Classifying IS Project Problems: An Essay on Meta-Theory

Andreas Munk-Madsen
Dept. of Computer Science, Aalborg University
amm@cs.aau.dk

Abstract. The literature contains many lists of IS project problems, often in the form of risk factors. The problems sometimes appear unordered and overlapping, which reduces their usefulness to practitioners as well as theoreticians. This paper proposes a list of criteria for formulating project problems. This list is grounded theoretically. The list is evaluated through a discussion of some classic lists of risks.

1. Introduction

There is a lack of a common systematic classification of information system project problems, regardless of whether they are described in advance as risks, in the present as crises, or in retrospect as failures. The unsystematic classification entails overlapping concepts and confusing root causes and symptoms when diagnosing problems in the individual projects. Also, the lack of communality impairs the exchange of experience between projects. The lack of common classification also impedes the further development of IS project theory.

In the field of project management there is a generally recognized need for more theory (Turner, 2006). Alter (2000) specifically states: “… a lack of clarity about basic concepts is one of the most fundamental problems in the IS field.”

In this essay we attempt to contribute to the building of a theory of problems in IS projects by establishing a list of meta-theoretical criteria. These criteria support the discussion of what characterizes good problem classifications.
Section 2 outlines a meta-theory of the science of medicine. Based on this outline a set of criteria for classifications of IS project problem is hypothesized in section 3. Also in that section, the relevance of the criteria is discussed theoretically. In section 4 the criteria are used in a critical discussion of selected lists of IS project problems. Section 5 sums up the discussion.

2. Analogy: Medicine

In this section we examine the meta-theory behind a more established science, medicine (Jensen, 1973). We only wish to get inspiration. We do not attempt “proof by analogy”. There are a number of differences between information and medical science. Notably a difference in the type of domain, since the human body – till now – has been a product of nature, whereas IS projects and their resulting artifacts and organizational changes are created by humans. Yet there are also similarities. Especially a common orientation towards theories that can improve professional work, i.e. normative theories such as methods and standards.

Thus when we see differences between the meta-theories of medicine and IS we should ask why? Can the differences be related to differences in the domains of the sciences; or do the differences stem from the relative immaturity of the field of IS project management?

2.1 Causality

There is a strong notion of causality in medicine. A problem may cause several other distinct problems, and a problem can be caused by several other problems that may be independent and may reinforce each other. Thus medical science faces not singular problems but chains of problems interwoven in causal networks.

This is reflected in the distinction between symptom and diagnosis. Doctors are not satisfied with just any problem description. There is a search for a root cause.

In the absence of a root cause symptoms that appear together can be clustered in syndromes.

Causality is also reflected in the theory of action, where the diagnoses are linked to treatments. Again the causal network is involved; several treatments may be applicable to the same diagnosis; and a treatment may have side effects. There is an underlying axiom that it is better to treat a root cause than to treat symptoms.
The causality is paired with uncertainty. The link between symptoms and diagnosis is not always certain, it holds true with some probability. The link between diagnosis and treatment is also subject to uncertainty; the prognosis is not always certain.

Causality is a complex relationship. It may involve time, e.g. an incubation period. And medical causality is often non-linear.

2.2 An Explanatory and Empirically Based Theory

The relations between symptoms, diagnosis, and treatment are founded in explanatory theories. A mere correlation between variables is not sufficient. The “workings” of the physiological mechanisms linking symptoms to diagnosis and diagnosis to treatment must be explained. The quality of the theory, and the acceptance in the profession, is related to empirical evidence. It is preferred that variables involved are measurable.

The theories of medicine are of course subject to scientific progress. Thus any accepted theory represents the knowledge at a certain time in history.

2.3 The Dynamics of Problems

The dimension of time is an important aspect in medical theory. Some diseases go through distinct phases of development with different symptoms. A small, but untreated, disease may develop into a major problem. Singular diseases may over time result in numerous complications.

2.4 Context Sensitive Problems

What is considered to be problem is not always independent of the situation of the patient.

A famous example is the gene for sickle hemoglobin. On the one hand side it is a problem as it causes anemia. On the other hand it is an advantage in some parts of the world as it offers better protection against malaria (Bridges, 2002).

The consequence is that a problem cannot generally be stated as a single fact. It must be formulated as a contradiction between two phenomena.
2.5 The Concept of Abnormality

Simply stating that some phenomenon is a problem implies a desire of change. This desire presumes that change is possible. Thus the concept of problem is linked to its antithesis, the concept of normality.

What is considered normal can be defined either as an *ideal* or as an *average*. In natural sciences normality is mostly associated with average whereas norms often describe ideals in technical sciences. The interesting point is that medical science use of a mix of ideal and average when defining what is normal.

2.6 An Authorized Catalog

Creating a systematic list of diagnoses involves some serious meta-theoretical problems. How should overlapping be avoided? When should a disease be described as a deviation from an average, and when should it be described as a deviation from an ideal? How should the catalog be sorted?

The medical profession cuts right through these meta-theoretical problems by creating a normative classification. This classification is authorized by an international organization and enforced by national governments. (WHO, 2003)

The classification of diagnoses is ordered systematically. The ordering mainly follows the different parts of the human body.

3. Criteria for Formulating Problems

The presentation of elementary meta-theory of medical science in the previous section can be summed up to the list of criteria in figure 1.

1. There should be an attempt to distinguish root-causes from symptoms.

2. There should be an attempt to model that an unsolved problem may be the cause of other problems.

3. There should be an attempt to reduce overlapping between problems.

4. The problems should be related to the phases in the project life-cycle where they occur.

5. The relation between problems and solutions should reflect uncertainty.
6. The problems should be observable and measurable.

7. The problems should be formulated as contradictions.

8. The problems should be founded in explanatory theories.

9. The problems should be related to normative statements. It should be clarified whether the specific norm reflects an ideal or an average.

10. The structure of a problem catalog should be made clear.

Figure 1. Criteria for formulating IS project problems

The relevance of these criteria to the science of information systems can be illustrated with fairly trivial examples. These examples will show that it is not possible to discard any of these criteria in general. This statement does not contradict that in a specific study some of these criteria may be irrelevant.

Goldkuhl (2004) argues for the need for multi-grounding design theories in the field of IS. We consider, that his meta-theory is also relevant for problem handling theories. The ten criteria in figure 1 reflect Goldkuhl’s grounding processes.

A problem classification should be theoretically grounded. An important aspect of this is the conceptual grounding. All ten criteria reflect this demand. In particular criteria 1, 2, 3, 7, 9, and 10 directly address the conceptualization of the individual problems.

Another sub-process within the theoretical grounding is value grounding. This involves explicating the relations to goals. Criterion 9 reflects this demand. The relation between risks and goals plays an important role in classic risk assessment (Feather, 2004).

A third sub-process within theoretical grounding is explanatory grounding. This demand is reflected in criterion 8.

The demand for empirical grounding is reflected in criterion 6.

Finally Goldkuhl talk about internal grounding, a process partially overlapping theoretical grounding. Internal grounding also involves knowledge reconstruction. This demands is reflected in criteria 1, 2, 3, 7, 8, 9, and 10. Finally internal grounding involves evaluation of knowledge cohesion. This demand is reflected in criteria 1, 2, 3, 4, 5, 7, 8, and 9.
Having argued that the ten criteria conform to the epistemology of IS, we shall apply them to discuss some well-known lists of problems in IS projects.

4. Application of the Criteria: Examples

In this section we evaluate the list of criteria in figure 1 through a discussion of some classic lists of risks. Lists of risks are often taken as checklists for identifying and discussing IS project problems. Thus the quality of these lists of risks is an important issue to discuss.

4.1 Alter and Ginzberg

Alter and Ginzberg (1978) presented one of the first lists of risk factors. The list is based on structured interviews with designers and users involved in 56 systems. The problems in the list are shown in figure 2.

1. Nonexistent or unwilling users
2. Multiple users and designers
3. Disappearing users, designers, or maintainers
4. Inability to specify the purpose or usage pattern in advance
5. Lack or loss of support
6. Lack of prior experience with similar systems
7. Inability to predict and cushion the impact on all parties
8. Technical problems and cost-effectiveness issues

Figure 2. Alter and Ginzberg’s risk factors (our numbering)

This list is grounded in an “ideal” project. Each risk factor represents a deviation from the following situation:

The system is to be produced by a single implementer for a single user, who anticipates using the system for a very definite purpose which can be specified in advance with great precision. Including the person who will maintain it, all other parties affected by the system understand and
accept in advance its impact on them. All parties have prior experience with this type of system, the system receives adequate support, and its technical design is feasible and cost effective.

There is an indisputable simplicity in this ideal. It serves the purpose of clearly structuring the catalog of problem, and – indirectly – it also yields some explanations.

The factors are explicitly placed in a seven-stage waterfall inspired project model. The factors are linked to a number of strategies. These strategies are classified as either “compensating” or “inhibiting”, both words imply some uncertainty.

A friendly interpretation would be that this almost thirty years old list of problems satisfies half of the criteria, namely criteria 4, 5, 8, 9, and 10.

Basically the list does not reflect causality. Despite the structuring, there is some overlapping in the list. For instance factor 1 and 6 could lead to factor 4, which might result in factor 7.

Some of the problems would be hard to measure. What is the difference between a “technical problem” and the technical challenges encountered in ordinary IS projects? And does our project have “adequate support” or “lack of support” when we encounter a normal pressure on resources?

All the problems are formulated as negations of the ideal situation, so in a sense they represent contradictions. However, they contradict a single, highly unnatural situation. We are not informed about situations where some of the risk factors are irrelevant.

Alter and Ginzberg’s model has problems fulfilling criteria 1, 2, 3, 6, and 7.

4.2 Boehm

Boehm (1991) presents an often quoted checklist of sources of risk to software projects. This list is shown in figure 3.

1. Personnel shortfalls
2. Unrealistic schedules and budgets
3. Developing the wrong functions and properties
4. Developing the wrong user interface

5. Gold plating

6. Continuing stream of requirement changes

7. Shortfalls in externally furnished components

8. Shortfalls in externally performed tasks

9. Real-time performance shortfalls

10. Straining computer-science capabilities

Figure 3. Boehm’s risk items (our numbering)

The checklist is actually a top 10 hit list “based on a survey of several experienced project managers”. This explains why the list appears unordered and untheoretical.

The list does not model causality. Item 1 and 6 could lead to item 3 and 4, which eventually would entail item 2.

There is no relation to life-cycle phases. There is no indication as to how the items can be measured. Most of the risk items are formulated as a negation of an ideal using words as “shortfalls” and “wrong”. However, the ideal is not described.

The only criterion from figure 1 that is fulfilled is number 5.

It should be noted that Boehm’s paper contains other, more specialized and stronger conceptualized, lists. A key point in the paper is that projects should develop their own, specific risk lists. Still, a better structured list in figure 3 would have served both practitioners and theoreticians better.

5. Discussion

The analysis of the actual lists of problems has been an interesting exercise. Many shortcomings of the lists have appeared. They underline the need for metatheoretical criteria as those in figure 1.
It is also interesting to compare the lists of problems with the advanced state of medical knowledge represented in the classification of diseases. Even though there are important differences between the two sciences, there is much room for improvement in the science of information systems.

In particular the “flatness” of the problem lists, the absence of conceptual depth, is disappointing. The lists do not reflect that the presence of one problem in a project may lead to another that in turn may create a serious crisis in the project. Thus the lists do not assist a project manager in distinguishing between symptoms and root causes. This would be helpful since you could ignore the symptoms if the root cause was treated effectively.

The reason for the “flatness” of the problem lists could be a preponderation of empiricist research approaches in this field. When a researcher asks practitioners what they consider the most important problems or risks the immediate result must be a flat unstructured list.

The need for more elaborate models of problems is recognized in the theory of errors and failures in human and engineered systems. Busby and Hughes (2004) apply some of this theory in project management. Here we find the idea of the incubation period and the idea that sometimes several factors must coincide to cause a problem. The systems are seen as complex. “Complexity typically involves the presence of unforeseen feedback loops that make a system’s behaviour obscure and unpredictable.” Another issue is tightness of coupling that tends to involve absences of slack that makes a project more vulnerable.

Let us assume that projects are complex. They should be since they solve complex tasks (Munk-Madsen, 2005) in a complex environment (Mintzberg, 1983). Then we may ask why non-complex models of project problems should suffice?

6. Conclusion

A meta-theory of IS project problems is a rather complex matter. This is not always reflected in the current theory.

7. Postscript

This paper is still in progress. All sections will be expanded.
References


