Information system development: From user participation to contingent interaction amoung involved parties

van Offenbeek, M.A.G.; Koopman, P.L.

published in
European Journal of Work and Organizational Psychology
1996

DOI (link to publisher)
10.1080/13594329608414868

document version
Publisher's PDF, also known as Version of record

EUROPEAN JOURNAL OF WORK AND ORGANIZATIONAL PSYCHOLOGY, 1996, 5 (3), 421–438

Information System Development: From User Participation to Contingent Interaction Among Involved Parties

M.A.G. van Offenbeek
Faculty of Management and Organization, University of Groningen, The Netherlands

P.L. Koopman
Department of Work and Organizational Psychology, Vrije Universiteit Amsterdam, The Netherlands

Over the last few decades many social scientists have pointed to the importance of user participation in information system development (SD). In study A the opinions of 314 Dutch system developers on this issue were examined. The conclusion is that the majority rate the value of user participation highly. However, these data confront us with three important problems. First, we cannot be sure that the stated preferences of system developers are reflected in their actual behaviour. Second, system developers may give a different meaning to the concept "user participation" than social scientists. Third, "user participation" does not necessarily imply "user influence" on SD. Moreover, the empirical relationship between user participation and the effectiveness of SD remains unclear. In our opinion this can be attributed both to the multi-dimensional nature of the former concept and to the contingency of this relationship to the context of SD. Therefore, in study B, we developed a multidimensional contingency model to study the interaction among parties in SD instead of the study of user participation. In this explorative study no linear relationship between the effectiveness of SD and the interaction between users and system developers was found. Instead, effective SD seemed to require a fit between the context of SD and the interactions with and among users.

INTRODUCTION

Employee participation is a mechanism for exchanging information and creating commitment, necessary features for managerial decision making (e.g. IDE, 1981). In studies concentrating on the organizational aspects of

Requests for reprints should be addressed to Dr M.A.G. van Offenbeek, Faculty of Management and Organization, University of Groningen, postbus 800, 9700 AV Groningen, The Netherlands.

information systems development (SD), attention has been directed towards the need for "user participation" (e.g. Cressey, 1989; Hedberg, 1975; Mumford, 1983; Söderberg, 1986). "User participation" refers to the participation of employees in the development process of the system of which they will be the future users. The discussion has been concerned with the question whether, and to what extent, system developers should consult future users during the design process. The aim of user participation is to enhance both the quality of the developed system and the acceptance of this system by the future users (Ives & Olson, 1984).

A variety of research has been undertaken to determine whether user participation does indeed lead to more effective SD. Research over the last decade suggests that user participation is not a panacea for effective SD. Although Baroudi, Olson, and Ives (1986) and Riesewijk and Warmerdam (1988) have reported positive findings, Heinbokel (1994) has found a negative correlation between user involvement and overall success. Ives and Olson (1984), Algera, Koopman, and Vijlbrief (1989), and van Oostrum and Rabbie (1988) have shown that user participation has no definite effect on the effectiveness of SD. According to these studies, the conditions under which user participation will lead to better outcomes must be explained. Furthermore, the type of participation with which one is concerned should be specified. Similar conclusions have been drawn for employee participation in decision making by Heller, Drenth, Koopman, and Rus (1988). Likewise, Cotton, Vollrath, Lengnick-Hall, and Froggatt (e.g. 1990) have propagated the use of a multidimensional view on participatory leadership because its effects can vary strongly according to the form and context of the participation.

First, it can be concluded that user participation should be looked at in a more differentiated way; a one-dimensional high–low concentration of user participation has proved to be too simple. Second, the amount and kind of user participation needed depends on the context of the SD process (e.g. Edström, 1977; Schonberger, 1980; Wijnen & van Oostrum, 1993). Thus a multidimensional contingency framework for the study of the effectiveness of user participation in SD is needed.

The study of the concept "user participation" has some dysfunctional consequences from a cognitive point of view as well. We will argue that the origin of the concept makes it less suited to modern SD practices.

It is not clear what precisely is indicated by the concept of user participation. It is a semantic device which refers to many different practices. We have to specify who participates in what kind of processes and to what purpose. Do system developers participate in the change processes of user-organizations, or do users participate in the design processes of system developers? Both statements contain some truth, whereas the term "user participation" seems to refer solely to a situation in which the application of information technology is developed in isolation to its social context. The leading role is for technology, and users are brought into the design process to establish the specifications for the software. As Clegg and Symon (1989) stated "it [user participation] assumes the issue is one of getting users and others to participate with the designers who are the legitimate experts on design, and who thereby often end up owning both the problem and the solution". As such the concept leads to inferences which are inappropriate in the light of the idea that the systems built should fit the clients, in this case the prospective users.

The concept of user participation was introduced at a time when automation was characterized by a technological push and the term bears testimony to this. Nowadays we find a much stronger demand pull: the automation market is shrinking and organizations have far more knowledge about information technology than in earlier days. Nevertheless, case studies of SD have shown that technicians too often still dominate the SD process (Björn-Andersen, Eason, & Robey, 1986; Child & Loveridge, 1990; Heming, 1992; Riesewijk & Warmerdam, 1988).

When system developers are told that they should allow for user participation, this in a way reinforces the domination of SD by automation experts. In fact, the concept of "users" only exists in the minds of automation experts. In their own eyes the former are workers (either operational, professional, or managerial) making use of an automated system in the accomplishment of their tasks. This reminds us of the fact that system developers are regularly heard complaining that users have no knowledge of their own needs. According to Mumford (1983), this problem frequently has to do with the abstractness of the context in which system developers try to elicit their clients needs. We would like to add that their own roles are often unclear to the users. Users may be committed in a passive way, but they do not directly see how they can claim an active role. Both the inadequate knowledge of the users and the indefinite nature of their roles stem from the fact that the orientation is all too often a technically determined one, whereas we can conceive of many SD scenarios in which the imperative does not lie with the technicians (e.g. Frese, Prümper, & Solzbacher, 1994; Robinson & Hayes, 1994).

Given the central role of system developers in SD, what is their own opinion of user participation? This question was treated in study A, which is reported in the next section. The results show that to a large extent system developers have embraced the concept of user participation. The majority are of the opinion that user participation is needed for effective SD. However, these findings confront us with three important problems. First, because of the technical imperative that was discussed in the last paragraph, user participation does not necessarily imply user influence on SD. Second, we cannot be sure that the stated preferences of system developers are reflected in their actual behaviour. Third, system developers may give a different meaning to the concept of user participation than social scientists.

Hence, we also needed study B in which the effectiveness of the interaction among parties in SD processes was studied. Studying the interactions among groups in SD has some advantages over studying user participation. First, because by doing this the technical orientation to which the term "user participation" bears testimony is replaced by a managerial orientation. Second, because it stresses the two-sided nature of the exchanges between system developers and their clients. The influence of future users can only be realized by interaction (whether written or oral, direct, or indirect), which is inevitably a two-sided phenomenon. The participation of a group in an SD process can have no direct effect on the outcomes of SD: It is the interaction among and within the groups involved that determines the outcomes. In this respect it is interesting to note that Brodbeck (1994) has reported that the negative correlation between user participation and project effectiveness was greatly diminished by communication density in the project. As Weick (1979) has pointed out, human interaction is crucial in situations where organizational members are faced with equivocality. While user participation may be common practice (Wijnen & van Oostrum, 1993), the involvement can still be largely symbolic (Baroudi et al., 1986).

A third reason is that we should look at users in a more differentiated way. "The users" as a homogeneous group of people does not exist and if it ever did, certainly does so no longer. Nowadays, the control of SD processes requires far more differentiated questions than whether or not users should participate.

Now we come to our fourth and final argument. As in decision making (Heller et al., 1988), the rationale for participation in system development has cognitive as well as political components: information is exchanged and power is exerted. *User participation refers to the interaction between* designer and user. The people who will work with the new system do not only have interests as users but also as employees (Doorewaard, 1985). As users they want systems that are (amongst other things) reliable and usable. As employees they are interested in subjects such as work security, payment, stress, and autonomy. If future users work together with system developers during the design phase, this interaction may not lead to the fulfilment of their interests as employees. Promotion of these interests calls for interaction between management and operational workers, and probably also between different intra- or inter-organizational groups. The concept of user participation diverts the attention from the political component of the interaction processes by sustaining a *unitary orientation* towards SD (Blacker & Brown, 1986), an orientation which does not fit practices that are potentially pluralist in nature and will be even more so in the future (for instance groupware, electronic data interchange, integration of information systems, business network re-engineering).

The organizational and cognitive psychological arguments that have been discussed thus far lead to the conclusion that an interaction model is needed that allows for:

- describing interaction in SD in a multidimensional way
- specifying the meaning of critical contextual factors
- adopting a managerial instead of a technical orientation
- stressing the two-sided nature of interaction
- making a distinction between different user and other groups involved
- displaying a pluralist orientation to SD as opposed to a unitary orientation
- discerning cognitive as well as political functions of interaction.

In study B, we will report the development of such a multidimensional contingency model in which both the interaction among the groups involved in SD and the context of the SD process can be described. The context is defined to consist of five types of risk to effective SD. Different risk profiles will call for different interactions among the groups involved in SD. We will describe some of the results of our explorative in-depth study of seven cases.

STUDY A: A SYSTEM DEVELOPERS AND "USER PARTICIPATION"

We administered a survey on the topic of organizing the information system development process. One of the aims of this study was to determine whether system developers favour a user-oriented working style, that is, to what extent do they prefer the participation of users. In this section we will present the data concerning the opinions of system developers about user participation in SD.

Method and Subjects

In order to determine whether system developers favour a user-oriented working style, we asked respondents to rate their personally favoured style of working on four five-point scale items with opposite poles: "Which working style do you yourself enjoy the most?" We wanted to know whether the differences in user orientedness are related to the type of employment (e.g. user organization, software firm) and type of automation (technical versus administrative). As can be seen in Table 1, other independent variables were the level and type of education, and years of experience.

The extent to which system developers enjoy a user-oriented working style does not tell us whether they perceive a positive relationship between user participation and the effectiveness of SD. Do system developers believe that user participation is needed for effective SD? We asked the respondents

TABLE 1
Variables in Study A

Dependent Variables	*Independent Variables*
User orientedness	Type of employment
Perceived ideal SD approach	Type of automation
Perceived critical causes of success	Level and type of education
Perceived critical causes of failure	Years of experience

whether they think that an ideal approach exists for system development. If the answer was "yes" we invited them to name the five most important elements of this approach. This question was open-ended, but the respondent could pick items from an enclosed list of 50 approach characteristics drawn from the SD literature. One of the listed characteristics was "active user participation". Furthermore, respondents were asked to report the perceived causes of the outcome of two automation processes in which they had participated; one successful, the other not. This question was also open-ended.

The addresses of 600 system developers with two or more years of experience were drawn in a random way from the database of a free journal widely read among automation practitioners. The usable response rate was 52% ($N = 314$). According to our comparison of the background variables of the database population and the respondents, we have no reason to believe that the non-response was selective. The fact that only a few of the questions concerned user participation reduces the possibility that subjects with a negative attitude to user participation did not respond.

Results of Study A

Almost 100% of the respondents see "good usability for the end users" as a critical criterion for the success of SD. An indicator "user-orientedness" was derived by summing the scores on the four items in Table 2 and dividing them by four. A high user-orientedness (score >3) was found for 226 respondents (72%). Table 2 shows that the majority (89%) personally prefer to make use of "active user participation", indicating that user participation is not often seen as a handicap by Dutch system developers. We did find that respondents in technical automation are less user-oriented than those who developed only administrative information systems (regression analysis, $P < 0.05$, 6% explained variance). A possible explanation may be found in the fact that subjects in technical automation reported cases with an average of 10 workstations, while subjects in administrative automation reported on average cases with more than 100 workstations. Because the former have to deal with fewer users, a user-oriented working style may be easier to

TABLE 2
Personally Favoured Working Style, Distribution of Percentages Per Item (N = 314)

*Personal Preference for . . .**	*Score 1, 2 (%)*	*Neutral 3 (%)*	*Score 4, 5 (%)*	*Personal Preference for . . .**
Minimal user participation	3	9	**89**	Active user participation
Few participating users	30	26	**44**	Many participating users
Automation expert as project leader	**43**	19	38	User as project leader
The role of an expert who devises the design and/or other products	26	15	**59**	The role of a supporter with expertise

*Five point scale: 1, 2 = preference left-side alternative; 4, 5 = preference right-side alternative; 3 = neutral.

realize in technical automation. However, it does not say anything about the intensity of user participation needed in technical automation. On the whole the independent variables (Table 1) explained less than 10% of the variance in preference for a user-oriented working style.

Two hundred and nine respondents think that an ideal approach for SD does exist. In Table 3 the most frequently mentioned characteristics of their "ideal approach" are listed. "Active user participation" is top of the list; 43% mention it as an essential element.

In 26% of the reported successful automation processes (N = 283) user participation was declared to have been a critical success factor. In 20.6% of the processes that were not successful, a shortage of user participation was attributed to be a critical cause of failure (N = 265). In only two of the processes reported was "too much user participation" attributed to be a cause of failure. More generally, the respondents referred to communication, including user participation, as a critical factor in 47% of the successful cases and in 28% of the failed cases.

TABLE 3
The Most Frequently Mentioned Characteristics of "The Ideal Approach" (N = 209)

Characteristic	*% of Respondents*
Active user participation	42.5
Realistic planning and alert monitoring	38.2
A good atmosphere in the team	23.7
Explicit definition of aims and expectancies	22.7
A multidisciplinary team	20.3

Discussion

From these data it can be concluded that the majority of Dutch system developers prefer a user-oriented working style. Moreover, many system developers think that user participation is needed for effective SD. This result is in accordance with Cressey and Williams (1991), who have reported that the user participation that did actually take place is by and large evaluated positively. Nevertheless, our study has some limitations.

The question remains whether the stated preference of the system developers is in accordance with their behaviour. Heinbokel (1994) has found that high user-orientedness does not necessarily lead to more interaction with users.

Further, system developers may have something different in mind when they speak about user participation than social scientists. The results of Gould and Lewis (1987) suggest that system developers in the United States identify, stereotype, or describe who the users are, rather than trying to understand them. They hear the users and read what others wrote about them instead of getting into direct contact with the users. The system developers present their design to the users and let them sign off on their design instead of interacting prior to the design phase.

Finally, user participation does not necessarily enable employees to have a substantive influence on the systems with which they will perform their tasks. This will depend on the interaction realized during SD. In a European survey, Cressey and Williams (1991) interviewed 7326 persons about user participation. They found that user participation became substantial rather late in the process, when many decisions had already been made and the room for influence had become limited (see also Vijlbrief, Algera, & Koopman, 1986).

The acknowledgement of system developers that user participation is needed is important, but as in the three problems mentioned previously show, this finding results in an array of questions: Which categories of potential users interact with whom, when, how, and to what purpose in effective SD, and which contextual factors influence these relationships?

STUDY B: FROM USER PARTICIPATION TO CONTINGENT INTERACTION

Method and Cases

In this study we have explored how a match could be realized between contextual characteristics and SD approach, in order to develop a successful information system. Existing theory was used to develop a contingency framework, composed of three main variables: context, approach, and outcome (van Offenbeek, 1993). Within this contingency framework, propositions about the matching of context and approach were formulated and tested preliminarily (van Offenbeek & Koopman, in press). One aspect of an SD approach is the choice of which interaction should take place among the groups involved. Within the scope of this article we will not draw upon the other aspects of an SD approach—see van Offenbeek and Koopman (1996).

The case material consists of the retrospective analyses of seven processes in which a data processing and/or operational and/or tactical management information system was developed. The cases were selected on the basis of two criteria: (1) the SD process should involve at least some social and/or organizational issues, and (2) variance among the cases in context variables. Because we were confronted with substantive changes in the context and/or approach variables in some cases, we subdivided those cases into different episodes to be analysed separately. This resulted in 10 episodes, of which five could be considered to be failures and five to be successes (Table 4).

Data collection took place during the last phases of SD and consisted of semi-structured interviews with key actors (developers, users, management) and by the analysis of documents. Six months after the implementation of the system, questionnaires were filled out by users and managers. The data from the interviews and the documents were used to determine the characteristics of context and approach. Three judges rated the cases according to the context variables. The mean interjudge agreement, corrected for coincidence, was satisfactory: 0.87. The questionnaires were used to evaluate the outcome of the SD process. The indicator for the success of SD was whether the resulting information system was actually implemented and used on a regular basis—a dichotomous measure.

TABLE 4
Total Scores for Risks and Measures Taken in 10 Episodes of SD

Case	*Total Score Risks (max. 8)*	*Total Score Approach Measures (max. 40)*
Successes		
A2	0	12
B2	1	21
C2	2	27
D2	6	30
E	6	35
Failures		
B1 (system rejected)	1	5–9
C1 (project cancelled)	6	16
D1 (major stagnation)	7	18
F	3	16
G	4	16

Because the scope of the article is confined to the interaction with and among users groups in SD, we will limit ourselves to the introduction of the context and interaction variables.

An Interaction Model in SD Processes

We will describe two components of the contingency framework: the interaction model and the context variables (see Fig. 1). Ashmos, McDaniel, and Duchon (1990) formulated a model to describe the type of participation in decison making. We adapted this model in order to make it suitable for the description of the interaction within SD. The number of context factors is almost infinite. We limited ourselves to those context factors that cause a risk in terms of the effectiveness of the SD process. According to contingency theory, these risks can be controlled by choosing a fitting approach. This means that the interactions, as one of the aspects of SD approach, should match the risks in a specific SD process. We distinguished five risk types that can be used to specify the context of the information system development process.

Attributes of the Interaction Model

First it should be established who could potentially be involved in SD. The *number of people* having to interact together can vary strongly depending on the context and the function of the interaction. For some

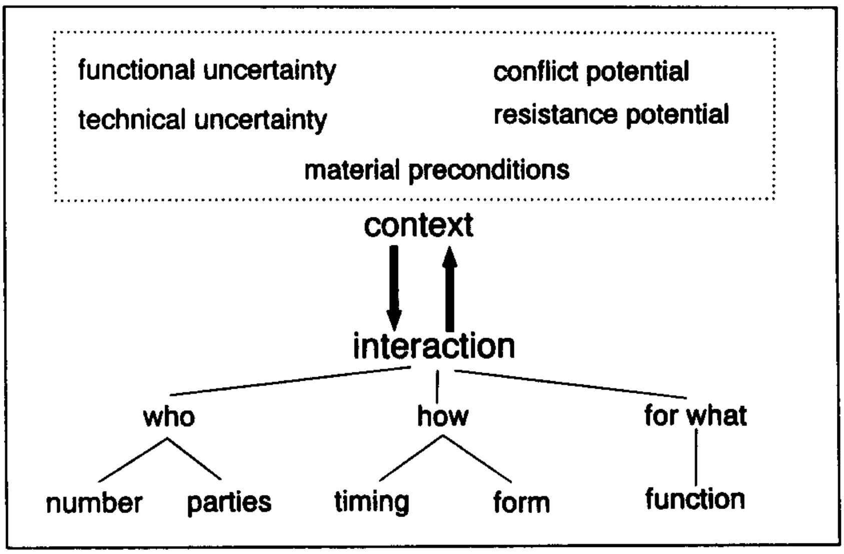

FIG. 1. Context factors and interaction model in SD.

activities in SD more participants will lead to a better result: the greater the number of direct users learning to handle the system, the more effective the implementation. For other activities there will be an optimum: when future users have to test the programs constructed, one user will not be sufficient to track down all the bugs and inconveniences. For most of the time, however, it would be highly inefficient if 50 future users were to test the system. Still, for some activities more interaction will be less effective: one person should be responsible for the conceptual integrity of the system.

The next attribute is formed by the *parties* who have to interact. This attribute also varies according to the context and the development activity involved. Parties that may be involved in SD are the board of directors, middle line management, automation experts, personnel managers, trade union representatives, and, of course, the (operational and managerial) workers who will use the system. The relevant groups should be assessed locally. Users should not be treated as a homogenous group. They will often have to be split into functional, geographical, vertical and/or horizontal categories. The same goes for the technicians: "why, with whom, and how" will be answered differently for a database specialist, a programmer, an information analyst, or an information systems manager.

We are not only concerned with the question of who has to interact with whom but also with the attributes describing how interactions are to be realized. The *timing of the interaction* determines in which phase(s) the interaction takes place. We have chosen a global life cycle model specifying the most important activities: initiation, analysis, design, realization, implementation, and use and maintenance. Sometimes implementation will be the starting point of a subsequent life cycle (incremental change). Moreover, in processes with many iterations among activities the distinctions between successive stages within one life cycle will be vague.

Many different *forms* of interaction can be found in SD. The form is defined by the extent to which the interaction is either formal or informal (Ashmos et al., 1990), direct or indirect (Blackler & Brown, 1986), and by its intensity (Vroom & Jago, 1988). The form of interaction can be very important. For instance, as formal task descriptions often do not describe tasks accurately, local observation and analysis of the actual task activities has been suggested (e.g. Frese, 1987; Rogard, 1990).

The last, but by no means the least, important attribute is the *function of interaction*. Ives and Olson (1984) mention two global aims of user participation: to realize increased system quality and to heighten user acceptance of the system. We have extended these into four functions of interaction, the first two cognitive, the last two political: factual exchange, learning, motivating, and negotiating.

The realization of high system quality calls for interaction directed towards the *factual exchange* of information. For example, exchanges between users and automation experts in order to establish the specifications, between programmers working on different subsystems to establish interface specifications, or between management and workers about the planned change. Most of the time *learning processes* are also needed to achieve this aim. They lead to a better insight for the parties into the specific system and a better understanding of information systems in general. Learning processes may be needed in order to reach a better understanding of (1) goals, problems, and the functioning of their own organization, (2) the meaning of the changes proposed, (3) the functions automation can and cannot fulfil, and (4) the consequences automation will have for the organization.

The second of Ives and Olson's aims concerns the *motivational function* of interaction. Interaction is critical in establishing a high level of acceptance. Organizational members should feel themselves to be the owners of the system and all parties should feel committed to the system's goals and concept. Usually, conflicting points of view, interests and incompatible needs and practices will occur. They can lead to conflicts about the development that have to be recognized and tackled. This brings us to the fourth function of interaction: Interaction can also facilitate processes of *negotiation and power*.

Of course this classification is an analytical one. Empirically the distinction between the functions can be blurred. Sometimes interaction will be directed towards different functions at the same time, sometimes the functions will concern interaction activities that are clearly separated in time and place.

The Context of SD: Five Risk Types

Once in a while SD processes will occur that are relatively simple and to which principles of linearity, objectivity, and technical rationality can be successfully applied. In these circumstances, the dominant actors can choose the most efficient approach, because the dominant actors do not have to take measures for risks that are low. That is what we have called the "efficiency proposition". However, perhaps more often than not, organizational reality is not simple and ordered in a known way, but complex, ambiguous, and unstable. Moreover, not everyone has the same objectives, interests, and views. In addition, system developers can be confronted by time and money constraints. Thus, risks can be cognitive, political, or material in nature. The following five risk types were distinguished: functional and technical uncertainty (cognitive), conflict potential and resistance potential (political), and material preconditions.

Functional Uncertainty refers to the risk that the actors will choose the wrong solution or solve the wrong problem. The magnitude of this risk is determined by the characteristics of the task system in the existing situation and of the (expected) changes to the task system. A high complexity, a low stability of the tasks, and having no acquaintance with the tasks to which the system development is directed, will heighten the functional uncertainty with which the system developers are confronted, as will obscurity of the problem(s), unknown goal(s) or needs, and the absence of criteria by which the solution will be judged. Two other potential factors are the anticipated extensiveness of the changes in the task system, and a lack of experience with SD on the part of the organizational members.

Technical Uncertainty refers to the risk that the conceptualized solution cannot be realized. The magnitude of this risk is determined by the characteristics of the technological aspect system in the existing situation, and by the technological aspects of the change. In system development this risk increases when the existing technological system is complex and relatively new; when technical experts are unacquainted with the software environment, the complexity of the system realization is high, and the quality and commitment of the technical experts is low.

Conflict Potential refers to the risk that incompatible needs and interests will hamper problem solving. It is determined by the degree of pluralism in the existing structure as compared with its desired uniformity. This type of risk is increased when more parties whose ideas, language, and/or interests are heterogenous are involved and when the scope of the SD process (in terms of people and finance) is large. This risk is also increased when the required integration among the parties is high and when the development is dependent on third parties or on the results or progress of other projects.

Resistance Potential refers to the risk that members of the organization will be dissatisfied with the realized solution, because they feel its implementation would decrease the quality of their working life. The magnitude of this risk is determined by the changeability of the organizational members concerned, compared with characteristics of the wanted change. The risk is increased when the workers (management) have a low change potential, a low willingness to change, and when the qualitative and quantitative impact on the work organization is high.

Material Preconditions refer to the risk that the SD process will not pay for itself or will be aborted prematurely due to lack of resources. This risk is defined as the amount of energy needed, as compared with the amount available, i.e. budget, in terms of human, machine, and computer resources, time pressure, and the importance of the SD process. Material preconditions define the extent to which an approach needs to be efficient.

Results of Study B

In this section we will first give a brief summary of the most important results of the study. More detailed results about the contingency framework as a whole and the individual cases can be found in van Offenbeek and Koopman (in press). Next, we will concentrate on the way in which a fit between context and interaction was realized. These results are tentative, as the study was explorative and consisted of only 10 episodes.

The analysis examined whether one or a combination of risk types, or one or a combination of approach characteristics, could explain the outcome of the 10 SD processes (Table 4). This was not possible. However, the match between the risk factors and the approach characteristics was able to explain the success or the failure of each case. Thus, we found some support for the efficiency proposition: In the case of one or more low risks, an efficient approach, that is an approach without measures to control these risks, led to success. When faced with high risks, a positive outcome was only found when measures to control them had been taken. Our findings suggest that when moderate risks are perceived, control measures should also be taken.

Interaction in the Case of High Functional Uncertainty. When functional uncertainty is high, users will have to interact at an early stage with the automation experts, because early in the process the capacity to process information will be highest (Davis & Olson, 1985); people are more open minded as many decisions have not yet been taken. The interaction must be directed towards the exchange of information and learning. However, when the resistance potential is also high (see later), and no room is created for negotiations, the interaction between users and developers during SD will turn into "mock participation". In this case interaction between users and system developers will be counterproductive (Algera et al., 1989; Markus, 1983; Söderberg, 1986).

Interaction in the Case of High Technical Uncertainty. This risk requires interactions among the system developers. Exchange of information and collective learning will be the most important functions of the interaction. As in the case of high functional uncertainty, the exchange of information alone will not be sufficient to control high technical uncertainty. Learning will have to occur because the participants have little experience with the technology and methods and/or have to improve their skills. Proven technology and/or methods are not available. Because of the learning function, we found that interaction cannot be confined to written communication and formal meetings. Both formal and informal interaction is necessary. Line management and SD management should interact to inform each other of the feasibility of the proposed solution as well as of the available alternatives.

Interaction in the Case of High Conflict Potential. When heterogenous groups are involved it is more difficult to create shared meaning and consensus because group members will be less prone to behaviour favouring and supporting other groups (Algera et al., 1989; Vroom & Jago, 1988). The values and norms will be questioned. Therefore, in cases with a high conflict potential, prior to interacting with system developers about specifications, representatives of the organizational groups involved should interact with one another. Representational forms of interaction will prohibit a Tower of Babel-like chaos. Formal co-ordination of the interactions is necessary to unequivocally take and record decisions and communicate them to the grassroots and others involved. This early interaction should be directed towards collective learning and negotiation. In this way ideas and experiences can be exchanged, differences in points of view and interests can be explicated, and the goals of the SD process can be negotiated. Negotiation will be the most important function of the interaction. It will be the responsibility of the dominant actors to manage the process towards shared meaning. Bouwen and Fry (1991), indeed, found a learning-confrontation strategy to be more successful in these circumstances.

Interaction in the Case of High Resistance Potential. The responsible line managers will have to interact with all the workers to generate the commitment needed (Vroom & Jago, 1988). The most important function of this interaction is enhancing motivation. Moreover, resistance has a cognitive base as uncertainty heightens resistance. Therefore, a sufficient exchange of information between the dominant actors and all those involved is necessary. Everyone should know what is going to happen when, and what consequences this will have on the quality of his or her work. Furthermore, in order to ensure that workers will be able to function effectively in a changing or new work environment, individual learning, e.g. courses, is necessary. A short time-lag between learning activities and implementation prevents people forgetting their new knowledge and skills and returning to old habits and methods.

Material Preconditions define the extent to which the SD process has to be efficient. If resources are limited, one is forced to be economical with the number of people that engage in interactions. When the preconditions are insufficient to realize the necessary interaction activities, the process should be redefined in order to make the SD context less risky. This could be accomplished by limiting the target groups or the functional purposes of the system.

CONCLUSIONS

The majority of (Dutch) system developers rate the value of user participation highly (study A). However, we have argued that additional questions about the issue of user participation would benefit from research into the interaction processes within SD, and that these interaction processes should match the risk factors of a specific SD process (study B). To sum up, by using the interaction model presented, some advantages are to be gained.

- Interaction can be specified between different groups at different phases for different reasons and in different ways.
- Alternative possibilities for interaction with and among users can be explicated.
- Responsibilities for interaction can be located in time and towards function.
- The interaction model offers a broader concept than that of user participation, fitting the pluralist nature of current development practices better.
- The model stresses the dual nature of interaction.

We believe that this model, together with the risk types, enables the formulation and testing of contingency hypotheses concerning the interaction needed for effective SD. In our study we have only made a start (van Offenbeek & Koopman, in press). The practical purpose of such research will be to offer practitioners more accurate contingency guidelines concerning the interaction with and among groups involved in SD. Nevertheless, the effectiveness of SD is not only dependent on the right kinds of interaction, but also on other aspects of the SD approach, notably the definition and orientation of, and the differentiation and co-ordination within, the SD process (van Offenbeek & Koopman, 1996). The interaction in SD should also be congruent with these other aspects of the approach.

ACKNOWLEDGEMENT

We would like to thank Prof. Dr M. Frese for his comments on an earlier version of this article.

REFERENCES

Algera, J.A., Koopman, P.L., & Vijlbrief, H.P.J. (1989). Management strategies in introducing computer-based information systems. *Applied Psychology: An International Review, 38*, 1.

Ashmos, D.P., McDaniel Jr., R.R., & Duchon, D. (1990). Differences in perception of strategic decision-making processes: The case of physicians and administrators. *Journal of Applied Behavioral Science, 26*(2), 201–218.

Baroudi, J.J., Olson, M.H., & Ives, B. (1986). An empirical study of the impact of user involvement on system usage and information satisfaction. *Communications of the Association for Computing Machinery, 29*(3), 232–238.

Björn-Andersen, N., Eason, K., & Robey, D. (1986). *Managing computer impact: An international study of management and organisation*. Norwood, NJ: Ablex Publishing.

Blackler, F.H.M., & Brown, C. (1986). Alternative models to guide the design and introduction of new information technologies into work organisations. *Journal of Occupational Psychology, 59*, 287–313.

Bouwen, R., & Fry, R. (1991). Organisational innovation and learning: Four patterns of dialogue between the dominant logic and the new logic. *International Studies in Management and Organisation, 21*(4), 37–51.

Brodbeck, F.C. (1994, July). *Empirical results about communication and performance in software development projects*. Paper presented at the 23rd international congress of Applied Psychology, Madrid, Spain.

Child, J., & Loveridge, R. (1990). *Information technology in European services: Towards a micro-electronic future*. Oxford, UK: Basil Blackwell.

Clegg, C., & Symon, G. (1989). *A review of human-centred manufacturing technology and a framework for its design and evaluation* (Sheffield Memo. No. 1036). University of Sheffield.

Cotton, J.L., Vollrath, D.A., Lengnick-Hall, M.L., & Froggatt, K.L. (1990). In fact the form of participation does matter—A rebuttal to Leana, Locke and Schweiger. *Academy of Management Review, 15*(1), 147–153.

Cressey, P. (1989). *Trends in employee participation and new technology*. Glasgow, UK: University of Glasgow.

Cressey, P., & Williams, R. (1991). *Inspraak in verandering; nieuwe technologie en de rol van inspraak van werknemers* (Report No. SY-58-90-384-NL-C). Luxemburg: Bureau voor officiële publikaties der europese gemeenschappen.

Davis, G.B., & Olson, M.H. (1985). *Management information systems*. New York: McGraw-Hill.

Doorewaard, J.A.C.M. (1985). Kantoren in actie. In *Technologie en arbeid in de jaren tachtig en negentig*. Nederland: FNV/cmhp.

Edström, A. (1977). User influence and the success of MIS-projects: A contingency approach. *Human Relations, 30*, 589–607.

Frese, M. (1987). Human–computer interaction in the office. In C.L. Cooper & I.T. Robertson (Eds.), *International review of industrial and organisational psychology*. Chichester, UK: Wiley & Sons Ltd.

Frese, M., Prümper, J., & Solzbacher, F. (1994). Eine Fallstudie zu Benutzerbeteiligung und Prototyping. In F.C. Brodbeck & M. Frese (Eds.), *Produktivität und Qualität in Software-Projekten*. Munich, Germany: Oldenbourg.

Gould, J.D., & Lewis, C. (1987). Designing for usability key principles and what designers think. In R.M. Baecker & W.A.S. Buxton (Eds.), *Readings in human–computer interaction: A multidisciplinary approach*. San Mateo, Calif.: Morgan Kaufmann.

Hedberg, B. (1975). Computer systems to support industrial democracy. In E. Mumford & H. Sackman (Eds.), *Human choice and computers*. Amsterdam: North Holland.

Heinbokel, T. (1994). Benutzerbeteiligung Schlüssel zum Erfolg oder Hemmschuh der Entwicklung? In F.C. Brodbeck & M. Frese (Eds.), *Produktivität und Qualität in Software-Projekten*. Munich, Germany: Oldenbourg.

Heller, F.A., Drenth, P.J.D., Koopman, P.L., & Rus, V. (1988). *Decisions in organisations: A three country comparative study*. London: Sage.

Heming, B.H.J. (1992). *Kwaliteit van arbeid, geautomatiseerd*. Unpublished doctoral dissertation. Faculteit der Wijsbegeerte en Technische Maatschappijwetenschappen, Delft, The Netherlands.

IDE. (1981). *Industrial democracy in Europe*. Oxford, UK: Clarendon Press.
Ives, B., & Olson, M.H. (1984). User involvement and MIS success. A review of research. *Management Science, 30*, 586–603.
Markus, M.L. (1983). Power, politics and MIS implementation. *Communications of the Association for Computing Machinery, 26*, 430–444.
Mumford, E. (1983). *Designing human systems for new technology: The ETHICS method*. Manchester, UK: Manchester Business School.
Offenbeek, M.A.G. van. (1993). *Van methode naar scenarios*. PhD thesis, Free University, Amsterdam.
Offenbeek, M.A.G. van, & Koopman, P.L. (1996). Interaction and decision-making in project teams. In M.A. West (Ed.), *Handbook of work group psychology*. Chichester, UK: Wiley.
Offenbeek, M.A.G. van, & Koopman, P.L. (in press). Scenarios for system development: Matching context and approach. *Behavior and Information Technology*.
Oostrum, J. van, & Rabbie, J.M. (1988). Inspraak en effectiviteit; een contingentiebenadering. *Gedrag en Organisatie, 1*(2), 55–70.
Riesewijk, B., & Warmerdam, J. (1988). *Het slagen en falen van automatiseringsprojecten*. Nijmegen, The Netherlands: Nijmegen instituut voor toegepast sociaal wetenschappelijk onderzoek.
Rogard, V. (1990, September). *How and why to improve the computer design methods by work analysis*. Paper presented at the workshop on Technological change process and its impact on work, Siofok, Hungary.
Schonberger, R.J. (1980). MIS design: A contingency approach. *MIS Quarterly, March*, 13–20.
Söderberg, I. (1986, May). *Office work and office automation—user participation under change*. Paper presented at the international scientific conference on Work with Display Units, Stockholm, Sweden.
Vijlbrief, H.P.J., Algera, J.A., & Koopman, P.L. (1986). *Management of automation projects*. Paper presented at the 2nd West European conference on the Psychology of Work and Organisation, Aachen, Germany.
Vroom, V.H., & Jago, A.G. (1988). *The new leadership managing participation in organisations*. Englewood Cliffs, NJ: Prentice-Hall.
Weick, K.E. (1979). *The social psychology of organizing*. Philippines: Addison-Wesley.
Wijnen, B.J., & Oostrum, J. van (1993). Een contingentiemodel voor de invoering van geautomatiseerde informatiesystemen. *Mans en Onderwering, 2*, 88–103.