10

Soft Systems and Hard Contradictions*

Lars Mathiassen
Peter A. Nielsen

Abstract. Checkland's Soft Systems Methodology for defining and solving problems in organizations is presented on the basis of an application of the methodology. In many ways this approach represents an improvement in relation to more traditional approaches to computer-based information systems.

Based on an experiment we explore the possibilities of combining the practical usefulness of soft systems thinking with the analytical power offered by dialectical thinking. Soft Systems Methodology already shares several basic elements with dialectical approaches to social theory and action. It does not, however, include the key element of dialectical analysis: explicit thinking in terms of contradictions.

The paper presents experiences, ideas and arguments, but it contains no final answers. At this stage it is our intention to engage others in critical and constructive reflection on further developments of Soft Systems Methodology and on dialectical approaches to computer-based information systems in organizations.

Keywords: computer-based information systems, Soft Systems Methodology, dialectics, contradiction.

* Published as: Soft Systems and Hard Contradictions - Approaching the Reality of Information Systems in Organizations. L. Mathiassen & P. A. Nielsen. In: Journal of Applied Systems Analysis, Vol. 16, 1989.

1. Introduction

Checkland's Soft Systems Methodology, SSM (Checkland 1981) has in general proved powerful as a systemic approach to organizational change, cf. (Checkland 1985). Therefore, it is only natural to consider SSM as a more specific approach to computer-based information systems in organizations (Wilson 1982; Schäfer *et al.* 1986), Within information systems we are confronted with challenges on several levels. Our first and most deep concern is the use of information systems in organizations. On this level SSM can be used to analyze, design, and change overall strategies and specific applications. Secondly, we are concerned with the process of changing organizations through development of specific computer-based systems. Here SSM can be used to understand the situations involved and to improve our ability to design and manage systems development projects. Finally we are concerned with improving the environments in which systems development takes place. Particularly, we want to understand and develop the competencies, traditions and working practices involved in information systems development. At this third level SSM can be used to introduce new methods and to change the working practice and tradition within the development organization.

Personally, we have become fascinated by the elegant way in which SSM combines in an elegant way insight into the nature of human activities and organizational learning with fundamental elements of systemic thinking. Earlier we have tried to enrich the area of computer-based information systems by using Schön's ideas on reflection in action (Schön 1983; Lanzara *et al.* 1985; Jepsen *et al.* 1989), and by using various dialectical approaches to social theory and action (Israel 1979; Mathiassen 1981, 1987; Mathiassen *et al.* 1988). From what we have learned through these studies, SSM looks as a constructive and attractive approach to some of the challenges within our field of interest.

SSM shares several basic elements with dialectical approaches, but does not include the key element of dialectical analysis: explicit thinking in terms of contradictions. Experiences using SSM shows that it is useful to think in terms of soft systems to support organizational learning, but organizational actors face many hard contradictions in their daily activities, i.e. the contradiction between efficient use of resources and quality of outcome in systems development or the contradiction between established organizational forms

and needs to change. We therefore raise the question: to what extend and by what means could one improve SSM by including explicit thinking in terms of contradictions? Based on this question we have set up a project within a Danish bank, and what we report here is mainly what we learned from this exercise.

Section 2 is an account of our use of SSM in its present form—in connection with the introduction of a new systems development method in a bank. This case demonstrates the usefulness of SSM in relation to computer-based systems and it illustrates at the same time the dialectical flavor of SSM. Section 3 contains an overview of SSM and a discussion of its dialectical elements. In section 4 we return to our case and show how we attempted to think in terms of contradictions. We also give an account of what we learned from this experiment. Finally, in section 5, we discuss the methodological challenge discussing whether and how one could use thinking in terms of contradictions within SSM. Section 6 contains our conclusions and we relate to the ongoing debate about the nature of SSM.

The reader should be aware that this paper addresses two audiences: the SSM community and the information systems community. Some of the material presented might be trivial to part of the potential readers. We ask each reader to judge and to concentrate on those parts of the paper which is of relevance.

2. Soft systems in practice

We are in The Merchant Bank of Provident, one of the largest Danish banks. Like most banks, it has a huge computer installation for on-line handling of the accounts in all branches, and there is an increasing number and variety of computer applications to support the activities of the bank.

We have been called upon to help the methodology group make sense of and sort out what they believe to be an unstructured and problematic situation. The methodology group is responsible for the methods used in the DP department by about a hundred systems developers. Their work includes such things as: giving courses in data-flow modeling and project management, attending work shops and seminars to learn about new or improved methods, supporting and advising projects on how to develop computer-based information systems, and finally they define and maintain a set of standards and guidelines for developing information systems (see figure 1). In

the last fourteen odd months the group has been engaged in the adaptation of a general method for data modeling, (cf. Flavin 1981), to the specific needs of information systems development in The Provident. The adapted method is based on entity-relation modeling and has recently been tried out in a few projects. The group now feels that they are as ready as they can be to make all projects in the department use the method. At this point, the methodology group (and others) perceive a variety of problems related to the data modeling method:

- How can the method best be taught to systems developers?
- Will the systems developers use it afterwards? Can it be ensured that they do? Should it be?
- What is the role of the methodology group in taking the adapted method into use?

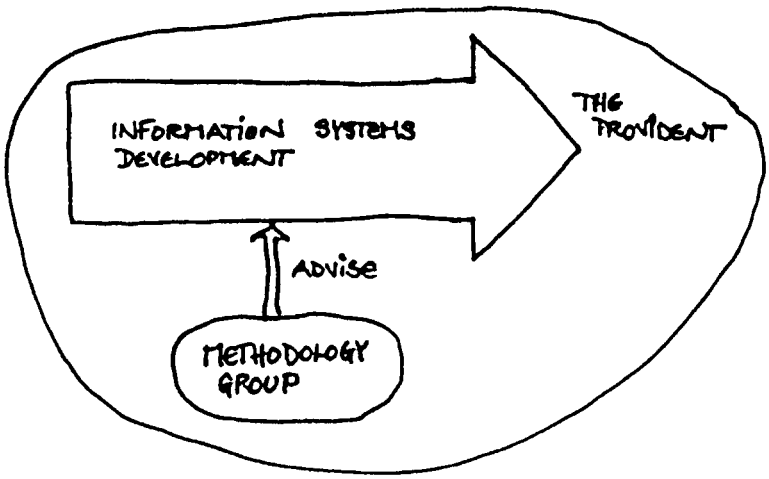

Figure 1. Methodological support and advise of ISD (after Checkland 1985).

Confronted with this unstructured situation the methodology group is most willing to let us help them in structuring and hopefully improve the situation. And applying SSM certainly is relevant in this situation (Checkland 1981, p. 149 ff.).

Following the ideas of SSM we immediately become reluctant to accept stated problems at face value: the starting point of SSM is

a problematic situation, i.e. a situation where some actors perceive that the situation needs to be improved. Being as open-minded as possible we find out about the situation by making what has become known as a *rich picture.* Without trying to impose a particular problem on the situation we look at elements of slow-to-change *structure* within the situation and elements of continuously-changing *process,* and from there we evaluate the *climate* by forming a view of how structures and processes relate to each other (Checkland 1981, p. 163 ff.).

This initial finding out in the DP department is done by interviewing systems developers, the methodology group, and DP management. After several attempts and iterations we arrive at five different rich pictures, one of which is found in figure 2. This rich picture describes our personal account of the situation without us-

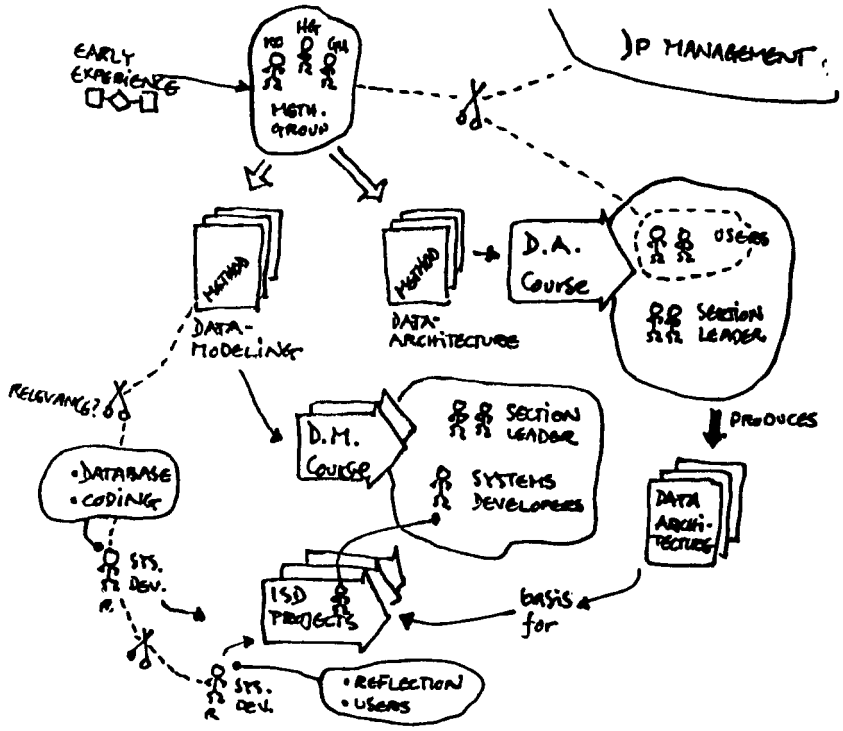

Figure 2. A rich picture of a problematic situation in the DP department of The Provident.

ing any explicitly defined syntax and semantics. The picture shows part of the situation where the methodology group have worked with and derived two methods: data modeling and data architecture (structure). Both methods are based on representing and understanding data as entities and relations. They are taught in courses (process), where systems developers are attending the data modeling course to learn the detailed modeling techniques to be used in development projects, and where users are attending the data architecture course in order to be able to produce the architecture of all data in The Provident. The two methods are intrinsically related: the idea is to make and maintain one large description that contains all relevant data in the bank, and this architecture is then to be used as the basis for data administration and data analysis in all development projects to come over the next decade. DP management finds that it is not a very good idea to let the users be too much involved in the making of the architecture while the methodology group finds that the architecture cannot be made properly without relying heavily on the users understanding of the bank. This actually constitutes an important climate as the relation between the methodology group and management is severely troubled by this dispute (the crossed swords). In the bottom left corner of the rich picture two roles are shown, illustrating two different archetypes of systems developers. One will speak about their work in terms like: “reflection”, “analysis and understanding of the use-situation”, and “involving users”. To this role the new method of data modeling is very appealing. The other will speak about their work in terms of: “coding modules”, “designing databases”, and “constructing systems”, and they find data modeling irrelevant to what matters in systems development. Again the climate between these roles and the data modeling course is important, since all systems developers are expected to attend the course and use the method afterwards.

Having done some finding out we proceed by selecting aspects of the situation to be analyzed in a more profound manner. We suggest a number of hopefully relevant *root definitions* (RD), each giving a precise account of a human activity system specifically relevant to the situation at hand. For instance, the following two root definitions are found relevant, but at this stage we do not know whether they are useful.

RD_3: A DP department-owned system to teach data modeling to systems developers by an interaction between the methodology group's courses and application of the method within appropriate areas of the bank, the interaction being controlled by an evaluation of the method.

RD_5: A DP department-owned system where the methodology group gives knowledge and advise about information systems development to systems developers in order to improve their possibilities for handling resources and product quality in systems development.

One of the most important elements of our root definitions are their *Weltanschauung* (or just plain W). The W is what makes the selected system meaningful—in RD_3 it is the belief that data modeling can be learned by an interaction between theoretical courses and practical application, without which RD_3 would not be worth doing. We are, in fact, making root definitions in order to see the consequences of taking a particular W, and more specifically we do this by building a *conceptual model* of each of the systems described in the RDs.

Conceptual models (CMs) are graphical descriptions of what a system does, where a RD describes what the system is. A conceptual model is made for each RD, and it must reflect the RD and not the situation. A conceptual model is made up of the minimal set of activities necessary to do the transformation described in the RD (as an intention), see figure 3. The model (not being a description of the situation) must be defensible against the root definition, which means that the *raison d'être* for each activity and logical dependency ($\rightarrow$) in the model can be argued from the RD. It is, for example, necessary in CM_3 to find an appropriate area of the bank before the method can be applied in it. At this stage we move back and forth between formulating RDs and building CMs sketching, detailing, and redefining both parts.

Until now we have mainly found out about the situation by interviewing actors and the RDs and CMs have so far merely been expressions of our personal systems thinking. We proceed by returning to the problematic situation by confronting the methodology group with the RDs and CMs. This makes us realize that some of our models (not shown) do not give rise to any debate, hence they are irrelevant and not useful and we throw them away. We iterate

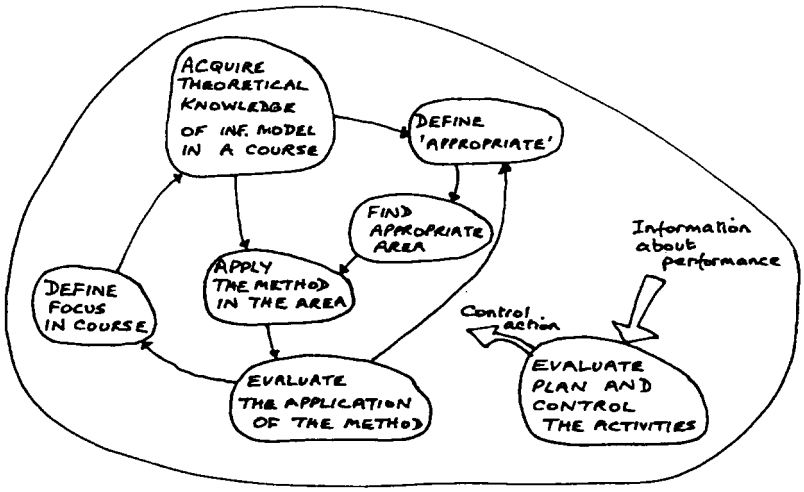

Figure 3. A conceptual model of RD_3.

by doing some more finding out, this time mainly in cooperation with the methodology group, and afterwards they suggest some new RDs and together we build the corresponding CMs. This close cooperation is possible in this case because the methodology group at the same time is eager to learn SSM.

We are using two approaches to organize the *debate* based on the models: discussions in the methodology group and public debates. In the discussions in the methodology group we try to use the models as a vehicle for learning about the situation:

- Are the properties of the system, seen as a whole, desirable?
- For each activity we ask:
  - How is it done?
  - Is it done well?
  - What are the criteria for "done well"?
  - How can it be improved?

The discussion of the RDs and CMs highlights many shortcomings and possibilities of the situation, and even the simplest question sometimes cause a major debate amongst the members of the methodology group. The *comparison* leads to the identification of differences between the models and the real world and to suggestions for improvements, e.g. in relation to RD_3:

- "Find appropriate area" is not performed as the method is now used without any evaluation of its applicability. A suggestion is therefore to find a way to do this activity.
- There is no real feedback from the evaluation of the application to "Define appropriate" and "Acquire theoretical knowledge of information modeling in a course".
- "Evaluate, plan and control the activities" is not carried out in any systematic manner. The performance of the activities is poorly monitored leading to poor evaluation and planning. A proposal for improvement is to monitor the activities more systematically.

The public debate could be organized like the discussions in the group, but we choose not to go into the same detail as it is a debate amongst all actors concerned in the DP department (systems developers, methodology group, and management). The ideas in the RDs and CMs are presented in a more popular form than shown in figure 3, and the following debate is therefore somewhat looser in its form, focusing on competing and even conflicting ideas and viewpoints.

The purpose of the debate is to decide on *action*. The immediate outcome is not a consensus on what actions to take to improve the situation. Instead, the debate results in a list of possible actions. From there we use two criteria to reach accommodation on what specific actions to take. The defined changes must at the same time be both systemically desirable and culturally feasible. For instance, it is found that "find appropriate area" allows for action to be taken. Whereas a more systematic monitoring of performance fails as it is not culturally feasible. In the present situation, systems developers are loosing autonomy and they find it unacceptable to reveal to the whole DP department how they do things.

Being now in the end of the inquiry we can, in retrospect, see that *iteration* of the different activities is an important feature of SSM. On the one hand, we have learned and debated by means of systems ideas. This debate, on the other hand, have enabled us to restructure and suggest new RDs and CMs which in turn causes new debate and insight. During the whole course of inquiry about twenty root definitions have been used and about ten of them were considered useful, each leading to between two and ten minor or major desirable and feasible actions to improve the situation. At the

end the situation is believed to have improved, even though it is still considered to be problematic.

3. The dialectics of soft systems methodology

We proceed by looking more closely at the activities of SSM—in its present form—and especially at those elements that gives SSM a strong dialectical flavor.

Checkland basically distinguishes between a hard systems approach as applied by engineers when facing well-structured, technical problems, and his own soft systems approach, which is an attempt to apply systems thinking to ill-structured problems in human activity systems. The activities of SSM, as illustrated by the case in section 2, are summarized in the form of a diagram, see figure 4. The presentation of the approach given in his book is, in

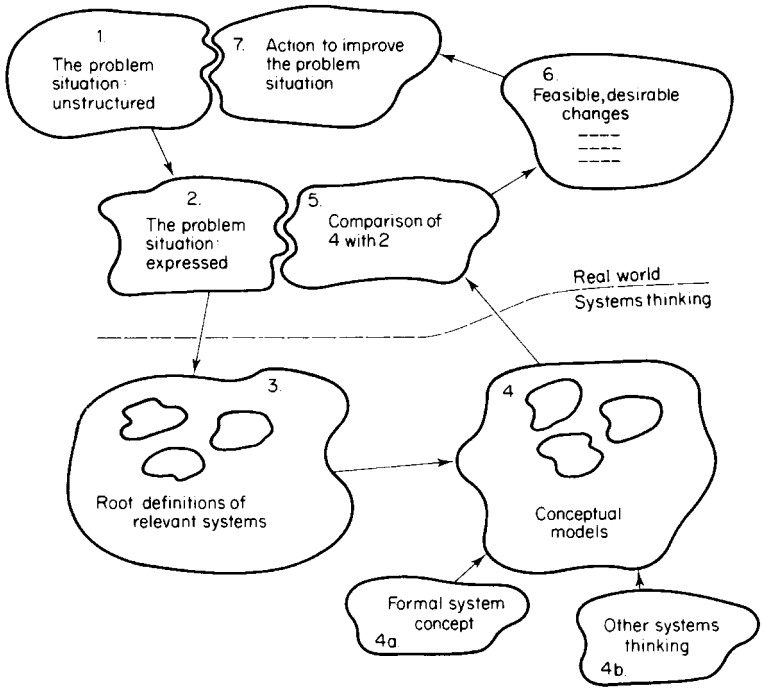

Figure 4. SSM in summary. A detailed account of the activities can be found in (Checkland 1981, p. 163–183).

Checkland's own words, itself a designed abstract system, and it should always be adapted and supplemented by concrete action. In real-life situations the most effective systems thinker will be working simultaneously at different levels of detail and on several activities (Checkland 1981, p. 163). In what follows we will look more closely at some of the key elements of SSM, and for each of these we will describe its dialectical nature.

The first element to be considered is the crucial distinction between taking action in the real world (activities 1, 2, 5, 6, and 7) and thinking about the real world in terms of systems (activities 3 and 4). The formulation of relevant root definitions and the building of corresponding conceptual models is called *systems thinking,* and what comes before and after are *real-world activities.* The purpose of the conceptual modeling is not to model phenomena as they are, but to model the Ws expressed in the RDs in order to learn about the consequences of taking that particular view. Hence the models are used to structure or orchestrate a debate amongst actors in the situation. This basic distinction negates a position where phenomena are spoken about and reflected upon as being systems. Within SSM we can chose to consider something as one or more systems, but we maintain the important difference between thinking about and being. Within dialectics this position is found in the fundamental concept of praxis. In the language of dialectics, praxis reveals man as a totality of a creative, productive being and a comprehending, reflective being who learns about the world through action (Israel 1979, p. 118–120).

The second element of SSM to be considered is the notion of *emergent properties.* At the very heart of soft systems thinking is the principle that whole entities exhibit properties which are meaningful only when attributed to the whole not to its parts (Checkland 1981, p. 74–82). In our case, examination of RD_3 shows that an emergent property is that the systems developers learn data modeling, a property not meaningful at a lower hierarchical level, e.g. simply by applying it, or by following a course on the subject. The principle of emergent properties negates a reductionist approach where all properties of a phenomenon are understood in terms of constituent parts of the whole (Checkland 1981, p. 45–47). Within dialectics we find a similar position expressed in the notions of concrete totality and explanatory emergence (Israel 1979, p. 62–78).

The third element of SSM to be stressed is the use of the concepts of *process, structure,* and *climate* in making rich pictures. In our case, this is illustrated in figure 2. This perspective negates a view, where the situation is seen as stable and balanced. Processes are stressed in their own right, and the climate between processes and structures indicates whether existing structures support or restrict performance, development, and change. As basic vehicles for reflection upon social situations these concepts correspond to the dialectical concepts of process, structure, and relation (Israel 1979, p. 115–124; Mathiassen 1987, p. 54–59). Checkland mentions that the climate of a situation frequently has been found to be a core characteristic of situations in which problems are perceived (Checkland 1981, p. 166), and we are not far from the dialectical notion of contradictions. One important difference, though, is that the concept of climate is somewhat peripheral in SSM and it is merely used to support real world activities. In dialectics, contradictions are at the very heart as they are seen as the driving forces of change and transcendence (Israel 1979, p. 72).

The fourth element of SSM to be considered is the suggested use of *several, different systems* to reflect upon the same situation. The point is illustrated by the phrase: “One man's freedom fight is another man's terrorism”. Hence, SSM stresses the different viewpoints and beliefs (Ws) of the actors involved and consequently it is recommended to make several root definitions and corresponding conceptual models (Checkland 1981, p. 166–177). In our case RD_3 and RD_5 describe different Ws and altogether we used about twenty root definitions. The idea of making several, different descriptions is an expression of the subjective nature of SSM negating the view that there exists objective facts about social phenomena isolated from the actors involved. In itself this is, of course, an idealistic position, but combined with the basic relation between real world activities and systems thinking, there are strong relations to the dialectical position expressed in the notion of praxis. Again, we should observe that we are close to the concept of contradictions in that there is no intention within SSM of enforcing coherence between different system views. In practice we can, therefore, easily have systemic views in mutual contradiction—even if there is no explicit notion of contradiction involved.

The important dialectics between thinking and being in SSM is stressed in a fifth element, the comparison in the form of *a dialogue*

between systems views and real world views (activity 5). In our case this comparison was organized as two discussions, one amongst the members of the methodology group, and one amongst all actors concerned. As a practical approach to organizational change this negates the idea of the organizational expert who designs solutions based on concrete analysis and general organizational knowledge. Instead SSM gives support to the idea of mutual learning and cooperation between different groups of actors, e.g. users and systems developers.

Finally, we want to stress a sixth element of SSM, namely the *criteria of feasible and desirable changes* (activity 6). In our case this leads to the acceptance of some actions and the rejection of others. This approach negates on the one hand the utopian idea of design by insisting that proposals for change must be culturally feasible. On the other hand it negates the somewhat pessimistic submission to the given through systemic generation of desirable changes. To focus on changes that are feasible as well as desirable corresponds to the dialectical view that social phenomena are produced and reproduced in the same way as man is both a creator of his social situations and submitted to them (Israel 1979, p. 76).

SSM shares many basic elements with dialectical approaches to social theory and action, and it is to a large extend these dialectical elements that represents an improvement in relation to more traditional approaches to computer-based information systems. In several respects SSM is in contrast with widely used methods, such as Structured Analysis and Design (DeMarco 1978; Yourdon 1982), i.e.

- SSM stresses that we have to interpret a situation to arrive at a problem, and also that we have to confront possible solutions with the specifics of the organizational situation we find ourselves in.
- SSM suggests that we use different systems concepts in reflecting on the problem situation. We could for example choose to include both organizational, behavioral and technical perspectives in the same systems development project.
- SSM is presented in terms of general ideas on how to reflect systematically in a problematic situation. The approach is in its form more like a framework for reflection and action than a cookery book to be followed.

SSM gives us the opportunity to understand a problem situation in many of its shades, it forces us to relate what we think to what we do and see, and it is not—like most conventional approaches—built on an unrealistic and naive view of computer-based systems in organizations (Mathiassen 1987). Still, criticism can be raised and supplementary approaches can be suggested.

4. Facing contradictions in situations

Returning to the situation in The Provident we have to admit that during the inquiry we found it irresistibly tempting to experiment with explicit and systematic thinking in contradictory wholes. Another account of this experiment can be found in (Nielsen 1989), and a different and more elaborate case is reported in (Mathiassen *et al.* 1989).

In the situation at hand it is relevant to ask about the purposes of using methods as, for example, data modeling. DP management sees the main purpose of methods as to supporting and promoting efficiency in information systems development, even though it is extremely difficult to assert whether systems developers in fact become more efficient when using data modeling. This *Weltanschauung* is captured in RD_{15}.

The users, on the other hand, look at the world of information systems from the point of view of quality. They do not pay much attention to what can be termed technical quality—what matters is quality in use, e.g. ease of learning, ease of use, and fitness of the information system to the context of work. The W of the users is shown in RD_{16}.

RD_{15}: A DP department-owned system for systems developers to develop information system efficiently, i.e. by a minimum of resources (time, manpower, and equipment), taking the information systems requirements as given.

RD_{16}: A Provident-owned system for systems developers to develop information systems of high quality with respect to ease of learning and use and to contextual fitness in the bank, the development being constrained by time.

These two root definitions are not different in nature from those shown in section 2 or advocated in SSM. It is common to have root

definitions in mutual contradiction and sometimes even in mutual contradiction. There is more than a purely abstract and theoretical difference between the two RDs, e.g. the users are motivated to learn part of the data analysis method if it can improve the quality of the information systems, but not if it mainly is to improve efficiency.

To fulfill our intention and desire to experiment with contradictions we attempt to go further than just having a contradiction between root definitions. Asking some of the systems developers about the purpose of using methods they cannot commit themselves fully to neither RD_{15} nor RD_{16}, since they believe they are trying to work efficiently and pursue quality at the same time. On the one hand, the systems developers have to live up to the rules imposed by management, for example, that time and other resources are limited. Each project have to be estimated and the time is fixed and can only be changed through tedious bureaucratic work, which they hate and therefore try to avoid. On the other hand, they work more and more in cooperation with users. They often know them personally and they know about their work in the bank. Therefore they have a strong commitment, not to say professional pride, to pursue quality in the computer-based information systems they develop. When they develop the systems and especially when they plan the projects, they are in this dilemma, as they call it. It is, they argue, inescapable. (This, however, is only an account of some of the systems developers—others tend to be in favor of efficiency). They maintain that this is a valid *Weltanschauung*—not two but one—and it should be treated as a whole, hence the formulation of RD_{17}.

RD_{17}: A contradiction between efficiency in the process of development and quality in the use of the new information system.

This is certainly a root definition of a different kind than all the others. It does not express a transformation and it cannot be modeled as shown in section 2. For example, the contradiction in RD_{17} cannot be replaced by the transformation: “A system to develop high quality information systems efficiently . . .” without severe loss of information. The proposed systemic root definition merely represents one out of many possible ways to handle the contradiction. For a detailed discussion see (Nielsen 1989).

In our experiment we used the three RDs in a major debate. At a public seminar it creates a rather heated and intensive debate amongst DP management and systems developers. DP management argues that using the quality-view is unrealistic in the sense that it would be too expensive to follow the quality demands of the users. The systems developers, now supported by the simplicity of the W's in the three root definitions, cannot see the point made by management. From their experience in the projects, the demands on efficiency put forward by management could be just as unrealistic as quality criteria of the users. Unfortunately, the users do not play an active role at the seminar. We therefore have to promote their viewpoint which is not as effective as if they did it themselves. DP management further argues that there is not a contradiction between efficiency and quality, and if systems developers were efficient enough quality would in the long run be free. The systems developers then go on and say that their experience is that, for example, tight and fixed deadlines (as a way of enforcing efficiency) are making it impossible to achieve just a bare minimum of quality. And, secondly, in their experience it always consumes more resources to increase quality of a specific system—it is simply cheaper to make a poor quality system. The systems developers, they argue, are the ones in the middle between DP management and users, so a choice or a compromise is always necessary. Furthermore, they are the ones that always have to make the decisions, which is probably why others cannot see the contradiction.

This is of course a condensed version of the real debate and the argumentation here only superficially reflects a one and a half hour debate. At the end of the debate it was evident that some accommodation was reached and that more than that would be unrealistic to expect. The main outcome can be summarized into:

- DP management realized that several systems developers believed the relation between efficiency and quality would demand specific considerations and actions by the systems developers in each project.
- The systems developers have learned about the limitations and the tradeoffs between efficiency and quality, not about how to balance the two aspects or make a choice, but what was at stake, and what was the assumptions made by DP management.

It started as an experiment, and it ended up being useful. The root definitions and especially the contradictory of them gave rise to a debate where no consensus was reached, but an accommodation where management moved their view-point towards a view of the practice of ISD as less straightforward than they had hitherto believed. The lesson we can learn from the project in The Provident is that it is useful to reflect and discuss in terms of contradictory wholes (as RD_{17}) and it is possible to do so while using SSM. Thinking in contradictory wholes is not a part of SSM, but it seems to work well within our practical use of SSM within The Provident. This is not to say that contradictory wholes can be integrated into SSM in general or that there may be an easy coexistence between the two—maybe that is impossible by their very nature.

5. The methodological challenge

We have experimented with a very simple and somewhat naive approach taking SSM as given and playing with the concept of contradiction. Let us now elaborate and explore the relation between SSM and dialectics.

We have already seen in what respect and to what extent SSM is in itself dialectical. Recently the systems metaphor have been extended to include notions such as a contradictory system (Atkinson *et al.* 1988). Turning to other systems approaches reveals that, for example, Churchman (1971) has dealt with dialectics where the idealistic dialectics of Hegel is interpreted in terms of systems, i.e. dialectical inquiring systems. On the basis of the work by Churchman a somewhat theoretical dialectical approach towards information systems was proposed (Mitroff *et al.* 1972). Along completely different lines we find our own contributions (Mathiassen 1981, 1987; Mathiassen *et al.* 1988) together with Stage (1988) criticizing traditional approaches and advocating complementary views as the fundamental conceptual basis for analysis and design, and Bjerknes (1988) giving an account of universal contradictions in information systems development. It is therefore not at all a new idea to look at information systems development from the point of view of dialectics.

Mao speaks of dialectics and contradictions in an early work of 1937: “On Contradiction” (Mao 1937). We do not want to outline a debate between different dialectical approaches and philosophies.

Instead, we have chosen a dialectical approach intended to and actually used in practice and with a certain impact on the real world (at least in China). There are at least two reasons for looking at the Chinese experience. Firstly, this approach has been used and described in a way that makes it useful in everyday situations (this cannot to the same extent be claimed about dialectical approaches as such). Secondly, Mao's ideas have been a primary source of inspiration for contemporary dialectical theories (Israel 1979).

In our interpretation of Mao there are five important elements in the dialectical approach. Firstly, the contradiction is a *whole.* Learning about the contradiction can only be done by looking at the aspects[1] of the contradiction and their relation, but basically the starting point is the contradiction as a whole. To phrase this in terms of systems, the contradiction is an emergent property at a level above the two aspects. As a consequence of this, it is not possible to learn about the contradiction by looking at only one of the aspects or at their relation.

> *"This is what we mean by looking at a problem one-sidedly. Or it may be called seeing the part but not the whole, seeing the trees but not the forest." (Mao 1937, p. 41).*

Seeing a contradiction as a whole was the only element of dialectics that was used properly in our experiment. RD_{17} is a description of a contradiction and RD_{15} and RD_{16} are descriptions of the aspects of RD_{17}. Looking at it in retrospect, we could have exploited the contradiction further. We could have constructed contradictions on the basis of other root definitions asking the question: What contradiction is this view-point an aspect of? This corresponds to the thesis-antithesis approach of Hegel (Churchman 1971, p. 170 ff.).

Secondly, sometimes the aspects of a contradiction are in *open* fight and sometimes the contradiction is only *latent.* Contradictions in organizations can appear as social conflicts. The introduction of various kinds of organizational filters (e.g. the application of specific approaches to develop information systems, the setting of agenda for project group meetings, the selection of members of steering committees) can prevent contradictions from surfacing as such. We did not use this element in the experiment. We could have learned about the willingness and the possibilities for handling the contradiction between quality and efficiency in The Provident by looking

at related conflict and by assessing how the contradiction was perceived by different actors.

Thirdly, in order to learn about and handle a given situation it does not give much insight to look at general contradictions. It is necessary to look at the specific and concrete contradictions *in the situation*.

> *"Processes change, old processes and old contradictions disappear, new processes and new contradictions emerge, and the methods of resolving contradictions differ accordingly." (Mao 1937, p. 38).*

As in SSM there is not a general solution that applies to all problems. The contradiction between efficiency and quality (RD_{17}) as experienced in The Provident was formulated as a general feature of ISD. Much effort went into discussing the contradiction instead of trying to deal with how those who experienced the contradiction could handle it. Thus, one should strive at explicitly stating in the definition which situation or concrete context we address. A better formulation would have been:

> RD_{17a}: A project-situated contradiction within The Provident between efficiency in the process of development and quality in the use of the new information system.

Fourthly, there is a *development* in any contradiction. Processes and their inherent contradictions are changing over time and it is important to inquire into how a specific contradiction has evolved and changed. We could probably have gained substantial insight in the situation in The Provident by looking into the history of the relation between quality and efficiency.

Fifthly, there is the distinction between the *main* contradiction and the subordinate contradictions and between the *dominating* aspect and the subordinate aspect. In any situation one contradiction will be more important than others. In any contradiction some aspects will at any time dominate or be most important. If RD_{17} is the most important in our experiment and if efficiency and quality are dominating it is a mere coincident. Maybe we could have found

other contradictions and other aspects by applying this principle of dialectics.

Mao summarizes:

> *"According to the dialectical materialism, contradiction is present in all processes of objectively existing things and in the subjective thinking, and it permeates all these processes from beginning to their end; this is the universality and absoluteness of contradiction. Each contradiction and each of its aspects have their respective characteristics; this is the particularity and relativity of the contradiction. In given conditions, opposites possess identity, and consequently can coexists in a single entity, and can transform themselves into each other; this is again the particularity and the relativity of the contradiction. But the struggle of opposites is ceaseless, it goes on both when the opposites are coexisting and when they are transforming themselves into each other, and becomes specially conspicuous when they are transforming into one another; this again is the universality and absoluteness of the contradiction. In studying the particularity and relativity of contradiction, we must give attention to the distinction between the principal contradiction and the non-principal contradictions and of the distinction between the principal aspect and the non-principal aspect of a contradiction; in studying the universality of the contradiction and the struggle of opposites in the contradiction, we must give attention to the distinction between the different forms of struggle." (Mao 1937, p. 72).*

The methodological challenge is to experiment further by developing the notion of contradiction and techniques for using it in the various activities of SSM. The lessons we have learned until now suggest the following minimal requirements:

- A contradictory root definition is an abstraction of the problematic situation denoting a Weltanschauung.
- A contradictory root definition expresses a whole by naming two opposite aspects and their relation.
- A contradictory root definition must describe its context, for example, using the phrase: An X-situated contradiction.

- Contradictory root definitions can be surfaced directly by looking at the problematic situation (a W in its own right), and indirectly by looking at systemic root definitions (taking two that are in conflict or taking one as the one aspect, and ask: what is the opposite aspect?)
- The relevance of a contradictory root definition is established through comparison and debate.
- A conceptual model of a contradictory root definition must describe the two aspects and their relation in such a detail, that a comparison makes sense.
- A comparison of a contradictory conceptual model results in assessments of: state of open fight/latent, related conflicts, state of development, dominating aspect, desirable actions (for example preferring or supporting one aspect, balancing the two aspects, hiding the contradiction through introduction of organizational filters, surfacing the contradiction, or simply letting it happen knowing it is there).
- A comparison must include an assessment of the relation between contradictions by pointing at the present main contradiction of the situation.

6. Conclusion

In conclusion, we relate to a debate of SSM initiated by Jackson's critique (Jackson 1982, 1983; Ackoff 1982; Churcman 1982; Checkland 1982; Rosenhead 1984; Mingers 1984). Based on the concepts of Burrell and Morgan (1979), Jackson claims that SSM falls within the interpretive paradigm. Firstly, this means that SSM is subjective in the sense that it does not assume the existence of objective facts about social phenomena. Instead the social world is seen as being the creative construction of humans—the assumption being that we use language to understand and act in social situations, but at the same time we produce our social situations in doing so. The social world has, in other words, no real existence outside the consciousness of human actors. Secondly, Jackson's characterization implies that SSM is regulative in the sense that the main concern is to understand how order and cohesion are achieved and maintained. There is, Jackson claims, no interest in SSM in radically going beyond the status quo. While the subjective nature of SSM is left un-

questioned, Checkland (1982) strongly disagrees with the view that SSM is regulative in nature. In his response Checkland argues that his approach facilitates a negotiation process between different actors, and that this process in principle could be regulative or radical, depending on the selections made in the modeling process and on the readiness of the actors to modify and change viewpoints and beliefs (Checkland 1982, p. 38). Moreover, Checkland criticizes Jackson for adopting a research method which relies merely on argument without giving evidence based on experiments and experiences.

The issue raised in this paper is certainly related to Jackson's theoretical viewpoints. We agree with Checkland that SSM in principle could be regulative or radical, but we still find it relevant to consider whether SSM could be modified to further stimulate organizational actors in challenging established traditions and beliefs. It seems obvious that the degree of change in a given situation depends mainly on situational characteristics and less on general features of the applied approach. We do believe, however, that the approach matters, and we find it interesting to explore means to reduce blindness and to support radical thinking in organizational learning and development. We see SSM primarily as an approach to *mediate* a learning process *requested* by actors in a problematic situation, and we ask ourselves: to what extent and by what means can we support an organizational consultant in *actively influencing* the actors' perception of the situation?

From a theoretical point of view one can convincingly argue, that SSM is subjective, while dialectics is objective. One could therefore suspect that a methodological marriage between the two would be impossible. This theoretical observation is, however, in contrast to our experience. From our point of view it is relevant to ask:

- What is the practical difference between the claimed subjectivity of SSM and the claimed objectiveness of dialectics that prevents one from doing both at the same time?
- Is it possible to do practical problem solving without relying on personal beliefs, experiences and interpretations (subjectivism) and without applying, maybe implicitly, assumptions about the reality of the situation (objectivism)?

Notes

1. These are also labeled contradistinctions and refer to the two aspects of the contradiction.

References

Ackoff, R. L. (1982): On the Hard Headedness and Soft Heartedness of M. C. Jackson. *Journal of Applied Systems Analysis,* Vol. 9 (31-34).

Atkinson, C. J. & P. B. Checkland (1988): *Extending the Metaphor "System".* Paper presented at the Systems Conference, University of Lancaster, July 1988.

Bjerknes, G. (1988): *Contradiction—A Discussion of the Concept.* Part of draft for Ph.D. dissertation. Oslo University. (In Norwegian)

Burrel, G. & G. Morgan (1979): *Sociological Paradigms and Organisational Analysis.* Gower.

Checkland, P. B. (1981): *Systems Thinking, Systems Practice.* Chichester: Wiley.

Checkland, P. B. (1982): Soft Systems Methodology as Process: A Reply to M. C. Jackson. *Journal of Applied Systems Analysis.* Vol. 9 (36–39).

Checkland, P. B. (1985): Achieving "Desirable and Feasible" Change: An Application of Soft Systems Methodology. *Journal of the Operational Research Society.* Vol. 36, Vol. 9 (821–31).

Churchman, C. W. (1971): *The Design of Inquiring Systems.* New York: Basic Books.

Churchman, C. W. (1982): Reply to M. C. Jackson. *Journal of Applied Systems Analysis.* Vol. 9 (35).

DeMarco, T. (1978): *Structural Analysis and Systems Specification.* New York: Yourdon Inc.

Flavin, M. (1981): *Fundamental Concepts of Information Modelling.* New York: Yourdon Press.

Israel, J. (1979): *The Language of Dialectics and the Dialectics of Language.* Copenhagen: Munksgaard.

Jackson, M. C. (1982): The Nature of "Soft" Systems Thinking: The Work of Churchman, Ackoff and Checkland. *Journal of Applied Systems Analysis.* Vol. 9 (17–29).

Jackson, M. C. (1983): The Nature of "Soft" Systems Thinking: Comments on the three Replies. *Journal of Applied Systems Analysis.* Vol. 10 (109–113).

Jepsen, L. O., L. Mathiassen & P. A. Nielsen (1989): Back to Thinking Mode—Diaries as a Medium for Effective Management in Information

Systems Development. *Behaviour and Information Technology,* Vol. 8, No. 3.

Lanzara, G. F. & L. Mathiassen (1985): Mapping Situations Within a Systems Development Project. *Information & Management.* Vol. 7 (3–20).

Mao Tsetung (1937): On Contradiction. (23–78) in *Four Essays on Philosophy.* Peking: Foreign Languages Press.

Mathiassen, L. & A. Munk-Madsen (1988): *Myths and Reality in Software Development.* Technical Report, Institute of Electronic Systems, Aalborg University.

Mathiassen, L. & P. A. Nielsen (1989): Surfacing Organisational Competence: Soft Systems and Hard Contradictions. In G. Bjerknes *et al.* (Eds.): *Organizational Competence in Systems Development.* Lund: Studentlitteratur.

Mathiassen, L. (1981): *Systems Development and Systems Development Method.* Ph.D. thesis, Oslo University. (In Danish)

Mathiassen, L. (1987): Systems, Processes, and Structures. In P. Docherty *et al.* (Eds.): *Systems Design for Human Development and Productivity: Participation and Beyond.* Amsterdam: North-Holland.

Mingers, J. (1984): Subjectivism and Soft Systems Methodology—A Critique. *Journal of Applied Systems Analysis.* Vol. 11 (85–103).

Mitroff, I. I., E. Rathswohl & J. Williams (1972): Dialectical Inquiring Systems: A New Methodology for Information Science. *Journal of the American Society for Information Science.* (365–378).

Nielsen, P. A. (1989): The Emergence of Contradictions in Soft Systems Practice. In R. L. Flood *et al.* (Eds.): *Systems Prospects.* New York: U.K. Systems Society, Plenum.

Rosenhead, J. (1984): Debating Systems Methodology: Conflicting Ideas About Conflict and Ideas. *Journal of Applied Systems Analysis,* Vol. 11 (79–84).

Schäfer, G., M. Domke, R. Hansjee & M. Harper (1986): FAOR Approach: Main report. Project report. Esprit Project 56: *Function Analysis of Office Requirements.* Essex: FAOR Project Library.

Schön, D. (1983): *The Reflective Practitioner—How Professionals Think in Action.* New York: Basic Books.

Stage, J. (1989): *Between Tradition and Transcendence. Analysis and Design in Systems Development.* Oslo University. (In Danish)

Wilson, B. (1984): *Systems: Concepts, Methodologies and Applications.* Chichester: Wiley.

Yourdon, E. (1982): *Managing the Systems Life Cycle.* New York: Yourdon Inc.