A Cognitive map of Maturity

Symptom-based improvement recommendations

Pries-Heje, Jan; Johansen, Jørn; Korsaa, Morten

Published in:
Systems, Software and Services Process Improvement - 27th European Conference, EuroSPI 2020, Proceedings

DOI:
10.1007/978-3-030-56441-4_33

Publication date:
2020

Document Version
Peer reviewed version

Citation for published version (APA):
Pries-Heje, J., Johansen, J., & Korsaa, M. (2020). A Cognitive map of Maturity: Symptom-based improvement recommendations. In M. Yilmaz, P. Clarke, J. Niemann, & R. Messnarz (Eds.), Systems, Software and Services Process Improvement - 27th European Conference, EuroSPI 2020, Proceedings: 27th European Conference, EuroSPI 2020, Düsseldorf, Germany, September 9–11, 2020, Proceedings (pp. 447-461). Springer. https://doi.org/10.1007/978-3-030-56441-4_33

A Cognitive map of Maturity: Symptom-based improvement recommendations

Jan Pries-Heje[1[0000-0003-2219-9332]], Jørn Johansen[2[0000-0003-1368-3640]] and Morten Korsaa[2[0000-0003-4521-454X]]

[1] Dept. of people and Technology, Roskilde University, Denmark
[2] Whitebox, Hørsholm, Denmark
janph@ruc.dk, jj@whitebox.dk, mk@whitebox.dk

Abstract. When an experienced assessor enters a company there are certain characteristic symptoms that reveals the maturity of the company even before the assessment. In this paper we start out from a list of 32 characteristic symptomatic problems generated by two experienced assessors after maturity assessments in more than 300 companies. We then use cognitive mapping asking five times why to get behind the symptomatic problems and reveal the underlying problems or causes of the problems. Our mapping revealed a number of interesting relationships between the symptomatic problems. We then evaluated our findings by building a tool in the form of a website where users could score statements - formulated from the symptoms - and from that we could point them to areas where they probably needed to improve. We improved the tool in three learning cycles of design evaluation and ended up in a summative evaluation where we compared the outcome of using the website tool with a CMMI maturity assessment. We conclude that looking for symptoms, and as an individual scoring statements build on the symptoms, can point to improvement areas. However, doing so is not a replacement for a maturity assessment, the scoring of statements cannot necessarily reveal the maturity of the organization.

Keywords: Cognitive map, Process Improvement, Maturity, Improvement, CMMI.

1 Introduction

Imagine that you are a manager in a product development company or department, in distress over some bad performance. Somewhere. It is your responsibility, but where is the real problem? Where should you direct your attention? What should you do?

Start looking, and you will find an overwhelming amount of processes, problems, causes of problems, symptoms of problems, tools, practices, people, organisational structures. Each of them could potentially be the problem. And they seem to relate to each other in many ways.

The authors of this paper are looking for a way to help you out. From our experience, there is a relatively limited number of symptoms that is of real importance in most organisations. There are also a limited number of relevant relationships between

the symptoms, the experienced problems, and the causes of the problems. In this paper we start our work from a list of 32 symptoms that we have meet when we have been doing assessments in more than 300 different product development organisations. We take each symptom and ask two things; (1) What are the causes of this symptom? – the problem behind so to say; (2) What does it cause or lead to or has as effect?

An example could be that the documentation for a product is not updated. That could be caused by lack of time for making documentation, and that again could be caused by a very tight development schedule or budget for product development. Further, when documentation is not updated it could lead to difficulties in maintaining and further developing the product, and that again could lead to the problem that it becomes excessively costly to maintain and further develop the product.

The example above has five levels of problems; a kind of problem-hierarchy that is. An old Japanese improvement technique called "5 times Why" operates exactly with five levels. The point being that you should never go with the first symptom of a problem but instead look for the root cause(s). The Japanese car company Toyota developed the "5 times Why" technique in the 1930s. It became popular in the 1970s, and Toyota still uses it to solve problems today [1].

The research question we aim at answering in this paper is: "How can we use a mapping of the symptomatic problems, causes and effects, and the relationships identified between them, to design a tool that can help determine what the most urgent improvement areas are in your company?"

Our approach for answering that research question is to use the "5 times Why" techniques on the 32 symptoms that we regularly have met in companies.

In this work we have used the CMMI [2] maturity model as our basis for discussions because two of the authors are very experienced in using this model. However, it could have equally relevant to use SPICE or Automotive SPICE [3] and the addressed practice capabilities in these models.

2 Existing research on problems

A problem can be defined as a perceived difference between what is and what ought to be [4]. There can be many aspects of a problem [5]. It can be a consciousness of a gap, a desire, or a need. On the other hand, an aspect can also be that something is undesirable and therefor implies the imperative for change. A third aspect can be that it is difficult as opposed to trivial. A fourth aspect that it is solvable as opposed to impossible to solve. And finally, there can be different perceptions by different stakeholders. One may see it as undesirable not to have updated documentation whereas another stakeholder may see it as no problem at all.

In the literature you find many problem analyses and problem-solving techniques. A well-known method by Peter Checkland [6] [7] [8] is called Soft Systems Methodology (SSM). Checkland had experience as a consultant for big international companies like Shell before coming back to academia as a professor in systems thinking. He distinguishes between hard and soft problems. A hard problem is a well-formed prob-

lem that can be solved with well-known engineering techniques. The problem presents itself so that it is easy to see what type of problem it is. A soft problem on the other hand may have many aspects, many humans involved, and many different stakeholder perspectives. Thus, it needs work and discussion to understand the problem – if it is a problem!

Checkland has two important points that we will use here. First, it pays off to distinguish between the real world and systems thinking at a meta- level about the real world. Second, it pays off to produce models of purposeful activity in the real situation and use the models as devices to explore the situations and structure a discussion.

Rittel and Webber [9] defined some problems as being 'wicked' in that you cannot solve the problem unless you have some knowledge that you can only get from solving the problem; the problem cannot be understood until after the formulation of a solution. Hence, the only viable strategy is to start solving the problem and learning in the process. Further, wicked problems can be considered to be a symptom of another problem. They are linked together.

Many years later Snowden and Boone [10] presented a framework for sense making to be used by leaders' for decision-making where they look at the relationship between cause and effect. If the relationship between cause and effect is known, it is a 'simple problem' that we can use best practice for. If the relationship is potentially knowable – through hard work by experts – then it is a complicated problem. And if the problem is wicked and thus the relationship between cause and effect only retrospectively coherent then it is a complex problem.

Looking at the relationships between problems and the causal relationships Colin Eden [11] came up with cognitive maps of problems or constructs as he calls them where the link between two problem constructs is in the form of an arrow to show the nature of the linkage; "an arrow out of a construct shows a consequence and an arrow into a construct an explanation" [11, p. 5]. Some years later Eden and Ackerman [12] developed cognitive or causal maps into a techniques [13] that could be used for making strategy. Finally, Venable [14] refined cognitive maps into coloured cognitive maps that can be used for creating a design of a solution. His idea was that each problem should be formulated with its opposing node. E.g. 'high employee turnover' has the opposing node 'low employee turnover'. And when you switch around a whole cognitive map – from the original nodes to the opposing nodes - you will end up with a potential design solution.

3 Research Method

To answer our research question, what are the most urgent improvement areas in your company?, we decided to apply DSR - Design Science Research [15]. DSR is a research approach where you build something and then learn from it (when evaluating). Thus, in order to answer our research question, we decided to build cognitive maps showing the linkages and relationships between problems and symptoms, causes and effects, to better understand what the most urgent improvement areas in a company

may be. And we decided not to do it for a specific company but instead do it at a systems-oriented meta-level.

A main reason for choosing DSR as our research methodology is that it combines the need for practical relevance and utility. DSR emphasizes that a design should address a need or a problem and at the same time should "stand on the shoulders" of existing research within the problem area [15]. Besides having a 'relevance iteration cycle' where you start by identifying a need or a problem you also have a 'rigor iteration cycle' where you identify all relevant academic literature; what do we actually know by now? The artefact that you are building in order to learn can be a product artefact or a process [16].

Hence, we developed causal cognitive maps. We started out from symptoms that can be seen in companies and asked what can this cause? (= Consequences), and what is causing this? As said above we decided to use the "5 times Why" techniques so we developed each symptom in five levels typically by starting with the symptom in the middle of the map and then eliciting two levels of consequences and two levels of causes.

What data should we use for eliciting the maps? Here we took advantage of the more than 600 assessments in more than 300 companies that the group of authors together have carried out. That has given us extensive knowledge of how things are related. So, we simply used the cognitive mapping techniques to make explicit what was in our cognitive mind. At first, we split the symptoms in three groups and mapped a group each. Then to avoid bias and give some inter-coder reliability we swapped the maps around among us until all three authors agreed of the linkages and relationships. We decided only to represent the most important linkage(s), one, two or three. So, a symptom can be caused by many things, but we decided to prioritize the causes and only represent the most important ones. In doing so we are following the principle of organizational learning from the SPI Manifesto [17].

This paper primarily supports the following principles in the SPI Manifesto: Create a learning organization; Support the organization's vision and business objectives; Use dynamic and adaptable models as needed – in the sense to bring insight for the principles.

4 Symptoms observed

In this paper we start out from a list of 32 characteristic symptoms generated by two of the authors of this paper being experienced assessors after maturity assessments in more than 300 companies. They were acknowledged in the following way.

Over the last 25 years as many as 600 assessments are performed in over 300 companies. During this work, the assessors got more and more trained to collect signals related to how "clever" the company are to develop new products or deliver projects for customers. It came so far, they the assessors started to guess on the maturity after looking at some of the main documents, development model and be welcomed in the reception.

Last year the assessors started to identify the most common symptoms. After several brainstorms and discussions, it ended in 32 symptoms, which was formulated as statements of the symptoms. Examples of symptoms were:

- We cannot tell how much effort an individual has used on an activity (#1)
- We do not know who and how many that have a say in the project and the results of the project (#8)
- Unfortunately, we do not find defects until the product is in operation with the customer or end-user (#13)
- We are often correcting the same mistake again and again (#19)
- Employees experience "bad" deliveries from colleagues (#28)

The 32 symptoms were grouped into 5 natural categories, based on the type of the symptoms. The five examples above are one from each of the following five categories:

1. To be in control of the projects across the organization
2. Knowing what do develop and deliver?
3. Projects having the adequate competences to run projects
4. Having project insight and status
5. Quality in work and work products

Having these symptoms defined, we had to qualify them and find a way to make this operational.

5 Cognitive Maps

To obtain a better understanding of the symptoms we decided to apply cognitive maps. We started out from a symptom. Then we asked, what is causing this? And then we asked again; what is the cause of the cause?, thereby identifying the underlying problem. Further we asked, what is the effect or result of the symptom? And then; what is the effect of the effect? For some of the problems we could probably have continued further back to an even more underlying problem or further forward to an effect of an effect of an effect. However, we had decided to apply the "5 times Why" heuristic so we ended our mapping of each symptom with a map that had five layers.

We did this mapping for all 32 symptoms. Furthermore, we circulated the maps among the three authors thereby neutralising any bias that any one of us may have had. We also had some discussions about certain problems; is this a cause or an effect? That was not always easy to answer.

An interesting observation that we made while going through the symptoms one by one was that some (causal) problems or effects started to reappear. We noted that and discuss it later in this paper.

In Figure 1 you find an example where we have mapped symptom #6: "We have difficulties correcting defects in something delivered".

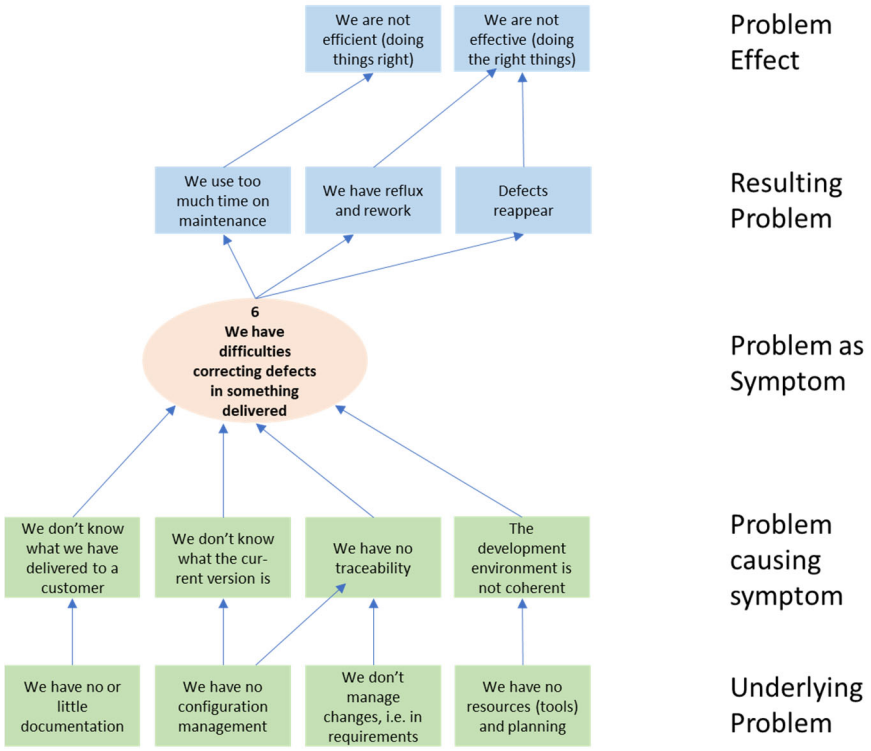

Fig. 1. An example of a cognitive map – here of symptom #6. As you can see the map has five levels of problems corresponding to the "5 times Why" heuristic that we have used.

One "causality track" that we find in Figure 1 is the following: We have no configuration management => We don't know what the current version is => We have difficulties correcting defects in something delivered => We use too much time on maintenance => We are not efficient (doing things right)

Another causality track found in Figure 1 is: We don't manage changes, i.e. in requirements => We have no traceability => We have difficulties correcting defects in something delivered => We have reflux and rework => We are not effective (doing the right things)

When we were eliciting and creating the cognitive maps, we also found that some symptoms are related to other symptoms. That was easily seen when two symptoms resulted in (or caused) the same effect or when two symptoms were caused by the same underlying problem(s). In figure 2 we have shown the same symptom #6 as we presented in Figure 1 – But now with two closely related symptoms represented as well.

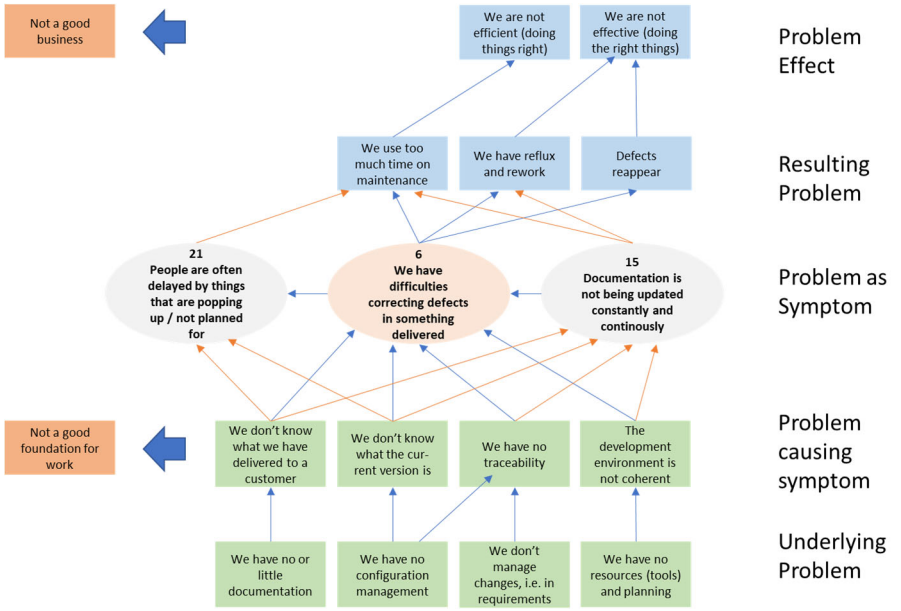

Fig. 2. A cognitive map of symptom #6 and the relationships to symptom #15 and symptom #21. Furthermore, two business-oriented effects of the whole is shown to the left

Thus, Figure 2 shows that symptom #15 ”Documentation is not being updated constantly and continuously” and symptom #21 ”People are often delayed by things that are popping up / not planned for” are closely related to symptom #6.

Another thing we identified was that there were problems at different levels. We have mapped things at the most concrete level – the problem instantiations. However, there is also an effect of the whole map or network of problems. For example, the two effects – not doing the right things and not doing things right – together will cause the business as a whole to be bad. And the four problems that together are causing the three symptoms in Figure 2 will altogether form a not so good foundation for work. In Figure 2 we have mapped these effects related to the ‘whole’ map in orange to the left of the cognitive map.

Let us take another example. Let us look at symptom #15. See Figure 3.

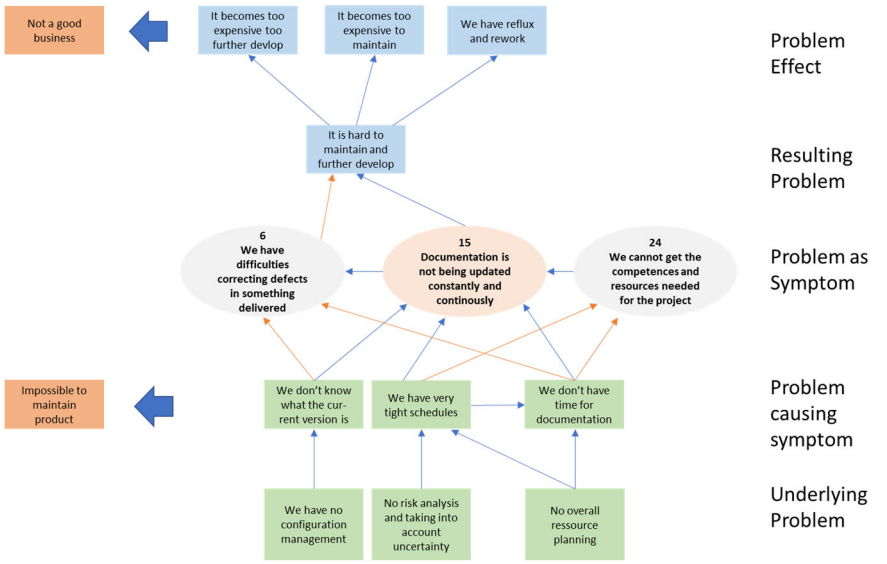

Fig. 3. Example of a cognitive map – here of symptom #15. And how it is related to symptom #6 and symptom #24.

For the 15th symptom; "Documentation is not being updated constantly and continuously", in Figure 3, we find again that the symptom leads to rework - but starting from different underlying problems. And again looking at the meta-level or the 'whole' we can conclude that the overall effect is that it will lead to bad business and that the level causing the symptom(s) as a whole will make it nearly impossible to maintain the product.

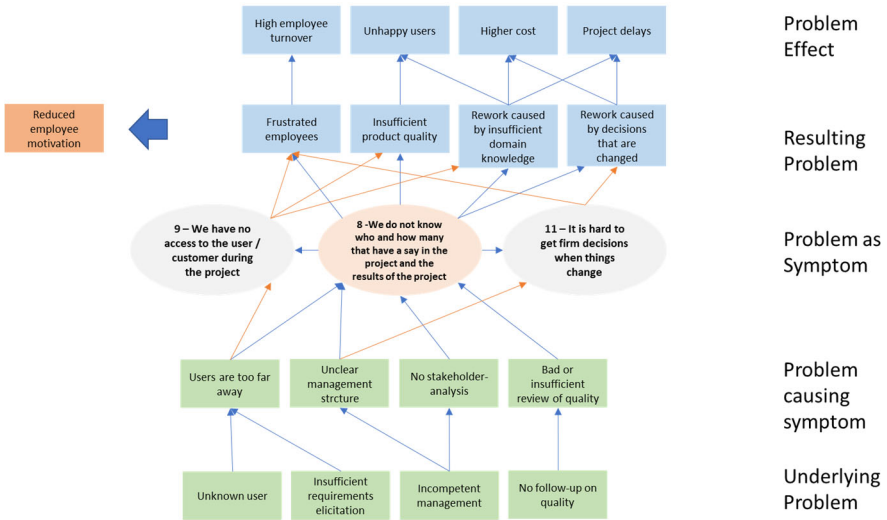

Fig. 4. An example of a cognitive map – here: Symptom #8

Another interesting observation in Figure 3 is that a problem can cause another problem directly or through another – more indirectly so to say. An example "We have very tight schedules". That can lead directly to the symptom #15 but it can also lead to "We don't have time for documentation" that then again can lead to symptom #15 "Documentation is not being updated constantly and continuously".

In the next example shown in Figure 4 it is even more obvious that the higher-level problems can be the same as some of the lower-level underlying problems.

Furthermore, for symptom #8 there is again a relationship to two other symptoms, namely #9 "We have no access to the user / customer during the project", and to #11 "It is hard to get firm decisions when things change". Finally, the 'whole' of Figure 4 will lead to reduced employee motivation

Let us now take a closer look at Symptom #9. See Figure 5

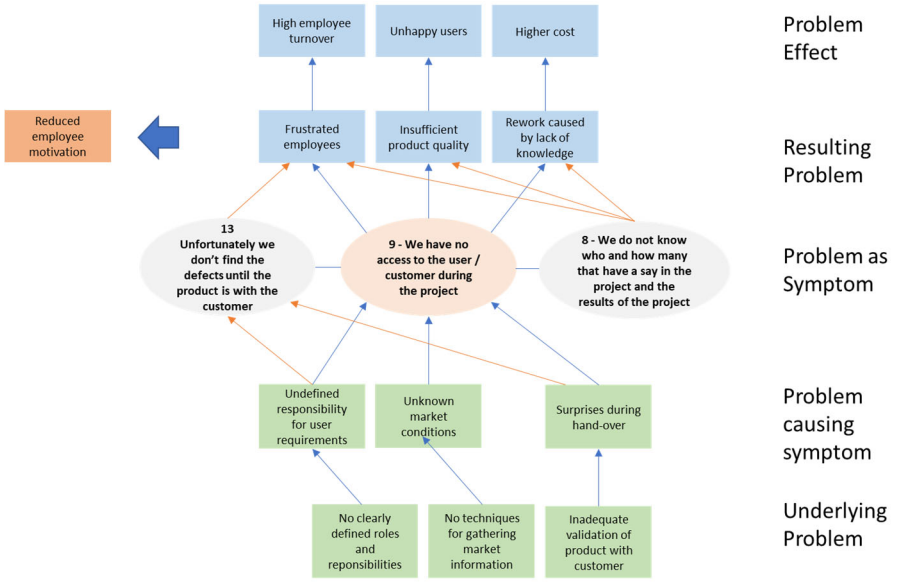

Fig. 5. An example of a cognitive map – here: Symptom #9

When one compares Figure 4 and Figure 5, we find that the layers above the symptoms are nearly the same for Symptom #8 and Symptom #9. Whereas the layers below are quite different for the two symptoms.

So to conclude this mapping section of the paper we have found that it was very useful to use cognitive maps to obtain an overview of the symptoms and the relationships either directly or through other symptoms.

6 Evaluation

An important part of using DSR as the research method is that you have to evaluate the design [18]. Typically, you start by having one or more formative evaluations

where the result is used to 'form' the artefact. And you at the end are having a summative evaluation that 'sums up' or conclude your research. In the beginning the first formative evaluations may be in artificial settings but later you aim at having the real users in the real context with the real problem – real, real, real – a so-called naturalistic evaluation [18].

We have used a framework for evaluation called FEDS by John Venable et al. [18] to plan and carry out our evaluation. For our first formative evaluation we developed a tool that we made available on a website. The tool presents a person in an organization for each of the 32 symptoms to "score" it, seen for the persons perspective.

Nr	**Assertions / Statement:**	Totally agree	Partly agree	Neither nor	Partly disagree	Totally disagree
	In control					
9	We have no access to the user / customer during the project					

Fig. 5. An example of "scoring" a symptom – here of symptom #9

We asked 9 persons from different companies to "score" all 32 symptoms and give their opinion about how easy it was to understand the symptom statements.

After having evaluated the website with 9 users we analysed the outcome and changed some of our statements and the build-in relationship between problems. The tool was updated, so a mail with the recommendations automatically was returned, when the "scoring" was finished. The website was then again tested also with 9 users. This time the focus was to analyse the algorithm in the tool to strengthen the recommendations. And again, this led to changes in and some small changes in the formulations of the statements.

Finally as our summative evaluation we decided to evaluate the web-site tool up against a classic CMMI [2] assessment of maturity. The two assessors fill out the questionnaire separately after a performed CMMI assessment to check if they had the same answer on the questions.

Same score	48%
1 next to (25% disagree)	37%
2 next to (50% disagree)	16%
3 next to (75% disagree)	0%
4 next to (100% disagree)	0%

Table 1: Differences in the score of one assessed company

During the exercise and following discussions and analysis, it became clear, that it is not possible to derive a maturity level for an organisation from the symptoms alone – but only pinpoint the main weaknesses important to address.

We started to identify which overall problem related capabilities the symptoms addressed in an organization showing the ability to run projects successfully and ended up with ten important themes: (1) Project management; (2) Control across the organization; (3) Proactive management; (4) Clear goals for the projects; (5) Capable development organisation; (6) Quality of results and products; (7) Control of work products and product parts; (8) Management insight and involvement; (9) Tools and support;

(10) Continuous improvement and learning. The algorithm was adjusted to calculate a score for the 10 themes as basis for recommendations: What is most important to focus on and to improve.

There is obvious links between the ten defined main themes related to the overall ability to successfully run projects and the processes. This link can be used to pinpoint the recommendations, but not the maturity level. The reason is, that the themes each are based on several processes. But it seems to be clear which themes are aggregated from which symptoms. This exercise was used to strengthen the questions.

Figure 6 shows the scoring of the themes presented in one of the company trials. In the left column you can see that "Control of work products and product parts" and "Project management" are scoring highest meaning that the example company have good control these two themes. Likewise, "Continuous improvement and learning" and "Management insight and involvement" are the two lowest scoring themes meaning that this is where the example company has to focus and improve.

Rank	Score	Ability to run projects successfully	No.
2	43%	Project management	1
8	22%	Control across the organization	2
4	31%	Proactive management	3
6	28%	Clear goals for the projects	4
5	30%	Capable development organisation	5
3	33%	Quality of results and products	6
1	47%	Control of work products and product parts	7
10	14%	Management insight and involvement	8
7	25%	Tools and support	9
9	21%	Continuous improvement and learning	10

Fig. 6. Example of a result scoring the symptoms for an organization

We also started to discuss a mapping between the lowest levels of problems and the specific practices in the CMMI model. We will continue this work, because we believe we can find connection between the 10 themes and specific practices at the processes in CMMI. We also believe this work will strengthen the symptom-based model, e.g. if we find some practices in CMMI not addressed at the lowest level of problems – then an important symptom may be missing.

7 Findings and discussion

We have two set of findings to report. First, we validate the use of cognitive maps.

Based on the links from symptoms to problems below and above we validated, that some symptoms where related – because the symptoms linked to the same problems/causes. It proved to us that the development of cognitive maps gave an insight in how symptoms are related seen from a problem/cause point of view. We even identi-

fied clusters of problems, which made good sense – and gained the understanding of the symptoms.

E.g. looking at symptom #6 in Fig 1, it is strongly related to symptom #15, as both symptoms share many problems and causes. Symptom #21 is seen to be slightly weaker coupled. As a sanity check we formulated the relationship between problems, causes and symptoms as:

"If you do not have a good foundation for work (the problem level just below the symptoms) then it is very difficult to keep the documentation up to date. And if you do not have an updated documentation, it is difficult to correct something that has been delivered. If maintenance is difficult, it typically generates ad hoc rework, which disturb people."

If we take another example. In Fig 3, symptom #8 strongly links to both symptom #9 and #11at the upper problem level. Sanity check: Does it make sense to explain the relationship?

"*If we do not know "who decides what" in the project, it is difficult to make clear decisions and will typically also include a lack of access to users or during the project (since we do not know who decides). All three symptoms lead to lack of motivation, initiated by frustration, failure to fulfill quality and a lot of rework. "*

We find the cognitive maps helps a lot to clarify and structure explanations of the main reasons for main problems. They also help to qualify and strengthen the symptoms, as well as the model. We believe, that over time the working with the model will identify "weak" symptoms, which then will be updated.

The second finding came out of our summative evaluation up against an organisation where two of the authors had performed a CMMI assessment. Here it was clear, that it is not possible from symptoms to derive a maturity level for an organisation – but only pinpoint the main weaknesses important to address. But we find there is a link between the 10 themes and the processes in CMMI. We will continue that research.

As a whole our approach of looking at symptoms are somewhat similar to the SPINACH method [19] developed by Information Promotion Agency in Japan. It has a checklist with about 150 potential symptoms that could trigger further analysis to produce affinity diagram of causal system.

We also tried to 'switch over' a map to the opposite following Venable [14]. In figure 7 we have switched around the map that was shown in figure 5.

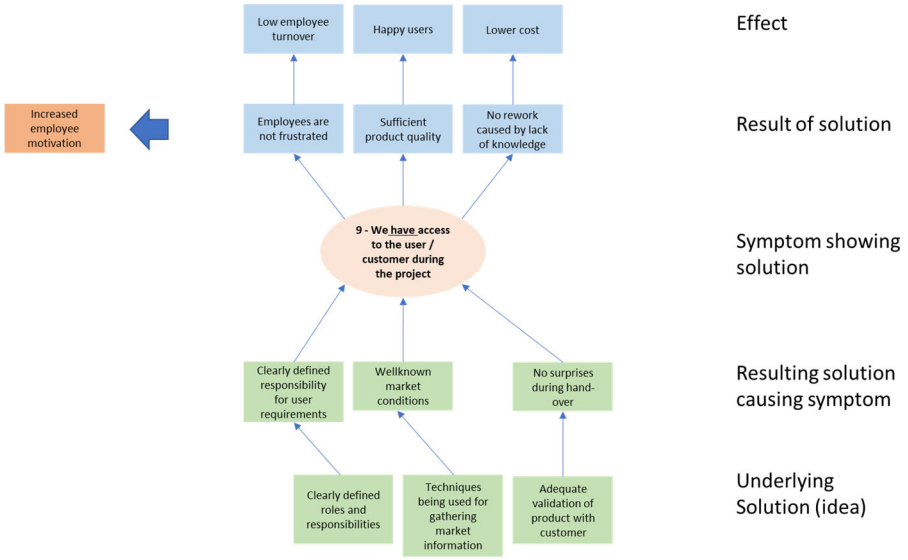

Fig. 7. An example of a design solution map – here: Symptom #9 from Fig. 5 has been switched around

In figure 7 we can then see that instead of having root causes at the bottom we have potential core solutions such as 'clearly defined roles and responsibilities', 'techniques being used for gathering market information' and 'adequate validation of product with customer'. Interestingly enough these core design solutions are very similar to practices in CMMI [2] like respectively Project Planning Specific Practice 2.4 Plan the Projects Resources, Requirement Development Specific Practice 1.2 Elicit Needs and Validation Specific Practice 2.1 Perform Validation. So, there is a clear connection to CMMI at practice level.

8 Conclusion

We have answered our research question; how can we use a mapping of the symptomatic problems, causes and effects, and the relationships identified between them, to design a tool that can help determine what the most urgent improvement areas are in your company?

We have presented a sub-set of the 32 cognitive maps we have created, one for each of the symptoms we started out from. We have shown how both causes, underlying problems and effects can be related and go across the symptoms. We have also found that a group of problems or effects seen as a 'whole' can lead to bad business, reduced employee motivation or other major meta-level effects for a company.

To evaluate our 32 cognitive maps, we have built a tool where people can "score" all 32 symptoms and get overall recommendations. We will ask the users about their experiences with the tool. We will use these data to strengthen the model and the recommendations.

Based on our research in this paper we have realized at least two interesting topics to further research.

The first thing is to use a tool, which enable the possibility to combine all the cognitive maps to one map, with all symptoms and different types of problems and causes. We believe this map bring new knowledge on how the symptoms are related and how problems and causes are related. It will give new possibilities to improve the model and strengthen the symptoms foundation in problems and causes. We will continue this work and write a new paper addressing the findings.

The second thing it to continue the investigation of how the 10 themes are related to the processes in CMMI. We can see that the lowest level of problems in the cognitive maps for each symptom has a relationship with the specific practices in CMMI processes. For example, in Fig. 3 the underlying problem "No overall resource planning" is the caused by not performing the process Project Planning Specific Practice 2.4 Plan the project's resources. We will map these relationships and see how the themes and symptoms are related to CMMI and use that knowledge to see if some CMMI practices are missed, which could indicate a missed problem, cause og symptom. With these relationships in place, we can guide a distressed manager towards the specific practices where improvements are highly likely to be the most beneficial.

References

1. Liker, J., *The toyota way*. 2004: Esensi.
2. Team, C.P., *CMMI for Development, Version 1.3, Pittsburgh, 2010.* Software Engineering Institute, Carnegie Melon University, 2018. **2**.
3. Hoermann, K., Hoermann, K., Mueller, M., Dittmann, L., & Zimmer, J. *Automotive spice in practice.* Rocky Nook, 2008: p. 75.
4. Kroenke, D., *Using MIS 2013*. 2013: Pearson Education UK.
5. Dumdum Jr, U.R.B., *An approach to problem formulation in ill-structured situations in information systems development.* 1993.
6. Checkland, P.B., *Soft systems methodology.* Human systems management, 1989. **8**(4): p. 273-289.
7. Checkland, P. and J. Poulter, *Learning for action: a short definitive account of soft systems methodology and its use for practitioner, teachers, and students*. Vol. 26. 2006: Wiley Chichester.
8. Checkland, P., *Soft systems methodology: a thirty year retrospective.* Systems research and behavioral science, 2000. **17**(S1): p. S11-S58.
9. Rittel, H.W. and M.M. Webber, *2.3 planning problems are wicked.* Polity, 1973. **4**(155): p. e169.
10. Snowden, D.J. and M.E. Boone, *A leader's framework for decision making.* Harvard business review, 2007. **85**(11): p. 68.
11. Eden, C., *Cognitive mapping.* European Journal of Operational Research, 1988. **36**(1): p. 1-13.
12. Eden, C. and F. Ackerman, *Making Strategy. The Journey of Strategic Management*. 1998, Londres: Sage Publications.

13. Ackerman, F., C. Eden, and S. Cropper, *Getting started with cognitive mapping: tutorial notes.* Strathclyde: Strategic Decision Support Research Unit, Strathclyde University. Retrieved November, 1992. **18**: p. 2008.
14. Venable, J.R. *Using Coloured Cognitive Mapping (CCM) for design science research*. in *International Conference on Design Science Research in Information Systems*. 2014. Springer.
15. Hevner, A.R., *A three cycle view of design science research.* Scandinavian journal of information systems, 2007. **19**(2): p. 4.
16. Walls, J.G., G.R. Widmeyer, and O.A. El Sawy, *Building an information system design theory for vigilant EIS.* Information systems research, 1992. **3**(1): p. 36-59.
17. Pries-Heje, J. and J. Johansen, *Spi manifesto.* European system & software process improvement and innovation, 2010.
18. Venable, J., J. Pries-Heje, and R. Baskerville, *FEDS: a framework for evaluation in design science research.* European journal of information systems, 2016. **25**(1): p. 77-89.
19. Norimatsu, S., T. Kishi, and N. Wada. *Development of "SPI Strategy Framework" and Its Application*. in *European Conference on Software Process Improvement*. 2018. Springer.