to be developed into a list of requirements. Parts of the requirement list were weak in detail and were specified during the development phase of the embedded system. Developers mentioned that early phases of the development were constrained through uncertainties of requirements. In addition, Gabriel (middle manager) claimed: "During the development of the EDR-AID we had to follow a moving target. Apart from the constraints on budget and human resources, unclear descriptions of the product requirements hindered straight-forward development." There was no working prototype to validate the design concept and to show capability of the device to the customer.

As a result of these circumstances the company faced some challenges for the development of the embedded system. Although some features of EDR-AID were well known to EC, there was one new technology, which had to be reengineered. Moreover, this critical part was shown to be working with the proof of concept prototype. This prototype was essential to show the customer the capability to achieve success with this development. Soon after the start of the EDR-AID development, other departments of the customer were aware of the potential of this device. As a result, the demanding attitude of the costumer increased. Initially the EDR-AID should have been produced in small numbers at a maximum of 90 devices. However, due to the request of the customer the number of total devices was increased to more than 250 within the first year. In addition, another clientele of customers asked for EDR-AID devices. This demand caused an obvious success in marketing. However, on the other hand the development was also constrained by this demand. As potential buyers had different requests this limited the development

Martin stated: "We never planned this as a giant product for EC. But we were lucky to be able to close a gap in our business field. Without the EDR-AID the EC would be totally different. The EDR-AID changed the EC and our business field." In addition, Michael mentioned: "The EDR-AID was not planned to be a large series product. So we talk now about 800 devices and we thought originally that we would produce a maximum of 80 or 90 devices. So, that's a huge difference, not only in the development, marketing, and production, but also in the maintenance." The fundamental assumption of producing a small series was shown to be wrong. The product has become a mass product instead of an easily controllable amount of devices. Besides the sheer amount of devices, the processes of maintainability needed to grow within the organization. The EDR-AID was becoming a mass product with a totally different market potential than initially intended. This progress affected the way EC was doing business. For example, Michael stated, "We needed to adapt our processes. We established a new service department and in-sourced some of the production. So it could be said, that the EDR-AID required some reconsiderations." Gabriel also claimed: "We had to put more emphasis on device testing. Every developer was used to test her or his part of the software. This will be reasonable if the device itself is easily manageable. We learned with the EDR-AID, that the usual way of testing was not useful anymore. So we needed to move forward in our way of thinking."

#### 4.3 Instances of Improvisation and Bricolage

We observed various instances of improvisation and bricolage during the development process. First, the research participants faced a phase of challenges and high risks. Raphael and Michael had some idea of realizing an important feature of the embedded system. In a defined system-environment they were able to trace the original software protocol and created a workable protocol for their purposes. This could be realized through the individual efforts of Raphael. In addition, Michael stated, “The new feature was a vital issue for the breakthrough of our 32-Bit platform. We <developers> had the hardware and the capability, we just needed to put it together and write some code.” Besides this new technology, the entire hardware platform was challenging. EC leapt from a 16-Bit single-processor platform to a 32-Bit multi-processor platform. This technological leap created some complications during the development. Besides the occurring inability to detect an error in the software or hardware, the complexity of the EDR-AID created some difficulties. Almost all developers focussed on their individual tasks with no holistic understanding. Only Raphael and the managers claimed to be aware of the complete architecture of the embedded device. This weak understanding of how to combine hardware and software caused problems during bug fixing, as developers mentioned.

Second, after the challenges during the initial development phase of the EDR-AID, the focus was on the system development. As a result, their way of programming changed. The developers stated that they had more time for looking at potential side-effects of changes. In addition they claimed to fix their hacks properly with some rudimentary documentation. Finally, the product maintenance phase resulted in bug-fixes and changes in the EDR-AID system. The hardware and software key features were settled, and therefore the development path was dependent upon its previous architectural outcomes.

#### 4.4 Structured Process Management

The EC processes were based on the ISO 9001 certification. However, there were various opinions on why the organization had ISO. There were basically two camps. Higher up the hierarchical ladder within the company the prevalent opinion was that internal motivation was the major reason why EC introduced ISO. In contrast, organizational members claimed that constraints around the EC were the reason for introducing the ISO at the EC. Developers voiced serious doubts about the practical help of ISO in their daily tasks. For example, one of the developers claimed, “Within a good company, ISO is not necessary!” “Individual motivation and common sense for quality is of greater value than ISO!” “ISO raises bureaucracy but is necessary for the EC products.” David (quality manager) mentioned distrust about the effectiveness of ISO with the EC developers. He stated that he did not appreciate the working style of people like some software developers, who perform only poor documentation. However, David mentioned also some uneasiness about the introduced ‘shallow quality management’. Besides this rudimentary established framework of ISO 9001 there are other pieces of structured process management. The use of SubVersion itself provides help around the management of various pieces of software during the development and maintainability process. Furthermore, the EC uses Bugzilla for the management of occurring bugs. In addition to these tools, Martin (project manager) introduced weekly meetings. In this particular meeting the responsible department managers discuss the possible future of the EDR-AID. This ‘change-control-board’ has been necessary to keep the various requests of future developments under control. According to Martin, this meeting improved the internal communication flow.



## 5 ANALYSIS

### 5.1 Improvisation and Bricolage

From the analysis of case study, we were able to identify three different instances of improvisation and bricolage during the development of an embedded system: breakthrough, innovative practices, and adaptive practices. First, a breakthrough occurred in one feature technology of the EDR-AID. We describe the development of this feature technology of the EDR-AID as a breakthrough of known boundaries, because this feature technology was new for everyone outside the original developer of this feature. However, since this technology was re-engineered, we still claim this as an instance of bricolage. This re-engineering constituted a disclosure of a secret software protocol of an existing technology. The developers relied on tinkering, which is a dimension of bricolage (Ciborra, 1996), as described above. In addition, the tinkering is more evident as the recycling during the breakthrough phase (Butor and Guynn, 1994). The EC developers used the material and tools available to them (recycling), without buying the software library of the original developer of this feature technology. In addition, Mirvis' (1998) improvisation characteristics (rehearsed spontaneity, anxious confidence, collective individuality, planned serendipity) were obvious to a large degree. For example, through previous training and experiences (school and other tasks), developers were prepared to be able to compete in a shifting environment (rehearsed spontaneity). The management hoped to overcome the uncertainties of this re-engineered protocol. Martin was confident about the skills of the developers, paired with some fear to neutralize problems. Raphael's role was critical in this breakthrough phase. His individual efforts cleared the way for other developers. The idea to pursue until the breakthrough shows some planned serendipity. Finally, it is interesting to note that the duration of this phase was about six months of four people within EC.

Second, a highly innovative output occurred early in the development. Innovative behaviour exhibited a great degree of flexible, unconventional and creative solutions. After the breakthrough of the prototype stage of the EDR-AID development, there was still an open field for EC developers available to be highly innovative. Improvisation was routine in this context, so that we were able to identify Ciborra's (1996) dimensions and the different levels of bricolage (Butor and Guynn, 1994). However, the degree of dimensions and levels was significantly different to the previous breakthrough period. The EDR-AID development concentrated on the requested features, which were still a notable challenge. Despite of this, fundamental breakthroughs could no longer have been expected. Nonetheless, the initial stage of the EDR-AID was an innovative period with bricolage action. Many claim that improvisation aims to be a harmonic synthesis of creativity, adaptation and innovation (Cunha, Cunha and Kamoche, 1999). Other than this ideal we observed strengths in the innovation degree in contrast to the adaptation ingredient during this development stage. In addition, we observed distortions of Mirvis' (1998) improvisation paradoxes. The importance of previous rehearsal of development activities grew. In contrast, the effects of spontaneity weakened during the post-breakthrough phase. There is a clear change in the confidence of managers towards developers and previous uncertainties were overcome which means that reliance of the outstanding deeds (accomplishments) of Raphael was no longer needed. An important factor during this time was '*Teamgeist*' (German: team spirit). After the breakthrough, less and less was left to chance. More planning and less serendipity was observed during this period. Yet, this phase took another six months with seven people within EC.

Third, a growing adaptive style arose during the development of the EDR-AID. Adaptive behaviour showed a high amount of rigid, conventional and uniform unfolding. Furthermore, we observed a third change in the improvisation and bricolage action. After the major development stage the EC re-adjusted its focus to maintainability. As a result, adaptive skills are far more important as in previous development stages. In the literature the adaptor is described as a person who has a preference for "doing things better" (Kirton, 1989). And this is what the EDR-AID developers mentioned to keep

control of their product. In addition, their possibilities to act were limited by the path dependence of their previous development stages of this product. So, the EC developers' way of doing was bricolage action (hacking and patching) with the emphasis on adaptive tasks. Eisenhardt and Tabrizi (1995) characterize this adaptation process experiential. In addition, we characterise this stage as being also full of the various dimensions and levels of bricolage (Butor and Guynn, 1994; Ciborra, 1996). However, the sum of bricolage action was significantly lower than in previous development stages. Nevertheless, the recycling aspect of bricolage was found to be higher than during the other phases. In reference to Mirvis' (1998) improvisation paradoxes, we found a strong distortion. Similar to the innovative post-breakthrough period, we find a clear representation of practiced rehearsal, confidence, 'Teamgeist', and planning. At last, this phase took more than 30 months and when time moved on the organization hired new developers and the EDR-AID developer team grew to 19 people.

In addition to these findings we observed a day-to-day improvisation and bricolage in the use of structures. With these structures we classify procedure templates like SubVersion, Bugzilla, and regular meetings. These templates were used intensely and as a result, their doings were restricted to the capabilities of the inhabited structures. So their daily improvisation and bricolage were composed of acting, to meet the requirements of the procedural templates. We highlight here the developers' use of Bugzilla as a mailbox for several issues. Besides errors they used Bugzilla as a programme to report special incidents. These incidents were a result of unclear requirement specifications and side-effects during the development. Our analysis draws attention to the need to differentiate our focus of improvisation and bricolage actions during embedded system development. On one hand it is purposeful to steer the analysis of the information system development. On the other hand it is useful to investigate the use of procedural templates, like database programmes and methods of personal interaction.

## 5.2 Reciprocal influences

In order to ascertain the role of structured process management we explore the linkage between structured process management and improvisation and bricolage during embedded system development. We identified the use of certain software tools combined with the associated processes for users as structured process management in this context. Especially two software tools and processes were shown to have positive results in managing the processes of the EDR-AID developers systematically. SubVersion is a version control system, which helps the developers to keep track of the multiple modifications made over time. The second tool is called Bugzilla, which is used to track bugs in the EDR-AID. Both software tools provided various processes for the users. These processes were inherited by the EC developers. First, we analyse the influential factors of structured process management on improvisation and bricolage. In this case, we observed that the motive for seeking certification was weak in general. However, higher up in the hierarchy the awareness grew of the positive contributions of a certified structured process management framework. Studies by Dick (2000) show, that the motive for seeking certification is an important factor for later performance of the implemented practices and techniques. As a result of the weak motivation of the developers, the direct influence of ISO onto developer' work on the embedded system was fragile. Software tools like SubVersion and Bugzilla influenced the developers' action significantly more than ISO, because of their daily interactions with these software tools. The use of software tools and processes and regular meetings were central to the developers' day-to-day activities, especially during the later stages of development.

Second, we investigate the effects of improvisation and bricolage on the structured processes. During the embedded system development, we observed that innovative improvisation and bricolage actions were more dominant over structured processes. However, at a later stage of the EDR-AID development, adaptive improvisation and bricolage practices were less important than structured processes. We highlight the early phase of the EDR-AID development, when Bugzilla was not really used, in contrast to the maintenance phase when Bugzilla was used intensively. In many cases,

developers reported various incidents (not errors) on their bug-tracking tool. As a result, this tool is becoming an incident reporter alongside its original purpose. Through continuous use, software tools and processes are transformed and established in their given context as a result of improvisation and bricolage actions. So improvisational and bricolage actions and the structured process management tools in use are mutually shaping each other. This usage of tools for improvisatory and bricolage actions helped to instantiate them and sometimes helped to modify or transform them in their use. In doing so, the use of tools made it more acceptable and they became established in the organizational context as acceptable practices, and then they became part of acceptable routine practices with each subsequent use.

## 6 DISCUSSION AND IMPLICATIONS

We have seen in our analysis that the degree of improvisation and bricolage varies during embedded system development. We analyzed different instances of improvisation and bricolage during the embedded development process. The analysis indicates that the early breakthrough phase caused a focus on creating a prototype. After the prototype stage, there were improvisation and bricolage actions with a tendency towards elaborating innovative solutions. This innovative phase was followed by an adaptive phase. The focal point moved to a series development of an embedded system with special reference to maintenance and service issues. These non-technical differences caused a transformation of the EC developers' way of thinking. Apart from this cultural change of symbols, we also analyzed a political shift of the control metaphor. Through different degrees of institutionalizations of the structured processes it was shown, that the EC gained more influence over their internal communication paths. Here we highlight the introduction of the weekly meeting of the content changes of the EDR-AID development.

In the analysis we explored theoretically and empirically the high degree of improvisation and bricolage actions in response to a range of constraints during the information system development process. These contrasting aspects of the EC developers are linked through their day-to-day enactment of their working life. This reproduction of actions and structure led to consequences, which were often unintended (although some were intended). Some of these replicated practices of improvisation and bricolage became acceptable practices. This tolerance created new methodological techniques and practices which were used according to a situation. In addition, the legitimization of this methodological practices was founded in the established structured process management of EC. The ISO certificate legitimized some techniques and practices, as well as prevented the organizational members from doing something that cannot be legitimized. Improvisational and bricolage actions were continuously reflexive monitored, rationalized, and motivations of action were sought according to the existing structured process management of ISO. Either way, the consequences influenced further activities of the organization. This reciprocity links the social context and social process in this study.

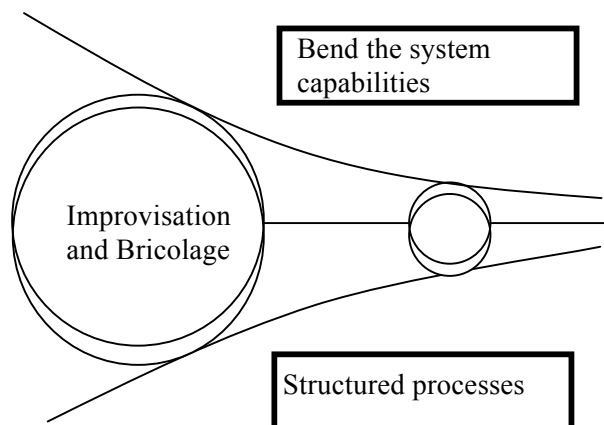


Figure 2. *Influence of structured processes on improvisation and bricolage*

Based on the analysis of the EC case study, figure 2 depicts the influence of structured processes on improvisation and bricolage. The complexity of this situation is that the developers moved from a predominantly innovative character to a more adaptive character of actions and practices and this is shown in figure 2 as a funnelling process. The change was influenced by various pieces of structured processes. These structured techniques and practices had a growing impact on the adaptive character of system development. These findings improve the available knowledge and add awareness of connection between innovative improvisation and bricolage and adaptive improvisation and bricolage in the context of structured processes. In addition, the technical limitations constrained the developers' actions of improvisation and bricolage. Therefore, we see it as essential for organizations within a turbulent and unstable environment to become conscious of a movement between innovative and adaptive characteristic of improvisation and bricolage. Structured processes heavily influence this movement. For practitioners this means that structural fragments need to grow to support the system development and to provide some structured process management. Brown and Eisenhardt (1995) claim that maintaining flexibility and learning quickly through improvisation and experience yield effective process performance.

## 7 CONCLUSION

This paper investigated the degrees, levels, and paradoxes of bricolage and improvisational activities in a structured process management environment. From the perspective of the software developers we tried to understand their interpretation of actions and perceptions. The research provided insights into the interaction of improvisational and bricolage actions and instances of structured processes. We claim that within a turbulent environment this interchange is an elementary factor of successful system development. Similar to Orlikowski's (1995) research on improvisation, we found that small adaptations led to considerable organizational change through continuous reproduction of practices. The contribution of this paper is the development of conceptualization that provides the reciprocal interdependence of bricolage along with improvisational activities and pieces of structured processes during an embedded software development process. We emphasize that some elements of structured processes may be necessary, for example, structured processes may serve as a scaffolding for improvisational and bricolage techniques and practices.

## References

- Akgün, A. E. and Lynn, G. S. (2002) *New product development team improvisation and speed-to-market: an extended model*, European Journal of Innovation Management, 5 (3), pp. 117-129.
- Amabile, T. (1988) *A model of creativity and innovation in organizations*, Research in organizational behavior, 10 123-167.
- Augier, M. and Vendelø, M. T. (1999) *Networks, cognition and management of tacit knowledge*, Journal of Knowledge Management, 3 (4), pp. 252-261.
- Avison, D. and Nandhakumar, J. (1995) *The Discipline of Information Systems: Let Many Flowers Bloom!*, IFIP Conference.
- Baker, T., Miner, A. S. and Eesley, D. T. (2006) *Improvising firms: bricolage, account giving and improvisational competencies in the founding process*, Research Policy, 32 255-276.
- Benner, M. J. and Tushman, M. (2002) *Process Management and Technological Innovation: A Longitudinal Study of the Photography and Paint Industries*, ASQ, 47 676-706.
- Boccardelli, P. and Magnusson, M. G. (2006) *Dynamic Capabilities in Early-Phase Entrepreneurship*, Knowledge and Process Management, 13 (3), pp. 162-174.
- Brown, S. L. and Eisenhardt, K. M. (1995) *Product Development: Past Research, Present Findings, and Future Directions*, The Academy of Management Review, 20 (2), pp. 343-378.
- Brown, S. L. and Eisenhardt, K. M. (1997) *The Art of Continuous Change: Linking Complexity Theory and Time-Paced Evolution in Relentlessly Shifting Organizations*, ASQ, 42 (1), pp. 1-34.

- Butor, M. and Guynn, N. (1994) *Bricolage: An Interview With Michel Butor*, Yale French Studies, 84 17-26.
- Ciborra, C. (1996) *The Platform Organization: Recombining Strategies, Structures, and Surprises*, Organization Science, 7 (2), pp. 103-118.
- Ciborra, C. (2000) *From Control to Drift*, Oxford University Press, Oxford, UK.
- Ciborra, C. (2002) *The Labyrinths of information*, Oxford University Press, Oxford.
- Cunha, M. P. (2005) *Bricolage in Organization*, FUENL Working Paper Universidad Nova de Lisboa.
- Cunha, M. P., Cunha, J. V. and Kamoche, K. (1999) *Organizational improvisation: what, when, how and why*, International Journal of Management Reviews, 1 (3), pp. 299-341.
- Currie, W. L. (2004) *The organizing vision of application service provision: a process-oriented analysis*, Information and Organization, 14 237-267.
- Dick, G. P. (2000) *ISO 9000 certification benefits, reality or myth?*, The TQM Magazine, 12 (6), pp. 365-371.
- Dybå, T. (2000) *Improvisation in Small Software Organizations*, Software, IEEE, 17 82-87.
- Eisenhardt, K. M. and Tabrizi, B. N. (1995) *Accelerating Adaptive Processes: Product Innovation in the Global Computer Industry*, ASQ, 40 (1), pp. 84-110.
- Giddens, A. (1986) *The Constitution of Society*, Polity Press, Cambridge.
- Guler, I., Guillén, M. and MacPherson, J. M. (2002) *Global Competition, Institutions, and the diffusion of organizational practices: The international spread of ISO 9000 quality certificates*, ASQ, 47 (2), pp. 207-232.
- Hmieleski, K. M. and Corbett, A. C. (2006) *Proclivity for Improvisation as a Predictor of Entrepreneurial Intentions*, Journal of Small Business Management, 44 (1), pp. 45-63.
- Innes, J. E. and Booher, D. E. (1999) *Consensus Building as Role Playing and Bricolage*, Journal of the American Planning Association, 65 (1), pp. 9-26.
- Kirton, M. (1989) *A theory of cognitive style*, In *Adaptors and Innovators styles of creativity and problem solving* (Ed, Kirton, M.) Routledge, London, UK.
- Lee, E. (2000) *What's ahead for embedded software*, Computer, 33 (9), pp. 18-26.
- Levi-Strauss, C. (1966) *The savage mind*, University of Chicago Press, Chicago.
- Miles, M. and Huberman, M. (1994) *Qualitative Data Analysis*, Sage Publications, Thousand Oaks.
- Mirvis, P. H. (1998) *Practice Improvisation*, Organization Science, 9 (5), pp. 586-592.
- Moorman, C. and Miner, A. S. (1998) *Organizational Improvisation and Organizational Memory*, The Academy of Management Review, 23 (4), pp. 698-723.
- Motwani, J., Kumar, A. and Cheng, C. H. (1996) *A roadmap to implementing ISO 9000*, International Journal of Quality & Reliability Management, 13 (1), pp. 72-83.
- Orlikowski, W. J. (1993) *CASE Tools as Organizational Change: Investigating Incremental and Radical Changes in Systems Development*, MIS Quarterly, 17 309-340.
- Orlikowski, W. J. (1995) *Improvising Organizational Transformation Over Time: A Situated Change Perspective*, Working paper; 3865-95 Sloan School of Management, MIT, Cambridge, MA.
- Orlikowski, W. J. and Baroudi, J. J. (1991) *Studying Information Technology in Organizations: Research Approaches and Assumptions*, Information Systems Research, 2 (1), pp. 1-28.
- Rahman, S. (2001) *A comparative study of TQM practice and organisational performance of SMEs with and without ISO 9000 certification*, International Journal of Quality & Reliability Management, 18 35-49.
- Singels, J., Ruël, G. and van de Water, H. (2001) *ISO 9000 series – Certification and performance*, International Journal of Quality and Reliability Management, 18 62-75.
- Vera, D. and Crossan, M. (2005) *Improvisation and Innovative Performance in Teams*, Organization Science, 16 (3), pp. 203-224.
- Walsham, G. (1993) *Interpreting Information Systems in Organizations*, Wiley, Chichester.
- Weick, K. (1993) *Organizational redesign as improvisation*, In *Organizational Change and Redesign* (Ed, Huber, H. P. a. G., W. H.) Oxford University Press, New York, NY.
- Weick, K. (1998) *Improvisation as a Mindset for Organizational Analysis*, Organization Science, 9 (5), pp. 543-555.
- Yin, R. K. (1994) *Case Study Research: Design and Methods*, Sage Publications, Thousand Oaks, CA.