

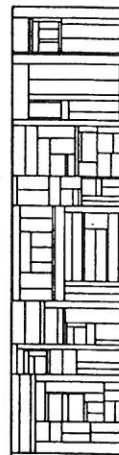
ISBN 87-88715-05-1
ISSN 0105-8517

Project Evaluation - Setting the Problem

Andreas Munk-Madsen

DAIMI PB-216

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



TRYK: RECAU (06) 12 83 55

MARS report No. 14, July 1986

Contents

1. Introduction	1
2. Project Evaluation and Project Situations	2
3. Project Graphs and Problem Networks	4
4. Maps, Graphs, and Networks	7
5. Denoting The Network Nodes	10
6. The Difficult Choice – Setting the Problem	14
7. The Relation between Project Graphs and Problem Networks	17
8. Exorcising the Evil of Excuses	19
9. Project Planning Should Make Theories Explicit	19
10. Cognitive Mapping and Problem Setting	20
11. Conclusion	21
References	21

Project Evaluation – Setting the Problem

Andreas Munk-Madsen
Computer Science Department
University of Aarhus
Denmark

Abstract

Evaluation of a system development project denotes those activities which aim at improving the understanding of the project and its environment. This understanding forms the basis for planning and regulating the course of the project. Project evaluation often takes place in a project situation which only deviates slightly from the plan. The project is then evaluated merely to determine the extent and consequences of the deviations. There are situations, however, where the system developers know that something is wrong in the project. But maybe they cannot identify the problem, or they disagree on what the problem is. The task is to set the problem. This paper discusses how project evaluation can be performed in situations like that. It discusses guidelines for defining problems, and claims that a problem should be formulated as a contradiction. Project graphs and problem networks are introduced as description tools in project planning and project evaluation, respectively.

1. Introduction

The objective of this paper is to give some guidelines for evaluating system development projects in problem setting situations.

Literature on system development management – e.g. (Boehm 81), (DeMarco 82), and (Dunn and Ullman 82) – give guidelines for evaluating projects, which are particularly applicable in situations where the goal or problem is well-known. These guidelines describe how it is possible to procure (especially quantitative) information about the project, and use this information to decide which method to apply to solve the problem or reach the goal.

There are, however, situations in system development projects where the system developers disagree on what the goal is; or situations where they have a notion that something is wrong without knowing exactly what it is. A decisive step out of a situation like that is a precise and relevant problem definition. Literature on system development management does not offer very many guidelines for this.

This issue may also be seen in another way: How is the theoretical knowledge about systems development applicable in a problem setting situation? Theoretical knowledge can typically be found in the form of a method – e.g. Structured Analysis and Structured Design (Yourdon 82) and Jackson System Development (Jackson 83).

But in practice system development projects seldom adhere to these methods (MARS 84a, MARS 84b, MARS 84c, MARS 84d). In a normal project there will be numerous deviations from any system development method. Which of these deviations constitute actual problems? To answer this type of question properly, you need a relevant project evaluation technique, and a suitable formulation of the theories underlying the system development method.

This paper will propose some answers to this type of question. The paper takes its starting point in a technique for mapping project situations (Lanzara and Mathiassen 85). The paper proposes requirements to the formulation of problems. These requirements may be applied as guidelines when evaluating projects.

The paper is structured as follows: Section 2 discusses evaluation activities and project situations. Section 3 discusses descriptions of projects, including different types of statements concerning projects and different graphical representations of projects. Section 4 takes a closer look at the relation between various tools for describing projects: maps, project graphs, and problem networks. Section 5 analyses the nature of the various statements represented in the problem networks, and proposes some requirements to problem formulations. Section 6 discusses criteria for determining the central problem in a given situation. Section 7 discusses the relation between graphical representations of projects and problem situations. Corollaries on project management and project planning are deducted in sections 8 and 9. Finally, the approach to problem setting in this paper is compared with the sociological method of cognitive mapping in section 10.

2. Project Evaluation and Project Situations

In this section the context of project evaluation is discussed. This context can be seen partly as the activities which surround and interact with project evaluation, and partly as the project situations in which project evaluation takes place.

In a system development project we may distinguish between system development execution and system development management. System development execution denotes the activities that aim at creating the project's deliverables: products and services. System development management denotes the activities that aim at creating and maintaining the project.

Within project management we may distinguish between different activities. Partly the regulating activities, which serve to form the project and its environment according to a plan. And partly the reflective activities which again may be divided into project planning, which aims at creating coherent visions about possible futures, and project evaluation which aims at understanding the project and its environment.¹

¹ This understanding of systems development has been developed in the MARS-project. See, e.g. (Munk-Madsen 85).

These activities cannot be separated in time and space in practice. There is a close interaction between the visions created when the project is planned, and the understanding of the possibilities and limitations obtained when the project is evaluated. Likewise there is a close interaction between reflection and regulation or action. " ... action on the situation is integral with deciding, and problem solving is a part of the larger experiment in problem setting" (Schön 83).

This conceptual isolation of project evaluation is an abstraction. The abstraction serves the analytical objective: that is focuses on the evaluation function and the techniques linked to this function. When you look at a specific technique, you must observe the fact that it may serve several different functions. Mapping – which we will discuss in more detail later in this paper – is a technique which both serves an objective within management, and an objective within communication. Communication activities run across our division into project management and execution.

If we take a closer look at specific evaluation activities, we can identify two types of activities: project auditing and problem investigation (Munk-Madsen 85). Project auditing is a routine examination of the situation in the project under the assumption that the project follows the plan. Problem investigation is performed in a situation where the project participants sense that something is wrong in the project. But they do not know what it is, or they disagree on what it is. Problem investigation serves to set the problem.

Inspired by (Lanzara 83), (Mathiassen and Munk-Madsen 85) distinguish between three types of situations: routine, problem solving, and problem setting.

A routine situation is characterized the fact that the task is well-defined, and that those who are going to carry out the task know a method for solving it.

A problem solving situation is characterized by the fact that the task (the problem) is well-defined, but the system developers do not see any obvious way of solving it.

A problem setting situation is characterized by the fact that the involved parties know that something should be changed, but cannot see or disagree on what it is that should be changed. How the change should be made is subsequently also unclear.

In relation to this distinction we can clarify that problem investigation takes place in problem setting situations.

The following will discuss concepts, statements, techniques, and description tools which are useful in problem setting situations.

3. Project Graphs and Problem Networks

This section will define two different tools for describing system development projects: Project graphs and problem networks.

Two characteristic types of statements may be seen in system developers' descriptions of their projects. One type is statements representing assessments of project components (e.g. "Person x is good at designing data bases". "Document y is a good analysis"). Another type is statements expressing the relations between project components (e.g. "We perform activity z to obtain results w_1 , w_2 , and w_3 ". "Activity x requires the conditions u_1 and u_2 ").

These relations may be graphically represented in what we call project graphs. The most general type of project graph links all types of project components to each other. This graph is illustrated in figure 3.1.

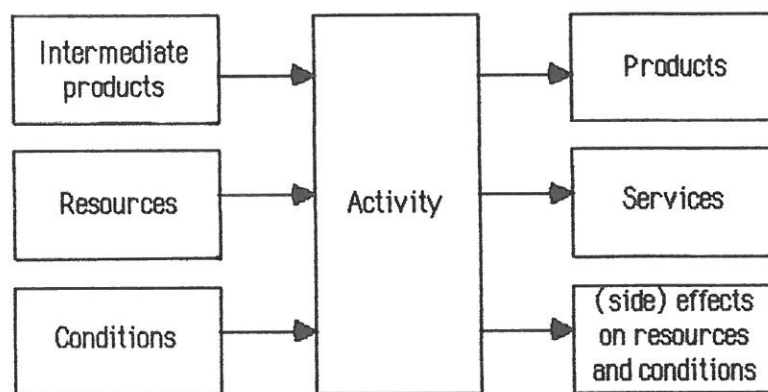


Figure 3.1. A general generic project graph

At the same time figure 3.1. illustrates the principal types of project components. The fundamental type of component in a project is work organized into activities. The activities are performed to create deliverables, which may be divided into products and services. Activities may result in side-effects. To be able to perform activities, resources are required (including man-power, tools, and facilities). Already finished work typically enters in the form of intermediate products. An activity is performed under conditions, among them the organization of the project.

Other project graphs may illustrate the relation between components of the same type, (see e.g. figure 3.2.), (Nielsen and Bjørner 85).

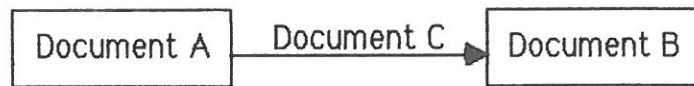


Figure 3.2. A project graph illustrating the relation between documents

This figure illustrates that document B has been worked out on the basis of document A. Document C expresses the relation between these two documents.

PERT-diagrams are another type of project graphs. They express temporally related interdependence between activities.

Project graphs express expectations to a project or the actual course of a project. Project graphs occur in project plans and in status reports. Figure 3.3. illustrates these two forms of project graphs.

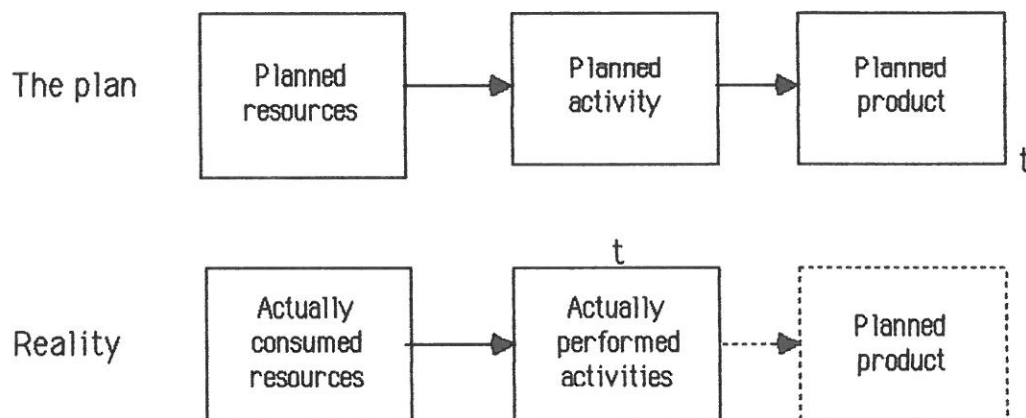


Figure 3.3. Two project graphs expressing expectations to a project and the actual course of a project

Project graphs are not directly applicable in problem setting situations. Problems "run across" the graphs. The starting point for project evaluation in a problem setting situation is statements about the project and causal relations between these. These statements and relations may be graphically represented in what we will call a problem network.

The following is an example of a discussion during project evaluation:

- What is the problem?
- We are behind schedule, the deadline will not be met!
- Why are we behind schedule?
- We did not get the necessary resources!
- We did not work efficiently!
- The schedule was unrealistic from the beginning!

The statements may be organized directly into a problem network as illustrated in figure 3.4.

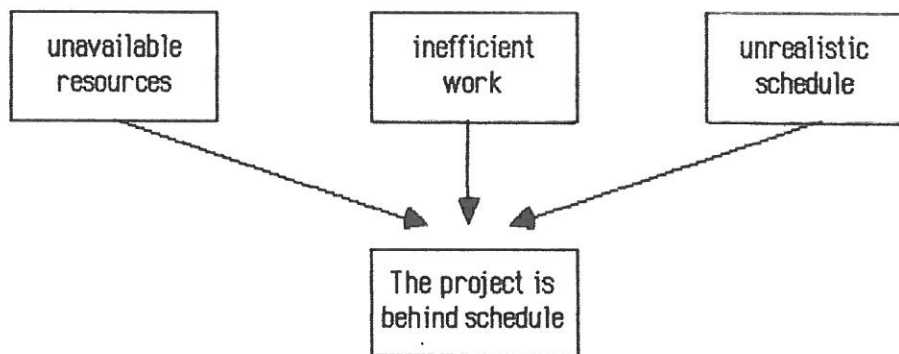


Figure 3.4. Example of a problem network

This example shows a network which is a result of a very superficial evaluation. Evaluation frequently stops here, and the project group ends up arguing about whether the problem is caused by one or the other reason. Later this paper will discuss how better problem networks are made, and the relation between project graphs and problem networks.

Thinking in networks is a profitable way to contemplate the problems. Networks stress the fact that there are several possible problem definitions. Questions of the following type may be formulated: What if we change this phenomenon – would that solve the problem? Mapping the situation of the project in the form of a network constitutes an important prerequisite for problem setting.

4. Maps, Graphs, and Networks

Maps as defined by Lanzara and Mathiassen (Lanzara and Mathiassen 85) may be transformed into graphs. Let us take a look at the relation between the various maps on one hand, and project graphs, and problem networks on the other hand.

Diagnostic maps describe the relations between causes, problems, and consequences on one hand, and the relations between the actual project situation and possible alternative situations on the other hand. A problem network like the one illustrated in figure 4.1. may be deduced from a diagnostic map. The alternatives do not fit into the picture. They express the first steps of planning a changed situation. So they really belong in a project graph.

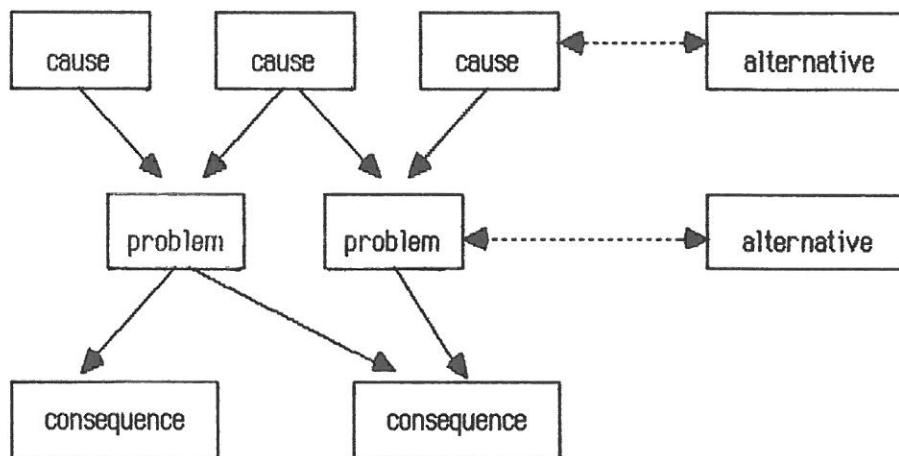


Figure 4.1. Diagnostic map represented as a problem network

Ecological maps elaborate the first kind of relation in the diagnostic maps by providing an outline of the causes of the problems. The causes are described as various types of conditions, and are organized in domains of action. These conditions are related to the situation (i.e. the current problem definition)². Ecological maps resemble problem networks in their graphical representation. Figure 4.2. illustrates an ecological map.

² A weakness in the graphic presentation of ecological maps in (Lanzara and Mathiassen 85) is that the arrows between the conditions and the situation are not denoted. This means that a major structuring criterion disappears. Ecological maps of this type will in practice often be fuzzy and tend towards inanity.

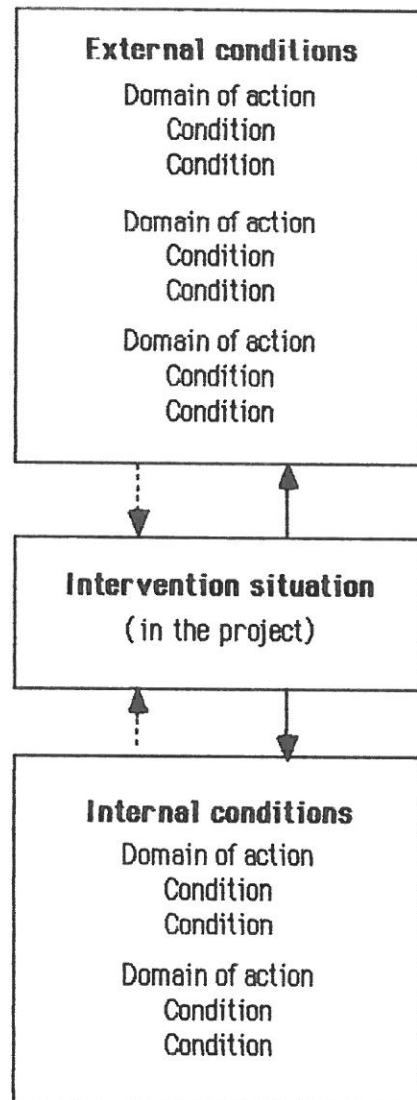


Figure 4.2. Ecological map

(After Lanzara and Mathiassen 85). Note that the arrows are not denoted, and that the situation is not expanded in detail.

Virtual maps elaborate the latter type of relation in the diagnostic maps by linking the actual situation (in the actual interpretation) to future desired situations through possible actions. Lanzara and Mathiassen do not show any illustration of virtual maps. Such an illustration would be very comprehensive, because virtual maps in reality are project plans.³ Project graphs illustrating expectations are thus a type of virtual maps.

³ Note the similarity between Lanzara and Mathiassen's question pattern for developing virtual maps (What do we want? What can we do? What would be the consequences? Do we get what we wanted?) and Polya's basic scheme for problem solving ("Understanding the problem. Devising a plan. Carrying out the plan. Looking back." (Polya 57)).

Historical maps, like diagnostic maps, describe the relation between phenomena. They differ from the latter by stressing the dimension of time, by classifying the phenomena into three categories (events, actions, and conditions), and by only describing actual phenomena (whereas the consequences in the diagnostic maps can be prognostications, and the alternatives hypothetical). Historical maps are project graphs illustrating the actual course of the project. Figure 4.3. illustrates a historical map.

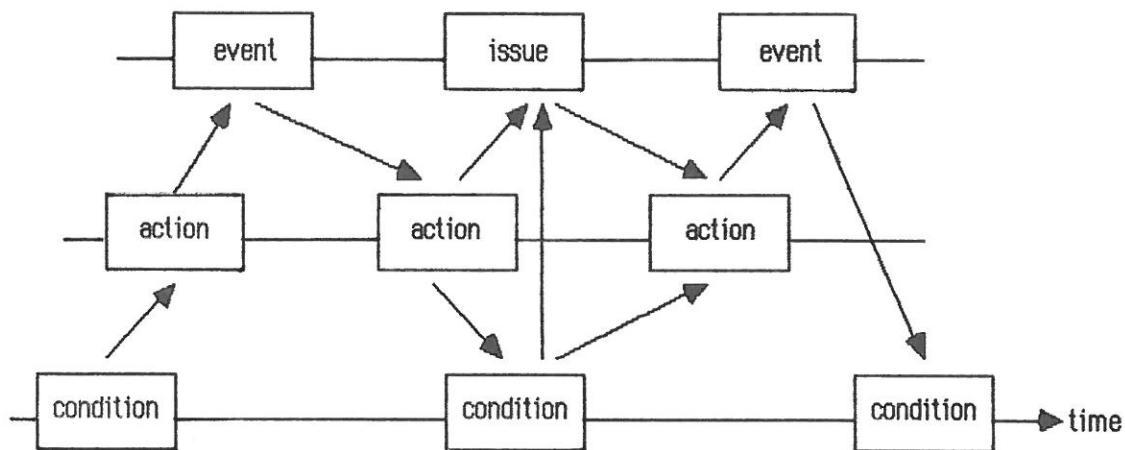


Figure 4.3. Historical map

We can summarize the discussion of maps, project graphs, and problem networks by stating that diagnostic and ecological maps have many things in common with problem networks, and that virtual and historical maps resemble project graphs.

Practice (MARS 84a, MARS 84b, MARS 84c, MARS 84d) has shown that maps in general, and diagnostic maps in particular, are useful tools in project evaluation. There are, however, certain difficulties connected to mapping. It can be difficult to distinguish between the various types of maps, and it can be difficult to determine when the situation has been analysed sufficiently.

Similar difficulties can be found in connection with networks. Two questions surface which - when looking closer - are intimately connected:

- Of what type are the nodes in a network? How are they denoted?
- What is selected as a problem? How is it possible to make sure that the problem has been thoroughly investigated?

Sections 5 and 6 will discuss these questions on the basis of a discussion of network semantics (what are the qualities of networks, what can you do with networks).

5. Denoting the Network Nodes

Of what form are the statements or the expressions which constitute the nodes in a problem network? Obviously some types of formulations are better than others.

Let us approach this question by considering how Lanzara and Mathiassen refer to the content of diagnostic and ecological maps. Their paper employs a number of general designations for the statements when going through diagnostic and ecological maps:

Diagnostic maps:

- problems, failures, anomalies
- causes
- actions, optional actions.

Ecological maps:

- conditions, domains of action.

We can come closer to the question by studying the examples in the paper.

The statements, figuring as nodes in the network corresponding to the diagnostic map in figure 5.1., are of the type:

- A (missing) property or functionality in an intermediate product (1, 2, 3, 10).
- An organizationally determined constraint for the project (4).
- A relation between an activity and a state of the project (5, 11).
- A relation between the way in which an activity is performed and a condition (6).
- A (missing) functionality in an applied description tool (7).
- A (missing) activity (8).
- A property of the resources (9, 15).
- A contradiction between the actual course of events and expectations (12)
- An expected event due to a (negative) condition (13).
- A (negative) side-effect (14).

The alternatives are formulated as a number of optional activities, which potentially negate the causes. (Note the use of the form "should be" in answer to the question: "What can be done".- we will return to that!)

What happened?	Why?	What are the consequences?	What can be done?
<p>1. The various system descriptions are incoherent.</p> <p>2. It is not possible to obtain detailed knowledge about the new system from the descriptions.</p> <p>3. The descriptions are unfinished and defective.</p>	<p>4. Meeting deadlines is given high priority.</p> <p>5. Time schedule produced before anything is known about the degree of change involved.</p> <p>6. New methods are applied, but no time for experimenting with them.</p> <p>7. The description tools are inappropriate for describing dialogues.</p> <p>8. No reviews have been performed.</p> <p>9. Many replacements of project members.</p> <p>10. No baselines are set up to indicate sub-products and criteria for evaluation of sub-products</p>	<p>11. New activities are initiated on a defective basis.</p> <p>12. At some point of development deadlines can no longer be met.</p> <p>13. The savings banks know too little about what they will get. A number of changes can be expected after implementation.</p> <p>14. Project members are often frustrated and unmotivated.</p> <p>15. Difficult for new project members to work efficiently.</p>	<p>16. Quality control should be improved during the early activities.</p> <p>17. Reviews should be performed with participation of both programmers and direct users.</p> <p>18. Time schedules should be based on descriptions of the degree of change involved.</p> <p>19. Time schedules should be changed as soon as the conditions on which they are based change.</p> <p>20. Versioning should have been applied as the basic strategy of the project.</p> <p>21. Different types of descriptions addressing different target groups should have been developed during the early stages of the project.</p> <p>22. There should be time for experiments when new methods are taken into use.</p>

Figure 5.1. An example of a diagnostic map
(The numbers are inserted by the author for reference reasons.)

External conditions

Relations to savings banks

1. Difficult to get access to internal savings banks routines.
2. JCN management wants the project to be a success.
3. No savings bank representatives within the project.

Project practices at JCN

4. Maintenance assignments are given higher priority than projects.
5. Little continuity of participants during project activities.
6. During the early activities participants have to finish other assignments.
7. JCN has no tradition concerning quality control.

Guidelines for project work

8. New methods are applied.
9. Few guidelines for quality control.

Existing equipment and system

10. Project based on new equipment.
11. New systems strongly depend on existing systems.

Special conditions

12. Time schedule and requirements specified in advance.
13. None of the participants took part in specifying time schedule and requirements.

Situation of budget project at JCN

Internal conditions

Values and norms

- 14 Participants do not acknowledge the necessity to experiment when using a new method.

Working practice

15. Disagreement to what extent the system should be based on existing programs.
16. Quality of products is identified with steering committees acceptance.
17. One-shot strategy as opposed to versioning.
18. Mainly formal contacts with savings banks.
19. Defective short-term management.

Figure 5.2. Example of an ecological map
(The numbers are inserted by the author for reference reasons.)

The conditions in the example of the ecological map in figure 5.2. are of the type:

- The organization frustrates an activity (1, 18).
- Expectations to the project (2).
- A (missing) resource, a (negative) characteristic of the resources, including the attitude of people (3, 5, 6, 10, 13, 14, 16).
- An applied criterion for allocating resources (4).
- (Missing) activities, characteristics of activities, (missing) descriptions of activities (7, 8, 9, 19).
- Properties of products (11).
- Properties of the plan or course of events (12, 17).
- A conflict in the project group (15).

To sum up, we can characterize the type of statements occurring in diagnostic and ecological maps as follows:

- Characteristics of project components (as described in the section on project graphs).
- In a minority of cases: relations, contradictions, or antagonisms between project components.

We can furthermore observe that there is a general lack of precision in Lanzara's and Mathiassen's example:

- The description of the situation is unbalanced; a deficiency of a product may be mentioned, but there is no comprehensive picture of positive and negative characteristics.⁴
- There is no attempt at giving products and activities general names.

These observations serve as inspirations to state some desirable characteristics of a node in a problem network:

Thesis 1:

A network node should be formulated as a contradiction, reflecting:

- 1.a. either a relation between properties of project components which negate a normative theoretical statement,
- 1.b. or an antagonism between the actual course of events and the expectations of people involved in the project,

⁴ In the examples space does naturally not allow for a comprehensive picture. The examples do illustrate Lanzara and Mathiassen's main message: the establishment of the cause-effect relations. The criticism here is primarily aimed at those who take the examples as paradigms.

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

In many of the cases where only one phenomenon or one property of a project component is formulated (e.g. "meeting deadlines is given high priority" or "no reviews have been performed"), the natural question is: so what? is it not an advantage? what is the problem? Naturally there is an underlying contradiction in every single case. But the contradiction is not explicitly formulated. And this makes it harder to discuss. It is not possible to see whether the contradiction can be solved by supporting one or the other opposite, or whether the contradiction reflects an insoluble conflict whose whole basis must be overcome in some other manner.⁵

Thesis 2:

The description of the project components should cover major positive and negative characteristics.

In the interest of evaluating solutions (plans) - (when you construct virtual maps) it is essential to document all important properties of project components. Otherwise it will be impossible to see whether the solution of one problem creates a new one. This documentation is also of important psychological value in intervention situations. What we deal with is primarily the actors, their actions, and their products. Even if 90% of a product is good, an atmosphere of resignation is easily created if discussions only deal with the 10% in which there are (more or less) insoluble problems.

Thesis 3:

Project components should, as far as possible, be described in standard terms.

This is a prerequisite to be able to apply normative theoretical statements.

6. The Difficult Choice – Setting the Problem

Practice has proved it difficult to determine whether a project situation has been sufficiently investigated. This may manifest itself in the fact that the cause-problem-consequence relation in a diagnostic map can be parallelly displaced so that causes turn into problems, and problems into consequences. The real difficulty is to identify the central problem. That is what problem setting is all about.

Which criteria can we use when we select the node in the network we see as the problem?

⁵ "In studying the particularities of the contradictions at each stage in the process of development of a thing, we must not only observe them in their interconnections or their totality, we must also examine the two aspects of each contradiction." (Mao 37).

Lanzara and Mathiassen speak generally about the common acknowledgement (collective, public), and the interpretative value of maps – especially as guidelines for action. They mention the desirability of being able to identify typical problems.

They mention other, more operational, criteria:

- the problem should entail serious consequences,
- the problem should be observable; there should be solid evidence behind it.

All these criteria are relevant. Some of them are, however, heavily related to the specific intervention situation. They are relevant to the function of communication. They may, however, disturb the function of evaluation. The common acknowledgement of a problem formulation, for instance, is decisive for the communication, but lack of direct common acknowledgement of a problem formulation need not constitute a poor starting point for further analyses.

In the analytical function there is a need for operational criteria for selecting candidates for the central problem. When a good problem formulation has been chosen on the basis of such criteria, you must stick to it – at least for a while. It cannot be rejected off hand, even if there are no obvious solutions. Operational criteria saves you from drawing up maps which center around each and every possible problem to determine its interpretative value.

Here we will propose five relatively operational criteria for analytically selecting the central problem in a problem-setting situation.

Thesis 4:

A candidate to the central problem

- 4.a. comes very close to fulfilling the formulation requirements stated in theses 1 to 3,
- 4.b. is the cause of many other problems in the network,
- 4.c. points towards possible actions (rather curative than preventive),
- 4.d. is relatively well-documented by evidence,
- 4.e. is a typical problem or an atypical problem. I.e. it is relatively easy to decide whether the problem is typical or not.

The first criterion was substantiated in the previous section. Criteria b, c, and d are self-evident.

The word "typical" in criterion e relates to the experience of the participants or to documented problems (Munk-Madsen 84). If the problem is typical (and most problems are – that follows from the definition of "typical"), a number of possible solutions are known. If the problem is atypical, and fulfils the other criteria, there is a good possibility that the reason which explains why the unexpected situation has developed is close at hand. If it, on the other hand, is difficult to decide whether the problem is typical or not (in some ways it resembles a known problem, in others

it differs from it), there is good reason to investigate the problem more closely before going into solving it.

It is not easy to prove the usefulness of guidelines like these through anything but practice, so let us have a look at what they imply in the examples in figures 5.1. and 5.2.⁶

First we can note that the description of the situation in the two maps are very rich. A total of 41 phenomena, relations, or alternatives are mentioned. There is, however, no mapping of the project participants' qualifications. The reason for this may be an axiom that there is "no substantial differences in cognitive skills" (Lanzara and Mathiassen 85, pp. 38-39). Or the reason may be that the subject was too emotional to be discussed. However, important information is missing when alternatives are to be evaluated.

The problem definition is formulated as negative qualities and missing functionality of an intermediate product. This product is not named. As a matter of fact, it was the overall technical and functional design, as well as a status report and a revised project plan. The next phase in the project included detailed design and programming.

The description of the intermediate product only covers negative characteristics, but this is not important until it is time to evaluate the alternatives, and the intervention did not get that far.

We deal with a potentially good problem. It relates to the normative statement that the quality of the overall design is very important. There is solid evidence of the phenomenon. The problem is a typical one. It has a high degree of explanatory value. Let us leave the criterion of action potential for a while, and turn to the fact that the problem is not formulated explicitly as a contradiction.

The problem ought to be formulated as follows:

- a. The overall design has serious defects
- +
- b. The project group plans to continue on the basis of this overall design
- +
- c. Development management expects the project group to continue, as management has approved design as well as plan.

Point b is mentioned in the consequences - issue 11 in figure 5.1. Point c is mentioned in the system developers' values and norms - issue 16 in figure 5.2.

⁶ The examples do not represent practice directly. In the actual intervention situation these maps were not made (this is the message in (Lanzara and Mathiassen 85)), they were drawn up afterwards. However, the maps represent the actual level of interpretation reached by the MARS-project in its first phase.

In this formulation points b and c are in harmony, while points a and b together negate the theoretical statement: that the quality of the overall design is important in systems development.

You can imagine a changed situation including a new point b^l:

b^l: The project group is going to change the overall design

Now points a and b^l are in harmony, but a conflict between expectations to the project has emerged. Hence a solution of the problem must also consider the expectations of the steering committee.

This aspect of the problem never becomes an item on the agenda. Therefore the system developers must reject b^l as being unrealistic. And this may explain why all the formulated alternatives are preventive and not curative. They are of value to the development organization, but for the project group they express misplaced hindsight. Seven proposals for what "should be done", or (even worse) what "should have been done" are mentioned, but nothing is said about "what can be done" in the situation in which the project group actually is.

If the problem was to be solved, an ecological map illustrating the group's possibilities for action concerning the relation to management would have been necessary: Can the project leader tell management that they were wrong when they approved the intermediate product? Can the project group secretly repair the overall design, while pretending to make detailed design? This domain of action is not described in the ecological map in figure 5.2., and the map is not very helpful in the actual intervention situation. It relates to the possibilities for action in the development organization.

This example obviously does not "prove" the value of thesis 4. But it illustrates how systematic application of the criteria for problem formulation and problem setting can bring the analysis a step further.

7. The Relation between Project Graphs and Problem Networks

Section 3 claimed that problem networks are not the same as project graphs. We will now try to explain the relation between these two description tools. Let us start by recapitulating the definitions.

A project graph consists of statements concerning project components and relations between these components, expressing concepts like "enters into" or "a prerequisite of". Project graphs may either express history or expectations (plans).

A problem network consists of statements concerning problems and relations between these problems expressing causal relations. Thesis 1 claimed that a problem should be formulated as a relation between project components, or between diverging expectations, or between expectations and reality.

If we, on the basis of these definitions, try to draw the problem network in figure 3.4. into the project graphs in figure 3.3., we would end up with figure 7.1.

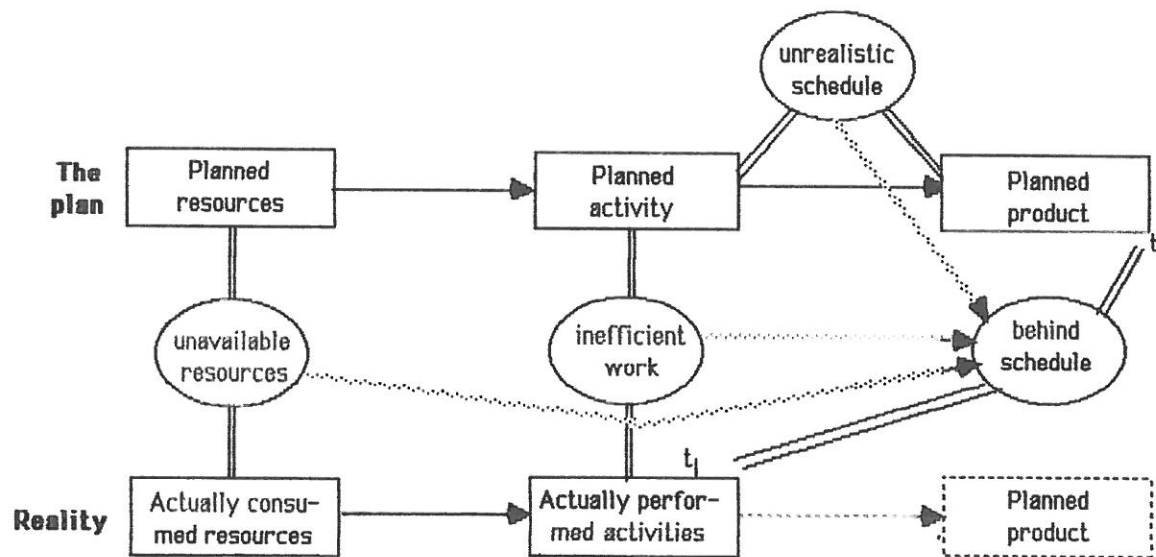


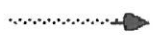
Figure 7.1. Project graph with added problem network



represents a problem



is the problem's relation to project components in the plan and/or in reality, and



represents causal relations between the problems.

This drawing exercise leads us to the following idea:

Thesis 5:

A problem network is – in a not strictly mathematical sense – the dual network to the existing project graphs. The graphs which express the history and the expectations of the involved people, respectively.

The lacking mathematical rigour must be stressed. It is doubtful whether all the expectations to, and relations in a project can be represented in a project graph. It is also doubtful whether all problems and their causal relations can be represented in a problem network. Thesis 5 merely expresses a way of looking at descriptions of projects which may be useful in some situations.

8. Exorcising the Evil of Excuses

The explicit formulations of problems in terms of contradictions have the advantage that they point more clearly towards various domains of action. Another advantage is that they expose excuses.

When a project group is in a tight spot it often develops a defensive attitude to the problems. Instead of solving them, the project group accumulates excuses for why they cannot act differently. The project group in the example has an excuse: management has approved the intermediate product. Consequently the system developers are in good faith when they continue on the basis of this product. The excuse only partly exempts the project group of responsibility. After all, it is the project group who made the intermediate product.

To make an excuse really good, responsibility must be placed completely outside the project group. An example of an excuse like that would be the formulation of a problem as "an agreed-upon resource was not available". A number of negative consequences can be explained by this phenomenon. And, most importantly, the reason is placed outside the project group's domain of action.

Or is it? If we formulate the problem a little more precisely: "An agreed-upon resource was not available as expected", we can see that we deal with an antagonism between reality and the project group's expectations. One aspect of the antagonism is easily documented, but investigation into the other aspect is often neglected – exactly because it is not formulated explicitly. The questions: "Could the project group at some time or other have anticipated that the negative event would occur? And could the project group have done something to mitigate the consequences?" are not asked.

A precise formulation of the problem in terms of an antagonism makes it harder to maintain a defensive excuse-accumulating attitude, and enlarges the domain of action.

9. Project Planning Should Make Theories Explicit

Project evaluation – and especially problem setting – becomes easier if the project plan fulfils certain requirements:

Thesis 6:

Expectations to the properties of project components should be extensively formulated in the project plan.

This will make it easier to determine whether the course of the project agrees with the expectations in the plan. And at the same time it becomes harder for the various groups involved in the system development project to maintain different expectations to the project.

Thesis 7:

Project components should be denoted by standard names.

This is a prerequisite for using theory in project planning and project evaluation.

Thesis 8:

The wellformedness of the project graphs should be motivated in normative theoretical statements.

The explicit formulation of the expected balance in the project makes it easier to disclose imbalances which have arisen during the course of the project.

The effect of better planning is naturally not only that it becomes easier to evaluate the project, but also that the project group ends up in fewer problem setting situations.

To make the recommendations of thesis 8 applicable, the system developers must have access to normative theoretical statements. Most system development methods do not say anything general about the wellformedness of project graphs. They only describe fragments of a generic project graph, from which a restricted class of project plans can be derived. The methods do not say whether alternative plans would be better or worse. The methods do not make the underlying theory explicit.

10. Cognitive Mapping and Problem Setting

Cognitive mapping is a sociological method for describing and analyzing a person's assertions about his beliefs with respect to limited domain (Axelrod 76). A cognitive map is a graph. The points represent statements about domain components⁷. The arrows represent causal relations between points. Axelrod describes a number of techniques for deriving and analyzing cognitive maps.

Cognitive mapping and the ideas of problem setting in this paper have the graphical representation of causal relations within a domain in common. There are, however, a number of differences between the two approaches. These differences will be discussed here.

First of all there is a difference in scope and purpose. Problem setting describes a project reality (including the participant's perceptions) in order to identify and solve problems. Cognitive mapping describes persons' perceptions for the sake of description or in order to improve decision making – e.g. through simulation.

From this difference follows a difference in guidelines. Where thesis 3 recommends the use of standard terms to enable application of theory in the analysis, Axelrod recommends that concepts should not be specified in advance in order to obtain authentic maps.

⁷Axelrod talks about concept variables.

Another major difference is only partly explained by the difference in scope. Whereas the problem setting approach allows for alternating perspectives on the totality and the details of a situation, the cognitive mapping stays at the detailed level. This is illustrated by Axelrod's concern about cycles (and especially the absence of negative cycles) in the cognitive maps. The concept of negative cycles is captured by thesis 1 which states that a network node should be formulated as a contradiction.

The emphasis on the empirically obtained details explains why cognitive mapping techniques to a great extent deal with ordering maps statistically and quantitatively. Conversely problem setting is concerned with qualitative conceptualisation of situations.

Finally, cognitive mapping only deals with one type of graphs, where the analysis of network semantics leading to thesis 5 explains the necessity of two types of graphs. The nodes of cognitive maps are domain component properties. This also holds for the nodes of project graphs - and for the nodes of poor problem networks. Thesis 1 states that a network node should be formulated as a contradiction involving several project components or expectations to project components.

11. Conclusion

This paper has drawn up some new concepts and ideas for evaluating and managing system development projects: The formulation of problems as contradictions, the duality between problem networks and project graphs, the use of standardized names for project components, and the use of normative theoretical statements. Further theoretical and experimental work will verify and validate these ideas.

Most of the paper has discussed project descriptions. Plenty of those have been made in the course of time. All method makers have described a class of possible projects. All project leaders have described a concrete project in their own plans and status reports. However, the interesting point is that new issues arise when you start in the most difficult description situation - namely the practical problem setting situation.

References

- R. Axelrod (1976), Structure of Decision, Princeton University Press.
- B.W. Boehm (1981), Software Engineering Economics, Prentice-Hall.
- T. DeMarco (1982), Controlling Software Projects, Yourdon Press.
- R. Dunn and R. Ullman (1982), Quality Assurance for Computer Software, McGraw-Hill.

- M. Jackson (1983), System Development, Prentice-Hall.
- G.F. Lanzara (1983), The Design Process: Frames, Metaphors, and Games, in U. Briefs et al. (eds.) Systems Design for, with, and by the Users, North Holland.
- G.F. Lanzara and L. Mathiassen (1985), Mapping Situations within a System Development Project, in Information Management, Vol. 8 (1985), # 1, pp. 3-20, North Holland.
- Mao Z. (1937), On Contradictions, in "Five Essays on Philosophy", Foreign Languages Press, 1977.
- MARS (1984a), Systemudvikling i praksis: Jydsk Telefon-Aktieselskab (Systems Development in Practice: The Jutland Telephone Company), MARS-report No.2, University of Aarhus.
- MARS (1984b), Systemudvikling i praksis: Regnecentralen af 1979, (Systems Development in Practice: RC-Computer), MARS-report No.3, University of Aarhus.
- MARS (1984c), Systemudvikling i praksis: Sparekassernes Datacenter, (Systems Development in Practice: The Computing Centre for Danish Savings Banks), MARS-report No. 4, University of Aarhus.
- MARS (1984d), Systemudvikling i praksis: ØK Data, (Systems Development in Practice: EAC Data), MARS-report No. 5, University of Aarhus.
- L. Mathiassen and A. Munk-Madsen (1985), Formalizations in Systems Development, in H. Ehrig et al. (eds.) Formal Methods and Software Development, Springer-Verlag.
- A. Munk-Madsen (1984), Practical Problems of System Development Projects, in M. Sääksjärvi (ed.) Report of the Seventh Scandinavian Research Seminar on Systemeering.
- A. Munk-Madsen (1985), Evaluation of System Development Projects, in Proceedings from the Working Conference on Development and Use of Computer-Based Systems and Tools. University of Aarhus.
- M. Nielsen and D. Bjørner (1985), Meta-Programs and Project Graphs, Dansk Datamatik Center.
- G. Polya (1957), How to Solve It, Doubleday and Company.
- D.A. Schön (1983), The Reflective Practitioner, Basic Books.
- E. Yourdon (1982), Managing the System Life Cycle, Yourdon Press.