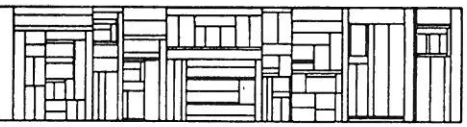
ISBN 87-88715-05-1
ISSN 0105-8517

Evaluation of System Development Projects

Andreas Munk-Madsen

DAIMI PB-218
MARS report No. 15, August 1986

DATALOGISK AFDELING
Bygning 540 - Ny Munkegade - 8000 Aarhus C
tlf. (06) 12 83 55, telex 64767 aausci dk
Matematisk Institut Aarhus Universitet



TRYK: RECAU (06) 12 83 55

ISBN 87-88715-05-1
ISSN 0105-8517

Evaluation of System Development Projects

Andreas Munk-Madsen

DAIMI PB-218
MARS report No. 15, August 1986

EVALUATION OF SYSTEM DEVELOPMENT PROJECTS

Andreas Munk-Madsen
Computer Science Department
Aarhus University
Denmark

Contents

Abstract

Many groups are involved in system development projects: System development management, system developers, user management, and users. All of them wish to influence the project. Influence, however, requires understanding. The process of obtaining this understanding is called project evaluation.

This article will discuss project evaluation. The most important conclusions are:

- Evaluation should not be seen as an appendage to planning, but as a complementary activity.
- There are two fundamentally different kinds of project evaluation.
- There are two types of documents constituting the key for understanding most system development projects.
- In practice evaluation of system development projects requires the active participation of the system developers.

1. Introduction

Participating in a system development project, I discovered that the project followed a curious line of action. Apparently the project did not move forward - at least not in any meaningful direction. It was, however, very difficult to describe what was wrong, and it was impossible to offer any reasonable answers to why the project took the course that it did.

I have seen researchers and shop stewards go to great lengths to evaluate the consequences of a future computer system, only to find that the system was never to be introduced. Inherent antagonisms in the system development project entailed that the project was never completed. But this fact was beyond the scope of the researchers and the shop stewards. Perhaps they would have made more of their time and efforts, if they had known.

These observations indicate that project evaluation is a non-trivial activity which is relevant in various situations. Project evaluation can be defined as an activity which creates a better understanding of a system development project.

Various groups are interested in a system development project: System development management, user management, the actual users represented by their trade unions, sub-contractors, and the system developers who carry out the project. These groups may pose different questions to the project in progress:

- Will the product and the change in working practices be of sufficient quality?
- Will the project finish on time?
- Will the project make a profit?
- In which ways can the project be influenced?

Formulating and answering these questions is part of project evaluation. (There are other questions concerning a project which are beyond the scope of project evaluation. The question "what should the product be like?", for instance, belongs in the design process.)

2. Towards a Theory of Evaluation of System Development Projects

The following section will call attention to the great practical need for more knowledge about project evaluation.

This article will then discuss project evaluation on the background of the MARS-project's (MARS 84a) findings. The mode of representation cannot, however, follow the process of cognition, but must take its starting point in general concepts for understanding system development projects. The system development project plays a double role in relation to project evaluation. It is both the object and the environment of project evaluation.

Therefore, I will first look at system development projects from a process-structure point of view (Mathiassen 81) in order to identify principal types of project components (Section 4).

Then system development will be examined from a more detailed functional point of view in order to identify important types of activities and their qualitative interactions (Section 5).

The issue of competence and discipline in project evaluation will then be addressed (Section 6).

And finally the article will discuss principles for project evaluation (Section 7).

3. State-of-the-Art of Project Evaluation

3.1 System Development in Practice

In the MARS-project (MARS 84a) we have studied system development in practice. In our experience project evaluation is carried out in an unmethodical and defective manner:

- estimates are made implicitly,
- prerequisites of the project plan are not documented,
- intermediate products are accepted on face value,
- the system developers' working practices are not discussed,
- resources (e.g. people and tools) are expected to be available even though it is known that other projects compete to get them,
- problematic situations are identified later than is necessary,

- there is no conscious distinction between planning, evaluation, and regulation in problematic situations, which means that a lot of time is spent on regulation, and too little time is spent on evaluation and planning.

3.2 Our Research Experience

The MARS-project has been in contact with many projects. On each occasion we have - as a subordinate activity - evaluated the projects.

In my experience my first evaluation is often based on cues which are interpreted as clues. I.e. I hear a remark concerning a specific phenomenon in a project. My experience or theoretical background tells me that it may entail a certain outcome of the project. I then pose some elaborating questions and present a preliminary evaluation of the phenomenon in question. I have often been proven wrong. The phenomenon was insignificant seen in relation to other project elements which I did not have the opportunity to examine. On other occasions I was right. Circumstances to which project members did not attach any importance - which I however had been aware of - proved to influence the course of the project significantly.

The MARS-project has produced a number of reports describing and evaluating system development projects (MARS 84b, MARS 84c, MARS 84d, MARS 84e, MARS 85a, MARS 85b, MARS 85c). Our research process is also described in (Munk-Madsen 86a).

We have tried to systematize our descriptions - among other things on the background of (Lanzara and Mathiassen 85). If our descriptions are sufficient this may, however, well be attributed to persistence rather than to systematism. It is not difficult to produce interesting project descriptions. But it is difficult to make them accurate and systematic.

3.3 Literature

The issue of system development management is generally acknowledged as important. There is a growing awareness of the uncertainty and complexity in system development. "... the picture of the software designer deriving his design in a rational, error-free, way from a statement of requirements is quite unrealistic. We believe that no system has ever been developed in that way, and probably none ever will."(Parnas and Clements 85).

In (Glaser 84) it is said: "Managing a well-planned and well-staffed project is challenging enough; with fuzzy objectives, an unrealistic schedule, an inadequate budget, and weak staffing, project managers would need a miracle to succeed. Yet projects with these weaknesses are launched every day. In fact, projects are routinely launched under circumstances that even further increase the chance of failure." Glaser

then mentions conditions as: overly ambitious objectives, crash schedules, cross-functional scope, and technological complexity.

This uncertainty motivates the discussion of project evaluation. How does the literature recommend that we handle the uncertainty? The main answer is prevention - i.e. improved project planning. Parnas and Clements suggest that we reduce the uncertainty by using a project model which they call a rational or an ideal design process. It is described in terms of standard intermediate products. However, Parnas and Clements say, that we should not expect to be able to follow this ideal process - only to fake it.

Glaser's solution is also primarily based on prevention: project start-up, project management philosophy, and project plans and schedules. The main elements in his recommended plans are milestones and tasks.

Regarding project evaluation it is characteristic that both papers only mention reviews. System development literature usually sees project evaluation as an appendix to planning. It is stated that the plans naturally have to be "followed up" upon, but the difficulties in doing so are not discussed (Mikkelsen et al 84), (Lööf 81). When the literature focuses on project evaluation as e.g. (DeMarco 82), evaluation is reduced to answering quantitative questions.

(Howes 84) "documents a philosophy for software development and the tools used to support it. Those management techniques deal with quantifying such abstract terms as "productivity", "performance", and "progress", and with measuring these quantities and applying management controls to maximize them". Later in his paper, Howes says: "There are three reasons why software development work does not progress in accordance with the plan. They are:

1) changes in the scope of work,

2) quantity deviations, and

3) productivity deviations."

This may be true at the end of an evaluation. But the statement represents a premature assumption of reduced complexity. This assumption may be found in many papers on quantitative evaluation. The point is, that problems do not present themselves with a fixed label. The difficult task is to identify the problem in an uncertain situation. You cannot quantify what you have not conceptualized.

We may note that the literature contains valuable solutions to improved understanding of system development projects: Better project plans, baselines, reviews, and techniques for quantitative evaluation.

But the literature does not give the whole answer. Prevention reduces, but does not remove the uncertainty and complexity. We must also be able to evaluate projects in those situations that are characterized by high uncertainty and complexity.

The generally recommended evaluation techniques - reviews - are not sufficient. They prescribe the organizational setting of the evaluation rather than the content of the activity.

Quantitative evaluation techniques are valuable - under the precondition, that the situation has been conceptualized.

Thus, it will be interesting to see, which answers a direct focusing on project evaluation may produce.

4. Principal Project Components

The core of a system development project is activities which consume resources (primarily man-power, tools, and facilities), and which result in products and a change in the users' working practices. Into the activities there will typically enter intermediate products which are the results of earlier activities. The system development activities are affected by the system development organization and (naturally) by the users' current working practices. The system development activities have side effects. Especially that the system developers' qualifications are improved, and that the system development organization is changed. This is illustrated in Figure 1.

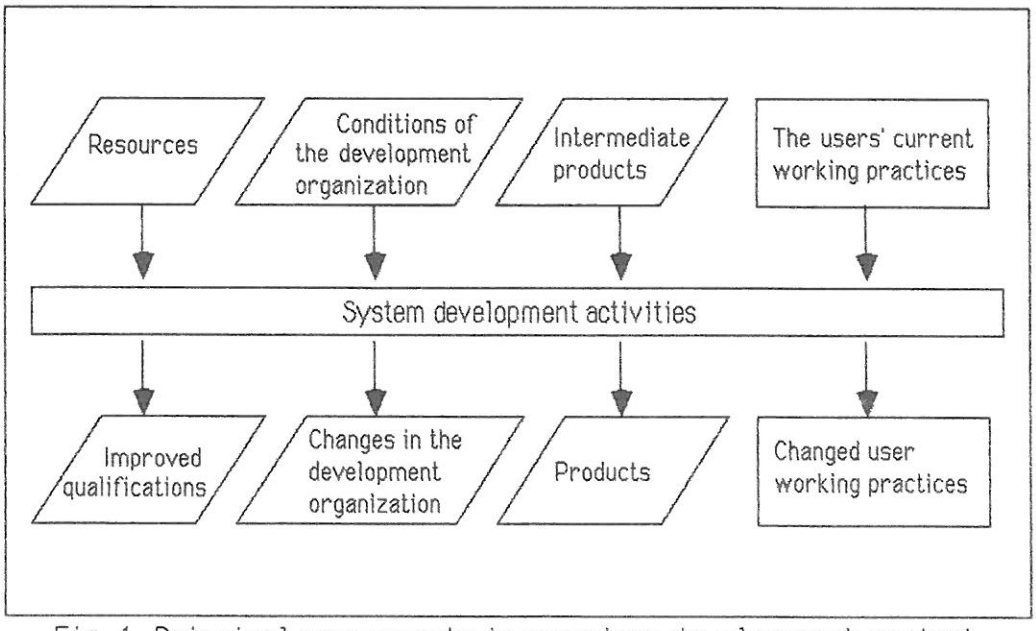


Fig. 1. Principal components in a system development project.

In this perspective on system development projects, the task of the project evaluation is to:

- examine whether the principal components of a project are in balance at any time,
- elucidate the expectations of various groups to the principal components,
- examine the relation between expectations and reality,
- examine reasons for and consequences of discrepancies.

Under the non-trivial assumption that the project is carried out under reasonably predictable circumstances - i.e. that there is agreement between expectations of various groups, and that these expectations are documented in the project plan - the task of project evaluation can be reduced to:

- examining whether the project plan's descriptions of principal project components are consistent,
- examining whether the plan's descriptions agree with reality,
- examining problems arising in the project and relating them to the plan's expectations.

5. System Development

(Mathiassen 81) defines the function concept, and identifies some sub-functions in system development. Project evaluation does not appear very clearly in this division into functions. System development may, alternatively, be decomposed into sub-processes - see (Munk-Madsen 84). This decomposition does not, however, illustrate the fundamental interrelations between specific sub-processes. The MARS-project has developed a modified description of the activities of system development, which will be presented below (Andersen et al. 86).

System developers perform two types of creative activities. They create a computer based system, i.e. the users' future working processes, the software, and other sub-products. And they create a project, i.e. their own working practices which are to result in the products and the changes in the users' working practices. We can denote the first type of activities as the system development proper, and the other type of activities as system development management.

Within both of these activities we may distinguish between descriptive activities and executive (changing) activities.

The descriptive activities may again be divided into activities describing things which must be established in the future (design of a product or planning of a project), and activities describing current situations and the consequences of changing them (analyses of the users' working practices, appraisal of technical options, evaluation of a project).

Figure 2 illustrates these activities and their interaction.

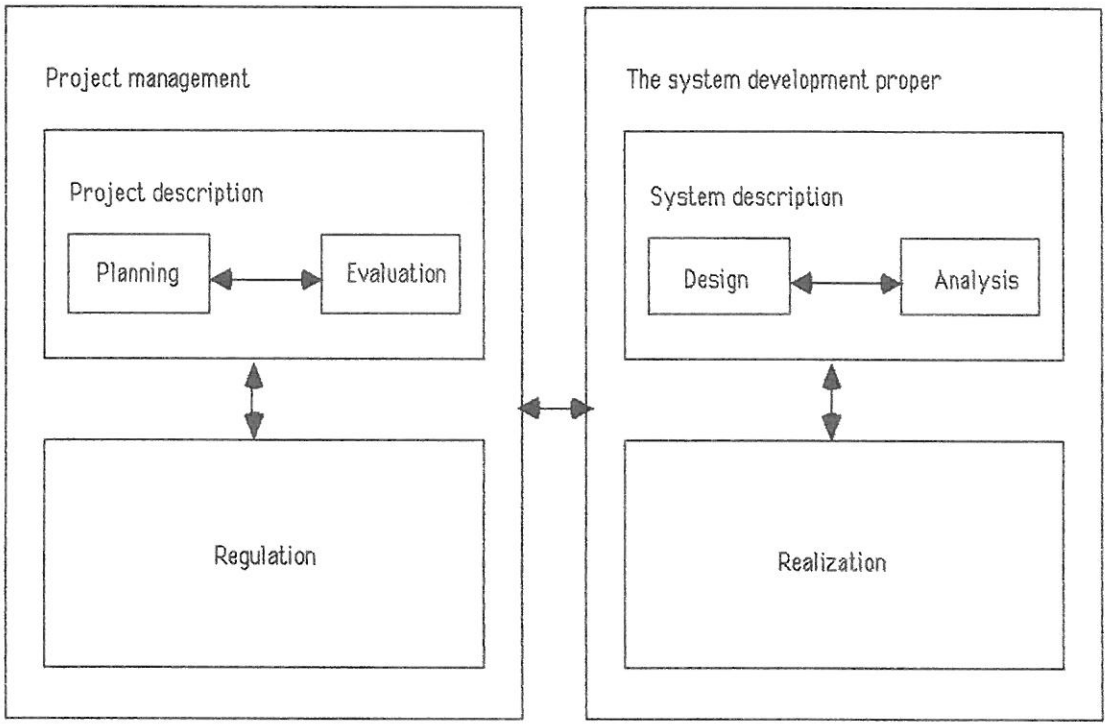


Fig. 2. Fundamental activities in system development.

The right-hand side of Figure 2 illustrates subordinate activities in *the system development proper*: *Analysis*, which deals with understanding and describing the users' working practices and organization, and with appraising technical options. *Design*, which deals with contriving and describing software products, and changes in the users' working practices. *Realization*, which deals with changing the existing computer system and the users' working practices and organization.

The interactions between these activities are illustrated by the arrows to the right in Figure 2. They can be interpreted as follows:

← Understanding the users' working practices and organization and the technical options is necessary to be able to contrive a realistic design.

→ A notion of desirable changes is necessary to be able to goal-orient the analysis.

↓ A description of both the actual situation, as well as the desired development is necessary to be able to carry out changes.

↑ Experience obtained by trying to introduce changes form the basis for renewed analysis and design.

From this perspective the most important intermediate product in the system development proper is the overall technical and functional design. It is important because it constitutes the interface which all the activities work from.

We can characterize project management in the same way, as illustrated in the left-hand side of Figure 2. We can identify three fundamentally different activities in project management: *Project evaluation*, which deals with understanding and describing the system developers' working practices and their organizational and technological frames. *Project planning*, which is aimed at designing the project, i.e. to contrive and describe the project and the necessary prerequisites for effecting it. *Regulation*, which aims at bringing the system development project and its environment in accordance with the project plan.

The interactions between these three activities are illustrated by the arrows to the left in Figure 2. They can be interpreted as follows:

← Understanding the current course and the conditions of the project is necessary to be able to contrive a realistic project plan.

→ A notion of desirable project courses is necessary to be able to goal-orient the project evaluation.

↓ A description of both the current project situation, as well as its desired development is necessary to be able to regulate the project.

↑ The experience gained through trying to regulate the project forms the basis for renewed evaluation and planning.

These considerations lead to some important statements concerning project evaluation and project management.

The most important intermediate product of project management is the result of planning - the current version of the *project plan* It is

important because it constitutes the interface which the evaluation and the regulation works from.

Poor (read: unsystematic) project management is characterized by the fact that planning and evaluation are given lower priority than regulation.

Planning must be especially thorough in those areas of the project where problems can be expected. It is the task of project evaluation to identify these areas.

Finally the arrows in the centre of Figure 2 illustrate the interaction between the system development proper and project management. They can be interpreted as follows:

← Information concerning the course of the project is necessary to be able to manage the project.

→ Management of the project and its environment affect the course of the system development proper.

There is a strong analogy between project management and the system development proper:

The plan is the key product in project management. The overall design is the key product in the system development proper.

Poor project management is characterized by poor planning and evaluation. Poor system development is characterized by poor design and analysis (entailing a lot of implementation and repairs).

Project evaluation must identify the areas in the project which require thorough planning. System analysis must identify the areas in the system which require a thorough design.

6. Project Evaluation - Discipline and Competence

Section 3.1 described how little effort system developers spend on evaluating their projects. One could ask the following question: Why do system developers not evaluate their own projects?

In my opinion the reason is a combination of lack of competence (i.e. the ability to do something) and the lack of discipline (i.e. the determination to do it).

In theory there are plenty of incentives to perform project evaluation. Most system development literature encourages it. But it often remains

a formality. One of the reasons for this is expressed by Freedman and Weinberg: Formal manuals "concentrate on the *how* and neglect the *why*" (Freedman and Weinberg 82).

Discipline is a question of tradition on the place of work and adaption of attitudes during training. In their training, programmers are thoroughly indoctrinated to test programs. Sometimes the discipline nevertheless fails. How can anything else be expected where project evaluation is concerned - a subject which takes up less time in most curricula.

However, discipline cannot be taught, it can only be practiced.

Let us therefore turn to the question of competence: How can projects be evaluated?

7. Project Evaluation Techniques

Project evaluation can be performed in different situations. We can identify two extremes. In the first type of situation the prevailing assumption is that everything goes according to plan. To check this assumption a routine inspection is performed. We call this activity project auditing. The second type of situation is a problem-setting situation (Mathiassen and Munk-Madsen 85, Lanzara 83). In a problem-setting situation the participants recognize that certain things in the project are out of balance. The objective of project evaluation is here to obtain a better understanding of the problem to be able to make a new plan and regulate the project and its environment. We call this activity problem investigation.

We may note the similarity to scientific research activities. Project auditing corresponds to testing hypotheses. Project investigation resembles conceptual and experimental exploration (Munk-Madsen 86a).

7.1 Project Auditing

Project auditing includes examining whether the plan is consistent, and whether the plan is in accordance with reality.

Examining the plan's consistency embraces the answers to the following questions:

- Can the activities be performed with the available resources and the planned intermediate products under the expected conditions?
- Will the activities lead to the desired products and the desired changes?
- Are the time schedules in the activity plan internally consistent?

Some of these questions are of a quantitative nature, and here various calculation techniques may be employed. The time schedule for the activities can be subjected to a network analysis, and estimation techniques may be employed when assessing the extent of the activities and the consumption of resources.

The most important questions are, however, of a qualitative nature (e.g. "can the system developers make a sufficiently good design in a process which only involves minimal interaction with the users?") Answering this type of questions requires knowledge about system development in the form of theory (Munk-Madsen 86a). It is also required that the project plan documents the expectations to the various project components in standard terms in order to make theoretical statements applicable (Munk-Madsen 86b).

Examining the plan's accordance with reality embraces the answers to the following questions:

- Have the planned resources been available, and have they been spent according to plan?
- Were the intermediate products finished on time, and did they have the right quality?
- Have the planned changes taken place?
- Do the current activities correspond with the plan?
- Will the future resources and conditions be as planned?

To obtain the necessary information different types of techniques may be employed: interviews, inspections, reviews, tests, and registration individually performed by the system developers. Of these techniques only tests and registration are real techniques in the sense that they prescribe the content of the evaluation activity. Interviews, inspections, and reviews especially prescribe the organization of the activities, whereas the actual evaluation is left to the persons performing the evaluation. They can be characterized as auxiliary techniques (Hansen and Jensen 85).

It is very difficult and expensive to evaluate ongoing activities. For that reason project evaluation is facilitated when intermediate products and other results are grouped in baselines. A baseline is a predefined project state where intermediate products and results are characterized by evaluation criteria and procedures. "Management of any process that is not described in terms of work products can only be done by mindreaders. Only if we know which work products are due and which criteria they must satisfy can we review the project and measure progress" (Parnas and Clements 85).

Again there are questions of both a quantitative and a qualitative nature. Quantitative questions especially base themselves on time-recording tables, and may often be supported by calculation tools. It is the qualitative questions, however, which are the hardest to answer. Consider a question like "Does this design document fulfill the criterion of being comprehensive?" Answering this question requires knowledge - theory or experience - in the form of a standard or checklist for this type of intermediate product.

A major practical problem in connection with project evaluation is that project plans are rarely good enough to support these analyses. There is only one thing to be done about that: the plan must be reconstructed from the system developers' notion of the course of the project. If this is not done, there is only one thing which can be said about the project with certainty: that nobody can say anything reasonable about its status.

Project auditing will often result in recognizing a problem - e.g. a discrepancy between the plan and reality. This brings us to the other type of project evaluation: problem investigation.

7.2 Problem Investigation

Problem investigation is carried out in problem-setting situations. The task is primarily to identify and analyse the problem so that the developers can generate ideas for solving the problem, and so that they can assess the usefulness of these solutions.

On the outset the problem is rarely well-defined. Different people have different perceptions of the problem. It is therefore necessary to start by mapping how the involved people see the situation, and from there to work towards a common understanding of the problem. The best way to do this is in a dialogue between the involved people.

Further analysis takes place as an iteration between re-defining the problem, identifying the reasons for and the consequences of the problem, and assessing the usefulness of possible actions.

It is important that these interpretations of the project are based on reliable information, preferably incontrovertible facts. To obtain this it will often be necessary to perform project auditing type of activities. The techniques mentioned here are described in detail in (Lanzara and Mathiassen 85).

The issue of defining the central problem in a problem-setting situation is further explored in (Munk-Madsen 86b). It is proposed that the central problem should be formulated as a contradiction reflecting

a. either a relation between properties of project components which negate a normative theoretical statement,

b. or an antagonism between the actual course of events and the expectations of people involved in the project,

c. or an antagonism between the expectations of different people involved in the project.

Problem investigation is facilitated by the use of standard terms in characterising project components and by the availability of relevant theory on system development.

It is worth mentioning that problem investigation in practice almost always is carried out in an unsystematic manner, and therefore is more time-consuming than is necessary, and gives poorer results than would be possible.

8. Conclusions and Corollaries

The major conclusion of the present discussion is that project evaluation should not be seen as an appendage to project planning. There are fundamental dialectic relations between evaluation and planning, evaluation and project situations, evaluation and knowledge on system development (both general theory and specific project information), and evaluation activities and evaluation theory and methods. Through examining the aspects of these relations separately and in their totality we may reach important statements on system development project evaluation.

This article has emphasized the interaction between planning and evaluation, the two different types of project evaluation, and the role of theory and the role of the project plan in evaluation. The article has placed existing evaluation techniques within a theory of project evaluation.

One important question remains: who should (or can) evaluate system development projects? We will address this issue now.

Section 5 argued that the project plan is a key document in project management, and that the overall design is the key product in the system development proper. The techniques for project evaluation described in Section 7 very much build on the project plan, and I claim that it has to be like that. I also claim that in going into describing the concrete qualitative relations between the principal components in a project (see Section 4), one will find that the overall technical and functional designs play a similar central role.

In the MARS-project we have found that project plans rarely are of sufficient quality. I.e. they do not describe all the principal components in the project, and above all, they do not account for the relationship between the components in a satisfactory manner. As mentioned in Section 7.1 this means that to be able to perform project evaluation, the plan must first be reconstructed from the system developers' notions.

This entails that project evaluation in practice requires the active involvement of the system developers, and that an organizational separation of project evaluation from the other system development activities is impossible.

What are the implications of this statement for all the many groups (e.g. management and users) that are interested in understanding and influencing a system development project? Their possibilities to influence depend on the active involvement of the system developers. An enforcement of external control without the system developers' consent may often lead to increased irrationality and unpredictability in the project.

References

(Andersen et al. 86): N.E. Andersen, F. Kensing, M. Lassen, J. Lundin, L. Mathiassen, A. Munk-Madsen, P. Sørgaard, Professionel Systemudvikling (Professional System Development), Teknisk Forlag, 1986.

(DeMarco 82): T. DeMarco, Controlling Software Projects, Yourdon Press, 1982.

(Freedman and Weinberg 82): D.P. Freedman and G.M. Weinberg, Handbook of Walkthroughs, Inspections, and Technical Reviews, Little, Brown and Company, 1982.

(Glaser 84): G. Glaser, Managing Projects in the Computer Industry, IEEE Computer, October 1984, p. 45-53.

(Hansen og Jensen 85): P.B. Hansen and P.J. Jensen, Estimering af system-udviklingsprojekter - praksis, teori, eksperiment (Estimating System Development Projects - Practice, Theory, Experiment), MARS-report no. 7, Aarhus University, 1985.

(Howes 84): N.R. Howes, Managing Software Development Projects for Maximum Productivity, IEEE Transactions on Software Engineering, vol. SE-10, no. 1, January 1984, p. 27-35.

(Lanzara 83): G.F. Lanzara, The Design Process: Frames, Metaphores and Games, in U. Briefs et al. (eds.), System Design for, with, and by the Users, North-Holland, 1983.

(Lanzara and Mathiassen 85): G.F. Lanzara and L. Mathiassen, Mapping Situations within a System Development Project, in Information Management, 1985, 8 (1), p. 3-20.

(Lööf 81): P.-O. Lööf, Effective Management of Systems, in Konferencedokumentation NordDATA 1981 I, p. 121-126, Dansk Databehandlingsforening, 1981.

(MARS 84a): MARS-report No. 1, Project Description. MARS - A Research Project on Methods for Systems Development, Aarhus University, 1984.

(MARS 84b): MARS-rapport nr. 2, Systemudvikling i praksis: Jydsk Telefon-Aktieselskab (Systems Development in Practice: The Jutland Telephone Company), Aarhus University, 1984.

(MARS 84c): MARS-rapport nr. 3, Systemudvikling i praksis: Regnecentralen af 1979 (Systems Development in Practice: RC-Computer), Aarhus University, 1984.

(MARS 84d): MARS-rapport nr. 4, Systemudvikling i praksis: Sparekassernes Datacenter (Systems Development in Practice: The Computing Centre for Danish Savings Banks), Aarhus University, 1984.

(MARS 84e): MARS-rapport nr. 5, Systemudvikling i praksis: ØK Data (Systems Development in Practice: EAC Data), Aarhus University, 1984.

(MARS 85a): MARS-rapport nr. 8, Forandring af arbejdsformer i systemudvikling: Jydsk Telefon-Aktieselskab (Changing Working Practices in Systems Development: The Jutland Telephone Company), Aarhus University, 1985.

(MARS 85b): MARS-rapport nr. 9, Forandring af arbejdsformer i systemudvikling: Regnecentralen af 1979 (Changing Working Practices in Systems Development: RC-Computer), Aarhus University, 1985.

(MARS 85c): MARS-rapport nr. 10, Forandring af arbejdsformer i systemudvikling: ØK Data (Changing Working Practices in Systems Development: EAC Data), Aarhus University, 1985.

(Mathiassen 81): L. Mathiassen, Systemudvikling og systemudviklings-metode (Systems Development and Systems Development Method), Aarhus University, DAIMI PB-136, 1981.

(Mathiassen and Munk-Madsen 85): L. Mathiassen and A. Munk-Madsen, Formalization in Systems Development, in H. Ehrig et al. (eds.), Formal Methods and Software Development, Springer-Verlag, 1985.

(Mikkelsen et al. 84): H. Mikkelsen, J.O. Riis, H.J. Schmidt, Styring af administrative systemudviklingsprojekter (Managing Administrative Systems Development Projects), PROMET, 1984.

(Munk-Madsen 84): A. Munk-Madsen, Practical Problems of System Development Projects, in M. Sääksjärvi (ed.), Report of the Seventh Scandinavian Research Seminar on Systemeering, 1984. II p. 280-298, Helsinki School of Economics, 1984.

(Munk-Madsen 86a): A. Munk-Madsen, Viden om systemudvikling (Knowledge on System Development), MARS-report no. 13, Aarhus University, 1986.

(Munk-Madsen 86b): A. Munk-Madsen, Project Evaluation - Setting the Problem, MARS-report no. 14, Aarhus University, 1986.

(Parnas and Clements 85): D.L. Parnas and P.C. Clements, A Rational Design Process: How and Why to Fake It, in H. Ehrig et al. (eds.), Formal Methods and Software Development, Springer-Verlag, 1985.