Jens Kaasbøll

FIRE report no. 13
1. september, 1994

SA++

How to bring organizational aspects into teaching analysis and design

Also in

Joze Zupancic and Stanislaw Wrycza (eds.)
Proceedings of The Fourth International Conference
Information Systems Development — ISD ’94
Methods & Tools. Theory & Practice
(Bled, Slovenia, 20–22 Sept., 1994)
Moderna Organizacija, Kranj, 1994
pp.559–568

Department of Informatics, University of Oslo
P.O.Box 1080 Blindern, N-0316 Oslo, Norway

Phone: +47 2 85 24 10
Fax: +47 2 85 24 01
Email: jens.kaasboll@ifi.uio.no

Abstract

Because of the close connection between computer systems and organizations, organizational aspects should be taught in system development courses. Dataflow diagramming is a frequently taught technique that can model formal aspects of information processing. Socio-technical methods are intended to cover relations between people and technology in an organization. To avoid learning two techniques, the dataflow technique has been extended with two organizational aspects from socio-technical methods: goals and exceptions.

Two versions of the extended technique have been used in a university course for systems analysis and design. Organizational issues were taught together with the technique. The students' work has been evaluated.

The technique is sufficiently easy to learn for students. The students also learnt to involve users in analysing their tasks through a wall graph session. However, many students did not learn sufficiently well to consider organizational impacts of computer systems or to design systems to fit organizational requirements. A possible way to cope is to give the students better background in knowledge of organizations.

1 Introduction

It is known that attention should be given to organizational aspects during development of information systems, and that user organizations should be involved in the development. Students should therefore learn to consider both organizational and technical aspects when they learn to analyse and design information systems.

Textbooks on systems analysis and design emphasize that computer systems should meet organizational needs. The techniques for development taught in courses often focus on technical aspects. For example, dataflow diagrams and data modelling are two widely taught techniques that focus on data processes and data.

Methods bringing together technical and organizational issues have been created (eg, Mumford, 1983; Avison and Wood-Harper, 1990). However, it takes time to learn a larger method, and there is not always time for this in academic or industrial courses.

Structured Analysis (SA) is widely used in courses. The method is also frequently employed in system development, although studies show that developers freely adapt the method and its techniques and break its rules (Dekleva, 1992; Bansler and Bødker, 1993). The spreading of CASE tools that support the method constitutes a reason for believing that SA will continue to be used both in development and education for a period of time.

Given that students will continue to learn to analyse and design information systems in SA courses, there is a need for a way to make the students consider organizational aspects while analysing and designing. In addition, the students should also learn a way to involve users in the development. This paper proposes and evaluates a way to integrate organizational analysis into courses in which dataflow diagramming from SA has a central role.

2 Course requirements

Textbooks in SA (eg, Delskov and Lange, 1990; Yourdon, 1989; Hawryszkiewycz, 1988) focus on analysis and design with dataflow diagramming as its central technique. In the original version (DeMarco, 1978), the diagrams were intended to describe four versions of an information system: current physical system, current logical, new logical, and new physical. Recent versions have emphasized the analysis of the new, logical (essential) structure of the information system.

The logical dataflow diagrams capture the formal flow, processing, and storing of data. The physical diagrams may be used for portraying the formats and means for flow and storing. In addition, the physical processes can represent the persons and the computers that perform the logical transformations. The flows and stores do not describe all data structures. This can be done by means of data modelling or object oriented modelling.

The requirements for a course where students learn to handle organizational and technical issues are:

1. During work with the dataflow diagrams, the students have to consider organizational impacts of current systems.
2. When shaping a new system, the students should learn to make the system fit organizational requirements.
3. There should be a way for students to learn how to engage user organizations in the development process.

3 Setting of the experiments

An extension of dataflow diagrams called SA++ was developed to be a remedy for meeting the requirements. The extension is motivated in section 4 and described in section 5.

The first evaluation of SA++ was carried out in 1992 in a university course on system analysis and design. This experiment is described and evaluated in section 6.

This course was intended to meet the requirements nos. 1 and 2 above. In addition, the test aimed at answering the following question:

0. Can SA++ be comprehended and used within an acceptable time limit?

The second experiment was carried out during the same course in 1993. This time the course intended to meet requirements nos. 2 and 3. Description and evaluation are found in section 7.

4 Organizational aspects

There are a multitude of organizational aspects that could be considered in analysis and design. The choice of aspects to be included in the dataflow analysis is based on the shortcomings of the dataflow diagramming and the opportunities for making improvements that can be learned easily and used effectively.

4.1 Exceptions

The processes in dataflow diagrams describe the routine transformations from input to output. Practitioners often say that they have minor problems with the routines; the design problems are created by all the exceptions. This is confirmed by research, which also emphasizes exceptions. Gasser (1986) has analysed the fit between work and computing in detail in two organizations. He has identified how people work around formal routines. Socio-technical studies also point at the need for finding the discrepancies or variances from routines (Mumford, 1981, p.12). The processes will therefore be extended with exceptions from the formal transformations.

Gasser has identified three areas of accommodation to misfit. These will be used to characterize misfit between routines and the actual work.

Fitting. Fitting work consists of making changes to computer arrangements or adjusting the routines of work. Fitting work constitutes a deviation compared to the routines. But after making the fits, the routines might have been changed.

Augmenting. Augmenting work is to undertake additional work to compensate for the misfit.

Working around is when using the computer in ways for which it was not designed, or when avoiding routines or computers by performing ad hoc. (Gasser 1986, pp.214–216)

Carrying out a process with exceptions presumes that the process is performed by a person, not by a machine.

4.2 Goals and results

The physical processes describe who or what performs a process and the logical processes describe the formal transformations. The question about why a process is carried out can be answered by referring to the rules of the process for the routine processing. There has to be another way of describing what is determining the exceptional processing. When behaviour in organizations cannot be explained by reference to a rule, the actors can be assumed to have intentions, goals, preferences, values, etc., as their reasons for action.

Whether or not an actor is behaving in a certain way because of a goal is not very relevant for the analysis and design of an information system. However, it is relevant to know whether a specific process contributes to fulfilling the goals of the organization, departments, stakeholder groups, users, customers, clients, or other constellations of people relevant to the organization.

Socio-technical methods emphasize the goals of people in organizations (Mumford, 1981, Mumford, 1983), focusing on efficiency and job satisfaction. Effectiveness and customer satisfaction are other categories of goals which may be linked to strategic business objectives.

Even if an effort has been made to make organizations rational by means of computer systems, decisions are made on grounds that may be all but rational and uniform (Feldman and March, 1981). The difficulties of understanding the complex pattern of goals may be used as an argument against considering goals at all in analysis and design. However, if strategic goals, competing interests, and personal motives are kept in the shadow, there is a risk that a designed system will be ignored.

When considering the goals of a process, there is an opportunity to compare whether the results of the process fulfils its goals, and if not, there may be a deviation that could be explained by an exception. To make the comparison, it is also necessary to know the results of the process. For the formal transformation of a logical data process, the result is as specified in the output. Also, when regarding the persons or computers in the physical data flow, other results may be noted, amongst them unintended effects. An example is given below.

4.3 Other organizational issues

There are many other factors missing in the diagrams. It might, eg, have been useful to consider social relations between people performing the processes, the resources available for keeping the various parts of the information system going, or the organizational culture. However, to keep the diagrams as simple as possible, exceptions and goals have been selected as their only extensions. An example will illustrate that some other organizational factors, including skills and resources, can be considered by the extensions proposed.

5 SA++

Since the diagrams are extended with two issues, exceptions from regular performance, and goals and results of the actual performance, the extension is called SA++. The extension has been presented in lecture notes for a university course (Kaasbøll, 1992). It will be illustrated by a case.

> The meteorologist at an airfield analyses the weather by reading observations from various sources, including numerical forecasts of a larger region produced at a central meteorological institute. The meteorologist draws a synoptical map showing the weather situation. He simultaneously forms an inner weather picture, which is his conception of the weather development in all three dimensions of the atmosphere over a period of time which stretches into the future. The meteorologist uses his inner weather picture for making forecasts, issue warnings, and briefing pilots. He needs a comprehensive understanding of the weather in order to do this. (Perby, 1987)

Figure 1 shows parts of an extended dataflow diagram for the case. Logical diagrams should model the formalizable data flow. The diagram therefore models the contents of the synoptical map that the meteorologist draws, Simultaneous, located observations. At the physical layer, it is appropriate to call it Synoptical map, since this tells how the data are arranged. Since the "inner weather picture" is the information which the meteorologist creates from the data, it neither appears in the processes nor in the flows.

Goal covers any goal of the process that may exist in the organization and any intention of the persons undertaking the process. Result describes the output, products and other changes produced by the process. The results may satisfy goals, they may be insufficient, or they may not be related to the goals at all, the latter being typical for side-effects.

Process no.1, Numerical analysis, is performed by a computer, as indicated in the physical diagram and the gray shading. Its goal is related to the needs of the meteorologist in this case. Others may have different goals for the same processing.

Process 2, Local weather analysis, is performed by the meteorologist. The two first mentioned goals and results of this process refer to the information and the data. The third result, Meteorological skills are maintained, is a longer term result that is emphasized by Perby in her study of the meteorologists (Perby, 1987, p.219–221). Similar side-effects are described in process 2 in the physical diagram. These results are described as side-effects of the processes, because Perby gives no hints that the meteorologists consider skill maintenance while doing their daily

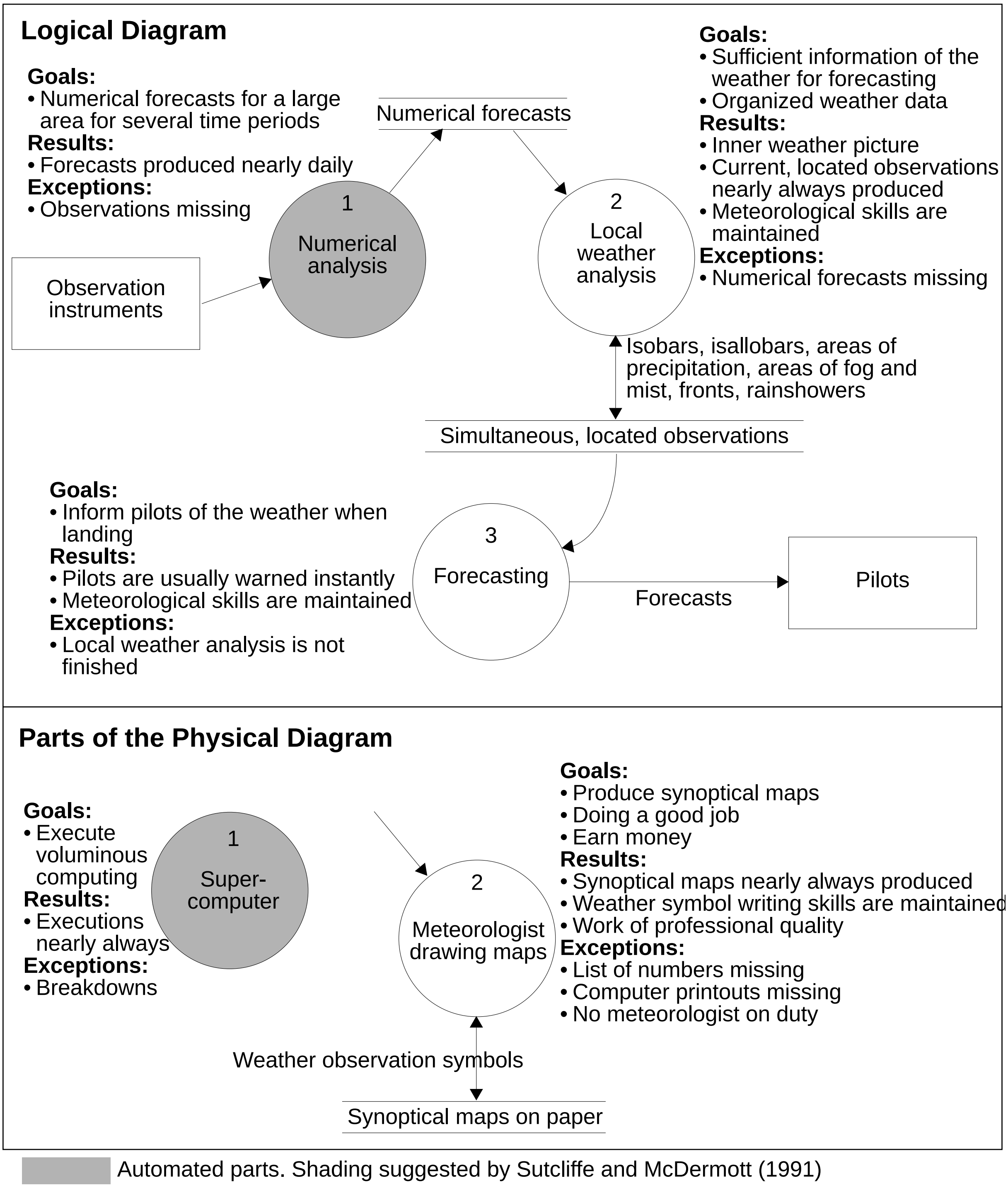

Figure 1: Extended logical dataflow diagram

analyses and forecasts.

The goals of the processes performed by persons, partly determine the transformations going on in the processes. When no exceptions occur and rules for performing the process exist, the dataflow diagrams express that the process is performed according to the rule indicated by the process name. For processes for which no rule exist, or when exceptions occur, the goals should determine the outcome of the processes.

According to the rules of dataflow diagrams, the persons, organizational units or machines that perform the processes are described as physical processes. Therefore, goals relating to the work of persons or organizational units in general are described in the physical layer. A goal like Doing a good job may be closely related to the forecasts given. That indicates that this goal could have been placed in the logical diagrams or in any process performed by the meteorologist. It is placed in the physical diagram, because it then becomes clear to whom this goal belongs. If an aggregated diagram had been made, in which the meteorologist just appeared in one process, such goals should be placed there.

5.1 Deviations

The goals, results and exceptions are compared during the next step of the extension: deviations and analysis of dependencies.

The goal deviations have already been indicated: insufficient results and side-effects. Current, located observations nearly always produced is an example of an insufficient result, while Meteorological skills are maintained is a side-effect.

The analysis should proceed in the following way:

1. Compare goals and results, and note goal deviations of the processes. Compare information about goals and results compiled from different sources. If the goals differ, is it because

   (a) the users come from different parts of the organization, and their goals correspond to those of their organizational unit?

   (b) the users have their own interests or represent groups of interest with goals that differ from those of the organization, or are in conflict with those of the organization?

   (c) the users have no explicit goals of their work?

2. The processes should be classified either as necessary or augmenting work or working around the computer system (Gasser, 1986). Augmenting work and working around should be compared with the goals. What seems superfluous in one setting, may be necessary for other goals.
3. Determine dependencies between the factors found. In particular, do routine deviations cause goal deviations? (Kaasbøll, 1992)

Four simple chains of dependencies for the example are shown below:

Deviation from routine, augmenting: Extra effort is needed in local weather analysis because numerical forecasts are missing due to breakdown in the super-computer.

Deviation from goal: Current, located observations are not produced in local weather analysis because there is no meteorologist on duty.

Deviation from goal: The meteorologists build inner weather pictures and maintain their skills.

Deviation from goal: The pilots are not warned instantly because local weather analysis is not finished.

5.2 Reducing the deviations

System development may have several aims. The extended dataflow analysis may be used in design to reduce deviations. Two designs are proposed to illustrate the consequences for results, exceptions, and deviations.

The first proposal aims at reducing the latter deviation from goals. The delay of the weather forecast is explained to happen because the local weather analysis has to be completed before forecasts are given. The local weather analysis lasts for one hour when performed by the meteorologist. A design option is therefore to substitute the meteorologist with a computer program that make numerical analysis and forecasts that are adequate for pilots. To avoid technological changes in aeroplanes, the meteorologist will still have to transmit the forecasts to the pilots on orally. Parts of the design proposal is sketched in figure 1.

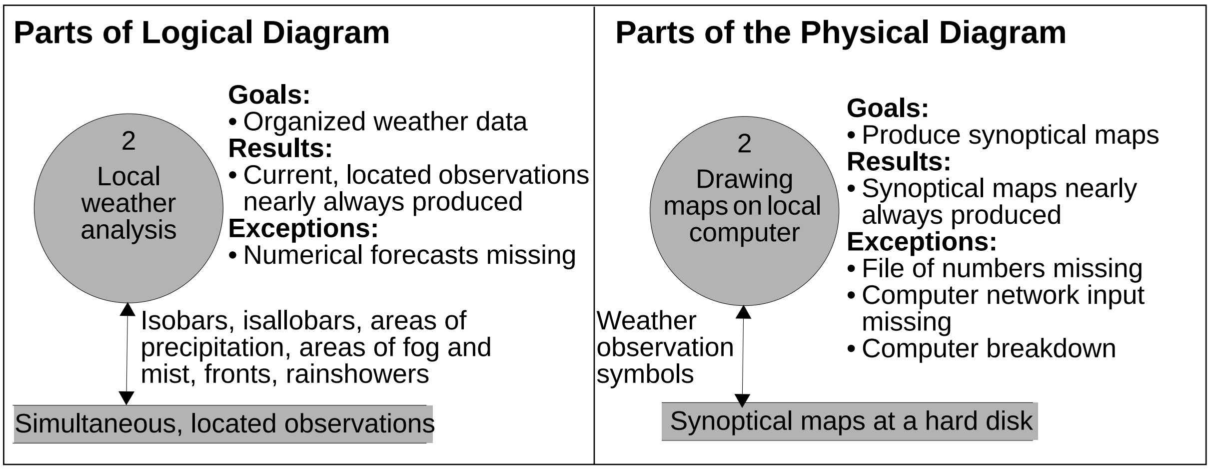

Figure 2: Redesigned system, 1. alternative: The local weather analysis is performed by

The new design has not altered the routine deviation of the old system. The goal deviation caused by No meteorologist on duty has only been slightly altered, because the new system is also dependent on meteorologists informing the pilots. The side-effect being that the meteorologists build inner weather pictures is removed, and their skill maintenance is reduced. Perby and the meteorologists warn against this solution, because they believe the meteorologists will loose their skills in the long run (Perby, 1987, pp.223–224).

Another design aims at reducing the latter goal deviation whilst keeping the skill maintenance. The proposed implementation is to replace the paper for drawing maps with a computer screen, and let the data transfer between process no.1 and no.2 be electronic. The deviations are as follows:

Deviation from routine, augmenting: Extra effort is needed in local weather analysis because numerical forecasts are missing because of breakdown in the super-computer.

Deviation from goal: Current, located observations are not produced in local weather analysis because there is no meteorologist on duty.

Deviation from goal: Meteorologists build inner weather pictures and maintain their skills.

5.3 Scope of organizational issues

The SA++ technique covered skills and contents of work in this example. The timeliness of the output is also considered, in addition to computer breakdowns. Job satisfaction in general can be covered, since persons are included. The efficiency of processes and the effectiveness of the results can also be considered. In other examples than the one chosen, the exceptions may be more dominating.

6 First course evaluation

The first evaluation of SA++ aimed at answering the following questions, corresponding to the requirements in section 2.

0. Can SA++ be comprehended and used within an acceptable time limit?
1. Did students learn to consider organizational impacts of current systems?
2. When shaping a new system, did the students learn to make the system fit organizational requirements?

The evaluation was carried out in 1992 in a university course on system analysis and design; 69 students participated. The course was scheduled to take half of the students' time during the spring term. The course material contained guidelines with explanations of how to extend dataflow diagrams with goals and exceptions, and of how to find goals and exceptions in organizations. The activities involving extended dataflow diagrams were class room exercises, description of an actual information system, and parts of a written exam. A class room exercise in SA++ was to make extended dataflow descriptions of the weather forecasting case, based on the text by Perby (1987). Three weeks were spent on introduction to SA and to the extensions.

The students knew that SA++ was a novelty. However, they were not informed that their learning process was part of an experiment.

The guidelines for finding goals encompassed a brief introduction to types of goals. Many of these issues were known from previous courses. The method for finding routines, goals, results and exceptions in organizations was divided into four steps, inspired by Hawryszkiewycz's search procedure (1988). It can be summarized as follows.

1. Read documents about the organization, for example, annual reports, organizational charts, documents describing goals, commercials.
2. Interview with managers. Ask the managers about goals for their organizational unit, and let them give an overview of the routines.
3. Observation and interview with those working with the information. Let them *show* you what they do in their actual work at their normal workplace. Ask what is going on, for instance, how do these data arrive to you, how do you know how to do it, how many copies do you make and who are they intended for? After a survey of their routine, ask about exceptions: eg, what if something is missing, what are the usual problems, what happens if you forget to sign, makes it extra work for you if someone is absent?
4. Observe a meeting concerning their objects of work. To understand what happens at the meeting without interrupting, make this your last technique of investigation. Prepare by studying the material for the meeting beforehand.

Groups of approximately ten students found organizations on their own, which they described with SA++. They were tutored by a graduate student instructor.

Each group produced a twenty page description of an information system in an organization. The descriptions were swapped between the student groups, and the groups were given the task to design a computer system for an organization which another group had described.

Method

A qualitative analysis of the knowledge that the students documented in their project reports was carried out (Larsen, forthcoming). Eight out of sixteen student designs were arbitrarily chosen to make the analysis a manageable task.

The method of analysis was to disclose the knowledge upon which the students had based their design decisions. 75 design decisions were found in two textbooks (Avison and Wood-Harper, 1990; Yourdon, 1989). It was analysed whether the students had made these decisions, and if so, on what ground they had made them.

Results

The students had made most of the decisions concerning functionality proposed in the textbooks. These decisions were supported by the diagrams.

Four of the eight groups had not analysed the goals sufficiently for making design decisions. Decisions concerning social aspects and socio-technical aspects were made in lesser extents than proposed in the textbooks. Some of these decisions were made without firm grounds, while others were grounded in the goals and exceptions in SA++. Other decisions, eg, concerning implementation, security, and maintenance were mostly neglected by the students.

Discussion

The four groups that had analysed goals properly had learnt SA++ within the time available during the projects. In many instances, the students mixed physical and logical concepts, however.

The students were given the task to make an extended dataflow description based on a text during the exam. The average grade at this question did not differ significantly from the average score for the total questions at the exam. The exam showed that extended dataflow diagramming technique can be learned and used during three weeks of teaching and some days training before an exam. Question no. 0 is therefore confirmed.

Only half of the groups had learnt to analyse organizational impacts sufficiently to answer confirmative to question no. 1. The remaining students were not capable of or spent too little effort in finding organizational aspects.

The same half of the students who learnt to find goals, also managed to propose an innovative design that would fit organizational requirements, according to Larsen (forthcoming).

The fact that the students neglected several types of decisions that were proposed in the method, is a consequence of the lack of attention concerning these issues in the course and in SA++.

7 Second course experiment

The technique was redesigned to improve creativity and user participation (Wiig, 1994), thus intending to fulfil requirements nos. 0, 2, and 3. Freely drawn wall graphs were used in initial descriptions. Exceptions were noted when appropriate, and goals were structured on separate descriptions. Dataflow diagrams were made later on.

The revised version was used in the course the following year. This time, the systems were described in sessions where both users and students participated.

The students responded mixed experience as to finding goals in a questionnaire. Eg, it was difficult because the organization had no clear goals, it was enlightening to discuss goals, it made little sense in a small organization. The actual users who participated in the wall graph sessions answered in a questionnaire that they experienced being engaged in the wallgraphing.

The designs seemed more innovative than the year before. A preliminary analysis shows that the proposed systems were designed to meet some of the organizational goals and to reduce exceptions.

The results of the exam did not differ from the previous course.

The conclusion of the second experiment is that requirements nos.0 and 3 were met, while no. 2 only was achieved by some of the students.

8 Conclusion

A technique has been developed to learn students in SA courses to consider organizational issues during analysis and design.

The technique is found to be sufficiently easy to learn for students. The students also learnt to involve the users in analysing their tasks through a wall graph session.

However, many students did not learn sufficiently well to consider organizational impacts of the computer systems or to design systems to fit organizational requirements. A possible way to cope is to give the students better background in knowledge of organizations.

Bringing organizational issues into dataflow diagramming is an example of modification of a technical oriented system development technique. If, eg, object-oriented methods are taught, organizational aspects could probably be included in a corresponding way.

Acknowledgements

Many thanks to Bodil Bye Larsen and Ingrid Wiig, who have provided invaluable help in the development and evaluation of the technique. Thanks also to Kristin Braa, who has supported the experiment through unlimited encouragement, and to Sara Selmark for improving the language.

References

Avison, D.E. and Wood-Harper, A.T. (1990) “Multiview: An Exploration in Information Systems Development” Blackwell, Oxford

Bansler, J. P. and Bødker, K. (1993) A Reappraisal of Structured Analysis: Design in an Organizational Context” *ACM Transactions on Inf. Sys.* 11, 2, pp.165–193

Dekleva, Sasa M. (1992) The Influence of the Information Systems Development Approach on Maintenance *MIS Quarterly* Sept.1992, pp.355–372

Delskov, Lis and Lange, Therese (1990) “Struktureret Analyse — Integreret Systemanalyse” Teknisk Forlag, København

DeMarco, Tom (1978) “Structured Analysis and System Specification” Yourdon, New York

Feldman, Martha S. and James G. March (1981) Information in Organizations as Signal and Symbol *Administrative Science Quarterly* Vol. 26, pp.171–186

Gasser, Les (1986) The Integration of Computing and Routine Work *ACM Transactions on Office Information Systems* Vol.4, No.3, pp.205–225

Hawryszkiewycz, I.T. (1988) “Introduction to Systems Analysis and Design” Prentice-Hall, NY

Kaasbøll, Jens (1992) “SA++” Lecture Notes, Department of Informatics, University of Oslo

Larsen, Bodil Bye (forthcoming) “Supporting Design Decisions by Means of SA++” Master Thesis, Department of Informatics, University of Oslo

Mumford, E. (1981) Participative Systems Design: Structure and Method *Systems, Objectives, Solutions* 1, pp.5–19

Mumford, E. (1983) “Designing Human Systems for New Technology. The ETHICS Method” Manchester Business School, Manchester

Perby, M.-L. (1987) Computerization and the Skill in Local Weather Forecasting *in* “Computers and Democracy: A Scandinavian Challenge”, Bjerknes, Ehn, and Kyng (eds.), Avebury, Aldershot, pp.213–229

Wiig, Ingrid (1994) “Methodological Support for User Participation and Creativity in System Development” Master Thesis, Department of Informatics, University of Oslo

Yourdon, Edward (1989) “Modern Structured Analysis” Yourdon Press, Englewood Cliffs, NJ