Bondarouk, T., & Sikkel, K. (2005). The role of Group Learning in implementation of a personnel management system at a hospital. In: A. Sarmento (Eds.), ***Issues on computer human interactions***, (pp. 334 – 361), Idea Group Publishing, Hershey, PA.

THE ROLE OF GROUP LEARNING IN IMPLEMENTATION OF A PERSONNEL MANAGEMENT SYSTEM IN A HOSPITAL

Tatyana Bondarouk

Faculty of Technology and Management

University of Twente,

PO Box 217, 7500 AE Enschede,

the Netherlands

phone: +31-53-4893666

fax: +31-53-4892159

t.bondarouk@utwente.nl

Klaas Sikkel

Faculty of Computer Science

University of Twente,

PO Box 217, 7500AE Enschede,

the Netherlands

phone: +31-53-4894003

fax: +31-53-4892927

k.sikkel@utwente.nl

THE ROLE OF GROUP LEARNING IN IMPLEMENTATION OF A PERSONNEL MANAGEMENT SYSTEM IN A HOSPITAL

ABSTRACT

A new HR system was introduced in a Dutch hospital. The system implied collaborative work among its users. The project planning seemed to be reasonably straightforward: the system's introduction was intended to take place gradually, including pilots in different departments and appropriate feedback. After some time, the system was successfully adopted by one group of users, but failed with another.

We conceptualize the implementation process of groupware as group learning to frame the adoption of the system, and analyze the qualitative data collected during the longitudinal case study. We found that in the user group with strong group learning, adoption of the system occurred effectively and on time. In another user group with rather weak group learning, the use of the system was blocked after a short time. The results provided a first confirmation of our assumption about the importance of group learning processes in the implementation of groupware.

Keywords

IT Management, Human Resources, Job Design, Groupware

INTRODUCTION

It is broadly recognized that the IT use often develops differently from the expected plans, and that the degree, to which use of technology corresponds to the anticipated rules and norms, can vary a lot, depending on an organizational context, type of IT, end-users' awareness of the system, etc. (Bardram, 1998; DeSanctis and Poole, 1994; Orlikowski, 1996).

Different research perspectives have developed their views on this issue in parallel. Orlikowski (2000) gives the following examples. Social constructivists refer to the IT "inscription", analysing further the role of debates, social interests, and conflicts in achieving a consensus in IT functioning (Akrich, 1992). Structurational traditions examine how technologies develop through the interplay between "embodied" and "embedded" structures (Orlikowski, 1992). Similarly, adaptive structuration theory focuses on the differences between "faithful" appropriation (use in line with IT intention) and "unfaithful" appropriation (actual use) (DeSanctis and Poole, 1994). Developing the structurational concepts, Orlikowski (1996) talks about "institutional" (prescribed) and "on-going", "enacted", or "situated" use of technology. All these views start with characteristics of technology, and analyze how those are used, appropriated, accepted, or adapted by the targeted employees.

Rather then starting with technology and examining how people appropriate, adapt, or accept it, we shall start with the employees and explore how they develop their work with the system. Whether through mistakes, or purposefully, users often ignore, alter or play around the "anticipated', "inscribed", and "institutionalised" technological characteristics. Even if technology is given and its use is mandatory, employees will influence their recurrent work with it through developing certain interpretive schemes like making preferences, new rules of the work being automated, new tasks facilities, norms (e.g. traffic regulation), interpersonal interaction via IT, modification of technological properties, choosing or ignoring optional properties, inventing new ones, etc. (Orlikowski, 2000).

Various studies have applied this to introduction of collaborative technologies, also known as groupware, which are intended to support interdependent tasks (e.g. Bikson and Eveland, 1996; Orlikowski 1996). Engagement of different employees in a common task through the system in fact decreases technological malleability. Anticipating any technological changes involves negotiations of all users if their interdependence is based on the functionality of IT. In other words, employees probably will look for a community consensus in order to develop interpretive schemes to work with the technology together, within given or created interdependent tasks. As a result, implementation of groupware may have “drifted” (Ciborra, 1996) from its intended use because of those negotiation processes that result in new collaborative interpretive schemes.

The question arises as to how group interactional processes are related to the adoption of groupware. Some interactional processes that influence IT adoption are emphasized in the literature: reflective group processes (Tucker et al., 2001; Hettinga 2002); sharing understanding (Mulder et al., 2002); collaborative knowledge building (Stahl, 2000).

We propose to look closer at groupware implementation from a learning-oriented approach, which focuses on the group interactional process as the core factor in adopting a new system.

Why would we want to consider a collaborative technology implementation process as a learning-oriented process?

- User groups adapt a novel way of working when a new technology is introduced. Not all groups do this in the same manner, and this adoption process, called appropriation (DeSanctis and Poole, 1994; Ruel, 2001) depends on the group processes. The terms in which one describes the appropriation process – sharing understanding, mutual adjustment – are closely related to learning theory.
- Changes in technology do not only allow more effective ways of doing the same work, but, in addition, lead to changes in various aspects of professional competency such as knowledge, skills, and attitudes. That, in turn, could influence on-going use of technology. Hence, in theory, there is an on-going evolutionary process of professional and technological development.
- While using collaborative technology in practical situations, user groups gradually discover the affordances provided by the system and come up with new, unforeseen ways of

working. We believe that lots could be gained from collaborative technology if users exploit their group learning potential to a large extent.

- In several accounts of case studies, the implementation process did not take place in an optimal way, and the cause of this has been attributed to a lack of reflective restructuring among the users. (Tucker et al., 2001; Hettinga and Schippers, 2001)

In the next section we present a theoretical framework for groupware implementation based on collaborative learning. Then we apply the framework to a longitudinal case study, involving implementation of the same system with two different users groups. The differences in success of the implementation processes can, at least in part, be attributed to the different learning processes involved. Finally we conclude that the case gives a first validation of the proposed framework.

GROUP LEARNING AS A FOCUS FOR GROUPWARE IMPLEMENTATION

The learning-based framework for groupware implementation is built upon different areas of knowledge. We will briefly summarize relevant characteristics from the different areas of research. Implementation of technology is considered from an organizational and management science perspective. Computer Supported Cooperative Work is a distinct interdisciplinary research area that provides understanding of the design and use of collaborative technologies. Group learning, finally, draws upon educational sciences.

Groupware Technologies

Our study focuses on a specific type of IT that aims at supporting collaborative work. These systems are commonly called groupware, or collaborative technologies. Keeping in mind that groupware exists in the research agenda since 1960s, we take the risk to broaden our understanding of it and introduce in this work our definition of groupware.

Holtman (1994) has recognized four generations of groupware: basis groupware (1960s), educational (1970s), commercial (1980s), and diverse (1990s). And probably, the beginning of the 2000s has brought the next, *multiple*, groupware bracket.

1960s. The basic functionalities of ‘shared multimedia’ started in 1960s from the groupwork experiment with very basic equipment initiated and sponsored by US Military and invented by Douglas Engelbart of the Stanford Research Institute. Engelbart had main problems with the display quality – VDUs were at the beginning of their development and the output was via the TV-style display. But that was first mix of text and video on screen. The term groupware was not in use by that time, but it was a shared technology two decades before people had even heard of the term.

1970s. The era of educational applications of groupware began. First, it was for university distance education purposes. Secondly, groupware products supported groups of teachers who wanted to communicate at different times. Holtman (1994) distinguishes three notable educational groupware areas. First, the EIES – Electronic Information Exchange System from the New Jersey Institute of Technology – provided information and conferencing to distance students based on conventional telephone lines via modem access. Then, the PLATO – Programmed Logic for Automated Teaching Operations, based at the University of Illinois – supported campus students with inter-site connectivity via private leased lines. And finally, there was the rapid growth of the Unix operating system. Unix appeared at the AT&T Bell laboratories, and later on it was taken up by universities and research establishments in North America. That system indicated a significant change in groupware orientation. Without any commercial purposes, it was used by scientists and researchers, both individually and in groups, to develop small team applications, but besides all, to connect to each other.

1980s. By the end of 1970s groupware was focused mainly on special functions in the science and research domain. The introduction of IBM PC in 1981/82 determined a new era in business computing, which is still continuing. It was not initially a revolution in groupware applications, but the growth of PC actually influenced groupwork. Several reasons motivated connecting PCs together. Firstly, it was the need to share databases; secondly, it was partly a top-down pressure from some IT departments to avoid the problems of individual PCs working in isolation. Thirdly, it was pressure from traditional hardware vendors seeking also to retain connectivity to their proprietary products. Fourth, there was bottom-up pressure from the new local area network vendors (LAN), each unfortunately with their own standard for communications. The mid to late 1980s was the period of significant growth of group-oriented software that mostly included group decision support systems. Introduction of the most innovative groupware products is usually credited to Lotus Notes. Notes improved the business performance of people working together by compressing the time and improving the quality of everyday business processes, such as customer service, account management and product development (Papows & Fielding, 1994).

1990s. During the 1990s groupware products clustered into several broad groups and attracted most of scientific and commercial attention. Coleman (1995) has noted that groupware never took off in the 70s and 80s because there was no sufficient network infrastructure. In the 90s infrastructure was put in place, and business was using groupware to restructure itself for global competition.

Numerous definitions of groupware were presented at various conferences (CSCW, ECSCW, GROUP). We've chosen two of them, the most illustrative in our view. Baecker (1993) defined groupware as "any multi-user software supporting computer-assisted coordinating activities". Ellis et al. (1991), considered groupware as "computer-based systems that support groups of people engaged in a common task and that provide an interface to a shared environment"(p.40).

The 1990s was an intensive period of exponential growth of a variety of groupware systems.Organizations were offered video- and audio conferences, Group Decision Support Systems, Electronic Meeting Rooms, Electronic mailing, Shared Document Applications, Shared Whiteboard Applications, Project Management Tools, Group Calendaring Systems, Collaborative Authoring Systems, etc. The research community offered a variety of groupware typologies based on locus of control (Coleman, 1995), level of support, group processes (McGrath and Hollingshead, 1994), time/space taxonomy (Ellis et al., 1991), application level (ex., Put, 1996), to name a few.

At the same time it became obvious that groupware lay at the convergence of a number of technical, economical, social, and organizational trends that had combined to propel groupware into the minds of managers in both the business and technical communities.

Meanwhile, the rise of the World Wide Web led to a revolutionary change in the possibilities, diffusion and perception of internet and intranet technology.

2000s. While traditional understanding of groupware, developed in 1990s, focuses very much on the support of group work in dedicated teams, nowadays in organizations the available IT infrastructure supports lots of fragments of cooperative work embedded in traditional tasks and group structures. Such cooperative fragments can be recognized often in different work situations ranging from document sharing, cross-functional and cross-departmental projects, to even incidental correspondence between employees linked by a given task. Stand- alone computers nowadays are limited to tests and experiments in organizations, while the norm is that workstations are linked in an organizational network. Common understanding of the way collaborative tasks are performed gets also a broader perspective: employees can work together virtually, intra- and inter-organizationally, globally, etc. Modern collaborative technologies have a role in almost all kinds of business and the public sector. Such a multiplicity of groupware in terms of its targeted sector, employees tasks and structure of collaboration, calls for a broader definition.

We define groupware as *any software systems that facilitate and/or induce collaboration between end-users. These can be either dedicated systems (traditional groupware), or embedded fragments that are part of more general applications such as ERP, CRM, or PDM.*

Group Learning

The concept of group, or collaborative learning strengthens our view on the social issues in the adoption of groupware. This is the core of the theoretical foundation for groupware implementation, in our view. We define learning as changing knowledge and behaviour, and focus not on learning in general, but learning “in the work place” (Watkins and Marsick, 1996), or on-the-job learning (Onstenk, 1995).

The findings from a number of studies (Onstenk, 1995; Dixon, 1994; Crossan et al., 1999) have validated that the fundamental characteristic of learning in the work place is work socialization. Socialization calls for collaboration, which includes mutual interdependence of individual and a group. Collaborative learning doesn’t consist of the arithmetical sum of individual learning contributions, but appears to be a more complex and integrated phenomenon. If employees work collaboratively and engage in a common task with the use of technology, on-the-job learning gets the features of group learning. We view group learning as behaviour that consists of actions carried out by team members through which a team obtains and processes data that improves cooperation. In other words, group learning consists of group interactional processes, like seeking feedback, asking for help, talking about errors, experimenting, discussing of failure, looking for information from outside, critiquing, comparing, evaluating, developing a collective vision, etc. (Edmondson, 1999; Schippers et al, 2001; Stahl, 2000).

Numerous studies have shown that implementation of collaborative technologies is a process that takes time. User groups do not change their ways of working overnight but gradually appropriate the available technology. In order to allow further support of such processes, a further understanding of the true nature of these processes is needed.

In order to build our understanding of collaborative learning we have transferred the experiential individual learning cycle of “acting–reflecting–thinking–deciding” (Kolb, 1984) to a collective one. On the inter-personal level, the mechanism of group learning is described with the following wheel: “collective actions – group reflection – knowledge disseminating – sharing understanding – mutual adjusting” (Figure 1).

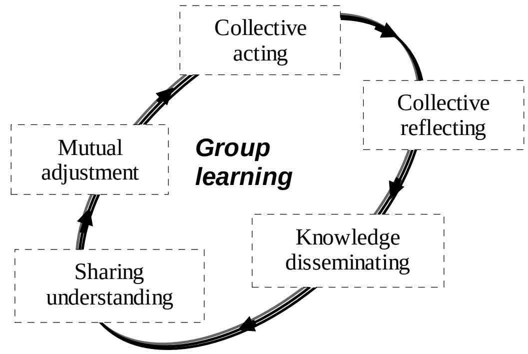

Figure 1. Group learning processes

At a group level, learning is conceptualized as on-going group interaction activities of group acting and reflecting (Edmondson, 1999).

A group learning cycle begins with the *collective experiences and actions*, when a group of people is given a certain task to perform. This step reflects apprehension of knowledge, when a group is expected to accept new knowledge through perceptions and direct experiences. According to West (2000), action refers to the goal-directed behaviours relevant to achieving the desired changes in team objectives and strategies. This stage is assumed to be important in all learning cycles as it helps to experience assumptions. Acting might lead to new information, which can lead to further reflection, planning, and again action as an on-going process (West, 2000).

When a new technology is introduced to the targeted employees who are networked together, they will start operating with the system in order to execute the tasks. This can develop through different activities, including operating with basic modules in the performance of everyday tasks, or searching for new techniques in the system. The employees can simply replicate the techniques they have learnt during instructions or try to find out new functionality in using the system. More experienced members of a group may take the initiative for testing new techniques.

The next stage is ***group reflection*** **-** the extent to which group members reflect upon, and communicate about the group's objectives and strategies (e.g. decision-making), and update them to the current circumstances (Schippers, 2003). A group is expected to move inward to reflect upon previously acquired knowledge. Reflection takes place through a variety of activities: discussions, asking questions, declaring difficulties, collective debates, presentations that aim at knowledge externalisation. It is considered crucial in learning from experience because it might help neutralize biases and errors in group decision-making.

A lot of research has been conducted on group reflective processes. Swift and West (1998) have identified three levels of reflection based upon its depth. Shallow reflection is seen as the first level of group awareness (for example, discussing aspects of the tasks). Moderate reflection viewed as a more critical approach towards tasks (for example, discussing strategies used by a group to accomplish the tasks). Deep reflection occurs when a group questions the norms and values of the group or an organization. Schippers (2003) summarises that reflective group behaviour includes evaluation of actions, ascertaining whether everyone in the group agrees about the way in which the task will be handled, discussing the effectiveness of methods of working and communication, and discussing the norms and values of the groups and organization.

A group may reflect on its knowledge before actions, during task execution, or after that. Reflection before task execution may include open dialogue about strategies and goals. Reflection during task execution mainly aims at identifying whether a group is still on track. It can be also achieved by organising dialogues, forum groups, and discussions (Schippers, 2003). Reflection after task execution is characterised by evaluation of the performance that might lead in our model to knowledge extension during the "deciding" processes.

In the situation with introduction of a new technology, group reflecting can take place at different stages, too: after some operations with the system, or along the way during implementation, but it can happen even before the system introduction when the future users discuss design issues of technology. In any case, group reflecting would include communicating upon the extent to which the system supports performing tasks. Discussions, open dialogue, focus groups, meetings with a project team might concentrate on speaking out difficulties in use of the system, comparing with another software experience and with another IT, declaring individual problems in use of the system. Users might express doubts and suspicions or trust and beliefs in existing ways of solving IT-related difficulties, consider possible reasons and outcomes of mistakes made during operating the system, discuss errors in working with different IT functionalities

The *knowledge disseminating step* brings the crucial difference between individual and group learning. When we are to transfer individual learning to the cooperative level, the act of knowing becomes more complicated. In a group environment people would think together, that means they would share results of their thoughts. But knowledge is not something that can be easily passed around (Hendriks, 1999). There is no doubt that some information can be codified, stored and reused to enable effective action at a later stage, but a representation is not equivalent to knowledge (Sutton, 2001). Let us clarify these processes.

With the assumption that the knowledge is created through conversion between explicit knowledge (that is transmittable and communicable in formal language, and often referred to as information) and tacit knowledge (that has a personal quality and is hard to communicate), there are four modes of the knowledge conversion processes which can take place in group learning: externalisation – from tacit knowledge to explicit knowledge; combination – from explicit knowledge to explicit knowledge; internalisation – from explicit knowledge to tacit knowledge; and socialisation – from tacit knowledge to tacit knowledge (Nonaka, 1994).

In other words, to break experiences into meanings, a group would need two phases: first, reconstruction and codifying of knowledge (externalisation and combination); and only then, knowledge can be shared, or transformed to a tacit form (internalisation and socialisation) (Hendriks, 1999). We label those phases knowledge disseminating and sharing understanding.

Knowledge disseminating can appear in many forms, including presentations, lectures, oral explanations of ideas, or "codifying it in any intelligent knowledge system" (Hendriks, 1999). This process is not necessarily conscious. For example, employees can learn by watching someone's performance, even if they are unaware of the specific knowledge needed for the task performance. But we are convinced that in almost all practical situations where knowledge sharing is going to occur it is important to stimulate 'knowledge owners' to externalise their knowledge in a way that is suitable for others.

Knowledge disseminating during the implementation process of a new information system would include behaviors of the group members that aim at externalization of ideas about the system in order to improve its usage. It might emerge in demonstrating of working with technical modules both in formal situations (workshops) and informal (work pauses), proposing new actions to improve the usage, clarifying difficulties and questions to the peers. Users may take the initiative to show their colleagues how to generate new options in the system or to come up with new suggestions to improve the system.

After that the wheel cycles to *sharing understanding*. That involves using insights to help people see their own situation better (Kim, 1993). Internalization also takes on a great variety of forms: learning by doing, reading books, etc. It is oriented to those people who look for acquisition of knowledge. It implies mutual informal acceptance and respectfulness of diverse ideas and suggestions. Nelson and Cooprider (1996) define sharing understanding as an appreciation of knowledge among the group members that affect their mutual performance (p.410). Appreciation among the group members is characterised by sensitivity to the frames of reference and interpretations of others in a group. Effective shared understanding can be viewed as a synergy between group members that mutually respect and trust each other. Appreciation and trust are two main components of shared understanding.

Knowledge internalization concerning new technology will lead to a shared meaning of the system among the users. They will share their understanding of the global role of IT in a company and its intentions for every member of a group, as well as design intentions of the developers of the system. Understanding of technical possibilities and different functionalities (main and optional) can be also considered as a result of this stage. A group would come up with common attitudes towards the technical functionality and content of IT – whether technology helps to accomplish job tasks and responsibilities and to which extent.

The last step in the cooperative learning is *mutual adjustment*, or arrangements initiated by the group members. In Kolb's model this step ("deciding") is related to the extension of knowledge when learners are expected to move beyond the selves to interact with an external environment. Reflections and knowledge sharing don't lead to changes in group learning. At this stage, the group engages in activities that lead to a choice to make decisions together, to reject or adopt, to evaluate or ignore tasks, strategies, or new rules.

Some adaptations need to occur. Joint regulations, planning, arrangement and deciding – these are activities undertaken by group members in order to move the learning cycle further. In this phase, goals are presented and ways to achieve them are planned. According to some authors, adjustment takes place not only before task execution, but also during it as well (Schippers, 2003).

In a situation with a new technology, this step in the group learning cycle will include activities that aim at collective agreements to improve the use of the system in the group. Group members may take initiative to arrange (request) additional training, instructions, manuals, and other learning activities. Developing regulations in order to improve the use of technology can become a crucial issue, especially if the users never worked before as a group. For example, this might involve decisions about dividing responsibilities in making inputs and schedules of making outputs. Decisions may be also made about sorts of documents to be submitted or about the data traffic and classification. IT might also concern group process issues like developing regulations for intermediate evaluations of the IT project, supporting on-line chat about hot issues in the project and news overviews.

These plans will be implemented in the action phase. After planning is completed, its implementing starts and this provokes a new wheel beginning with collective acting.

A new learning cycle will be based on the previous group experience and knowledge. Planning can also take place during the action, or executing of a task, when plans are developed and shaped by seeking feedback, group reflecting processes. This strengthens the importance of group reflexivity.

It should be noted that the five steps in group learning do not necessarily take place in consecutive order. The decomposition into five steps is not a temporal but a logical decomposition, which serves to understand and analyze group learning processes.

To summarise, group learning is understood in IT implementation as negotiations among the targeted employees aimed at developing implementation of a new system: they practice with the system and discuss the experience, experiment and search for new possibilities and communicate upon it, ask for help, clarify difficulties, talk about errors while working with it, propose new actions to improve its use, plan further implementation, develop common rules on working with the system, evaluate its use at different stages, and sometimes reject it.

Group learning in Groupware implementation is defined as all interactional processes through which group members develop interpretive schemes about a newly introduced system that help them to implement it, i.e. to work together with it skilfully and task-consistently.

METHODS

We have conducted a case study research in one of the larger hospitals in the Netherlands, called Medinet, where a new Personnel Management System was introduced.

The case study lasted 10 months and was based on qualitative methods like semi-structured interviews, observations, field notes, and documents analysis. 34 interviews were conducted lasting from 45 minutes to 2 hours, in total of 48 hours. During interviews we asked employees to describe how and why the new system was introduced, what kinds of job tasks were supported by the system, characteristics of the system, etc. Such questions allowed to listen for understanding of technological features and functionality, attitudes towards technology, examples of group learning behaviour, and learning climate in the company. Postscripts of all 34 interviews were again discussed with interviewees for verification.

The qualitative approach supported an analysis of different actors' interpretations of the technology and their actions around it. In order to analyze the qualitative data, we operationalized group learning processes for groupware implementation. The definitions are given in Table 1.

Table 1: Operationalization of adoption of groupware through group learning

Dimensions of group learning	**Components**
1. Collective acting – task-related operations with the system undertaken by members of a group.	§ operating with basic modules in everyday tasks performance § searching for new techniques in the system
2. Group reflecting – communicating upon extent to which the system supports performing tasks.	§ discussing difficulties in use of the system § comparing with another software experience § declaring individual problems in use of the system
3. Knowledge disseminating – behaviors of the group members that aim at externalization of ideas about the system in order to improve its usage.	§ demonstration of operating with technological options § proposing new actions in order to improve on-going use § clarifying difficulties to the team members
4. Sharing understanding – the level of common meaning of the system regarding the role of the system and its functionality.	§ clearness about the purpose of the system § users' needs in the system § understanding of operating with the modules in the system § attitudes towards functionality of the system § attitudes towards future state of the system
5. Mutual adjustment – activities that aim at collective agreements on on-going use of the system in the group.	§ arranging (further) learning activities to improve use of the system § developing regulations § evaluating intermediate results

CASE STUDY

Our case study reports on the implementation of a personnel administration system – Beaufort – in one of the larger Dutch hospitals, called Medinet, which has 1070 beds and around 3700 employees. The project, involving acquisition of a new information system, development of the project plan, and realisation – started in June 1999 and was expected to be completed in December 2001.

The project had two planned phases: introduction of the system to the central Personnel and Salary Administration (PSA) department, and introduction of the system across the entire Medinet. From our theoretical perspective, these became two distinguishable sub-cases. The PSA department implemented Beaufort effectively, efficiently, and in accordance with the initial plan (sub-case 1). The introduction of the same system to the personnel specialists in other departments failed, which led to the blocking of the whole project in October-November 2001 (sub-case 2).

Organizational context

There is a tight cooperation between the PSA and the local managers: every day the latter send information in special paper-based forms about all changes in personnel data to the PSA. Day-to-day communication between all representatives of the personnel service in all departments and units was made via internal paper-based mail, e-mail, fax and telephone.

The idea of the new system was that local managers should input the personnel data straight into the system and could share that information across departments. At the same time PSA employees could immediately use these data to make any salary mutations.

System Specification

The Beaufort system, developed by the Dutch software company Getronics, is a personnel and salary administration system, extensible with modules for time registration, human resource management, financial management, etc. One of the strengths of the system is that it allows decentralized use. Data entry can be done locally in each department. Department managers can have access to management information for their department.

Beaufort is a system that provides a company with the opportunity to improve and decentralize its internal personnel management processes. It is a module-based personnel and salary administration system that contains technical options for publishing, composing, structuring, improvisation, and storing personnel data. There are seven modules, with which users can perform document administration: personnel management, salary administration, sick leave administration, formation and organization, time registration, office link, and report generator.

The basic module is Personnel Management, through which the users input and update all the information concerning personnel data (see Table 2). These inputs do not require specific codification as they are registered using normal words.

The Sick Leave Administration and Time Registration modules are very important in salary calculation. All inputs in those two modules are coded using special numbers, consisting of 3–5 digits. Any changes in the code numbers might indicate changes in the working conditions (for example, less or more working hours per week, or urgent working hours, or differences in types of sickness including professional sicknesses) that will automatically modify the salary in the Salary Administration module.

The Salary Administration module also requires codified inputs. The users (salary administrators) combine all the personnel data in this module (such as sick leave days, participation in the optional schemes for fringe benefits, flexible and urgent working hours, types of professional qualification, and medical authorization). Any small mistake in numerical input would lead to an incorrect salary for an employee.

Beaufort's Formation and Organization module provides the structure of the company in a hierarchical manner: subdepartments and units, clusters, divisions, etc. It gives an overview of the whole company and allows one to see the place of any employee in this structure. Only Medinet's IT department is authorized to make changes in this module and update the information, other users can only read it.

Office Link is a special HRM module that allows HR administrators to send letters to employees using mailing lists within Medinet, for example to a certain department, or to all nurses. Such letters may concern a range of personnel information – changes in work contracts, invitations to special events, up-dating labor conditions, information, etc.

The final module – "Informer" – provides the possibility to generate non-standard reports upon requests from the HR managers: reports about different expenses on yearly or monthly bases (such as travel expenses or telephone bills), salary and premium overviews, etc.

Specification of the Beaufort functionalities is given in table 2.

Table 2: Specification of the Beaufort modules

Module	Specification
Personnel Management	Registration of: • Personnel data: name, title, address, family status, date and place of birth, employee number, type of contract, department and function, special authorization issues, participation in the fringe benefit options, etc. • Career development data: educational background, professional experience, on-going professional development (courses, education, etc.), and social activities. Inputs are not coded.
Salary Administration	Operating with all inputs from other modules in order to calculate salary. All inputs and outputs are numerically coded.
Sick Leave Administration (SLA)	Registration of absence (total or partial) due to sickness, and notification of this absence to the various external administrative bodies related to the social security system in the Netherlands. Inputs are based on the date, type of sickness, necessary treatment, pregnancy, frequency of sickness, and relationship with the occupation in the hospital, etc. Inputs are crucial for salary administration. All inputs are numerically coded.
Formation and Organization	Detailed picture of the organizational structure and employees within the hierarchical order: divisions, clusters, departments, subdepartments, sub-units, etc.
Time Registration (TR)	Registration of working hours in accordance with the Collective Agreements for Dutch Hospitals (special registration of weekend and holiday working hours, emergency hours, day and night shifts, etc.). Inputs are essential for calculating monthly salaries. Inputs are numerically coded.
Office Link	Administration of various types of letters to employees (invitations, congratulations, bulletins, etc.)
Report generator "Informer"	Creating non-standard reports.

The Beaufort project's strategic plan (January 2000) contains information about the reasons for Beaufort's introduction at Medinet. It states that the introduction of Beaufort aimed at improving the efficient processing of HR administrative data, simplifying admission to strategic information, and improving the protection of sensitive personnel information. Other goals were described as follows:

- To increase the efficiency of personnel administration by restructuring the HRM processes, from a highly centralized approach to a decentralized one. Local HR managers were expected to carry out data processing directly using the system.
- To create shared information files, leading to the use and exchange of personnel information among local managers.

In the Beaufort project at Medinet, two modules were selected for decentralized use: sick leave administration and time registration. Sick leave administration involves registration of absence (total or partial) due to sick leave and notification of this absence to various external administrative bodies related to the social security system in the Netherlands. It is important that these notifications are timely and correct; failure to do so may lead to a situation where Medinet is held liable for a financial compensation that could have been claimed elsewhere. Time registration is essential for calculating the monthly salary. For doctors and nurses the salary is a function of the number of hours worked on different kinds of duties.

FINDINGS

In this section we present our findings from the case study in the following order: first results of the implementation in the PSA department, after that the results among the decentralized users.

Beaufort and the PSA Department

The group learning processes in the PSA department in order to adopt Beaufort were characterised as moderately high. We provide the description of these processes based on the textual analysis of the interview transcripts.

The PSA employees operated with the system very actively, in their day-to-day task performance. Mainly it was based on the running basis modules, while searching and testing new techniques were exceptional.

They used to critically reflect upon their experience with the system. Every morning they discussed different problems in on-going use during special sessions. Also informal discussions took place often. They had special notebooks, where they noted every nuance from Beaufort that must be discussed together. It led, for example, to a long chat about rules for sending the salary data away. At the beginning the system used to make some unexplainable errors (e.g. mixing up the numbers, or miscalculating working hours). An employee who first found that immediately pointed out those errors.

Everybody felt free to declare their individual difficulties and lack of skills in use of some modules. They knew each other's difficulties with operating the system.

Knowledge disseminating was rather intensive and based on two streams. Firstly, some active members stimulated, proposed and demonstrated new ideas with the intention to improve the usage of Beaufort. Secondly, at a more modest level, colleagues clarified for each other different aspects of Beaufort.

Sharing understanding among the PSA employees was moderate. Interesting is that they all had similar ideas concerning the role and functionality of Beaufort, but their understanding did not reflect the real purpose of the system.

Mutual adjustment was moderate and mainly related to arranging further learning activities and suggestions concerning improvements of the system. Collective agreements and developing new regulations to apply new ways of working with new system were not initiated.

In sum, collaborative learning processes within the PSA group members can be characterised as strong. Task-related operations with Beaufort, communicating about different aspects of it, activities oriented towards knowledge externalisation and achieving collective agreements were strong. Only the group understanding of the role and functionality of Beaufort was moderate.

The PSA employees valued the system as very helpful and advanced in supporting their tasks. Especially they rated highly that all the personnel information was placed on one screen. They estimated that they could perform the documents and administration procedures faster than with the previous system.

Also they found valuable that the system helped them in communicating with their clients (employees of the Medinet): during telephone calls it was enough to use only one screen without difficult paper-based searching processes.

Based on the observations and interviews we may conclude that PSA members have adopted the newly introduced system with the high level of efficiency. All employees got used to Beaufort in accordance to the scheduled plan – within three months.

Beaufort and Decentralized Use

We identified group interactional processes among HR local managers as low: group acting, reflecting, sharing understanding, and mutual adjustment hardly took place, and only under strong pressure from the management. Only knowledge disseminating was observed as promising. Below we illustrate it.

Every time when decentralized users met even small technical difficulties, they stopped operations with the system. They were not clear about the idea behind the decentralized use.

Actually they did not need Beaufort for their usual job tasks. Operating with the system brought only additional duties and complexity into their tasks. Collective acting did not develop through exercising, instead, end-users had to start working with a new system immediately. Decentralized users did not try to search for any new techniques in the sick leave administration module.

We did not identify group reflecting at all. They did not want to discuss any problems, but passively waited for the external help. They did not communicate about errors in the system with each other, and preferred to talk about it directly at a higher level – to the project management.

Knowledge disseminating was initiated by the PSA employees, who used to give advice anytime upon the request of the decentralized users. The low level of sharing understanding resulted in unclarity about even the content of the sick leave inputs. Mutual adjustment was observed as absolutely low. Tasks and rules were not written down – there wasn't any agreement on how to work together.

The HR managers were of opinion that the system did not facilitate their tasks, but rather brought new ones for them. They acknowledged the importance of Beaufort for the salary administration, but did not find their participation in it essential. They stressed that time registration and sick leave administration were just small administrative responsibilities among their HR work, but the system made them pay too much attention to those tasks.

At the same time the users even lacked some data necessary to make inputs to the system.

The system required changing the usual way of performing the tasks (new collaborative responsibilities, sharing the data, duplication or triplication of the task performance, new schedule for making inputs).

The local HR managers have not adopted the newly introduced two modules of the system in accordance to the project plan. They were struggling with the implementation process, described above, during 7 months, and finally decided to stop it. All end-users (100%) shared the opinion that it was necessary to suspend the project until better times.

DISCUSSION

The perspective of group learning provides us with interesting notions about different outcomes between the two sub-cases. We discovered that group learning processes did take place in both cases, but the content of them was rather different. In the PSA department these processes helped to improve adoption of the new system and led to the stable use of it. In the group of decentralized users learning processes blocked adoption of the new system and contributed to termination of the whole project.

While we credit success and failure of Beaufort adoption to differences between group learning, we also realize importance of the organizational circumstances for those processes. First, we summarize and conclude about the content of group learning in adoption of Beaufort, and after that we discuss the organizational environment for the system implementation.

To estimate group learning we gave qualitative labels ranging from 'weak' to 'strong' (active-passive, high-low, intensive-fuzzy, etc.). Giving such labels we kept our operationalization scheme, where 'high' learning meant the intensity of the users' activities and orientation towards improvement of system adoption. We have categorized group learning in the PSA department as relatively strong towards adoption of Beaufort. In the group of decentralized users group learning was labelled as weak towards adoption of Beaufort.

PSA employees used to communicate and discuss different aspects of Beaufort implementation with the aim to improve its use. We discovered the leading role of group reflecting and knowledge disseminating. 'Activities-based' group learning processes (collective acting and mutual adjustment) were lower. Sharing understanding – the content of the shared meaning of Beaufort among PSA members – at the beginning slowed down implementation. This has changed while using the system: PSA members transformed their perceptions of the system from ignoring to acknowledging its advantages.

Decentralized users also communicated actively about different aspects of Beaufort. Their discussions aimed at sharing negative ‘feelings’ concerning the system and the future introduction of it in the whole company. They exchanged their experience and evidence against using Beaufort, and suggested to terminate the pilots. They perceived the system as un-reasonably difficult and complex to operate.

It was interesting to find the development of employees’ needs in a new technology. We think that in an ideal situation, employees should need a new system before its introduction. These personal needs can differ from the main goal of a system in a company. However, in both Medinet sub-cases we have observed an absolute disregard of the individual needs in introduction of a new system. PSA employees developed and realised their needs in Beaufort while implementing it (this even helped them to clarify the intention of the system). Decentralized users kept on lacking any individual needs for more than six months.

The most illustrative opposite results were discovered in the processes, which we labelled ‘mutual adjustment’. In the PSA department employees arranged educational activities to learn more about Beaufort; they strived to reach new departmental rules and agreements in order to ease the use Beaufort. Decentralized users put efforts to arrange different sessions to convince the project team to stop the pilots. We have characterised the group learning process in the two settings in table 3.

Table 3: Group learning processes in two settings

	Group learning in PSA	**Group learning among decentralized users**
Collective acting	Moderate to Active	Passive
Group reflecting	Mostly strong	Moderate to weak
Knowledge disseminating	Mostly intensive	Fuzzy
Sharing knowledge	Moderate	Low
Mutual adjustment	Moderate to Strong	Weak

Although the insights that come out of the learning perspective are remarkable, the results need to be considered from a broader perspective.

The two sub-cases lead us to the notion that functionality played an essential role and even to some extent predicted the results. The system aimed to carry organizational changes in the whole company regarding task design and collaboration among employees. In fact it is not a novel idea to say that the organizational change can be hardly realized by technical introduction of a new technology. Beaufort did not bring any task changes to the work of the PSA specialists. But the decentralized local managers had to change their work a lot. They had to learn new tasks, which were just secondary; and to take higher responsibilities to perform those tasks. The PSA employees did not face changes in the way they used to cooperate before introduction of Beaufort. But the local managers faced a new, very complex collaboration in a new situation. They faced the necessity to serve the system instead of getting support from it. At the same time the content of the tasks appeared to be crucial. The Medinet case study convinced us that the tasks related to the personnel information – administering and managing – were very sensitive. They are associated with the privacy and the security of very sensitive information, and therefore require strong responsibility if to be transferred.

In the PSA case Beaufort played a role of the intensive groupware and supported reciprocal interdependence within one department. In the decentralized case there was a need for a higher cooperation between the departments, within the local communities, and with the PSA specialists. Beaufort became a multichannel groupware and supported associated interdependence. A complexity of the groupware contributed to the negative results in the decentralized sub-case.

We suppose that before Beaufort was implemented for decentralized users, there was also a need to create collaboration among them. It doesn't mean that groups of users must have perfect collaborative prerequisites in advance in order to adopt the system. As we have said earlier, group processes do improve over the use of groupware. But essential group characteristics must be built up in advance. Those are interdependence, individual accountability, task division. Such prerequisites prepare the basis for interactional processes, through which implementation of groupware, in our view, develops.

CONCLUSIONS

We have proposed a model for implementation of collaborative technologies, which regards it as a learning process. The longitudinal case study in the Dutch hospital confirmed our theoretical assumptions that adoption of collaborative technology developed through group learning; when the system was introduced to the users they had to collaborate to perform the tasks. However, we should notice that the paper presents the results of only one organization's experience. In order to validate the model, IT implementation in different organizations should be studied. That must include different types of companies and different types of information technologies.

Group learning includes interactional processes through which group members develop implementation of technology: they practice with the system and discuss this experience, experiment and search for new possibilities and communicate upon it, ask for help, clarify difficulties, talk about errors while working with it, propose new actions to improve its use, plan further implementation, develop common rules on work with the system, evaluate its use at different stages, etc. We have found that the five steps of group learning in accordance to our operationalization scheme – collective acting, group reflecting, knowledge disseminating, sharing understanding, and mutual adjustment – existed in reality in both settings, PSA and decentralized users.

An important finding is that in both settings group learning emerged immediately after a new collaborative technology was introduced to the targeted users.

At the same time the content of group learning in sub-cases was opposite. In the PSA department it was categorized as strong as it helped improve adoption of the new system and led to the stable use of it. Decentralized users blocked adoption of the new system and initiated termination of the whole project – also through group learning, which was oriented towards blocking system usage.

The technology may trigger group learning, requesting redirecting of its scope towards alignment with a new user group. We have seen that the higher the level of interdependency between the users was requested by the system, the more efforts were needed to redirect group learning from a smaller group to the entire group of users across different departments.

Investigation supports the idea that organizational support does influence adoption of the system. The idea is not new. However, based upon our research we propose that organizational support should include special practices to advance group learning in order to promote implementation of collaborative technologies. We realise that this itself it is not a guarantee of successful implementation yet, but ignoring of group learning processes by projects managers may lead to slowing down or even terminating it.

To summarise our discussion we may conclude that group learning processes do play an important role during adoption of the newly introduced groupware system and can explain its implementation success or failure. Independently of the organizational conditions, group learning emerges immediately after introduction of a new groupware system. But the direction of group learning can differ depends on the conditions, in which it takes place.

REFERENCES

Arkich, M. (1992). The de-scription of technical artifacts. In: W.E.Bijker, J.Law (eds.), *Shaping Technology/Building Soiety: Studies in Sociotechnical Change* (pp. 205–224). Cambridge, MA: MIT Press.

Baecker, R.M. (Ed.) (1993). *Readings in Groupware and Computer-Suppored Cooperative Work: Assisting Human-Human Collaboration.* San Mateo, CA: Morgan Kaufman.

Bardram, J. (1998). Designing for the Dynamics of Cooperative Work Activities. *Proceedings ACM 1998 Conference on Computer Supported Cooperative Work ,* (pp. 89–98), New York, NY: ACM Press

Bikson, T.K. & Eveland, J.D. (1996). Groupware Implementation: Reinvention in the Sociotechnical Frame. In: M. Ackerman (Ed.), *Proceedings of the ACM 1996 Conference on Computer-Supported Cooperative Work* (pp. 428–437), New York, NY: ACM Press.

Ciborra, C.U. (1996). Introduction. In: C.U. Ciborra (Ed.) *Groupware & Teamwork: Invisible Aid or Technical Hindrance?* Wiley, Chichester, U.K.

Coleman, D. (1995). Groupware technology and applications: an overviewof groupware. In: D. Coleman, D. & R. Khanna (Eds.), *Groupware: technologies and applications* (pp. 3–41). Prentice Hall PTR.

Crossan, M.M., Lane, H.W., & White, R.E. (1999). An organizational learning framework: from intuition to institution. *Academy of management Review, 24,* 522–537.

DeSanctis, G., & Poole, M. (1994). Capturing the Complexity in Advanced Technology Use: Adaptive Structuration Theory. *Organization Science, 5*, 121–147.

Dixon, N. (1994). *The Organizational Learning Cycle.* London: McGraw-Hill.

Edmondson, A. (1999). Psychological safety and learning behavior in work teams. *Administrative Science Quarterly, 44*, 350–383.

Ellis, C.A., Gibbs, S.J., & Rein, G.L. (1991). Groupware: some Issues and Experiences. *Communications of the ACM, 34*(1), 38–58.

Hendriks, P. (1999). Why share knowledge? The influence of ICT on the motivation for knowledge sharing. *Knowledge and process management, 6,* 91–100.

Hettinga, M., & Schippers, M. (2001). Invisible forces in favor of the status quo: Stimulating reflective restructuring. In: K. Sikkel, T. Bondarouk, M. Hettinga, J.G. Schuurman (Ed.), *Learning Groups – report of a workshop at the 7th European Conference on Computer Supported Cooperative Work*, Telematica Instituut, Enschede, the Netherlands.

Hettinga, M. (2002).*Understanding evolutionary use of groupware*.Telematica Instituut Fundamental Research Series, No. 007 (TI/FRS/007), Telematica Instituut, Enschede, The Netherlands.

Holtman, C. (1994). Groupware: its past and future. In: P.Lloyd (Eds.), *Groupware in the 21st century*. *Computer supported cooperative working toward the Millenium.* London: Adamantine Press Limited.

Kim, D.H. (1993). The link between individual and organizational learning. *Sloan Management Review*, *35*, 37–50.

Kolb, D. A. (1984). *Experiential Learning. Experience as the Source of Learning and Development.* Englewood Cliffs, NJ: Prentice-Hall.

McGrath, J.E., & Hollingshead, A.B. (1994). Groups interacting with technology. Ideas, evidence, issues, and an agenda. London: Sage Publications.

Mulder, I., Swaak, J. & Kessels, J. (2002). Assessing group learning and shared understanding in technology-mediated interaction. *Educational Technology and Society*, 5 (1), 35-47.

Nelson, K.M. & Cooprider, J.C. (1996). The contribution of shared knowledge to IS group performance. *MIS Quarterly*, *20*, 409–429.

Nonaka, I. (1994). A dynamic theory of organizational knowledge creation. Organization Science, 5, (1), 41–60.

Onstenk, J.H.A.M. (1995). Human Resources Development and On-the-job Learning. In M. Mulder, W.J. Nijhof & R.O. Brinkerhoff (Eds.), *Corporate Training for Effective Performance*, Boston, MA: Kluwer Academic Publishers.

Orlikowski, W. (1992). The duality of technology: Rethinking the concept of technology in organizations. *Organization Sciences*, 3, (3), 398–427.

Orlikowski, W.J. (1996). Improvising Organizational Transformation Over Time: a Situated Change Perspective. *Information Systems Research* 7, 63–92.

Orlikowski, W. (2000). Using technology and constituting structures: a practice lens for studying technology in organizations, *Organization Science*, 11, (4), 404–428.

Papows, J., & Fielding Sr, J. (1994). The future of Groupware. In: P.Lloyd (Eds.), *Groupware in the 21st century. Computer supported cooperative working toward the Millenium.* London: Adamantine Press Limited.

Ruel, H.J.M., (2001). *The non-technical side of office technology; managing the clarity of the spirit and the appropriation of office technology*. Ph.D. Thesis. Enschede, the Netherlands: Twente University Press.

Schippers, M. C., Den Hartog, D. N., & Koopman, P. L. (2001). *Reflexivity in teams: The relation with trust, group potency, team leadership, and performance in work teams.* Paper presented at the Academy of Management, 3-8 August, Washington DC.

Schippers, M. (2003). *Reflexivity in teams*. PhD Thesis, University of Amsterdam.

Stahl, G. (2000). A model of collaborative knowledge building. *Proceedings of Fourth International conference of the Learning Sciences*, Ann Arbor, MI, 70–77.

Sutton, D.C. (2001). What is knowledge and can it be managed? *European Journal of Information Systems*, 10, 80–88.

Swift, T.A., & West, M.A. (1998). *Reflexivity and group processes: Research and practice.* Sheffield: The ESRC Centre for Organisation and Innovation.

Tucker, A.L., Edmondson, A.C., & Spear, S. (2001). *When problem solving prevents organizational Learning*. Harvard Business School working paper 01-073.

Watkins, K. & Marsick, V. (Ed.) (1996). *Creating the Learning Organization*. Alexandria, VA: ASTD.

West, M.A. (2000). Reflexivity, revolution and innovation in work teams. In: M.M. Beyerlein, and D.A. Johnson (Eds.), *Product development teams* (Vol. 5, pp. 1 – 29). Stamford CT: JAI Press.