



A multi-dimensional approach to deal with complex project management

Master Thesis Report

Roy Dulam

Faculty of Technology, Policy and Management

A multi-dimensional approach to deal with complex project management

Author: Roy Dulam
Student ID: 1103792
Email: r.v.dulam@student.tudelft.nl
Track: Systems Engineering, Policy Analysis and Management
Date: May 2011

Abstract

Project failure occurs due to several reasons, but complexity and underestimation of complexity are often seen as the cause. Because of the interdisciplinary character of projects, it seems that project management approaches are not sufficient to deal with the complexity of projects. Therefore the interdisciplinary aspect has to be considered and managed appropriately, which brings us to the following research question: “How can project managers combine tools and techniques from project management approaches and discipline oriented approaches, for managing multidimensional complex projects successfully?”

First a number of management approaches are selected and classified, in order to identify the focus and steps of the approaches. Next, complexity is taken into account, where it becomes clear that complexity in construction projects can be assessed by the TOE framework, while on a meta level the complexity elements of the ASC framework can provide insight in the complexity of projects. Expert interviews were conducted to identify the extent to which different management approaches support mapping and managing project complexity elements. The results point out that the Project management body of knowledge scored weak in dealing with actors, support tools and involving the context elements of a project. Finally, some techniques of other management approaches which score better are presented in order to enhance the project management approach for dealing with complex projects.

Keywords

Project Management, Complexity, Management approach, Framework, Interviews, Dimensions

Graduation Committee

Prof.dr.ir. A. Verbraeck, Section of Systems Engineering, Faculty of TPM

Dr. Ir. G. Kolschoten, Section of Systems Engineering, Faculty of TPM

Prof.dr.ir. P. Herder, Section of EWI, Faculty of TPM

PREFACE

This report is written as completion of my master's research project at the Faculty of Technology, Policy and Management from the Delft University of Technology. The research is conducted at the section of Systems engineering, which is one of the cornerstones of the study Systems Engineering, Policy Analysis and Management. It was a long journey, but with finalizing this research, I realize that my time as a student has come to an end.

The goal of this research was to analyze how different management approaches can be combined to assess the complexity of projects. Therefore several management approaches are classified and analyzed, followed by expert interviews which were based on a complexity framework. For this I am grateful to all experts of the Faculty of Technology, Policy and Management, who have been so kind and cooperative to give their visions during the interviews, which serve as the basis for the conclusions of this thesis.

This thesis could not have been completed without the guidance and support of my first supervisor, Gwendolyn Kolfshoten, who provided me the opportunity to conduct this research and guided me during this project. Secondly, I am thankful to Professor Alexander Verbraeck, for providing his comments and reviewing my work even during a flight. His comments definitely made me being critical on my own work and being more consequent. I would like to thank Professor Paulien Herder, for her input in how I could approach this research effectively and come to useful conclusions.

I am grateful to my parents, whom have always encouraged and supported me in every possible way during my study period abroad. Secondly, I would like to thank family Autar-Jaggoe, whom have supported me from the beginning of my journey in the Netherlands. My thanks go also to my friends Vikash Mohan and Nitesh Bharosa, not only for their sensible discussions about the political situation in Suriname and football on Saturdays, but especially for their inspiring words. Finally I would like to thank my wife, Sushma, who have always motivated me when I needed it the most.

LIST OF TABLES AND FIGURES

Tables

Table 1 Overview of different management approaches	10
Table 2 Selection of management approaches	14
Table 3 Roles in Lean Six Sigma Projects	18
Table 4 Principles of Agile Development	24
Table 5 Comparison table 5 ways model	27
Table 6 Definitions of project complexity	30
Table 7 Four dimensions of complexity	31
Table 8 complexity elements of the actor dimension	33
Table 9 complexity elements of the context dimension	33
Table 10 complexity elements of the support dimension	34
Table 11 Subcategories of TOE Framework	35
Table 12 Layers of Multi-Actor Systems	36
Table 13 Seven stages of an interview research	38
Table 14 Respondents List	38
Table 15 Definitions of a Project	40
Table 16 Project complexity factors	40
Table 17 Maturity Levels (CMMI)	54
Table 18 Levels for analyzing management approaches	55
Table 19 Lean Six Sigma scores	57
Table 20 Process Management scores	57
Table 21 Collaboration Engineering scores	58
Table 22 Software Engineering scores	59
Table 23 Agile Development scores	59
Table 24 Systems Engineering scores	60
Table 25 System Dynamics scores	61
Table 26 Stakeholder Management in PMBOK	67
Table 27 Human Resource Management in PMBOK	68
Table 28 PMBOK support system tools	72

Figures

Figure 1 Overview of the research steps	12
Figure 2 The five ways model	15
Figure 3 Steps for process improvement with Lean Six Sigma	19
Figure 4 Design principles for a process	20
Figure 5 Elements of the FPM Model	22
Figure 6 Stages of the Waterfall approach	23
Figure 7 Actor, Support, Context Framework.....	32
Figure 8 Scores of management approaches on actor dimension	63
Figure 9 Scores of management approaches on support dimension	64
Figure 10 Scores of management approaches on context dimension	65
Figure 11 Enhancements on dealing with actor dimension	70

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES.....	4
Tables	4
Figures	5
1. INTRODUCTION.....	9
1.1 Problem delineation	9
1.3 Knowledge Gaps	11
1.4 Research Objective and Questions.....	11
1.5 Research Setup and Expected Outcomes	11
1.6 Thesis structure	13
2. LITERATURE STUDY	14
2.1 Selection of Management Approaches	14
2.2 Comparison model for different management approaches.....	14
2.3 Project Management Body of Knowledge.....	17
2.4 Lean Six Sigma	18
2.5 Process Management	20
2.6 Collaboration Engineering	21
2.7 Traditional Software Development	22
2.8 Agile Development	24
2.9 Systems Engineering.....	25
2.10 System Dynamics.....	26
2.11 Comparison of Approaches	27
3. FRAMEWORKS FOR ASSESSING PROJECT COMPLEXITY.....	30
3.1 Project Complexity	30
3.2 The Actor, Support and Context Framework.....	31
3.3 The Technical, Organizational and Environmental Framework.....	34
3.4 Framework choice	36
4. EXPERT INTERVIEWS	37
4.1 Qualitative Research	37
4.2 Interview Research.....	37
4.2.1 Thematizing.....	38
4.2.2 Designing.....	38
4.2.3 Interviewing	38
4.2.4 Transcribing.....	39
4.2.5 Analyzing	39

4.2.6 Verifying	39
4.3 Results of the interviews	39
4.3.1 Project Definitions.....	39
4.3.2 Project complexity factors	40
4.3.3 Project Management Body of Knowledge	41
4.3.4 Lean Six Sigma	42
4.3.5 Process Management.....	44
4.3.6 Collaboration Engineering.....	46
4.3.7 Traditional Software Development	47
4.3.8 Agile Software Development	49
4.3.9 Systems Engineering	50
4.3.10 System Dynamics	52
5. ANALYSIS OF RESULTS.....	54
5.1 Analysis framework	54
5.1.1 CMMI: Maturity Levels.....	54
5.1.2 Score Levels.....	55
5.2 Scores of Project Management	56
5.3 Scores of Lean Six Sigma.....	56
5.4 Scores of Process Management	57
5.5 Scores of Collaboration Engineering	58
5.6 Scores of Traditional Software Development.....	59
5.7 Scores of Agile Development	59
5.8 Scores of Systems Engineering	60
5.9 Scores of System Dynamics	61
6. SCORES OVERVIEW PER DIMENSION.....	63
6.1 Actor Dimension	63
6.2 Support Dimension.....	64
6.3 Context Dimension	65
7. ENHANCEMENTS FOR THE PROJECT MANAGEMENT BODY OF KNOWLEDGE.....	66
7.1 Actor Dimension	66
7.1.1 Verification of actor management in PMBOK.....	66
7.1.2 Enhancements on Actor Dimension.....	70
7.2 Support Dimension.....	72
7.2.1 Verification of support tools in PMBOK	72
7.2.2 Enhancements on the Support Dimension	73
7.3 Context Dimension	74
7.3.1 Verification on the context dimension in the PMBOK	74
7.3.2 Enhancements on the context dimension	74
8. CONCLUSIONS	76

9. REFLECTION	78
9.1 Limitations	78
9.2 Future Work	78
REFERENCE LIST	79
APPENDICES	83
Appendix 1 Interview protocol	84
Appendix 2 Interview Questions	86
Appendix 2.1 Questions in English	86
Appendix 2.2 Questions in Dutch	88
Appendix 3 Interview transcript Project Management	90
Appendix 4 Interview transcript Lean Six Sigma	93
Appendix 5 Interview transcript Proces Management	96
Appendix 6 Interview transcript Collaboration Engineering	100
Appendix 7 Interview transcript Traditional Software Development	103
Appendix 8 Interview transcript Agile Development	107
Appendix 9 Interview transcript Systems Engineering	110
Appendix 10 Interview transcript System Dynamics	114

1. INTRODUCTION

Project failures in terms of cost overrun and time delays occur frequently and are being investigated for a long time now (Flyvbjerg et al., 2003). The failure is often linked to a lack of basic processes (Whittaker, 1999), but increasing complexity of projects (Williams, 2005) and the underestimation of project complexity (Neleman, 2006) are also mentioned. There are different factors that can make projects complex, e.g. the number of involved stakeholders, risks, uncertainties, dynamic conditions, new technologies or a combination of different factors. This means that the complexity of projects can be influenced from different dimensions, which makes it difficult to assess projects adequately. However, several specific project management approaches are available, but they might no longer be able to deal sufficiently with complexity as projects become increasingly interdisciplinary. The problem that can occur is that a specific project management approach does not take all factors that can make a project complex into account, because the influence of actors, support systems and different context elements play a greater role for managing projects successfully.

These factors have to be considered and managed adequately within projects, which means that project management approaches possibly should be combined with other management approaches, e.g. for quality improvement, process design, software development and systems engineering. In this sense techniques from other management approaches can be combined to enhance specific project management approaches for dealing with complexity elements, in order to reduce project failure. Consequently, this research is carried out to present possible guidelines to project managers in terms of possible techniques that could help them to deal adequately with project complexity elements and manage complex projects successfully. The starting point of this research is to get insight in widely used management approaches, for managing projects, for actor negotiations, software development or quality improvement. The choice to start with a theoretical approach is deliberately made, because first it is necessary to identify if different approaches offer possibilities for combining techniques to enhance project management. Secondly, interviews with people who are experienced with a specific management approach provide starting points for possible combinations. Thus, the goal is to present guidelines for project managers to deal adequately with project complexity from a theoretical perspective that is supported by expert interviews. This study can be seen as a step towards the development of a framework that could help assessing project complexity in the future.

1.1 Problem delineation

The focus of this research is first on complexity of projects and on the other hand widely used management approaches. This denotes that in this research the term project complexity and management approaches have a central role. Several researches (Pavard and Dugdale, 2000) have been conducted on project complexity, but the definition of Mikulecky (2008) offers a view on how complex systems can be studied from multiple disciplinary

perspectives and are different for all practical purposes. In another research (Xia and Lee 2005), four dimensions of complexity are identified: structural complexity, dynamic complexity, organizational complexity and technology complexity. By combining different views on complexity, frameworks were developed that can be used to analyze and classify the complexity of projects. In this sense, two important frameworks will be discussed further in this research.

There are several project management approaches available, but their focus and methodology are different. The term “project” can be viewed from different angles. A project that is executed for improving the quality of processes is totally different from a project that is carried out to develop a product or information system. But generally a project is defined as a planned set of interrelated tasks that will be executed over a fixed period and within certain cost and other limitations (business dictionary 2011). In this research, a distinction is made between specific project management approaches, which are generally aimed at quality improvement of processes and other discipline oriented management approaches, which are aimed at actor negotiations, software development or modeling. For the simplicity, all approaches will be denoted as management approaches further in this research. A quick scan delivers the following list of widely used management approaches, along with the approaches that are used in the other dimensions.

Table 1 Overview of different management approaches

Quality Improvement	Actor Management	Software Development	Systems Modeling
Prince 2 (Prince 2, 2011)	Negotiation Harvard Method (Fisher et al., 1981)	Agile methods (Agile Alliance, 2011)	System Dynamics (System dynamics Society, 2011)
PMBOK (PMI 2004)	Collaboration Engineering (Kolfshoten et al., 2006)	Extreme (Extreme Programming, 2011)	Hevner Design Science (Hevner, 2004)
CMMI (Software Engineering Institute, 2011)	Process Management (de Bruijn et al., 2010)	Scrum (Scrum Alliance, 2011)	QFD (qfdonline, 2011)
Continuous Improvement (Asq, 2011)	The fifth Discipline (Senge, 1994)	Spiral Model (Boehm, 1988)	IEEE Std 1220-1998 (IEEE Standards Association, 2011)
Lean/6 σ (Sigmapro, 2011)		Waterfall approach (Sommerville, 2000)	Systems Engineering (NASA, 2007)
TQM (Pmhut, 2011)		Object oriented modeling (Engels & Groenewegen, 2000)	Meta Model (Herder, 2004)

This table forms the basis for approaches that will be explored, but due to the constraints of this research only two widely used approaches of each dimension are chosen to explore further.

1.3 Knowledge Gaps

First it is necessary to identify some knowledge gaps that have to be explored, before research questions can be formulated. As stated earlier, project complexity and management approaches are central in this research. So it is necessary to explore further on the following knowledge gaps:

1. It is necessary to classify management approaches in order to get an overview of differences in their methodologies.
2. It is necessary to elaborate on project complexity and the elements that play a role in making projects complex.
3. It is necessary to identify how different management approaches deal with project complexity.
4. It is unclear how these findings can be translated into useful guidelines for enhancing project management.

1.4 Research Objective and Questions

This research focuses on how combinations of different management approaches can be accomplished in order to enhance project management. First the research question is formulated, followed by the sub questions.

The research question can be formulated as follows:

“How can project managers combine tools and techniques from project management approaches and discipline oriented approaches, for managing multidimensional complex projects successfully?”

In order to achieve the required information that is needed to answer the research question, several sub questions are derived from the knowledge gaps:

1. How can a classification of different management approaches be accomplished?
2. What is meant by project complexity and how can it be assessed?
3. How does different management approaches support mapping and managing project complexity elements?
4. How can this level of support used to enhance project management?

1.5 Research Setup and Expected Outcomes

This research is conducted in different stages. First a desk research, where the classification of selected management approaches is conducted, followed by an exploration on project complexity frameworks. Next, expert interviews are performed to gather insight in the extent to which complexity elements are supported and

managed. Finally, the results of the interviews are used as starting point for presenting guidelines for enhancing project management. Figure 1 provides an overview of the steps that are followed in this research.

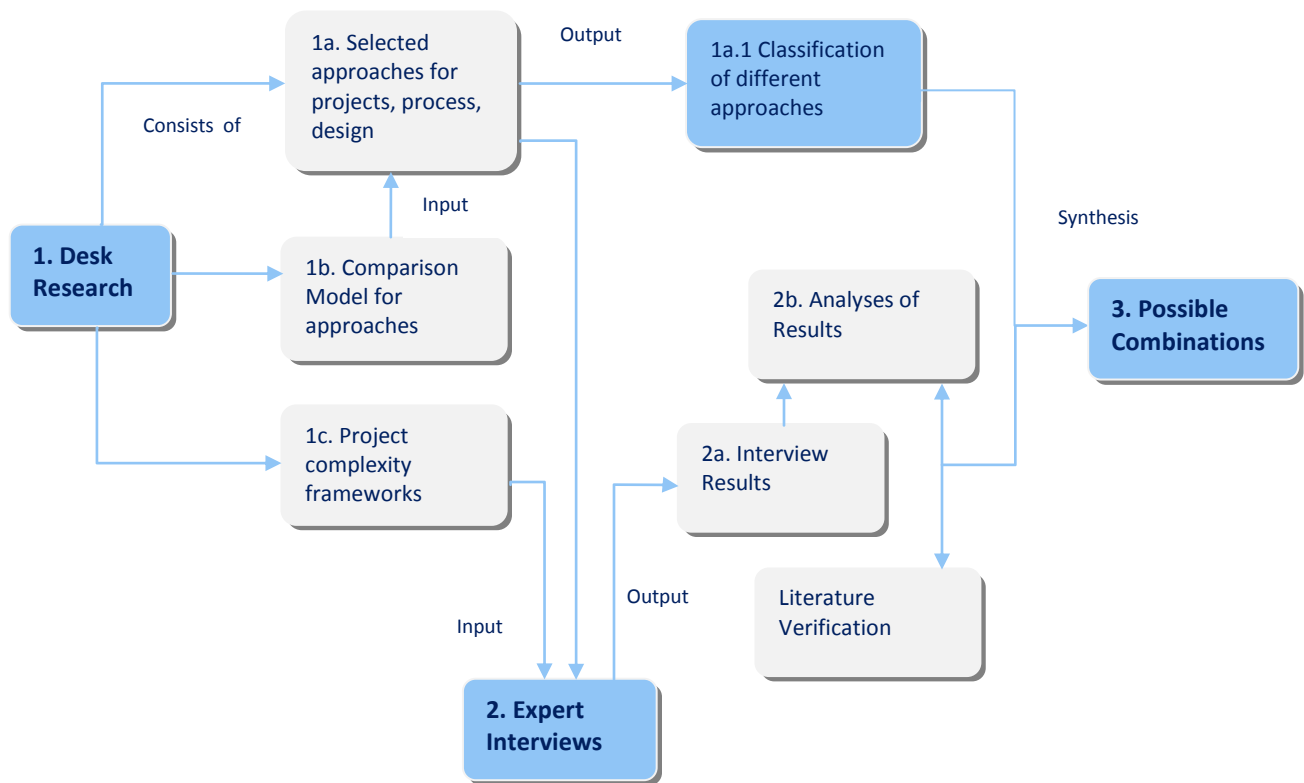


Figure 1 Overview of the research steps

The approach that is used to generate data is the Grounded theory approach, which can be defined as theory that is derived from systematically collected and analyzed data during the research process (Strauss et al., 1998). This approach is particularly chosen, because in this research purely qualitative data from several angles is collected and analyzed.

First a framework is presented that is created to classify information systems development approaches, but is suitable to compare and classify the selected management approaches. The classification of different management approaches is needed to understand and identify differences between them. Further this comparison method can be used for another set of management approaches in the future. Next, the term complexity needs to be elaborated and explored on frameworks that can help to assess complexity of projects. Further information need to be gathered on how the management approaches deals with complexity. Therefore expert interviews are conducted, where experts are asked to which extent the management approaches can deal with complexity elements. The insights provide possible starting points for enhancements for project management. A literature verification on possible issues that occur in project management provides insight in where necessary

enhancements need to be made. Conclusively, supported by the results of the interviews possible techniques from other management approaches are presented to enhance project management.

1.6 Thesis structure

A literature study on different management approaches is elaborated in chapter 2. First the selected approaches presented, followed by a comparison framework and the results of the comparison analysis. Chapter 3 follows with two different studies on project complexity and the framework that is chosen for this research. Chapter 4 consists of the interview research that is conducted to collect data on how different approaches support the complexity elements of a project, followed by the results of the interviews. The analysis of the results is described in chapter 5, where is discussed how the approach score on complexity elements based on levels that are created to measure the scores. In chapter 6 an overview of the scores is provided in order to identify starting points for possible combinations. This research ends with chapter 7, where possible techniques are presented for dealing with complexity in projects and enhancing project management. Finally the conclusions are presented, followed by some limitations of this research and recommendations for future work.

2. LITERATURE STUDY

This chapter consists of a comparative analysis of different management approaches, based on a literature study. First, the selection of management approaches that is used further in this research is presented. For comparing and classifying the selected management approaches, the five ways model of Seligmann et al. (1989) is proposed. This model is applied to each management approach and finally a comparison overview is presented with the main differences.

2.1 Selection of Management Approaches

As stated in the introduction the involvement of actors, tools and context elements plays an important role in managing complex projects. Some management approaches focus on quality improvement of processes within a project, while others deal with consensus building amongst actors. Other approaches are specifically developed for managing software development projects, while some approaches focus on systems modeling. In table 2, the selection of approaches is presented.

Table 2 Selection of management approaches

Quality Improvement	Actor Management	Software Development	Systems Modeling
Project Management Body of Knowledge	Process Management	Traditional Software Development	Systems Engineering
Lean Six Sigma	Collaboration Engineering	Agile Development	System Dynamics

It is clear that for this research, not all management approaches are taken into account, due to time constraints of this research and the availability of experts during the period of the research. Therefore, the selection is based on a sample of two widely used approaches per dimension and the aim for a balanced mix of management approaches from different disciplines. Next, a framework is presented that is used to classify and compare different approaches, followed by the application of this framework on each management approach.

2.2 Comparison model for different management approaches

From the selected management approaches, the only explicit project management approach is the Project Management Body of Knowledge, while the other approaches are focused on process improvements, actor negotiation processes, software development and systems modeling. Although the different purposes of each management approach are known, it is necessary to find out what the deeper differences are in terms of the philosophies and methodologies behind the approaches. This is needed to identify what the possible strengths of the approaches are and to identify possible starting points for combinations. But to conduct this comparison, a comparison model has to be developed or found in order to classify the different approaches.

This model can be developed by formulating different comparison criteria, but deliberately is chosen to use the five ways model of Seligmann et al. (1989), because this provides a better theoretical foundation and a structured methodology. When information systems are developed, the problem that developers come across is that the development is too complex to handle it straight forward. This complexity can be dealt with by using a model that can classify the different aspects of the development and focus to one certain aspect. The theory of Seligmann (1989) provides a structured way to look at information systems development processes. This theory contains a framework, where five important parts of a development approach are described, consisting of a way of thinking, a way of working, a way of modeling, a way of supporting and a way of controlling. The elaboration of the five ways is presented in the next sections and is supported with some elaborations of another research (Wijers, 1991).

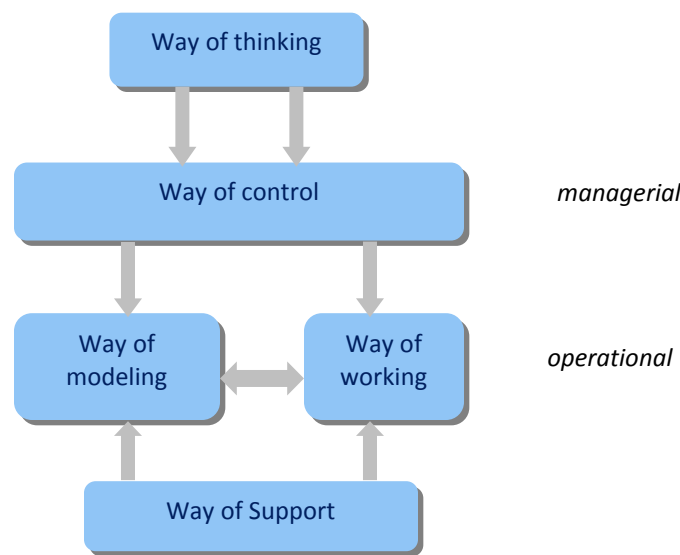


Figure 2 The five ways model

Way of thinking

According to Sol (1983), for really understanding a development approach or methodology, it is important to look at the way of thinking or the philosophy behind the approach. The way of thinking forms the basis for the important features that are considered for the processes that take place and the methodologies that are used. The basic assumptions and viewpoints of the method are described, including their relationship with the environment and components.

Way of control

The way of control comprises means that an approach offers at managerial level. Project management elements, such as control of time, costs and quality within the scope stand central here. Controlling an information systems

development consist of dividing the project in a logical sequence and parts, defining a project organization which includes the involvement of stakeholders and internal resources. Furthermore it is important to denote points where decisions and controls are necessary. The way of modeling and the way of working depend on the way of control, because they are both driven from the way of control.

Way of modeling

The way of modeling encapsulates the model concepts that are used and a description of the relations that it consists of. For example, in the development of information systems, there are many different aspects and interrelated problem areas that have to be dealt with. Therefore a series of models are needed to represent all the aspects of the development process, which is defined by the way of modeling. The models that are used can represent parts of the reality, organization or business systems, information systems or stages of the process. Specifically mentioned the way of modeling describes (Wijers & Heijes, 1990) the models and their relationships and the elements and their relationships. Some development methodologies present models with a clear elaboration on how to construct or verify a product, while other offers practical propositions for obtaining a model. If no specific models are available, some principles that have to be taken into account are presented.

Way of working

This part describes the tasks that have to be carried out for developing an information system. Subtasks are distinguished, arranged and described in terms of steps that should be carried out. Further, it presents guidelines for creating and working on the operational level. Besides this, it offers a means of describing the structure of the work that have to be carried out, including tasks and definitions at the operational level, procedures and informal indications.

Way of support

The way of support describes the tools that are needed to support the way of working, the way of modeling and the way of controlling in the best possible way. It can consist of simple templates, charts, but also computer aided software engineering (CASE) tools (Hoffer et al., 2002). Here it must be taken into account that the usefulness of the tools is related to the level in which the way of support is integrated with the other ways.

In the following sections, the described ways are applied to each management approach. It is not the intention to explore each approach extensively on the content, but more to emphasize on the differences between the management approaches. Therefore each of the “ways” is briefly described.

2.3 Project Management Body of Knowledge

Way of thinking

The term project management is sometimes used to describe an organizational or managerial approach for the management of projects and some ongoing operations. According to the PMBOK guide (PMI Institute, 2004); project management is the application of knowledge, skills, tools and techniques to project activities to meet project requirements. A project is described as a temporary endeavor that is undertaken to create a unique product, service or result.

Way of control

The project manager has the responsibility for accomplishing the project objectives (PMI Institute, 2004). Management of projects include, identifying requirements, establishing clear and achievable objectives, balancing the competing demands for quality, scope, time and cost. Further the project manager has to adapt the specifications, plans and approach to the different concerns and expectations of the various stakeholders. Further the project team members are the group that is performing the work of the project and the project management team, which are the members of a project team that are directly involved in project management activities.

Way of modeling

All the necessary processes that have to be conducted for managing a project successfully are modeled as discrete events with well-defined interfaces. The process interactions and data flow among the processes are modeled through flow diagrams. In the Project Management Body of Knowledge, the inputs, outputs and necessary tools of all processes within a specific knowledge area are modeled by using process flow diagrams.

Way of working

Project Management according to the PMBOK guide consists of nine knowledge areas and five process groups, each allocated to different management processes. For example, the processes and activities that integrate the various elements of project management are clustered together in project integration management and the processes that organize and manage the project team are described in project human resource management. So, the way of working in PMBOK, consists of all the activities of each knowledge area and achieving the required outputs.

Way of support

The necessary support tools for carrying out the processes of each knowledge area are mentioned in the PMBOK guide. The main support system is the Project management information system (PMIS), which is standardized set of automated tools available within a system. Two subsystems of the PMIS are the configuration management

system and change control system. Other support tools that are used are decomposition and other diagramming techniques, templates, forms and standards and (cost) estimation tools.

2.4 Lean Six Sigma

Way of thinking

Lean Six Sigma is derived from two different points of views. The focus of Lean is on the increase of product flow speed by eliminating processes that do not add value, while Six Sigma deals with variances in production processes for improving product quality. The goal of Lean Six Sigma is to ensure a logistical efficient process and products of high quality.

Way of control

The way of control in Lean Six Sigma projects is explained through the different roles and responsibilities that actors have in these projects. The objective of any design for Lean Six Sigma projects is to deliver better results more quickly with the help of fewer resources. Lean Six Sigma can be a sustainable approach to improvement, but it requires that people with different roles with clearly defined responsibilities have to work together (Sigmapro 2011). Lean Six Sigma is about improving an organizations performance, so people are needed that lead the improvement activities. In the table 3, different roles and responsibilities in a Lean Six Sigma project are specified.

Table 3 Roles in Lean Six Sigma Projects

Role	Area of Operation	Tasks
Black belt	Management level	Tackle complex, difficult and strategic projects
Green belt	Supervisory level	Part time improvement agent, not cross functional or strategic tasks
Yellow belt	Operational level	Help green and black belts on larger projects
Process owner	Local manager, Problem owner	Support belts with team members and expertise
Project champion	Senior manager	Make sure that project is strategic and necessary resource and focus is applied
Master black belt	Masters of methodologies	Train and coach the management team and the belts.

Way of modeling

The way of modeling of Lean Six Sigma can be described in terms of the design principles (Sahwney 2010):

- Specify values in the eye of the customer
- Identify the value stream and eliminate waste and variation
- Make value flow smoothly at the pull of the customer
- Involve, align and empower employees
- Continuously improve in pursuit of perfection.

Way of working

Quality improvement according to Lean Six Sigma can be achieved in several ways, such as DMAIC – Define, Measure, Analyze, Improve and Control. But the overarching method is always the same and consists of the following four aspects (Arthur, 2007). It is focused on improving quality, productivity and profitability.



Figure 3 Steps for process improvement with Lean Six Sigma

The steps can be defined as:

1. Focus on key problem areas by counting and categorizing delays, defects, misses, mistakes, errors and variation
2. Improve by eliminating delays, defects and variation
3. Sustain the improvement by monitoring key measures and respond if they become unstable. Keep the new, higher level of performance
4. Honor the progress through simple rewards

Way of support

Microsoft Excel is a powerful tool for Lean Six Sigma, but people have to be trained to use the quality improvement macros for Excel. Other tools are also used to support and making statistical analysis in Lean Six Sigma projects.

2.5 Process Management

Way of thinking

The philosophy behind process management is about change in complex issues, which often takes places in a network of multiple actors (de Bruijn et al., 2010). They depend from each other for the implementation of changes and have to negotiate with each other, which are complex in themselves. Mostly there are many negotiations needed to shape the change that is needed, so a number of meetings are required to form the change. Process management provides a view on how to deal with change that is achieved by negotiations in multiple actor networks.

Way of control

A good negotiation process needs to be designed on basis of some core principles. The way of modeling a process elaborates on what these principles are when a process is designed:

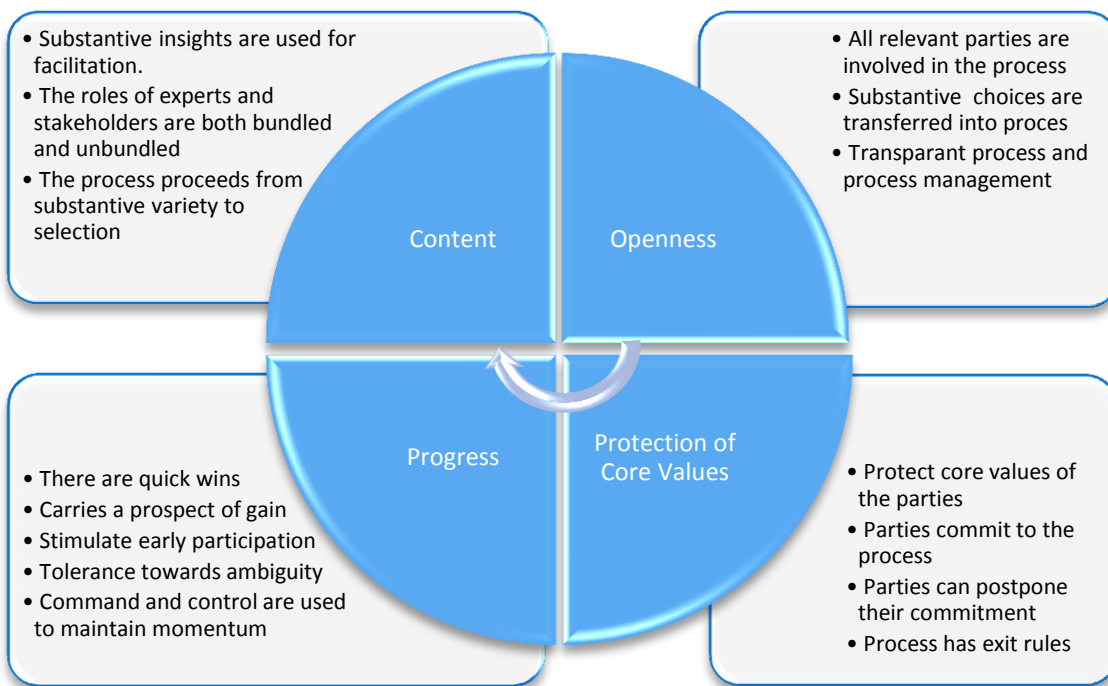


Figure 4 Design principles for a process (de Bruijn et al. 2010)

It is difficult to manage a process, because parties are only willing to reach agreements if these agreements give them the opportunity to serve their own interests. The design and implementation of a process may have a major impact on substantive outcomes.

Way of modeling

The process design is described in terms of an activity diagram (de Bruijn et al. 2010), where the process of dealing with substance and the process is modeled.

Way of working

The way of working in process management is described in terms of the tasks of the process manager in relationship with the design principles of a process. This cannot be standardized, because the configuration and the substance of the issue are different in every situation. The process manager has to ensure the openness of a process, protect core values of actors, transform substance into process tactically and create awareness of sense of urgency among stakeholders. So the process manager has to convince parties that there is an issue that needs to be solved and that their cooperation through a process is needed to solve the issue.

Way of support

There is no information systems involved to support the process. Support in this sense can be done through performing a simulation of the process that is designed to get more results and make adjustments where is needed.

2.6 Collaboration Engineering

Way of thinking

“Collaboration engineering is an approach for designing reusable collaboration processes and technologies meant to engender predictable and success among practitioners of recurring mission critical collaborative tasks” (de Vreede & Briggs, 2005). Collaboration is essential for value creation (Hlupic & Quereshi, 2002) and is often used for mission critical tasks. Working as a team has advantages, such as higher productivity and chances for success, but involves a challenge that leads to unproductive processes and failed efforts (Nunamaker et al., 1991).

Way of control

The collaboration process consists of a sequence of interventions that create a specific pattern of collaboration that are intended to direct the group towards an aimed goal. A collaboration engineer designs a reusable and predictable collaboration process for a recurring task, including technical support. The practitioners take this design and execute them without the ongoing help of facilitators, who are group process professionals. So the practitioners do not need to be experts in designing new collaboration processes for themselves, but they execute the designed collaboration processes as part of their regular work.

Way of modeling

The activities that take place in a collaboration session are modeled in a Facilitation Process Model (FPM). This model is used to document the process flow and critical elements of a collaboration process design. The elements of the model are presented in the figure 5 (Briggs et al., 2006).

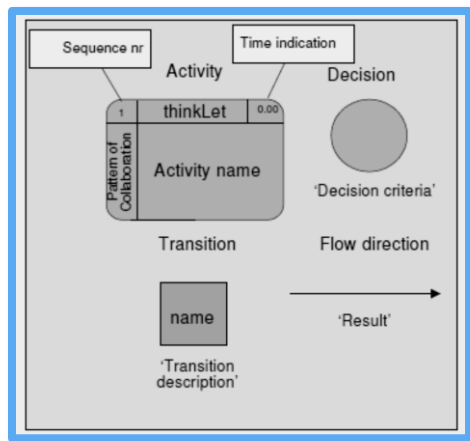


Figure 5 Elements of the FPM Model

Way of working

The way of working of collaboration engineering can be described as a sequence of patterns, consisting of a selection of thinkLets, which are used to generate certain patterns of thinking. The proposed sequence is; generate, converge, identify, evaluate, advocate.

Way of support

The use of Group Support System and software, e.g. Thinktank for performing collaboration sessions and CACE tools for engineering collaboration processes.

2.7 Traditional Software Development

The software development method that is chosen for this research is the waterfall method and is discussed according to the five ways model.

Way of Thinking

The philosophy behind the waterfall software development is that the development proceeds sequentially through a series of phases. It starts with system requirements and ends with product release and maintenance. Every

phase has a defined start and end point and the requirements and design are made clear in the beginning of the project.

Way of Control

The software development lifecycle is divided into five distinct and linear stages. The phases in a waterfall development are more linear than other models.

Way of Modeling

The way of modeling according to the waterfall method can be described in terms of how the end product is modeled through every phase of the approach. First the requirements that the end product should have are captured from the end-user by consultation. A Requirement Specification Document consists of analyzed and validated requirements for the next stage of the process. These requirements are used for the system design, where the hardware and system requirements are specified. The system design specifications serve as input for coding processes, which are translated in the actual development of the system. For coding purposes different high level programming languages like C, C++, Pascal and Java are used.

Way of Working

The waterfall development model provides a high-level framework that consists of five stages of developing a product (Huo et al. 2004). Developers that work according to the waterfall method have to go through the following deployment activities:

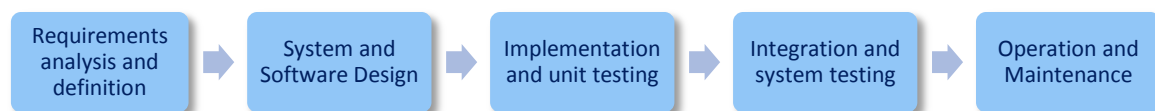


Figure 6 Stages of the Waterfall approach

Every stage can start only if the previous stage is finished and the results of the previous stage are approved.

Way of Support

The information systems that are used for the generation of code can be programming tools like, compilers, interpreters or debuggers. Testing methods and software is used to detect bugs that are inherited from previous phases.

2.8 Agile Development

Way of thinking

The philosophy behind Agile Development is that Agile methods apply iterative and evolutionary development within a time box, adaptive planning, promote evolutionary delivery and encourage to response flexibly to changes (Larman, 2004). The manifesto for Agile software (Agile Alliance, 2011) presents four main values for this approach:

- Individuals and interactions over processes and tools.
- Working software over comprehensive documentation.
- Customer collaboration over contract negotiation.
- Responding to change over following a plan.

Way of control

Agile projects are executed within small teams, where collaboration, coordination and communication are important. The decision making is spread out to the developers in Agile software projects, but management is still needed to remove barriers for making progress. The technical team however can make decisions on their own on basis of their expertise. The strategy for planning in Agile development projects is to make detailed plans for the short term and rough plans for the long term (Highsmith, 2000). The people factor in an Agile project is considered as the most difficult implication and managers need to consider the talent, skills and communication between the team members.

Way of modeling

There is no specific modeling technique for Agile Development, but the nine principles for the agile project manager could be helpful to give the Agile process the shape and direction that is preferred (Highsmith & Cockburn, 2001).

Table 4 Principles of Agile Development

Actor based
•Cultivate committed stakeholders •Employ a leadership-collaboration style •Build competent, collaborative teams •Enable team decision making •Encourage adaptability
Process based
•Deliver something that is useful to the client •Use short time boxed iterations •Champion technical excellence •Focus on delivery activities, not process compliance activities.

Way of working

Agile methods use an empirical process rather than linear process for developing software. The development process cannot be a linear process, because a high number of changes occur during the process. The philosophy behind Agile development states already that it is based on collaboration, interactions and iterations, making the way the working an iterative process.

Way of support

Collaboration plays an important role in Agile Development projects, because customer collaborates with developers by providing feedback on their efforts. On the other hand collaboration between Agile team members is important and they have to communicate often. The development tools are based on tacit knowledge of the developers or depend on the organization. There are tools for visualizing the scope of the project, orchestration of the processes and developments tasks and enforce process.

2.9 Systems Engineering

Way of thinking

Systems Engineering can be defined as the science of designing complex systems by enabling that (sub) systems are designed, composed, controlled and used on the most efficient way that is possible (van Daalen & Thissen, 2003). The system should meet the requirements within the given imposed constraints. Systems Engineering is about tradeoffs and compromises, about generalist rather than specialists (NASA SE handbook, 2007).

Way of Control

The way of control in Systems Engineering can be described in the way a product development is managed through this approach. Systems engineering seeks a safe and balanced design in the face of opposing interests and multiple, sometimes conflicting constraints. The systems engineer directs, communicates, monitors and coordinates tasks. He reviews and evaluates the technical aspects of the project to ensure that the engineering processes are functioning correct and evolves the system from concept to product. Further, the systems engineer usually plays an important role in leading the development of the system architecture, defining and allocating requirements, evaluating design tradeoffs, balancing technical risks between systems, defining and assessing interfaces, providing oversight of verification and validation activities (NASA SE Handbook, 2007).

Way of Modeling

The way of modeling in systems engineering can be described in terms of all requirements of the users or stakeholders. These requirements are described in requirements/specification documents, verification and validation documents, certification packages and other technical documentation.

Way of Working

This is described by elaborating the Systems Engineering process, which is an iterative process of technical management, acquisition and supply, system design, product realization and technical evaluation at the level of the system (IncoSE 2000). The process starts at the system level and is propagated through a series of steps to come to the preferred system solution or product. Iterations at every level help to analyze, postulate and evaluate, because decisions on one subsystem can influence other subsystems.

Way of Support

There are several types of tools that can be used to support the systems engineering processes. There are simple drawing tools, like PowerPoint and Visio. Other specialized text processing tools are used also for applying structure to documents. Classical Systems Engineering tools, like Core and Cradle, support the full life cycle and are more often used, but software engineering tools, featuring e.g. UML, are used also.

2.10 System Dynamics

Way of thinking

System Dynamics was developed because existing methods did not provide necessary insights for strategic problems in complex systems (SD Society, 2011). The philosophy behind System Dynamics is that the behavior of a system is being influenced for an important part by the structure of the system. System Dynamics is a methodology and simulation modeling technique for framing and understanding the dynamic behavior of complex systems (van Daalen & Thissen, 2003).

Way of control

Here it is important to have a look at the approach from the managerial side. As there is no managerial side in this approach, the way of control is considered from a modeler's viewpoint.

Way of modeling

The systems are modeled by drawing causal loop diagrams in collaboration with the problem owner and other actors and transferring into flow diagrams, consisting of levels and rates.

Way of working

This consists from predefined steps that need to be followed in order to come up with a model:

- Define the problem.
- Conceptualization: Making diagrams of the qualitative model and analyzing the model.
- Specification: Filling in the diagrams with quantities data and determine the relations between the variables and diagrams.

- Verification: Check if the model is coded correctly and the dimensions of the equations are consistent
- Validation: Used to check whether the model deals adequately with data of the system.

Way of Support

Representing and solving Systems Dynamics models can be done with different software packages, e.g. Powersim, ithink and Vissim. By using these packages, a system dynamics model can be represented simply as a computer model.

2.11 Comparison of Approaches

The management approaches that are described in the previous paragraphs are classified, but an overview is needed to present the differences and identify possible starting points for combinations. Table 5, presents an overview of the comparison according to the five ways model.

Table 5 Comparison table 5 ways model

	Way of Thinking	Way of Control	Way of modeling	Way of Working	Way of Support
Project Management (PMBOK)	Organizational or managerial approach to the management of projects	Clear defined role and responsibility of project manager and project team	Process Flow diagrams	Activities based on knowledge areas	PMIS, CMS, CCS, templates, estimation tools
Lean Six Sigma	Reduce waste and improve quality	Defined roles with tasks	Design principles	Focus, Improve, Sustain, Honor	Excel Power Tools
Process Management	Change in complex issues, network of multiple actors	A process design is based on four design principles	Activity diagram	Tasks of the process manager	No information systems
Collaboration Engineering	Create sustained collaboration	Reusable and predictable	Facilitation Process Model	Perform a sequence of selected thinkLet	Group Support

	support	collaboration process		patterns	Systems and Software
Software Development (Waterfall approach)	Development proceeds sequentially through a series of phases	Lifecycle is divided into five distinct and linear stages	Requirements specifications and coding	Linear deployment process of different phases	Software for compiling, interpreting, debugging and testing
Agile Development	Apply iterative evolutionary development	Collaboration, communication in small teams	Nine principles of Agile	Empirical, Iterative process	Software for designing, coding.
Systems Engineering	Developing a system that meets requirements	Seeks a Safe and Balanced design	Requirements, specifications, certification packages	Iterative process evaluation	Software: Drawing, SE software, UML
System Dynamics	Understanding the dynamic behavior of complex systems	Viewpoint of the modeler	Causal loop diagrams, Flow diagrams	Predefined steps, Linear	Software: Powersim, Vissim

The way of thinking points out that there is only one explicit project management approach, the PMBOK. Lean Six Sigma has the goal of improving quality and reducing waste from an organization, so it can be seen as a quality improvement approach rather than a project management approach. The way of control in both approaches is based on clearly defined roles and responsibilities. PMBOK thrives on process flow diagrams that have to be followed for completing the tasks within each knowledge box successfully, while Lean Six Sigma is based on design principles. So PMBOK is the only project management approach here, and the enhancements will be focused on this approach.

Process Management and Collaboration Engineering are clearly focused on managing actors, their stakes and solving conflicts between them. Process Management specifies how a negotiation process between actors can be shaped on basis of 4 principles, while collaboration engineering focuses on reaching a joint group goal and is

supported by electronic tools. This could mean that in projects where a network of actors plays a role; there two approaches can be helpful to manage their involvement.

The waterfall approach and Agile Development were developed for software engineering. They differ largely in the sense of way of working, because the waterfall approach is based on a linear way of working while Agile Development is based on an iterative way of working. Both methods can be useful, the waterfall approach in situations where all requirements are clear from the beginning and Agile Development when there are uncertainties and rapid changes and quick input of the team is required.

Systems engineering is based on balancing the requirements that are derived from the context, so that the design consist of all possible requirements. System dynamics is a modeling approach for understanding the behavior of systems. Both methods take the context elements into account, which means that when a project result is largely context based, these two approaches can be taken into consideration.

3. FRAMEWORKS FOR ASSESSING PROJECT COMPLEXITY

Now the differences in management approaches are clear, it is necessary to identify which elements contribute to project complexity and how this can be assessed. This section starts with describing different views on project complexity, where it becomes clear that there are different dimensions that can make a project complex. This is followed by an elaboration of the Actor, support, context framework of van der Lei et al. (2010) and the Technical, organizational, environment framework of Bosch-Rekvelde et al. (2010). Further, the choice of framework for this research is elaborated briefly.

3.1 Project Complexity

The term complex is derived from the Latin plexus (Luoma, 2006), meaning breaded together. In a complex system, the different elements interact and the outcomes are unpredictable and nonlinear. Complex means “composed of many interconnecting parts”, while complicated is defined as “composed of many elaborate interconnecting parts” (Maylor et al., 2008). Still complexity was seen as a black box and the factors that cause complexity in projects were not elaborated. However some interesting definitions and key aspects of project complexity found in project management literature give an indication of the way complexity could be threatened.

Table 6 Definitions of project complexity

Authors	Complexity
Baccarini (1996)	The number of physical elements of a project and the way they depend on each other.
Williams (1999)	Is characterized as structural uncertainty (number of elements and their independence, multiple objectives and stakeholders) and uncertainty (of goals and methods).
Parwani (2002)	Complexity refers to the study of complex systems, of which there is no uniformly accepted definition because, well, they are complex.
Maylor (2003)	The product of organizational, technical/novelty and scale/resource complexity.
Xia and Lee (2005)	Four dimensions of complexity: structural complexity; dynamic complexity; organizational complexity; technology complexity, involving aspects such as the number of elements, interdependence uncertainty and change or dynamics.
Whitty & Maylor (2009)	Project complexity should include structural, dynamic and interaction elements.

Apart from these different views on project complexity, there was no well-defined framework in the literature that could be used to analyze and systematically describe the key dimensions and characteristics of project complexity (Xia and Lee, 2005). The need for a complexity assessment became clear, as this could help managers to analyze and response to the practical project complexities. The certain characteristics of different projects are needed to determine the specific actions that can be taken to manage them successfully. Baccarini (1996) states that organizational, technological and informational complexity are such project dimensions and proposed that project complexity can be measured in terms of differentiation and interdependencies. Some projects are complex because of many stakeholders involved, while others are complex because unproven technologies are experimented or dynamic conditions are taken into account. The definition of complexity by Mickulecky (2008) followed the idea that every complex thing is different from another viewpoint. He stated that “complexity is the property of a real world system that is manifest in the inability of any one formalism being adequate to capture all its properties”. This definition makes it clear that complex systems can be studied from multiple disciplinary perspectives and can be different for the purpose it serves.

3.2 The Actor, Support and Context Framework

As stated in the previous paragraph there was a clear need for a framework for assessing complexity of projects. The definition of Mickulecky offered the opportunity of viewing complex systems from different angles. Manson (2001) specified aggregate complexity as one of the different types of complexity. This type of complexity can be related to the behavior and performance of socio technical systems, because it deals with individual elements and their relations and how a system responds to change. Xia and Lee (2005) took a same approach as Manson, by distinguishing four dimensions of complexity, which are described in table 7.

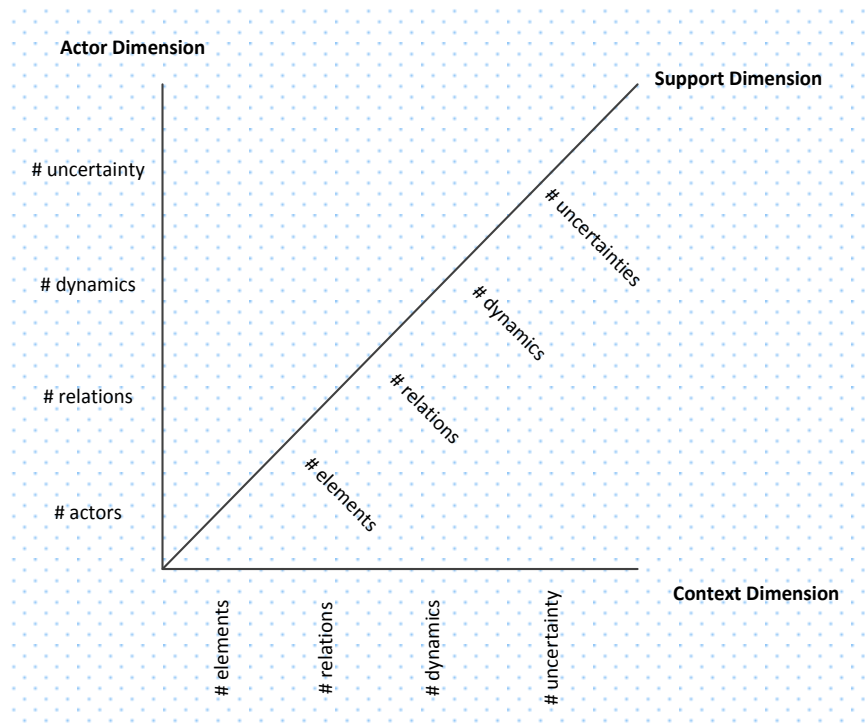
Table 7 Four dimensions of complexity

Type of complexity	Aspects
1. Structural complexity	Number of elements, scope, variety, multiplicity and interdependence
2. Dynamic complexity	Ambiguity, uncertainty and change or dynamics
3. Organizational complexity	Complexity of the knowledge environment or organization
4. Technology complexity	Technological complexity and uncertainty

Van der Lei et al. (2010) combined both perspectives by using the aggregate complexity (Manson 2001) and adapting the four dimensions (Xia & Lee, 2005) to the actor and system dimension of multi actor systems. The framework that is developed by Van der Lei et al. (2010) is the result of this combination and consists of four aspects of complexity, namely, elements, relations, dynamics and uncertainty. These four aspects are applied on a

multi-actor perspective and a system perspective, which are translated into an actor dimension and context dimension. In a multi-actor setting, the different actors can have different stakes and they interact with each other and the systems they are surrounded with (de Bruijn & ten Heuvelhof, 2010). The multi-actor perspective is applicable for every discipline where actors are involved. The context dimension is based on a system, which is defined as a set of entities, comprising a whole where each component is related to at least one other component and they all serve a common purpose. The framework that is developed by van der Lei et al. (2010) is used to perform a meta-analysis to classify the complexity of multi-actor research. Initially the framework consisted of two axes, the actor dimension and the context dimension, but the support dimension was added on later. This was done because the support dimension seemed an important factor, as the query of the study consisted a wide range of articles where tools or social software were mentioned to facilitate actors or test context models. The study resulted in the Actor, Support, Context (ASC) framework as presented in figure 7.

Figure 7 Actor, Support, Context Framework



The different complexity elements are depicted in the framework, but it is necessary to know how these complexity elements influence the complexity of each dimension and how they are influenced. The complexity elements are elaborated to provide a view on how they cause the complexity to increase per dimension. The elements of the actor dimension are described in table 8.

Table 8 complexity elements of the actor dimension

Complexity elements of the actor dimension	Definition
# actors	The number of actors involved. If more stakes or perspectives are involved, the complexity level increases.
# relations	When actors depend on each other, have long term relations, are bound by contracts, the complexity level increases.
# dynamics	Amount of change or dynamics in the multi-actor dimension. This is caused by interaction, changing positions or collaboration between actors.
# uncertainty	Uncertainty with respect to actor behavior; problems of trust, cultural differences and hidden agendas.

The context complexity is described by the number of elements in the context, their relations, dynamics and uncertainty in the context and is described in table 9.

Table 9 complexity elements of the context dimension

Complexity element of the context dimension	Definition
# elements	The number of elements or components the context system exist of.
# relations	The way the components of a system are related or depend on each other.
# dynamics	Complexity increases if a components or relations change and affects other components and relations through interfaces. Dynamics increase when the timeframe of the project increases or when the system is in design, development or construction.
# uncertainty	Plays a role in innovative settings, where the technology is still unproven or potential risks exist.

The support dimension concerns an intervention system that is used with the goal to change the status quo within the context and/or the actor dimension. The complexity of the support dimension depends on the amount of elements, their relations, dynamics and uncertainty in the system. By the term system is meant the support system itself. The complexity elements of the support dimension are described in table 10.

Table 10 complexity elements of the support dimension

Complexity element of the support dimension	Definition
# elements	Amount of software components and other components the support system consists of.
# relations	When the support systems exits of several components, their interoperability and interdependency are an important determinant of complexity
# dynamics	Dynamics increase when support systems are subject to change, design, development or improvement.
# uncertainty	Plays an important factor when functioning of the system depends on third parties, or when new or innovative tools have to be developed.

3.3 The Technical, Organizational and Environmental Framework

The Technical, Organizational and Environmental (TOE) framework is developed by Bosch-Rekvelde et al. (2010) for characterizing project complexity in large engineering projects. They state that the importance of complexity is made clear by a large number of project complexity related papers, but there was no framework that could provide support to characterize and analyze project complexity in large engineering projects. Consequently the TOE (Technical, Organizational, and Environmental) framework was developed based on a literature survey and a number of interviews in the process engineering industry. This framework makes it possible to assess complexity and subsequently to manage the complexity of the project in an early stage. Bosch-Rekvelde et al. (2010) aimed to make a detailed description of project complexity, by using an inductive approach. They first performed a literature study to gather elements that are assumed to contribute to project complexity, followed by case studies where elements that contribute to project complexity were found by performing interviews in six different projects.

They found a total of 40 elements contributing to project complexity from the literature search, by studying relevant articles by searching on the keyword 'project complexity'. The conclusion of this was that not only the

technical or technological aspects play a role in determining the complexity of a project, but as de Bruijn et al. (1996) had distinguished complexity exists of three dimensions: technical complexity, social complexity and organizational complexity. The proposed framework that followed is named the TOE framework, consisting of the technical view which is focused on the content of the project, the organizational view which includes the softer aspects and the environmental view which includes the environmental influences. Further on, they performed case studies by interviewing different members of six projects to identify elements that contributed to project complexity in the practice of project management. The results were that almost all the elements that were found in the literature were independently confirmed by the interviewees. The development of the framework for project complexity, the TOE framework, is based on a combination of elements of the literature and the case studies. The elements were grouped into subcategories and are described in table 11.

Table 11 Subcategories of TOE

Technical	Organizational	Environment
Goals	Size	Stakeholders
Scope	Resources	Location
Tasks	Project team	Market Conditions
Experience	Trust	Risk
Risk	Risk	

The elaborate TOE framework consists of 3 levels; categories, subcategories and elements. It provides the opportunity to discuss with involved parties and stakeholders about the complexity elements on various levels of aggregation. The primary objectives of the framework are to get better understanding of project complexity and enable the identification of complexity areas, subsequently pay attention to the management of these areas. It was remarkable that size as a complexity element, in terms of engineering hours and capital expenditure, which are seen as the dominant measure of project complexity was mentioned only by a few interviewees. But size related aspects, like the number of project management methods and tools and number of stakeholders were mentioned more often. Other elements of a structural character, like the number of goals, largeness of scope, number of tasks, dependencies between tasks, but also uncertainties concerning goals and methods were covered in the technical category of the framework. However other structural elements, like the number of project management methods and tools, the number of different disciplines are covered in the organizational category. The environmental category consists of elements like, the variety of stakeholder's perspectives, political influences, level of competition, strategic pressure, required local content and weather conditions.

The application of the TOE framework could for example be useful to support risk assessment in early project phases. But to cope the dynamics of project complexity, it is recommended to use the framework during different stages of the project life cycle to grasp the complexities that occur which were not taken into account in the initial

stage. The TOE framework provides the management of a footprint in terms of where complexity is expected in a project. This gives the management to react in an early stage, for example by allocating a project manager with technical knowledge in a project where technical complexities are expected. The framework could also help to decide if a project needs extra efforts in process management, stakeholder management or risk management. But as the authors describe the framework has some limitations. First the qualitative character, where the empirical results suggested data saturation and second, the study was focused on engineering projects in the process industry.

3.4 Framework choice

Both frameworks have some similarities, because both take into account the involvement of actor and context elements. The TOE framework specifies this into internal and external actors and the technological aspect of the TOE framework can be linked to the context dimension of the ASC framework. The complexity elements, size and uncertainty are mentioned in both frameworks, but the generic form of the ASC framework provide a better starting point for mapping a broad scale of approaches. So, the framework that is chosen for further research in this thesis is the Actor, Support, Context framework of van der Lei et al. (2010). As described earlier in this research, the focus is on different dimensions of complexity levels and this framework provides a foundation for the multi-dimensional character of the research. The TOE framework of Bosch-Rekvelde et al. (2010) provides a detailed view of complexity in engineering projects, but does not provide starting points to measure the extent to which different management approaches support the management of complexity elements. Another important aspect is that the support dimension is stated explicitly in the ASC framework, as this is considered as important. The five ways model of Seligmann (1989), which is used to classify different approaches, also incorporates the support dimension in the model. This research is conducted at the section of systems engineering, where the design of multi-actor system from a socio-technical perspective is researched (Faculty TPM, 2011). They describe that a multi-actor system consists of three key layers, which are also incorporated in the ASC framework.

Table 12 Layers of Multi-Actor Systems

Layer	Description
1. Actor System	Here actors and stakeholders interact, make decisions, share resources and collaborate
2. Support System	Divided into content support (e.g. information systems, simulation models) and process support (e.g. collaboration systems)
3. Context System	Represents a demarcated application domain which can have a real life context (e.g. factory or company) or a virtual context (e.g. simulations, virtual worlds)

Consequently, the ASC framework is chosen to conduct this research further and serve as basis for the interviews that are conducted in the next section.

4. EXPERT INTERVIEWS

This chapter provides an overview of the steps that are performed to conduct the interviews. First an elaboration is made on qualitative research and interview research, followed by the results of the interviews, which form the basis for the next sections.

4.1 Qualitative Research

Qualitative research works up from the data. A project is qualitative because you don't know in advance what you may learn from the data. In qualitative research it is often heard that themes or theories emerge from data, but actually theories do not emerge spontaneously. Qualitative interviewing is based on conversations (Kvale, 1996), with the emphasis on researchers asking questions and listening and respondents answering. The purpose of most qualitative interviewing is to derive interpretations, not facts or laws, from respondent talk. Perspective is especially significant in qualitative interviewing, where meaning making is center stage in the interpretative process (Gubrium & Holstein, 2001). This research is qualitative, because the researcher don't know in advance what the answers are and derives interpretations from the answers of the respondents.

4.2 Interview Research

Interviewing can be described as an activity that is steered by the research issue and the operationalizing of key concepts (Verschuren & Doorewaard, 2010). An interview can be characterized by a limited degree of pre-structuring or an open style of interviewing. The interview type that is performed for this research is semi pre-structured, because the questions are consistently and precisely defined in the same way for all experts and they have a degree of freedom to decide the way to answer the questions. However, the interviews had a certain open character, because in some cases or questions more elaboration from the interviewer was required or some unplanned useful interactions occurred during the interview. The research objective of requires to gather data on experts views, where interviews are an appropriate and efficient way for. The seven stages for conducting an interview research (Kvale, 1996) are presented in table 13.

Table 13 Seven stages of an interview research

1. Thematizing	• Formulate the purpose of the investigation • Describe the context of the topic
2. Designing	• Plan the design of the study
3. Interviewing	• Conduct the interviews with a reflective approach to the knowledge sought
4. Transcribing	• Prepare interview material for analysis • Include transcription from oral speech to text
5. Analyzing	• Decide which methods of analysis are appropriate • On basis of the purpose and nature of interview material
6. Verifying	• Ascertain the generalizability, reliability and validity of the interview findings
7. Reporting	• Communicate the findings of the study and the methods applied in a form that lives up to scientific data • Results in a readable product

4.2.1 Thematizing

The purpose of the interview is to get insight in the extent to which different management approaches support to map and manage project complexity elements of the Actor, Support, Context framework.

4.2.2 Designing

The interview starts with a short introduction, where the interviewer starts with a short introduction followed by the respondent. The three dimensions of the Actor, Support, Context framework (van de Lei et al. 2010) is used as basis for this research. The respondents are asked questions about how the approach deals with each dimension.

4.2.3 Interviewing

The respondents are chosen on basis of their experience in different management approaches. All respondents are professors at the faculty of Technology, Policy and Management of Delft University of Technology.

Table 14 Respondents List

Approach	Expert	Section
Project Management	Prof.dr.ir. A. Verbraeck	Systems Engineering
Lean Six Sigma	Dr.ir. Z. Lukso	Energy & Industry
Process Management	Dr. W. Veeneman	Policy, Organization, Law and Gaming
Collaboration Engineering	Dr.ir. G. Kolfchoten	Systems Engineering
Traditional Software Development	Dr. S. Lukosch	Systems Engineering

Agile Development	Dr.ir. M. Janssen	ICT
Systems Engineering	Dr.ir. A. de Haan	Policy Analysis
System Dynamics	Dr.ir. C. van Daalen	Policy Analysis

4.2.4 Transcribing

The interviews were recorded and translated into text, which can be found in the appendix of this research. An abstraction of the interviews is used for further analyses in this research. The text is structured in such a way that it gives a short introduction of the expert, a definition of a project, project complexity elements, followed by the view on the actor, support and context dimension.

4.2.5 Analyzing

Content analysis can be classified in two main categories, the qualitative and the quantitative content analysis. This research contains only qualitative data and elaborates on qualitative content analysis. A framework is created to analyze the scores of each management approach. The results are coded levels, so that can provide a basis for identifying possible combinations.

4.2.6 Verifying

The goal of this research is to enhance project management in such a way that it can manage the complexity elements of a project properly. Before suggestions can be made on basis of the interview results, first a verification is made of the possible weak points that could be enhanced. This is followed by the identification of possible techniques from other approaches, based on the scores of the interview results.

4.3 Results of the interviews

This paragraph consists of summaries of the results that were achieved during the interviews. The transcripts of the interviews are included in the appendices.

4.3.1 Project Definitions

Every respondent was asked to provide a definition of a project. This is particularly done to identify if there is a uniform understanding of a project amongst all respondents. The different definitions of a project that were mentioned are summarized in this section. There are some differences, but most important is that the involvement of actors is mentioned mostly by all respondents. Further the goal can be seen as an important factor, where it can be an end product, a solution, an enhancement or an agreement.

Table 15 Definitions of a project

#	Definition
1.	A project is a set of activities where there are resources and budget for needed, with a limited duration and a clear beginning and end.
2.	A project is a discrete set of activities with a clear goal of reaching something. This goal has to be specified and defined in terms of necessary resources that help to reach the goal.
3.	No specific definition for a project, like other clear definitions of what a project is. A project may well be linear if the scope of what is to be build can be predicted well.
4.	A project is the standardization of a process with a collaborative approach.
5.	A project has a beginning and normally should have an end, where you are trying to create something. The end product of software development processes are normally a product based on a clearly defined goal and specifications
6.	A project is part of a total program that you want to achieve by running several projects. Every project contributes to reaching the goal of a program.
7.	A project starts at the moment an actor or group is not satisfied with a situation. The goal is to create a useful product.
8.	A project can be an issue, where people are trying to come up with solutions within a certain time period and scope.

4.3.2 Project complexity factors

The complexity factors are presented to get insight in complexity elements of projects, according to the respondents. This can help to identify approaches that are strong in solving this complexity factors, providing a basis to combine techniques from these approaches with project management.

Table 16 Project complexity factors

#	Complexity factors
1.	Involvement of many stakeholders, time pressure, lack of resources or technological complexities
2.	Multiple actors, different interests that have to communicate with each other. A joined goal has to be translated to sub goals that fit in the organization
3.	Ill-defined boundaries of a project, adjusting actors to each other
4.	Many stakeholders, different interests, especially when their interests, information or knowledge conflicts, the subject or the context of the project.
5.	Different actors involved in the project, the intended outcome, complex end product, the more lines, more relations and interdependencies there are, the more complex the product becomes

6.	Understanding the environment, uncertainties, dynamic environment, high ambitions, embedding to the whole
7.	Interaction between actors, dynamics in the context
8.	The size of the problem, the uncertainties in the system, finding data because everything has to be quantified

Here, most experts mentioned the involvement of many actors and stakeholders as the main complexity element. Secondly, dynamics and uncertainties in the system or context play a great role of importance in making a project complex.

Next, the results of the management approaches in relation to the ASC framework are presented.

4.3.3 Project Management Body of Knowledge

This interview is conducted with Professor Alexander Verbraeck, professor at the section of systems engineering and expert on systems and simulations (appendix 3). He has a mathematical background and experienced on the field of project management approaches as this is one of his teaching areas.

In his vision there is not one ideal project management approach, as every approach consist of a checklist and the result should be good if the work is done according to the checklist, but in practice things are different although. Many project management approaches are weak in stakeholder management, risk management and contracting. The approaches need to deal with complexity factors, by dealing with “soft” complexity factors with process management and for technical complex factors with project management. PMBOK for example, consist of 9 aspects and by completing them your entire project result should be good. Things seems to be different in practice, as risk management, stake holder management, interface management are naïve and contracting is left out in the 9 boxes. This means that this approach is not sufficient for the complexity that is faced in some projects.

The approach does not describe how to map and manage the actors, but describes that the project manager can make a stakeholder map. Within certain boundaries it is possible to look at the relations of the actors, but another technique should be used as PMBOK does not describe how to do this. Dynamics and uncertainties of actors are not reflected in PMBOK. One of the strong points is interface management that includes relations between actors and how actors are linked with the work break down structure.

The support dimension cannot be seen as a factor that can make a project complex, but as a solution for complexity. PMBOK states that support is necessary, but does say specifically which tools are needed. Project management information systems (PMIS) are stated frequently, because all stakeholders should have insight in the

project information. Support tools are always necessary, e.g. scheduling tools, costing tools, risk register tools or quantitative risk analysis tools, otherwise values cannot be calculated. For instance a small project could be completed with MS Project, but for a larger project tools as Primavera are necessary. For each of the 9 boxes of PMBOK, it is described which type of tools can be used. So, tools and techniques are stated explicitly in PMBOK, but it is not enough to manage projects of very high complexity.

The context dimension is missing in most of the project management approaches. Each of the 9 boxes supports to create insight in the context, but do not state that critical success factors must be defined on beforehand. When searching for critical success factors, drivers or project goals, it seems that they are not stated explicitly in the PMBOK guide. Context is mentioned briefly, but is not elaborated with techniques. So context variables in which most of the project take place is missing in most of the approaches, as for PMBOK. There is checklist, but there are no tools or techniques that can help to map the context dimension of the project. Dynamics and uncertainties are covered in the risk management box, but for example when something changes in the context of the project, meaning that changes have to be made in the project management, there are no feedback loops in PMBOK.

In complex technological research projects where experts are needed, PMBOK does not state anything about how to negotiate with the experts. Integration of collaboration engineering or stakeholder management would be helpful, as these are missing in PMBOK. Also projects where specific risks are involved and risks analysis are needed, for example safety checks in the airplane industry, are missing in the approach. Yet again, working the checklist down, does not mean that a good risk management process has been conducted, given the complexity of the project. So, for making good risk analysis, the risks have to be treated separately by using, for example the ATOM methodology. Still PMBOK is widely used because it has some strengths, as a clear structure, developed in details, internationally standardized, and the weak sides could be covered by other approaches.

4.3.4 Lean Six Sigma

This interview is conducted with Dr. Zofia Lukszo, professor at the section of Energy and Industry (appendix 4). She is the leader of the project intelligent infrastructures, part of Next Generation Infrastructures, where intelligent infrastructures are viewed and modeled by using agent-based modeling. Lean Six Sigma is one of her teaching areas at the faculty.

Lean Six Sigma started with Six Sigma only, a philosophy that is aimed at reducing the variance of repeated processes. With Six Sigma people try first to get insight in the variance of the output of the activities and secondly to control this variance in order to optimize repeated processes. Lean is aimed at reducing waste in an organization. Lean Management is about making an inventory of the waste in the organization and reduces it to the minimum. It

can also be quality given away or spending time doing things with a better added value for the organization. A combination of both offers all for controlling processes in organizations. Both methods are based on statistical tools and data analysis, only the focus lies differently. At the moment a company reaches the six sigma quality standards, the organization will deliver products that match always to the specifications. Lean Six Sigma is a complete project management approach; it describes how to allocate tasks and roles to all actors.

This implies that Lean Six Sigma is capable of coping with actors. Before a Lean Six Sigma project is started, a clear structure is needed in the organization to determine which actors can be used for which tasks. Lean Six Sigma offers the possibilities for coordination and cooperation between actors, e.g. operators have to make measurements and green belts have to estimate the models. With dynamics in the actor dimension she understands that a situation can occur where actor positions can change. In Lean Six Sigma projects, every role is predefined, meaning that when for example a green belt is not available, he or she is replaced by another green belt. When it is found while monitoring a project that it does not contribute to the main goal, the scope of the project can be changed which is the responsibility of the black belt. With intermediate evaluation rounds the projects and the actors can be monitored on their progress. Uncertainties in the actor dimension are almost excluded, because Lean Six Sigma is seen as a rigid method, where every project starts with clear responsibilities and tasks for every actor in the project. This is one of the reasons that Lean Six Sigma is used mostly in companies instead of network like situations where actors are dependent of each other. What companies try to do is when they work with other parties is to learn them the advantages and strengths of Lean Six Sigma, which helps to optimize joint processes.

Support tools are seen as a means to achieve something. From the viewpoint of complexity theory, a complex system can never be described by one paradigm. You need different viewpoints, e.g. psychology, sociology or mathematics, to describe a complex system. Transferring this to the support dimension of the framework, it can be said that different types of tools are needed for interaction, for monitoring data or for measuring economic factors. If it is clear that different types of tools are needed, it can be said that it contributes something to the complexity. But tools do not add complexity and are always involved in every project, just as actors are involved. So it is obvious that the number of needed support tools does not say something about the complexity of a project, but if different types of tools are needed, it could suggest that a project is complex. If hundred administration tools are needed for a project, then it is not necessarily a complex project, but when different types of tools, e.g. mathematical, conceptualization and simulation tools are needed a project could be classified as complex. Lean Six Sigma describes what type of tool to use for which situation, as is described how to do experiments in case of uncertainties.

Lean Six Sigma is applicable in a concrete context where in all domains, as long as there is a tangible result out of the processes, e.g. bank sector, process industry, and logistic sector. The approach does not say something about the physical internal elements of the system, except when this are processes. Depending on what type of project it is and the scope of the project, more or less interactions have to be taken into account. But it is for sure aimed on processes, where dynamics can play a role also. The basics of Lean Six Sigma are to look at different processes, try to understand the results, formulate the expected behavior and the averages should be. Here there are some uncertainties, in terms of why the range is that wide or what the causes could be of the deviations.

There are some small weak points of this approach, such as the Japanese naming's that people of companies do not want to be associated with. Secondly it is a management technique, forms are developed for it, data needs to be updated and sometimes it can be a large administrative task. Further on the focus lies only on repeatable processes and single companies, while it is less capable to be used in network based contexts with multiple actors and interests.

4.3.5 Process Management

This interview is conducted with Dr. Wijnand Veeneman, professor at the section of Policy, Organization, Law and Gaming (appendix 5). He focuses on two topics: The role of governance in public transport and managing large technical projects. Further he is interested in how managing approaches affect the process management side of a project. He has a social geographical background and did his PhD on the governance of public transport at the faculty of Technology, Policy and Management.

In terms of project management, it starts with defining the boundaries of a project, where the project will be managed with and predict and control approach. At some moment the initial project is split up in small projects with a predict and control approach, but not everything can be coped within this boundaries. This is the point where the process management perspective comes across because keeping things within boundaries does not work anymore.

Process management says that when projects are escalated vertically, a part of the solution space is missed. The problems that are faced in complex project can be solved in two ways: hierarchic control or to look how actors can adjust to each other. The advantage of process management way is that collaboration between parties is made possible, allowing more possible solutions to emerge. On the other hand it weakens the responsibilities between parties, while a more hierarchical approach control people more on doing their work and carrying out their responsibilities. In process management it is not about implementing a substantive solution, but enabling interaction between actors working for a clear solution. Processes could be events that can be used for thinking

about e.g. who have to participate for something, why they have to participate what the exit rules are should everything be covered in one or more rounds.

Some process management literature describes well how to map the actors, by focusing on the dilemmas that exists between the different actors. Managing dilemmas is about controlling in such a way that parties come to a solution, without standing each other in the way. So the starting point of process management is to search for the actors with dilemmas, map the dilemmas, and reasoning from the dilemmas and try to create a solution that a project manager could not have come up with. Process management does not have problems with dynamics in the actor dimension, whereas project management struggles to cope with dynamics. In process management dynamics are used to create solutions, but it does require a lot of alignment between actors. So the dynamics lays in the fact that actors come to a solution, by moving and shifting their positions about certain issues in projects. When there are uncertainties in the actor dimension, process management instruments are necessary to be built in.

Process management can be supported by tools if actors have modeled for example the construction processes which can help analyzing the needed alignment between actors and getting quickly to solutions. Simulating or modeling can help reducing uncertainties and making choices by the involved actors. Process Management books also describe tools that help strengthening commitment or realizing a sense of urgency. If support tools work correctly, they can help to solve the complexity and provide more certainties, but you don't know this on forehand. The application of tools and its success depends on how the tools are used and whether they fit in the environment where they are used. If the complexity of a project is mapped, you can think about the tools that can be used to approach the project. This gives an indication of how complex the project is.

Process management is like project management applicable for just a simple bathroom renovation or a large oil platform installation for Shell. But for sure it is a different story when process management is applied for a small alignment for something that already exists or when process management is used for decision making during the whole project. So, process management is applicable in every context, but not even important everywhere. Sometimes process management can make simple projects unnecessarily complex, but the complexer a project the more freedom a project manager has to build in process management instruments. When there are dynamics in the context dimension, it means that relations between internal elements of the system changes and when there are uncertainties it is unclear how the dynamics will occur and what changes will take place. When dynamics occur and actors are not agreeing with each other, the process management approach tells that interaction between actors has to be enabled.

The shortcomings of process management are that when the created boundaries are reformed, it might bring protest and disagreements. A project must be managed in such a way that it is not controlled by processes,

otherwise the risk will be that the project will be out of its timeframe. So the trick is to find the right balance between project and process management in such a way that they not disturb each other.

4.3.6 Collaboration Engineering

This interview is conducted with Dr. Gwendolyn Kolfschoten, assistant professor at the section of Systems Engineering (appendix 6). Her research field is the support of collaboration processes between actors in different organizations.

The essence of collaboration engineering is to create a process that enables group support for reaching a specific goal. Ideally this process should be repeatable, independent of the group or subject that it is used for. The idea is to use the same process for multiple collaboration projects. Collaboration engineering can hosts frameworks that help to summarize data of the participants, such as goals, characteristics, education, relations with each other and to structure the knowledge of a group. This enables a better view on the participants, which help to create a better collaboration process initially which can be fine-tuned for specific situations. Sometimes the group needs to be focused on the goal if other issues are playing in their mind.

The relations between the actors are highlighted in case there are conflicts between them. For every assessment there is a default approach, which copes with the involved participants, their goals and how to handle this. For example, there was a case where the participants had to do an integrity assessment and they repositioned themselves after a learning moment about integrity. So the dynamic lies in the fact that positions and perspectives are changed in specific situations as not every group is the same as the group of the initial assessment. There are always uncertainties present in every collaboration project, as not everything can be spelled on forehand. The practitioners have limited resources to cope with changes, but the facilitator can deal with changes by making adjustments in the design of the tool. So dealing with uncertainties in the actor dimension lies partly in collaboration engineering, but is not accessible for everyone.

In Collaboration Engineering the support system is described as an electronic meeting system that consists of a network of computers, where every participant has an input device and there is a central device to give feedback. This main station can be used to coordinate the input device, as tools of the input devices or laptops can be enabled or configured. The support tools can cope with dynamics, because they can be configured differently according to the situation in case there are flexible steps necessary. Special knowledge is needed to cope with uncertainties in performance of the support system, which is only possible from the viewpoint of the facilitator.

The context dimension here is the specific organization where collaboration engineering is used for. Collaboration sessions are often about a topic or changing a system, but it does not hold a possibility to change something in the context in real. So, for all the complexity factors of the context dimension, the approach just offers actors the possibility to brainstorm and think about solutions, but not to make changes physically in the context elements. What actors say about the elements of the context is used to fine tune the collaboration session. In case where there are many uncertainties in the context dimension, this information can be used to perform a risk analysis in the session.

One of the weak points of collaboration engineering is that the scope for which it can be used is limited, as it is always about collaboration processes, analysis processes or decision support. The focus is more on the actor dimension, which makes it only useful for projects where all actors have to participate in creating a solution. Collaboration Engineering could be supplemented with tools that help to get more insight in the internal complexity of systems or context, so that it can be used as input for decision making, improvements or changes. Now everything is based on experience of the participants, where results of simulations or analysis of the context system could be used to adjust processes better for each context.

The advantage of collaboration engineering is that it can be used for collaboration between large groups of actors with a specific goal, where other creativity techniques and workshop support are not aimed at a specific goal. For large projects where there are multiple collaboration processes needed there needs to be a general project management approach, where goals, planning and steps are defined. Collaboration engineering in this case is used to tackle a specific part of the complexity and should be integrated in the general project management approach.

4.3.7 Traditional Software Development

This interview is conducted with Dr. Stephan Lukosch, associate professor at section of Systems Engineering (appendix 7). His research is about support systems, improving collaborative design and engineering of complex projects. He has a PhD in computer science and experience in software development approaches.

What is done in the more traditional software development approaches, e.g. the waterfall approach, is that the stakeholders come together in the beginning and write down their specifications for the tool. Normally there are different stakeholders involved in a software project. There is always a prospective customer with requirements, there are software developers who understand the requirements of the customer and there are designers who are responsible for the design of the product. There are no feedback loops or iterations at a later stage of the projects, as a result of what the requirements are not reconsidered. Even though this works for some kind of projects, where the requirements are clear from the beginning and are not subject to change at a later stadium. The

relations between different actors in software development projects are stable, because everyone has a clear role and responsibility. Traditional software developments approaches does not cope with dynamics and uncertainty in the actor dimension, because the process is based on phases and every phase is tackled one after another. The complexity of handling a software product development with an Agile approach has more difficulties, because the relations between the stakeholders have to be taken care off every day.

Considering the support dimension, there is always some kind of modeling language involved to make sure that the product is developed in the end. For example UML is a means of enabling communication between the actors, but the choice of the language depends on what the developers is familiar with. The approaches do not prescribe or oblige to use a specific language, but some kind of formal specification of the product is needed when you work with different actors. The support dimension in software development approaches can be seen as the modeling or the programming language and the developer has the flexibility to choose between several languages. There are certain dynamics in every modeling language, but this is more or less stable for traditional software development approaches. This depends on the requirements of the product, which are specified from the beginning and the programming language is adjusted to it. Uncertainties in the support dimension are not taken into account here, because all the requirements and roles are defined clearly from the beginning.

The complexity of the context dimension depends on what kind of software product is being developed and if physical aspects are considered. The modeling languages can cope with the elements and relations of the context dimension. For example, UML is a modeling language that can be used to describe and visualize different aspects. The context of a project gets more complex if there are a lot of different classes, objects and relations. For the more traditional software development approaches deal less well than Agile methods when it comes to dynamics and uncertainties in the context dimension. But UML, which is used for traditional software development approaches, can help to deal with project complexity in terms of specifying entities, objects, relations and interactions, but it does no tell you anything how to run or manage the project, specify milestones or distribute workloads among different developers. UML can be adjusted to your project context by making adjustments to the modeling language. SYSML is an addition to UML and it can be used to specify whole systems, so for example, a chemical plant could be modeled with SYSML. When dynamics are considered, you could say that the traditional approaches do not allow to cope with it, because there are not iterations. In the waterfall method you just go through the different phases, test the software en delivers it. When you consider a more Agile approach you basically run through every phase every day. Approaches with a shorter iteration cycle cope better with dynamics and uncertainties, meaning that the waterfall method is completely useless when you have to cope with dynamics and uncertainties.

The problem is that software development is basically also about a project that needs to be managed. A software development project is a real project, because your goal is to develop a real product in the end, which you can do on your own or with a lot of other actors, but the project complexity increases. When you know that there will be more dynamics and uncertainties, I would say you need to choose a more Agile approach. Waterfall method has the benefit that the requirements are already specified and the software company says that they are going to deal with the requirements within a certain time and money.

4.3.8 Agile Software Development

This interview is conducted with Dr. Marijn Janssen, associate professor at the department of Information and Communication Technology (appendix 8). He focuses on the design of architectures within organization networks of the government, by proposing changes in the organization and the ICT infrastructure. He is also the director of the master's curriculum, Systems Engineering, Policy Analysis and Management. With a number of integration projects where he has worked on, he provides a view on working with the Agile method in projects.

Most of the time there are several small projects being carried out in governmental organizations, but at a certain time they have to be brought together. For example, a problem in the beginning was that every organization used an own identification service, but the tax service suggested for a standard id, called digiD. The problem in such projects is not how to design such a system, but mainly embedding it in the whole. The environment of the project is ignored or limited by the government very often.

Agile development starts with a small prototype. The risk is that the developers do not think about the requirements on forehand or define the design space in which they can move properly. Developers make choices that not deal with the actors or standards of the organization but on basis of their used technology, meaning that a product is created that does not fit in the organization. Another pitfall is that a data model is created independent from another model, and in the end the models cannot be connected to each other. Agile works fine, but the boundaries around the design process have to be defined properly, because the design has to fit within the framework of the architecture.

Agile Development provides an overview in the involved actors, because a lot of interaction is necessary. So there are interactions and short iterative rounds between actors, after every progress that is made. Actors can be coordinated as Agile does not exclude process management, but the coordination is informal. Sometimes the users are part of the team or the main user provides input to the team. There are no specific deadlines and it is less focused on process support. The dynamics of actors can be coordinated easy, because Agile is used when it is not obvious what the end product will be exactly. A small flexible team is therefore easy to manage, but of course the

boundaries have to be created. So Agile is used when there is a lot of uncertainty, when it is unclear how things work exactly. The goal for using Agile is to set down a design and then look around it, but a key requirement is to work with a small team.

The way Agile deals with the support dimension is an issue of dependent and independent variables. The degree of support is determined by the numbers of actors and context variables, as the problem is leading and let think about resources, actors, finance and support. It is fine to look what type of support there is already available, because the involved actors often use their tools based on tacit knowledge and can bring them in quickly. There is no modeling involved in Agile, the only thing that is known is the requirements. That is why Agile is more something that is used in the preliminary phase, where it is not exactly known what is wanted exactly. So there is high degree of uncertainty, the delineation and system boundaries have to be defined optimal.

The context dimension acts as a guide for the architecture that has to be developed, but the developers have some freedom in it. This depends from the project, because at some projects the developers don't look what there already are and in other projects it is made explicit that the design has to fit in the organization. One of the goals of Agile development is to stay close to the context and the users, meaning that the context is taken into account for development. When uncertainty would mean that there are a lot of actors involved Agile would not work in that situation, but if means that the actors do not clearly know what they want it becomes a different story and Agile can be used. Dynamics in the sense of the context dimension could be external influences, new technologies and it keeps changing during the project. Agile does not prescribe how to deal with changes in the context, but to respond quickly. Uncertainty has to do with proven technology.

There are shortcomings in Agile development, for example, prototypes are developed that not get much appreciation from the users, because the expectations differed too much. So it is important to do good think work and not start immediately agile. The strong side of Agile is that it is very flexible, the developers can respond quickly to the wishes of the users, the system get shape quickly and a large user participation. Still Agile remains a small element of a total package, as it is suitable for small projects, but for large projects it can be incorporated in a project management approach.

4.3.9 Systems Engineering

This interview is conducted with Dr. Alexander de Haan, assistant professor at the section of Policy Analysis (appendix 9). He teaches policy analysis courses, including systems engineering, at the Faculty of TPM. This includes the basic ideas of thinking in terms of actors and factors and thinking in terms of solutions spaces instead of solutions. He has done research in the aviation sector and focuses now more on policy analysis, systems thinking,

systems engineering or other methods of working effectively. He did his PhD at the faculty of Aerospace Engineering and studied psychology in Leiden.

Systems' thinking is a thinking method of how to bring everything together and getting grip on the situation. Systems engineering follows from a system thinking approach, when the problem is considered complex enough that Systems Engineering is a clever method for letting actors communicate in such a way that they can make progress. One actor specifies what another actor has to deliver or has additions to a product, so with systems engineering it is easy to track all change management activities. Systems thinking and systems engineering are not necessarily applied together. When looking at the Actor, Support, Context framework, with systems thinking it would be impossible to leave out any element, because with systems thinking you have to consider everything. Systems Engineering is somewhat more concrete in how it deals with the elements.

Considering the actor dimension, with systems engineering the involved actors and their relations become clear, making it easy for the project manager to distinguish how the actors depend on each other. Dynamics and uncertainty can be taken into account with systems engineering, in case of projects with a large time span, but cannot be defined in advance. In case of a large engineering project, e.g. Delta works, where dynamics in the actor dimension are likely to occur, it is not possible to define this beforehand with systems engineering. However with policy analysis tools like context scenarios it is possible to identify this in advance and analyze possible outcomes, which is not possible with Systems Engineering.

However, in the context dimension, systems engineering takes dynamics into account as it is aimed at reducing uncertainties. All changes are recorded in a retrievable system and can be tracked down in cases actors come up with changing requirements and costs increase. Systems Engineering helps to manage the uncertainty, not to remove it from the system. With Systems Engineering the context of project could be specified more in detail as addition on project management, because it is strong in the context dimension. Context scenarios can help to get a representation of the situation over a certain period of time, caused by the influence of external or internal factors, and help to make more detail decisions gradually during the project. His idea of dynamics and uncertainties is actually scenario thinking. In the context dimension, the systems engineering approach constantly tries to capture the environment in which is operated and how this changes. Scenario analysis and change management are techniques to cope with this dynamics and uncertainties.

Systems Engineering is aimed at making a workable situation and in that sense it can be seen as a support method. The support dimension is one of the elements of a project, in his view and helps to determine the view on practical problems. When the actors are described, one looks already which type of tools will be necessary. Specific tools of systems engineering are causal loop diagrams, system diagrams, actor charts. These are explicit techniques that

help to determine the roles in a project or how to execute tasks. In this sense systems engineering offers nearly a project management like structure, where all the elements on a system level can be described, for example, the construction sector where many software packages for systems engineering have been developed especially since the government have obliged the application of systems engineering.

The obligation from the government can be seen as a drawback in some cases, especially when there is simple choice to be made in projects, now due to this an elaborate specification scheme has to be set up. Another drawback of systems engineering is that it is not described in what situations it can be used. But the biggest disadvantage of the approach is that it does not match culturally with every country, as it designed typically for industrialized countries. It has a typical positive approach to the world, but it does not work all the time. It does not take into account the way people interact or other factors that are important also. But big infrastructural projects cannot be carried out without systems engineering, otherwise it becomes chaotic. Still it remains important to guide the processes. This gap can be filled up with Process management, because it deals well with the richness of human behavior and its complexity. Yet the combination between systems engineering and process management will have difficulties, due to the distance that exists between people and systems engineering. They see it as an analytical and exact tool.

His choice for systems engineering was based because as an engineer he had a narrow view on the world and in systems thinking he found a compromise. It does not ignore complexities and predictability of people, but it starts with a situation where an actor or a group is not content with a situation and tries to create an useful product.

4.3.10 System Dynamics

This interview is conducted with dr. Els van Daalen, associate professor at the section of Policy Analysis, experienced and teaches system dynamics courses (appendix 10). She has a mechanical engineering background and with system dynamics she tries to reduce the complexity of systems or make it more insightful in order to come up with solutions.

The essence of the system dynamics approach is to represent a system in such a way that more insight can be obtained in the system and policies can be formed to solve issues. Getting insight in complexity of a system is one of the intentions of system dynamics, but also to make the complexity manageable in order to formulate solutions.

The actors of a system cannot be mapped as individual actors, but with an actor representation it is more about the actions the actors can take or how many resources there are needed. What actors do is possible to model in a system, but their interests have to be translated into activities, for example: the number of fines that the police spread weakly. Relations between actors can therefore not be modeled in system dynamics and dynamics within

the actor dimension should be translated in shifting interests in order to be modeled. So, uncertainties within the actor dimension could not be modeled, except if the quantified interests of the actors would be subject to a sensitivity analysis.

The system dynamics approach involves 4 different types of software, all with the same essence. There are two phases: a qualitative phase, where text is translated into diagrams and a quantitative phase, where data is translated in stocks and flows. Sometimes a qualitative model is sufficient to model a system, but for a complete system dynamics project both phases are used. System dynamics does not say anything about the number of elements of the support system, because this the elements are a representation of the context of the system. It is defined of which elements each support tool is build up with, but not how many as this depends on the context it is used for. System Dynamics says in terms of support systems which software packages can be used for modeling a system. The support dimensions dynamics and uncertainties are not supported in system dynamics.

With system dynamics the physical elements of the context can be modeled as long as they can be described in levels and flows. In system dynamics insight in a system is understood by making a model and by implementing different solutions. It is all about providing insight in the dynamics of the system, which is the reason that the approach is named system dynamics. So the dynamic is mapped by modeling the elements and relations between them, both for the context and actor dimension as long it is translated in elements and relations. Uncertainties in the context dimension are dealt with by making sensitivity analysis.

When looking at the complexity of a system, it can be said that system dynamics cannot deal well with uncertainties. Originally System Dynamics was designed in such a way that the analyst made the model, performed analyses and presented it to the problem owner. This method did receive little commitment from the problem owner as things seem to work out differently than intended by the model. System Dynamics seemed not is able to deal correctly with conflicting interests. This problem is solved by group model building, by committing every actor in the process. So, collaboration processes are incorporated to cope with this problem.

The strong side of System Dynamics is that the notation is quite simple, easy to understand for people with little experience and policy measures can be taken easily that would be difficult in practice. Scenario analyses and risk management are combined with system dynamics to measure the robustness of measures.

In the next section all interview results are analyzed in order to aggregate useful information and identify starting points for answering the research question.

5. ANALYSIS OF RESULTS

In this chapter an analysis is made of the interview results. First an analysis framework is derived from Capability Maturity Model Integration (CMMI), where levels are presented to analyze the maturity of processes within organizations. These levels are adapted and derived to analyze the interview results, followed by the scores of each management approach on the extent to which it support to map and manage the complexity elements of the ASC framework.

5.1 Analysis framework

The qualitative data from the interview results needed to be analyzed and translated in such a way that it could provide useful information for answering the research questions. CMMI presents a way of categorizing the organizations processes in such a way that organizations can be judged and use this information to improve their processes. This way of representation is adapted to analyze the extent to which management approaches support to map and manage complexity elements of the ASC framework.

5.1.1 CMMI: Maturity Levels

CMMI (Capability Maturity Model Integration) presents collections of best practices that help organizations to improve their processes. CMMI consists of a capability and maturity dimension which are used for benchmarking and appraisal activities, or guiding the improvement efforts (Software Engineering Institute, 2010). Capability levels can be applied to achieve organization process improvements in individual process areas and are a means to improve processes incrementally, within a given process area. On the other hand, maturity levels are applied to gain an organization process improvement across multiple process areas. The levels can be used for improving the processes that correspond with a given set of process areas and are seen as foundation layers for ongoing process improvement.

Table 17 Maturity Levels (CMMI)

Maturity Level	Definition
Level 1. Initial	Processes are usually ad hoc and chaotic.
Level 2. Managed	Processes are planned and executed in accordance with policy, with skilled resources and adequate monitoring and evaluation.
Level 3. Defined	Processes are well characterized and understood, and are described in standards, procedures, tools, and methods.
Level 4. Quantitatively Managed	The organization and projects establish quantitative objectives for quality and process performance and use them as criteria in managing projects.
Level 5. Optimizing	An organization continually improves its processes based on a quantitative understanding of its business objectives and performance needs.

Every level consists of related specific and generic practices for a predefined set of process areas that improve the performance of the organization. The maturity levels are measured by the achievement of specific and generic goals associated with each predefined set of process areas.

The content of each level is not important for this research, but the way of leveling the maturity of processes within organizations. This concept is adapted to formulate a score level for measuring the extent to which management approaches provide means for mapping and managing complexity elements of the actor dimension, support dimension and the context dimension.

5.1.2 Score Levels

The concept of the maturity levels of CMMI is adapted to analyze the results of the interviews. Each level defines the extent to which a management approach support to map and manage the complexity elements of the actor, support and the context dimension (van der Lei et al., 2010). The scores are based on the following scale; level 1, level 2, level 3 and level 4. The primary reason that is chosen for this scale is that the qualitative data has to be decoded in such a way that sensible conclusions can be drawn from it. Secondly it was sometimes hard to get clear information during the interviews or the expert was not always able to mention clearly how it was dealt with the elements. Therefore the scale consists of the levels as described in table 18.

Table 18 Levels for analyzing management approaches

	<i>Level 1</i>	<i>Level 2</i>	<i>Level 3</i>	<i>Level 4</i>
<i>Actors or Elements</i>	Actor or elements are considered important	Contains techniques for mapping actors or elements	Contains also techniques for managing actors or elements	The approach is focused on managing actors or elements
<i>Relations</i>	Important, but contains no means for describing the relations between actors or elements	Contains techniques for mapping relations between actors or elements	Contains also techniques for managing relations between actors or elements	The approach is focused on managing relations between actors or elements
<i>Dynamics</i>	Important, but contains no means for describing the dynamics between actors or elements	Contains techniques for mapping dynamics of actors or elements	Contains also techniques for managing dynamics of actors or elements	The approach is focused on managing dynamics of actors or elements
<i>Uncertainty</i>	Important, but	Contains techniques for	Contains also techniques	The approach is

	contains no means for describing the uncertainties between actors or elements	mapping the uncertainties of actors or elements	for managing uncertainties of actors or elements	focused on managing uncertainties of actors or elements
--	---	---	--	--

Next the scores of the management approaches are presented in terms of the described levels.

5.2 Scores of Project Management

There are several project management methods available, but for this research the PMBOK is selected. As this is the only explicit project management approach, the enhancements are focused on this approach. The scores are derived from the interview and are presented in table 19.

Table 19 Project Management Body of Knowledge scores

Actor dimension	Score Level	Explanation
# actors	3	Stakeholder maps, Resource Activity diagrams
# relations	2	Interface Management
# dynamics	1	Dynamics in actor dimension is not covered
# uncertainty	1	Uncertainty regarding actors is not covered
Support dimension		
# elements	3	In every knowledge area there are specific tools
# relations	1	Their interoperability/interdependency is not described
# dynamics	1	Dynamics of tools is not covered
# uncertainty	1	Uncertainty regarding tool is not covered
Context dimension		
# elements	2	Mentioned briefly in 1.5.3
# relations	1	Elements of context, critical factors not mentioned
# dynamics	2	Mapped via risk management
# uncertainty	2	Mapped via risk management

5.3 Scores of Lean Six Sigma

Lean Six Sigma is a quality improvement approach, rather than a project management approach as described in the literature study. Still the approach is researched on the extent to which it consists techniques to describe and

manage the complexity variables of the different dimensions. The scores that are derived from the interview are presented in table 19.

Table 19 Lean Six Sigma scores

Actor dimension	Score Level	Explanation
# actors	3	Allocate tasks and roles to all actors
# relations	3	Cooperation and coordination
# dynamics	2	Replacements of belts is possible
# uncertainty	1	Excluded, Lean Six Sigma is too rigid for this
Support dimension		
# elements	3	Different types of tools for each situation
# relations	1	Describing relations between tools is not defined
# dynamics	1	Not defined
# uncertainty	1	Not defined
Context dimension		
# elements	3	Described in terms of processes
# relations	2	Interactions in terms of processes
# dynamics	2	Change of processes
# uncertainty	2	Formulate expected behavior

5.4 Scores of Process Management

Process management is focused on managing dilemmas and negotiations between actors. The approach is strong in managing actors, but weak on the support side as support tools are not necessarily involved in process management. The results are derived from the interview and are presented in table 20.

Table 20 Process Management scores

Actor dimension	Score Level	Explanation
# actors	3	Actors with different dilemmas
# relations	4	Managing dilemmas between actors
# dynamics	3	Are used to create solutions
# uncertainty	4	Process management instruments needs to be built in
Support dimension		

# elements	2	Mentioned that tools are useful, but not how to manage them
# relations	1	Not defined
# dynamics	1	Not defined
# uncertainty	1	Not defined
Context dimension		
# elements	3	Applicable in every context
# relations	2	Can be input for negotiation process between actors
# dynamics	2	Can be input for negotiation process between actors
# uncertainty	2	Can be input for negotiation process between actors

5.5 Scores of Collaboration Engineering

Collaboration Engineering is an approach for helping a group effort towards a joined goal. Consequently the approach is strong in the management of the actor dimension. Besides this, the approach defines the way the support system is setup and managed. The elements of the context dimensions can be used solely as input for the actor collaboration processes. The scores are presented in table 21.

Table 21 Collaboration Engineering scores

Actor dimension	Score Level	Explanation
# actors	4	Focus is on involved participants and their goals
# relations	4	The relations are highlighted and conflicts are tried to solved
# dynamics	4	Deals with changing positions and perspectives in specific situations
# uncertainty	2	Described, but cannot be managed during a collaboration session
Support dimension		
# elements	3	Setup before the collaboration session
# relations	3	Interconnected devices and software managed during session
# dynamics	3	Support Software can be configured according to the situation
# uncertainty	2	Functioning depends on third parties, cannot be managed
Context dimension		
# elements	2	Elements of the specific organization are described
# relations	2	Relations are only subject for brainstorming or thinking of solutions
# dynamics	2	Can be mentioned as subject of collaboration process
# uncertainty	2	Can be mentioned as subject of collaboration process

5.6 Scores of Traditional Software Development

There are several Software Development approaches available, but deliberately is chosen for the waterfall approach for the interview, as this is the basis for most of the following software development approaches. It presents a linear way of developing software, where the requirements of the end product are specified in the beginning. The scores are presented in table 22.

Table 22 Software Engineering scores

Actor dimension	Score Level	Explanation
# actors	2	Stakeholders, roles and responsibilities are described
# relations	2	Described from beginning, clear dependencies and relations
# dynamics	1	Not involved, no iterations or feedback loops in the process
# uncertainty	1	Not involved, actors and relations are defined on forehand
Support dimension		
# elements	2	Described in terms of modeling/developing/testing software
# relations	3	In accordance with different phases of development process
# dynamics	1	Stable in traditional software development approaches
# uncertainty	1	Support tools are chosen on basis of requirements in the beginning
Context dimension		
# elements	2	Specified in the requirements, elements the end product consists of
# relations	3	Can be described as objects and classes. Pre specified, but manageable
# dynamics	1	Changes in elements are not considered, requirements are predefined
# uncertainty	1	Uncertainties are not taken into account, no feedback loops

5.7 Scores of Agile Development

Agile methods were developed to deal with a number of problems that the waterfall method inherited. The interview results confirm this more or less, specifically for the actor and context dimension where clearly improvements can be noted. The results are presented in table 23.

Table 23 Agile Development scores

Actor dimension	Score Level	Explanation
# actors	3	Involved actors are described

# relations	3	Their dependencies are managed by interactions and short iterative rounds
# dynamics	3	Managed easy, flexible team
# uncertainty	2	Is taken into account, key requirement is to work in small team
Support dimension		
# elements	2	Described, but based on tacit knowledge of developers
# relations	1	Not described how elements are related
# dynamics	2	Support systems can be subject to change or improvements
# uncertainty	2	Mentioned in terms of uncertainty about which tools are used
Context dimension		
# elements	3	Considered, acts as a guide for the end product
# relations	3	Relations within the context are managed, they are coupled
# dynamics	3	Are considered in iterative rounds, based on the requirements and changes
# uncertainty	3	Applicable in innovative settings

5.8 Scores of Systems Engineering

Systems Engineering is a clever method for letting actors communicate in such a way that they can make progress, it helps to reduce uncertainties and making a workable situation for every actor. Systems Engineering is strong in the handling the context dimension as the goal is to reduce uncertainties by specifying all the requirements on forehand. The results are presented in table 24.

Table 24 Systems Engineering scores

Actor dimension	Score Level	Explanation
# actors	3	Actors are clear, everyone has clear goals
# relations	3	Specifies how actors depend on each other, change management
# dynamics	1	Not possible to define in advance
# uncertainty	1	Not possible to define in advance
Support dimension		
# elements	2	Described in terms of different available software packages
# relations	1	Interdependencies not described
# dynamics	2	Support systems are subject to context (e.g. construction sector), dynamic character
# uncertainty	2	Developed by third parties, or innovative tools have to be developed based on the context of the project

Context dimension		
# elements	3	Provides a means of describing the context elements specifically
# relations	2	Relations between different elements can be described
# dynamics	3	Captured by change management and scenario analyses
# uncertainty	4	The goal of SE is to manage uncertainties about the context dimension

5.9 Scores of System Dynamics

System Dynamics can be used to reduce the complexity of systems by getting insight in the dynamics and adjusting the solutions accordingly. In the scores, presented in table 25, it is clear that System Dynamics is strong in the context dimension.

Table 25 System Dynamics scores

Actor dimension	Score Level	Explanation
# actors	2	SD translates the goals of the actors into activities
# relations	1	Relations between actors cannot be mapped with System Dynamics
# dynamics	1	Not considered, except for modeling changing interests in terms of new goals
# uncertainty	1	Unconsidered, except if the quantified interests of the actors would be subject to a sensitivity analysis
Support dimension		
# elements	3	Specially developed software packages
# relations	2	Description of elements the software consist of
# dynamics	1	Not considered
# uncertainty	1	Not considered
Context dimension		
# elements	3	Modeled in terms of levels and flows. Focus is to manage these levels and flows
# relations	2	Relations are modeled in diagrams
# dynamics	3	The approach is focused on providing insight in the dynamics of a system
# uncertainty	4	Manage uncertainties, by conceptualization, simulation and sensitivity analysis

The results provide the basis for combining techniques that are strong in handling complexity elements to enhance the PMBOK. A literature verification of the scores in PMBOK and the high scores of other management approaches will be performed at a later stage in this research, after the possible combinations are identified. The results are mostly in the line of expectation that was suggested after the comparative analysis in section 2.

In the next chapter the analyses of the results will be plotted in order to get an overview of how the project management approach could be supported by the other approaches for dealing properly with the complexity elements of each dimension.

6. SCORES OVERVIEW PER DIMENSION

The goal of this research is to present techniques from other management approaches to enhance project management. Therefore, the extent to which the management approaches support describing and managing the different complexity elements is presented in graphical forms for each dimension. This provides a better view on the strengths and weaknesses of the PMBOK against the other approaches.

6.1 Actor Dimension

The scores of the management approaches on the complexity elements of the actor dimension are presented in figure 8.

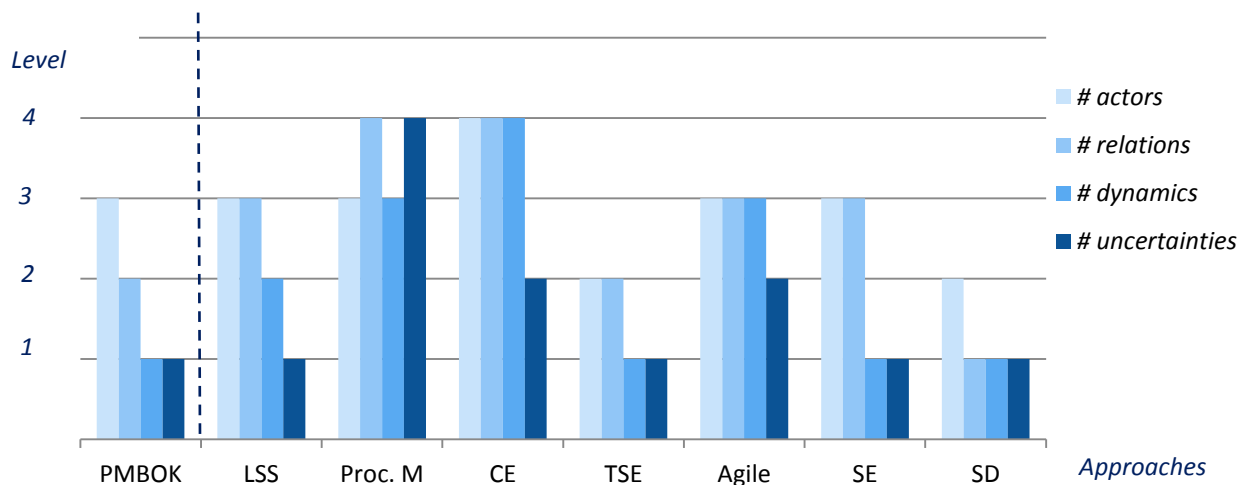


Figure 8 Scores of management approaches on actor dimension

Based on the graph, the following conclusions can be drawn on the way the actor dimension is handled in the management approaches. Every approach presents a mean for describing the actors that are involved in a project, but not every approach presents a technique to manage the actors. The relations between actors can be described in every approach, except Systems Dynamics. Describing relations between actors is only dealt in Lean Six Sigma, Process management, Collaboration engineering and Agile approaches.

The PMBOK scores low on management of relations between actors, dynamics of actors are not described and uncertainty of actors is also overlooked. Possible techniques that can be used to enhance these specific points can be derived from Process management, Collaboration engineering and Agile approaches. In chapter 7 specific techniques from these approaches for mapping and managing actors are verified with the literature and suggestions to enhance the possible issues in PMBOK are mentioned.

6.2 Support Dimension

The scores of the management approaches on the complexity elements of the support dimension are presented in figure 9.

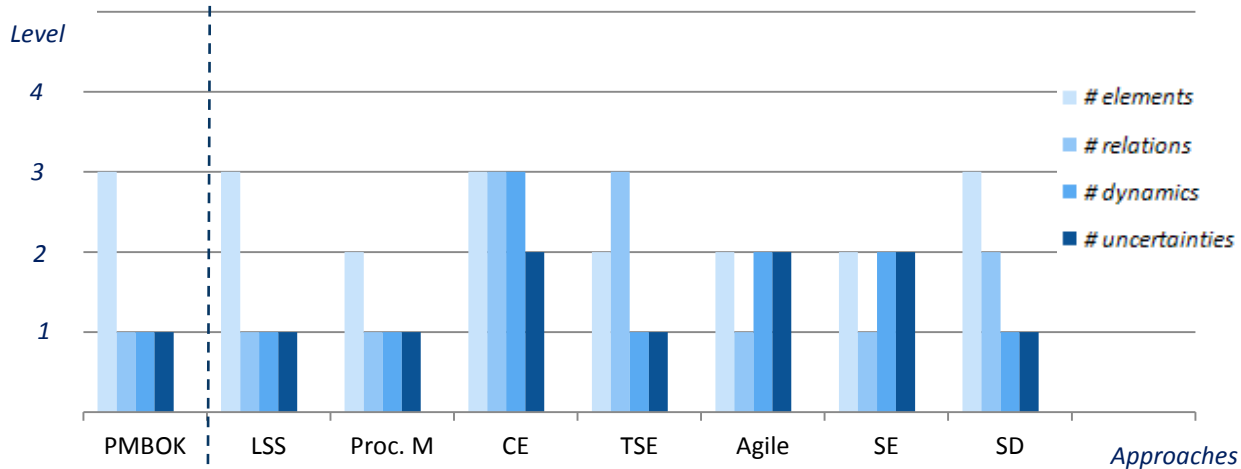


Figure 9 Scores of management approaches on support dimension

The PMBOK contains no means of managing the elements of the support system. Here it can be concluded that the way the support system is described is very inflexible, as it is only described which tools can be used for each section. In general, all the other approaches score very low in describing and managing the complexity elements of the support dimension. This could mean that not much effort is spent in developing tools for the different approaches or the mentioned tools are static.

All the approaches describe the necessary tools, but mostly the elements and relation between the components that the tools are build up with, are not mentioned. This could be caused that most tools are not modular and flexible. Therefore they are also less capable of coping with dynamics and uncertainties. This means that there are not much possibilities for making combinations on the support dimension. However, in the next chapter, first a verification will be made on the available tools in PMBOK, followed by the identification of possible issues and suggestions to enhance them.

6.3 Context Dimension

The scores of the management approaches on the complexity elements of the context dimension are presented in figure 10.

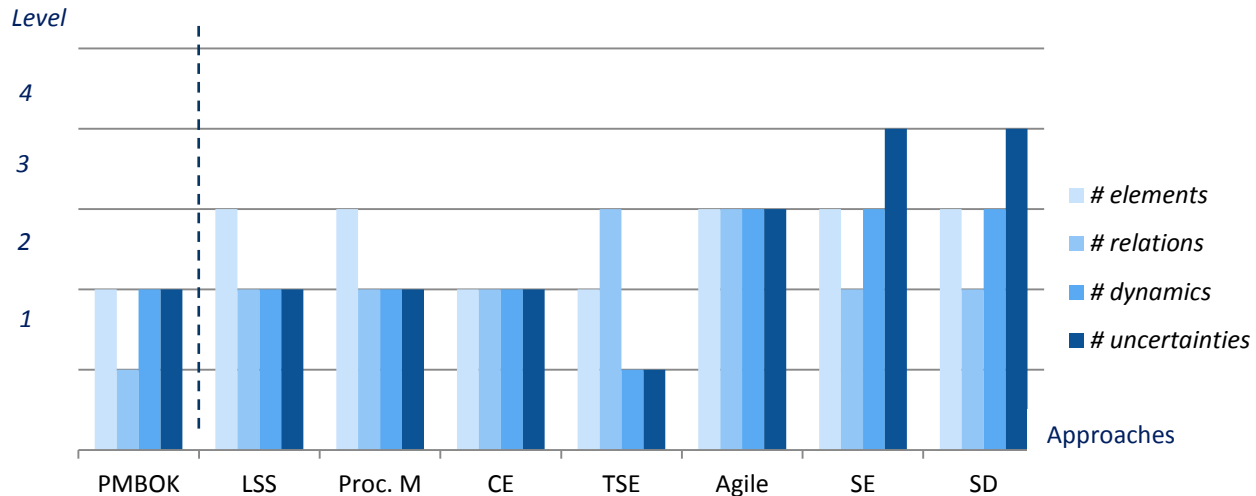


Figure 10 Scores of management approaches on context dimension

The scores on the context dimension shows that the PMBOK, scores low on the context dimension. The score means that the approach does provide techniques for describing the elements of the context dimension, but the management is overlooked. However, there are some approaches that are strong in management of the complexity elements of the context dimension, like Agile development, Systems Engineering and System Dynamics. Thus the graph provides a clear direction to look for searching techniques for dealing with the complexity elements of the context dimension and enhancing the project management approach, where necessary. But first a literature verification will be conducted in the next section, to identify possible weak points and following enhancements.

In the next, some possible enhancements for PMBOK on basis of the insights that the above presented graphs provide are mentioned.

7. ENHANCEMENTS FOR THE PROJECT MANAGEMENT BODY OF KNOWLEDGE

In this chapter some possible techniques for enhancing the project management approach, PMBOK, in terms of dealing with the complexity elements are presented. This is the result of what the experts mentioned about the different dimensions and a verification on the literature to find out how the possible combinations can be achieved. First an elaboration is made on the actor dimension, followed by the support dimension and the context dimension.

7.1 Actor Dimension

As was concluded from the expert's interviews, the PMBOK is limited in the way it supports the involvement of actors. In the following subparagraphs the shortcomings of these approaches are considered, followed by some suggestions for using techniques from approaches that are strong in actor management.

7.1.1 Verification of actor management in PMBOK

The actors that are involved in a project can be divided in project members and staffing that are involved in initiating, planning, executing, monitoring, controlling and closing a project, and at the other hand the stakeholders (PMI Institute, 2004). The way that is dealt with actors is described in Human resource management, where is described how to deal with human resources of the project and in project communications management, where is described how to deal with the involvement of project stakeholders.

Analysis of stakeholder management

Project stakeholders are individuals and organizations that are actively involved in the project and their interests can be affected by the results of the way the project is carried out or completed. They have the ability to influence projects objectives and outcomes. The project management team has to identify the stakeholders, determine their requirements and try to balance the influence of the stakeholders in relation to the requirements of the project. Stakeholders can be involved in different ways in projects, from occasional contributions to full project sponsorship including financial and political support. Project managers who ignore or not identify stakeholders can expect a damaging effect on project outcomes. Positive stakeholders benefit normally from the outcomes of a project, while negative stakeholders experience negative effects from the project execution or outcomes. Negative stakeholders can demand more reviews during the project, because they want their interests to be served better. However, these stakeholders are often overlooked by the project team and in some cases the project cannot be successfully finished.

Stakeholder Management refers to managing communications to satisfy the needs of and resolve issues with project stakeholders.

Table 26 Stakeholder Management in PMBOK

Activity	Explanation	Tools	Goals	Problems
Map stakeholders	Identify stakeholders	Stakeholder map	Understanding of stakeholder goals and objectives and influences	Subjective expectations entail a high risk of being successfully accomplished
Manage stakeholders	Determine requirements and expectations	Communication methods, face to face meetings, electronic tools	Satisfy the stakeholders	Negative stakeholders can form an obstacle
Map relations	Positive and Negative stakeholders	Stakeholder map, Communications Management Plan	Identify influences of stakeholders	Hidden Agendas, alliances, informal power
Manage relations	Manage their influences	Conflict Management	Maintain good, constructive working relationships among various stakeholders	Unresolved issues can be a major source of conflict and project delays
Map dynamics	To get insight in changing perceptions of stakeholders	Create environment where stakeholders can contribute appropriately	Get insight in changing positions of stakeholders	Limited till the planning effort, feedback and refinement process stops
Manage dynamics	Manage changing perceptions of stakeholders	-	-	No technique defined
Map uncertainty	Identify unexpected behavior, trust problems, hidden agendas	-	-	No technique defined
Manage uncertainty	Cope with uncertain issues	-	-	No technique defined

Analysis of human resource management

Project human resource management includes the processes that organize and manage the team. The role describes the portion of a project that a resource is accountable of, while responsibilities contain the work that is expected to be performed by a member. Managing a project team is the process that is necessary for tracking team performance, give feedback, coordinate changes and resolve issues in order to secure and enhance the project performance. The internal actors that are necessary to carry out the project are divided in different project management process groups, which are dependent from each other.

Table 27 Human Resource Management in PMBOK

Activity	Explanation	Tools	Goals	Problems
Map resources	Obtaining the human resources needed	Human resource planning	Roles and responsibilities	
Manage resources	Determine requirements and expectations	Develop project team, timetables, leadership	Improving the competencies and interaction of team members	Logistical problems, distance between resources
Map relations	Relations between project team members	Project organization charts, Process flow diagrams	Position descriptions and basic flow and interaction among process groups	
Manage relations	Process groups are linked by the objectives they produce	Conflict Management, Group Interactions, Negotiation Management	Greater productivity and positive working relationships	Overlapping activities that occur at varying levels of intensity throughout the project
Map dynamics	Type and number of project team members can often change as the project progresses	Observation and conversation, performance reports	To stay in touch with the work and attitudes of project team members	
Manage dynamics	Can occur due personal work styles,	Networking, motivation,	Guarantee quality	Remote locations

	scheduling priorities	collaboration		
Map uncertainty	If project team members do not possess required competencies	Establish clear expectations regarding acceptable behavior by project team members	Identify mismatches	Performance can be jeopardized
Manage uncertainty	Cope with uncertainties regarding behavior, trust, cultural differences	Training, hiring, schedule changes, scope changes are initiated, team building activities	Enhance competencies of members	Differences in competencies

In the next paragraph some techniques of the approaches that score high in the actor dimension are presented, which can help to solve the issues or guide project managers into the right direction.

7.1.2 Enhancements on Actor Dimension

The possible issues of the actor dimension and techniques from approaches that are strong in actor management are linked to the different issues in figure 11. The techniques were achieved by a brief verification of the literature, but provide useful insights for addressing issues in the actor dimension from a project management perspective.

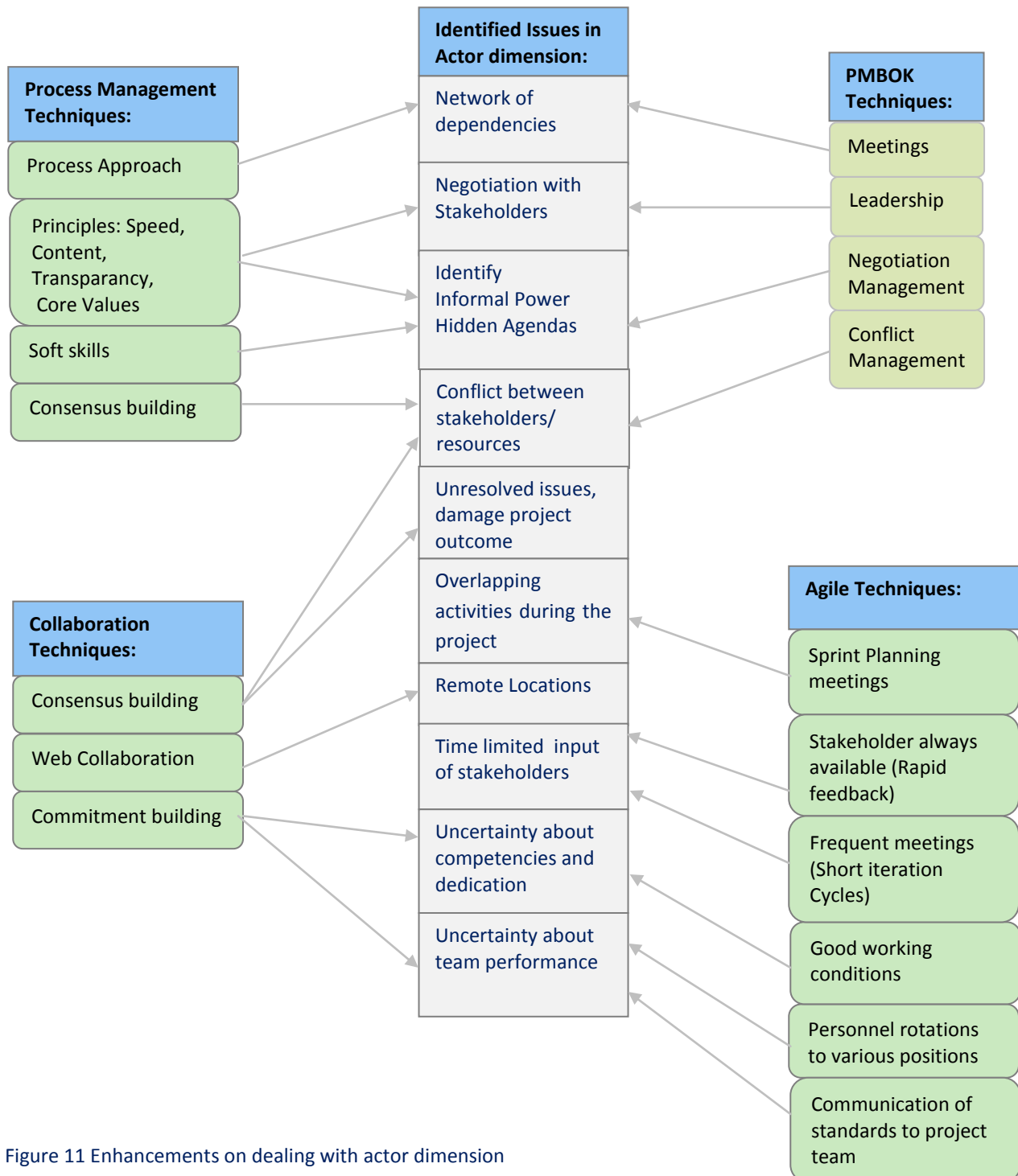


Figure 11 Enhancements on dealing with actor dimension

Next the techniques that are presented in the figure are briefly elaborated.

Process Management techniques

In most project a network of actors is involved, who are mostly dependent of each other. Project Management approaches have a limited meaning in a network of dependencies. So, a process approach is more appropriate for dealing with a network of dependencies between actors. Process designs can be made for negotiation rounds between stakeholders, but the four principles of process management; speed, content, transparency and protection of core values have to be taken into account. Further some soft skills are required to negotiate with stakeholders and to get them in the right direction.

Collaboration Engineering techniques

Collaboration techniques can help to create consensus between stakeholders, meaning that a state is reached where all stakeholders are willing to commit to a proposal or solution. Commitment building can be used for letting a group of mission-critical stakeholders arrive at mutually acceptable agreements. Another issue occurs when team members are not in a face to face setting, but collaboration is needed. Therefore web collaboration techniques are proposed, creating a virtual work setting where team members can collaborate and cooperate from distance.

Agile Development Techniques

A sprint planning meeting can be used to prioritize tasks for the team. During this meeting the team can ask questions and afterwards prioritize tasks. One of the problems of PMBOK was that the feedback of the stakeholders was limited till the project planning session. After all the stakeholders value have to be maximized, thus Agile methods propose that the stakeholders should be available frequently and closely consulted for feedback and understanding requirements. Uncertainties about team performance can be addressed by creating good working conditions, rotating personal to various positions and communicating project standards clearly to the team.

7.2 Support Dimension

The scores of the support dimension indicate that the approaches in general score low in managing relations between the elements together with the dynamics and uncertainties of the support dimension. The same applies for the project management approach, PMBOK, which provides only a means of mapping the tools and their purposes. But Collaboration Engineering and the Waterfall approach provide a means for managing the elements of the support dimension, so possible techniques can be borrowed from these two approaches. But first it is necessary to find out and make an inventarisation of the available support tools in PMBOK, particularly software and information systems. It is important to denote that it is not the intention to combine the support tools, but the techniques that help to map and manage the complexity elements of the support dimension.

7.2.1 Verification of support tools in PMBOK

The support tools that are selected can be technological systems or software (Pmroadtrip, 2011). Simple support charts, guidelines, techniques are not considered further in this verification.

Table 28 PMBOK support system tools

Tools	Purpose
Communication tools and technologies	Distribute information, Manage stakeholder expectation, Report performance
Facilitated workshops	Collect requirements, Define Scope
Forecasting systems	Control Costs
Group decision making techniques	Collect Requirements
Information gathering and distribution tools	Identify risks
Internet search	Conduct procurements
Payment systems	Administer procurements
Quality control tools	Perform quality assurance
Estimating software	Estimate costs
Project Management Information System (PMIS)	Direct and manage project execution
Project Management software	Control Schedule and Costs
Scheduling tool	Develop project schedule

The support tools that are mentioned in the table form the elements of the total support system that is necessary for managing projects based on the PMBOK. According to the Actor, Support, Context complexity framework, these tools have relations and are subject to dynamics or possible uncertainties.

Each support tool has a specific purpose, but is interrelated in some way to each other. Some of the tools provide the input for another tool or system, e.g. forecasting systems could be used as input for estimation software. It is likely that the Project Management Information System, which is in general intended to direct and manage project execution, should be linked with the other support tools. A verification with the PMBOK guide (PMI Institute, 2004) points out that the tools are mentioned for each knowledge area, but not specified how they are connected with each other.

7.2.2 Enhancements on the Support Dimension

The support system as described in the PMBOK guide is limited in two ways. First, the relations between the elements are not elaborated and dynamics and uncertainties regarding situations where innovative tools are needed are not considered. This could mean that the tools that are mentioned are designed and used statically. In this sense a simple diagram of how the tools are related to each other would be helpful, because it makes it easier to identify bad performance of tools and replacing them or to adjust and select the needed support tools according to the project that has to be carried out.

The second aspect is that it is not described how the tools have to cope with the complexity elements of the other dimensions. This is an important issue that has to be addressed, because the support dimension is seen as the link or the dimension that enables communication between the actor and the context dimension. Further the experts denote the importance of the support tools, but more in the sense of how they could facilitate or support the activities of the actor dimension and the context dimension.

The following examples elaborate that no further emphasis is put in how the tools support the complexity elements of the actor and context dimension.

- Communication tools and methodologies are proposed for the distribution of information and managing stakeholder expectation, but the way the tool has to cope with changes is not taken into account.
- Group decision making techniques are proposed to collect requirements, but it is not described how the tool supports commitment or consensus building.
- Information gathering tools are required for identifying risks, but as risk management is of great importance more detailed execution of this part is required. So some risk analysis and management tools should be developed.
- It is not stated how stakeholders (actor dimension) can have insight in project information and progress, which should enable feedback from them. The Project Management Information System should have a functionality for this.
- Scheduling tools are mentioned. However, a small project could be completed with MS Project, but for larger projects tools as Primavera are necessary.

It can be concluded that not much emphasis is put down on the development of tools in project management, in particular the PMBOK, in such a way that the tools are able to cope with relations, dynamics and uncertainties. The scores provided that nearly every approach has tools that are weak in handling this, except Collaboration engineering where a central device and software is used to adjust the input from individuals, making it easier to cope with dynamics and uncertainties.

Tools have to be developed in such a way that functionalities can be added or connections with other tools are possible in a later stage. An example of a support system that is built up in a modular way is Blackboard, in which functionalities can be added, e.g. for enabling communication between users.

Another example of a tool that deals with dynamics is SharePoint, developed by Microsoft. The idea behind this tool is that nowadays highly collaborative, mobile and virtualized work styles are expected by people and the business. In this way people can work simultaneously by sharing documents, data and information with colleagues, partners and suppliers. Further they can respond accurately and quickly with information that is acquired from experts across the organization.

7.3 Context Dimension

The results that were drawn from the expert interviews on the context dimension point out that the PMBOK has a low score on managing the elements of the context dimension. On the other hand Agile Development, Systems Engineering and System Dynamics score high in managing the complexity elements of this dimension. Next a brief verification follows on the context dimension according to the PMBOK, followed by some techniques from the other approaches that could be helpful.

7.3.1 Verification on the context dimension in the PMBOK

Each of the nine knowledge areas support to get insight in the context in which the project takes place, but do not provide a means for describing and handling the elements of the context. When elements are effected through interfaces, no means are provided to cope with the dynamics that occur. Further, in innovative settings where is worked with unproven technology and potential risks exist and there is a high level of uncertainty, the PMBOK does not describe how to deal with this. The critical success factors or drivers of a project could be derived from the context dimension, but it is not mentioned how this is covered. Risks can be covered via risk management, but when changes have to be made during the project, some feedback loops are missing.

7.3.2 Enhancements on the context dimension

Agile Development methods assume a largely emergent, rapid change context, while plan driven methods such as the PMBOK, assume a stable context where the context requirements are predefined. However, in a project it is

likely that dynamics will occur and requirements will change during the project, meaning that an Agile approach is preferred. In this way dynamics in requirements or uncertainties could be discussed frequently and rapid adjustments can take place on time. Another technique that is used in software development, including Agile Development is the use of modeling languages, in particular UML. This exist largely of drawing class and object diagrams and relating them to each other. Further a decomposition of classes can be made, so that it becomes clear of what (sub) elements a system should exist. In projects this is a helpful tool to get insight in the elements of the context dimension and their relations.

Systems engineering scores high on the management of dynamics of the context dimension, and is focused on managing uncertainties in this dimension. This is achieved by change management and bringing all the requirements together from the viewpoint of different actors. Risk sessions are organized, and actors are in touch with each other about the requirements of the project. The systems engineer has a specific task of enabling cooperation between the different actors and information flow about necessary requirements. In this way all requirements are taken into account and dynamics are not overlooked, because the input of every actor during different phases of the project is processed.

System dynamics is aimed at providing insight in the dynamics of a system. Relations between the context elements are modeled in diagrams and uncertainties are managed by conceptualization, simulation and sensitivity analysis. This way of working could be used to gather insight in dynamics of the context in a project, where elements could be modeled in terms of levels and flows and relations described via diagrams. By this the behavior of context elements or (sub) systems that can influence the project can be analyzed and addressed properly.

8. CONCLUSIONS

Before presenting the conclusions of this research first a short recap is given of the problem definition, the research question and the steps that were taken to answer the research question.

“How can project managers combine tools and techniques from project management approaches and discipline oriented approaches for managing multidimensional complex projects successfully?”

First a comparative analysis is made on the different approaches based on the five ways model. This provided insight in how the approaches differed in terms of the way of thinking, the way of control, the way of modeling, the way of working and the way of support. This pointed out that the PMBOK was the only approach with a project management structure and the goal to manage a project. Lean Six Sigma was qualified as a quality improvement approach, while Process management and Collaboration engineering were aimed at enabling cooperation and collaboration between actors. The Waterfall approach and Agile Development were particularly developed for managing software project, but they differ largely in their methodology. The waterfall approach proposes a very linear way of working, where no feedback or iterations are possible during development, while Agile Development propose short iteration cycles and feedback of the stakeholders. Systems Engineering and System Dynamics provide a means of getting insight in the context of a project, specifically by requirement analysis and simulations.

Secondly, it was necessary to find out how complexity plays a role in projects. The framework that was chosen to analyze how different management approaches deal with complexity elements consists of an actor dimension, a support dimension and a context dimension. To gather data on how the different management approaches scored on the complexity elements, interview research was conducted with 8 experts of the faculty of Technology, Policy and Management.

The results that were gathered provided insight in the scores of how the different approaches scored on the different dimensions. It was clear that the project management approach scored low on every dimension, meaning that enhancements were necessary. Therefore it was necessary to look at how other approaches scored on the complexity elements of the different dimensions. Some of the approaches have a high score and provided techniques to enhance project management.

Actor Dimension

PMBOK scored low in managing relations, dynamics and uncertainties of the actor dimension and it was clear that process management and collaboration engineering scored high, together with Agile development which was surprising. A process approach is proposed to incorporate in project management, as it becomes more and more an environment with a network of dependencies between actors. Therefore negotiation processes between actors

have to be made possible, taking into account four design principles of processes. These are speed, transparency, content and protection of core values of the actors. Collaboration Engineering is proposed for facilitating group interactions, negotiations and consensus building among actors. Web collaboration can be used for enabling group interaction when people operate from remote locations. Agile techniques include short iteration cycles and feedback from stakeholders.

Support Dimension

The selected approaches scored low in general in the way they handle the complexity elements of the support dimension. This means that the elements of the support system, which can be the necessary software or information systems, are described briefly but not elaborated in terms of how they are related or interdependent of each other. Further no emphasis is laid in the dynamics of the support systems, which means that support systems are subject to change, design, development or improvement. Most of the approaches assume a static functioning of the support systems, as a result of what the tools are not able to cope with dynamics and uncertainties of the actor dimension and the support dimension. To enhance the PMBOK approach in dealing with dynamics a modular structure of software is proposed, where functionalities can be added if necessary.

Context Dimension

Systems Engineering and System Dynamics score high on handling the complexity elements of the context dimension. As proposed, techniques from these two approaches can be used to enhance the project management approach in managing the complexity elements of the context. Taking into account all specific requirements, splitting up the context in different (sub) systems can help to prevent that the elements of the context are overlooked. Finally modeling, according to system dynamics, can provide in system behavior. In this way project managers can gather insight in the influence of (sub) systems of the context on the project and adjust proper measures to it.

9. REFLECTION

This chapter consists of the limitations of this research. Some limitations were caused by the boundaries in which the research is carried out, while others occurred during the different phases of the research. Therefore based on these limitations some future work has to be carried out to elaborate further on the outcomes.

9.1 Limitations

The research was constrained by some limitations, as time and the availability of experts during the research. Also, only a selection of management approaches was taken into account, where there was only one explicit project management approach included. Next, it was difficult to abstract useful data out of the results of one interview per management approach. It was deliberately chosen to interview experts of the Faculty of Technology, Policy and Management. Interviews in the practice field were not performed, because this research can be seen as a step in the direction of enhancement of project management and based on theoretical foundations. However, some experts had practice experience and could provide extra useful information about how the approach is used in practice, while for others the management approaches were more part of their research or education field. Another limitation was that the interviews were only based on the ASC framework, which was sometimes criticized during the interviews. But the ASC framework has a generic character and is generalizable to other fields also.

9.2 Future Work

Based on the mentioned limitations, future research on this topic should take the following points into account:

- Use another set of management approaches.
- Include more explicit project management approaches.
- Conduct interviews with experts who practice the management approaches in the field.
- Try to gather quantitative data, which can support the findings
- Validate results in practice
- Conduct a specific study for e.g. the construction sector and base the interviews on another framework, e.g. the TOE framework.

REFERENCE LIST

- [1] Agile Alliance, 2011. Manifesto for Agile Development. www.agilealliance.org, accessed November 2010.
- [2] Arthur, J., 2007. Lean Six Sigma Demystified, A self-teaching guide. McGraw-Hill Companies.
- [3] ASQ The Global Voice of Quality, 2011. Overview, Continuous Improvement. <http://asq.org/learn-about-quality/continuous-improvement/overview/overview.html>, accessed December 2010.
- [4] Baccarini, D., 1996. The concept of project complexity: A review. *International Journal of Project Management*, 14(4), 201–204
- [5] Boehm, B., 1988. A spiral model of software development and enhancement. TRW Defense Systems Group, Computer, vol. 21/5.
- [6] Bosch-Rekvelde, M., Jongkind, Y., Mooi, H., Bakker, H., Verbraeck, A., 2010. Grasping project complexity in large engineering projects: The TOE (Technical, Organizational and Environmental) framework, *Int. Journal Project Management*.
- [7] Briggs, R., Kolfchoten, G., Vreede G-J, Douglas, D., 2006. "Defining Key Concepts for Collaboration Engineering". *AMCIS 2006 Proceedings*. Paper 17.
- [8] Business Dictionary, 2011. Project Definition. <http://www.businessdictionary.com/definition/project.html>, accessed January 2011.
- [9] De Bruijn, H., ten Heuvelhof, E., in t Veld, R., 2010. *Process Management: Why Project Management Fails in Complex Decision Making Processes*, Second Edition. Springer-Verlag Berlin Heidelberg.
- [10] Engels, G., Groenewegen, L., 2000. Object-Oriented Modeling: A Roadmap. The 23d International Conference on Software Engineering. Presentation, June 2000.
- [11] Extreme Programming, 2011. Extreme Programming: a gentle introduction. www.extremeprogramming.org, accessed January 2011.
- [12] Faculteit Techniek, Bestuur en Management, 2011. Afdelingen, Section Systems Engineering. <http://www.sk.tbm.tudelft.nl/>, accessed March 2011
- [13] Fisher, R., Ury, W., Patton, B., 1991, *Getting to Yes, Negotiation agreement without giving in*, second edition, Harvard university. Penguin Group Publisher.
- [14] Flyvbjerg, B., Bruzelius, N., Rothengatter, W., 2003. *Megaprojects and Risk. An Anatomy of Ambition*. Cambridge University Press, Cambridge.
- [15] Gubrium, F., Holstein, J., 2001. *Handbook of interview research: context & method*, Jaber, Sage Publications, Thousand Oaks.

- [16] Herder, P.M., Stikkelman, R.M., 2004. 'Methanol Based Industrial Cluster Design: A study of Design Options and the Design Process'. *Industrial & Engineering Chemistry Research* Volume 43, pp. 3879 - 3885
- [17] Hevner, A., March, S., Park, J. and Rahm, S., 2004. 'Design science research in information systems', *Management Information Systems Quarterly*, Vol.28
- [18] Highsmith, J., 2000. *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*. Addison Wesley
- [19] Highsmith, J., Cockburn, A., 2001. "Agile Software Development: The People Factor", *IEEE Computer*, Society Press Los Alamitos, vol.34 Issue 11, November
- [20] Hoffer, J.A., George, J.F., Valacich, J. 2002. *Automated Tools for Systems Development, Modern Systems Analysis and Design Third Edition*, Prentice-Hall Inc.
- [21] Hlupic, V., Quereshi, S., 2002. What causes value to be created when it did not exist before? A research model for value creation. In: *Hawaii International Conference on System Sciences*. IEEE Computer Society Press, Los Altos
- [22] Huo, M., Verner, J., Zhu, L., Babar, M., 2004. "Software Quality and Agile Methods," *compsac*, vol. 1, pp.520-525, 28th Annual International Computer Software and Applications Conference (COMPSAC' 04)
- [23] IEEE Standards Association, 2011. 1220-1998 - IEEE Standard for Application and Management of the Systems Engineering Process. <http://standards.ieee.org/findstds/standard/1220-1998.html>, accessed January 2011.
- [24] Incose, 2000. *International Council on Systems Engineering, Systems Engineering Handbook, A how to guide for all engineers*. Version 2.0, Incose
- [25] Kolfschoten, G., Briggs, R., de Vreede, G-J., Jacobs, P., Appelman, J., 2006. A conceptual foundation of the thinkLet concept for Collaboration Engineering *Int. J. Human-Computer Studies* 64 (2006) 611
- [26] Kvale, S., 1996. *Interviews: An introduction to qualitative research interviewing*. Thousand Oaks, CA: Sage
- [27] Larman, C., 2004. *Agile & Iterative methods: From business case to successful implementation*, A Managers Guide, Pearson Education, Boston.
- [28] Luoma, M., 2006. A play of four arenas: How complexity can serve management development. *Management Learning*, 37(1), 101–123.
- [29] Manson, S.M., 2001. 'Simplifying complexity: a review of complexity theory', *Geoforum*, Vol. 32, pp.405–414.
- [30] Maylor, H., Vidgen, R., Carver, S., 2008. Managerial complexity in project based operations: a grounded model and its implications for practice. *Project Management Journal* 39, S15–S26 Supplement.
- [31] Mikulecky, D.C. (2008). 'Definition of complexity', Medical College of Virginia Commonwealth University; <http://www.people.vcu.edu/~mikuleck/cmpxnat.html> (October 2010).

- [32] NASA, 2007. NASA Systems Engineering Handbook, National Aeronautics and Space Administration NASA Headquarters Washington, D.C. 20546
- [33] Nunamaker, J.F., Dennis, A.R., Valacich, J.S., Vogel, D.R., George, J.F., 1991. Electronic meeting systems to support group work. *Communications of The ACM* 34 (7), 40–61.
- [34] Pavard, B., Duggdale, J., 2000. 'The contribution of complexity theory to the study of socio-technical cooperative systems', *InterJournal of Complex Systems*, Vol. 335, New England Complex Systems Institute, Cambridge, MA.
- [35] Prince 2, 2011. What is Prince 2? <http://www.prince2.com/what-is-prince2.asp>, accessed December 2010.
- [36] Project Management Hut, 2011. Six Sigma vs. Total Quality Management, <http://www.pmhut.com/six-sigma-vs-total-quality-management>, accessed January 2011.
- [37] Project Management Road trip, 2011. PMBOK process index. http://www.pmroadtrip.com/pmpv4_r01.html, accessed April 2011.
- [38] Parwani, R., 2002. Complexity: an introduction. Singapore: National University of Singapore
- [39] Project Management Institute, 2004. A guide to the Project Management Body of Knowledge (PMBOK Guide) fourth edition, Newtown Square, Pennsylvania
- [40] QFD Online, 2011. House of Quality (QFD tutorial). <http://www.qfdonline.com/qfd-tutorials/house-of-quality-tutorial/>, accessed January 2011.
- [41] Sawhney, R., 2010. Lean Six Sigma growth Presentation. Department of Industrial and Information Engineering, http://www.cis.tennessee.edu/library/Solutions_conference/sawhney.pdf
- [42] Scrum Alliance, 2011. Scrum Is an Innovative Approach to Getting Work Done, www.scrumalliance.org/pages/what_is_scrum, accessed January 2011.
- [43] Seligman, P., Wijers, G., Sol, H., 1989. Analyzing the structure of I.S. methodologies; an alternative approach. In: R. Maes (ed) *Proceeding of the First Dutch Conference on Information Systems*, Amersfoort
- [44] Senge, P., 1994. *The Fifth Discipline: The Art & Practice of the Learning Organization*. Doubleday Business, 1st edition.
- [45] SigmaPro, 2011. Training, certification and consulting. www.sigmapro.com, accessed January 2011.
- [46] Sol, H.G., 1983. "A Feature Analysis of Information Systems Design Methodologies: Methodological Considerations", Amsterdam, The Netherlands.
- [47] Sommerville, I., 2000. *Software Engineering*, sixth edition.
- [48] Software Engineering Institute, 2010. Improving processes for developing better products and services, technical report v. 1.3. CMMI Product Team. Carnegie Mellon University

- [49] Software Engineering Insitute|Carneggie Mellon, 2011. Capability Maturity Model Integration (CMMI), Overview. <http://www.sei.cmu.edu/cmmi/>, accessed December 2010.
- [50] Strauss, A, Corbin, J., 1998. Basics of Qualitative Research, Techniques and Procedures for developing Grounded Theory, page. 12
- [51] System Dynamics Society, 2011. What is System Dynamics? www.systemdynamics.org, accessed January 2011.
- [52] Van der Lei, T.E., Kolfshoten, G.L. and Beers, P.J., 2010. 'Complexity in multi-actor system research: towards a meta-analysis of recent studies. Journal of Design Research, vol. 8, number 4. pages 317-342.
- [53] De Vreede, G.J., Briggs, R.O., 2005. Collaboration engineering: designing repeatable processes for high-value collaborative tasks. Hawaii International Conference on System Science. IEEE Computer Society Press.
- [54] Whittaker, B., 1999. What went wrong? Unsuccessful information technology projects. Information Management Computer Security, 1999; 7(1):23–9.
- [55] Williams, T., 1999. The need for new paradigms for complex projects. International Journal of Project Management, 11, 269–273.
- [56] Wijers, G.M., 1991. Modeling support in information systems development. Thesis Publishers Amsterdam, The Netherlands.
- [57] Wijers, G.M., Heijes, H., 1990. Automated support of the modeling process: A view based on experiments with expert information engineers, Lecture Notes in Computer Science, Springer Link, Volume 436/1990, 88-108
- [58] Xia, W. and Lee, G., 2005. 'Complexity of information systems development projects: conceptualization and measurement development', Journal of Management Information Systems, Vol. 22, pp.45–83.
- [59] Van Daalen, C., Thissen, W., 2003. Dictaat spm2310, Continue Modellen, Deel A – System Dynamics. Faculteit Techniek, Bestuur en Management, Technische Universiteit Delft, juli.
- [60] Verschuren, P., Doorewaard, H. (2007), Het ontwerpen van een onderzoek. Uitgeverij LEMMA, Den Haag, Nederland
- [61] Whitty, S.J., Maylor, H., 2009. And then came Complex Project Management (revised). International Journal of Project Management 27 (3), 304–310.

APPENDICES

Appendix 1 Interview protocol

1. Background

This interview protocol is setup as a guideline for the interviews that are conducted as part of the research, with the objective of combining techniques from different management approach to enhance project management for complex projects. The respondents are selected on basis of their background, experience and availability at the Faculty of Technology, Policy and Management.

2. Purpose of the interview

The primary goal of this interview is to collect empirical data that will help answering the research questions. The questions are largely based on the selected Actor, Support, Context framework for assessing project complexity. An effort is made to identify possible combinations of techniques of different management approaches to enhance project management, in order to be able to deal better with complex projects.

3. Results of the interview

The interview delivers information about the following aspects from the viewpoint of the respondent:

- Background
- Project definitions
- Reflection of specific management approach on Actor, Support, Context framework.

4. Steps for interview

- The respondents are approached to participate in the interview. They receive an explanation by email about the context of the interview and when they agree, an appointment is made for the interview.
- When the appointment is made, some important aspects of the interview is mentioned:
- It will be a semi-structured interview, where not the structure of the interview is important but the dialogue that is started.
- It is about an inventarisation of personal experience, visions and tradeoffs on different management approaches.
- The time constraint of the interview will be approximately 45 minutes, including 15 minutes of slack.
- The interview will be recorded with the permission of the respondents, as support material for writing this report. The recordings and notes will only be used for this graduation project.
- The interview consists of the following parts:
 - Introduction of the interviewer

- Introduction of the expert
- Project definition and complexity factors
- Essence of selected management approach
- Management approach vs. Actor, support, context dimension
- Shortcomings of the approach
- Suggestions for combinations
- Closure

5. Introduction of the interview

An important part of the interview is a good mutual introduction of both the interviewer and the expert. Therefore a good introduction should consist of the following aspects:

- Introduction of the interviewer
- Explain why the contribution of the project is important for the project.
- Explain the background of the interview, check of the necessary and purposed information is received at the appointment.
- Location check.
- Explain the reporting of the interview , including recording
- Ask for introduction of the expert
 - Short intro of current function
 - Background/Education
 - Working experience

6. Context of the interview

This consists of the interview questions and is presented in appendix 2.

Appendix 2 Interview Questions

The questions that were used as basis for the interview research are presented here. First the questions are presented in English, followed by questions in Dutch as most of the interviews were conducted in Dutch.

Appendix 2.1 Questions in English

#	Part	Questions
1.	Introduction	• Can you tell something about your current function at the faculty? • Can you tell something about your education and background?
2.	Project Complexity	• What is your definition of a project? • What are the elements that can make a project complex?
3.	Management Approach	• Which approach do u use for managing projects? • Can you describe this approach briefly? • Does this approach provide the support to understand and manage the complexity of a project?
3.	Actor dimension	• Does the approach provide the possibility to get an overview of all relevant actors? • Does the approach provide guidelines for mapping and managing relations between actors? • Does the approach present techniques for mapping and managing dynamics between actors • Is it possible to get an insight in the uncertainties in the actor dimension and manage them?
4	Support dimension	• Does the approach incorporate techniques for mapping and managing elements of a support system(s)? • Are there techniques in the approach for mapping and dealing with the relations of the elements of the support system(s)? • Does the approach provide techniques to deal with (needed) dynamics in the support dimension and deal with this? • How does the approach support to get insight and deal with uncertainties

regarding the support system(s)?

- | | | |
|-------|--------------------------|---|
| 5. | Context dimension | <ul style="list-style-type: none">• What techniques does the approach provide to get insight and manage the elements of the context dimension?• Does the approach incorporate techniques for managing the relations between the elements of the context dimension?• Does the approach present techniques for mapping and managing dynamics within the elements of the context dimension?• How does the approach deal with getting insight and managing uncertainties within the context dimension? |
| <hr/> | | |
| 6. | Suggestions | <ul style="list-style-type: none">• What are the shortcomings of this approach?• Do you have suggestions for combining this approach with other approaches for dealing better with complexity of projects?• What were the decisive factors for you to use this approach? |
-

Appendix 2.2 Questions in Dutch

#	Deel	Vragen
1.	Introductie	• Wat is uw huidige functie aan de faculteit Techniek, Bestuur en Management? • Kunt u iets vertellen over uw opleidingen en achtergrond?
2.	Project Complexiteit	• Hoe zou u een project willen definiëren? • Wat voor factoren zorgen ervoor dat een project complex wordt?
3.	Management Aanpak	• Kunt u een korte beschrijving geven van de specifiek gekozen management aanpak? • Biedt deze aanpak mogelijkheden om de complexiteit van projecten in kaart te brengen en te managen?
4.	Actor Dimensie	• Biedt de aanpak technieken om de actoren die betrokken zijn in kaart te brengen en te managen? • Biedt de aanpak technieken om de relaties tussen de actoren te beschrijven en te managen? • Zijn er in de aanpak technieken waarmee je de dynamiek van actoren in kaart zou kunnen brengen en managen? • Biedt de aanpak technieken om inzicht te krijgen in onzekerheden in de actor dimensie en om in te kunnen spelen hierop?
5.	Support Dimensie	• Biedt de aanpak technieken om de elementen van een benodigde support systeem in kaart te brengen en te managen? • Biedt de aanpak technieken om de relaties tussen de elementen van het support systeem in kaart te brengen en te managen? • Zijn er in aanpak technieken waarmee je de (benodigde) dynamiek binnen het support systeem in kaart kan brengen en managen? • Zijn er mogelijkheden om onzekerheden die kunnen optreden binnen het support systeem in kaart te brengen en op te vangen?

-
- | | |
|--------------------------------|---|
| 5. Context
dimensie | <ul style="list-style-type: none">• Biedt de aanpak mogelijkheden om de context elementen in kaart te brengen en te managen?• Zijn er technieken in de aanpak, waarmee de relaties tussen context elementen in kaart gebracht en gemanaged kunnen worden?• Biedt de aanpak de technieken om de dynamiek tussen context elementen binnen in kaart te brengen en in te spelen hierop?• Zijn er technieken in de aanpak om de onzekerheden van context elementen in kaart te brengen en te managen? |
| <hr/> | |
| 6. Suggesties | <ul style="list-style-type: none">• Wat zijn volgens u de tekortkomingen van deze aanpak?• Heeft u suggesties om deze aanpak te combineren met een andere aanpak om zodoende beter om te kunnen gaan met de complexiteit van een project?• Wat zijn voor u de doorslaggevende factoren geweest om voor deze aanpak te kiezen? |
-

Appendix 3 Interview transcript Project Management

Respondent: Prof.dr.ir. Alexander Verbrack, Hoogleraar systeem en simulatie.

Achtergrond: wiskunde

Definitie van een project: een set van activiteiten waarvoor resources nodig zijn, waarvoor budget nodig is, met een duidelijk begin en duidelijk erin, dus heeft altijd een beperkte doorlooptijd.

Complexiteit van een project: hangt van heel veel factoren af. Ene project is complex omdat er bijvoorbeeld veel stakeholders betrokken zijn, terwijl een ander project complex is omdat er weinig geld en tijd voor is, of technologisch complex is. Dus er zijn heel veel dimensies zijn die een project complex kunnen maken.

Beschrijving Project Management aanpakken:

Veel van de methoden gelooft hij niet erg in, zijn naïef. Zijn allemaal methoden die een lijst met deliverables moeten opleveren, dus op het moment dat die resultaten zijn opgeleverd zou je kunnen zeggen dat het werk goed is gedaan. Veel project management aanpakken zijn zwak in stakeholder management, risico management of contracten in een project regelen. Standaard boeken over PM, zijn zwak op die punten. Aanpakken zijn best goed en nuttig, maar ze hebben deficiënties. Voor zachte complexiteitsfactoren is proces management nodig en voor technische aspecten meer project management en beide moet je goed vervlechten.

PMBOK: 9 aspecten waarop je moet letten en als je alle 9 gedaan hebt is je project goed, maar daar gelooft hij niet in. Dit is een start, maar geen garantie voor succesvol project, ook Risk management, stakeholder management, interface management is vrij naïef en contracting komt eigenlijk niet voor in de 9 vakjes van PMBOK. Deze aanpak is niet voldoende voor de complexiteit die je bij sommige projecten tegen komt, dus is niet volledig.

Op punten waar PMBOK gegeven de complexiteit zwak in is, bijv. complexe research projecten met nieuwe technologie waarbij je met diep ingevoerde experts moet werken, verteld PMBOK je helemaal niet hoe je met die experts moet aanspreken bijv. als ze divergeren ipv convergeren, hoe je in dit geval het project binnen gegeven tijd en budget moet afronden. Voorbeeld Integreren van collaboration engineering of stakeholder management, of hoe je experts aanpakt (zit niet in de PMBOK). Voor die complexiteit heeft PMBOK geen methoden voor in de scala aan arsenaal aan hulpmiddelen en technieken. Projecten die hele specifieke risico's heeft, waar er risico analyses gemaakt moeten worden zitten niet in de PMBOK (bijv. veiligheidscheck wanneer je een vliegtuig bouwt, wordt niet verteld door PMBOK). Vertelt niks over risico management approach je moet gebruiken, er is wel een checklist,

maar als je overal je een vinkje bij kan zetten betekent niet dat je een goed risico management proces hebt doorlopen gegeven de complexiteit van de risico's van het project.

Actor dimensie:

In PMBOK is er wel aangegeven dat je een stakeholdermap kan maken. Welke partijen moet je betrekken die een hoge interest erbij zitten (zitten wel bij PMBOK), maar de PMBOK biedt wel handreikingen, maar als het complex wordt met veel stakeholders schiet het tekort om hiermee om te gaan. Als het binnen bepaalde grenzen blijft biedt PMBOK voldoende handreiking om ook grote projecten te managen. Boven een bepaalde grens schiet het tekort en is zelfs naïef op het gebied van stakeholder management. Binnen de bepaalde grens kan je ook de relaties tussen de actoren bekijken, maar je zou een techniek buiten de PMBOK moeten gebruiken om te weten hoe je dit moet doen want de dynamiek en onzekerheden zijn niet sterk terug in PMBOK. Interface management is wel sterk in PMBOK (1 van de 9 vakjes), betekent dat je niet alleen over de actoren nadenkt maar ook over de relaties tussen de actoren, hoe actoren bijv aan je WBS gekoppeld zijn, overdracht tussen actoren. Interface management is wel apart benoemd. Risk management is ook apart aandacht gegeven (bijv. handbook of proj management schiet tekort hierin). Verschillende methoden schieten tekort om bijv. stakeholders aan te pakken.

Support dimensie:

Dit is juist een oplossing voor project complexiteit. Tools zijn niet op zichzelf complex, maar bedoeld om complexiteit op te lossen. Het zegt alleen een mate van support dat er nodig is, maar zegt niet welke tools er nodig zijn. PMBOK biedt geen handreiking om de noodzakelijke support te beschrijven (zit niet in het vocabulaire van PMBOK). Wel PMIS wordt vaak genoemd, alle informatie moet je opslaan en voor stakeholders inzichtelijk maken. Tuurlijk heb je tools nodig, zoals scheduling tools, costing tools, risk register tools, of kwantitatieve risk analysis tool, want anders kun je een waarde uitrekenen. (Ze worden onder de kreet: Project management information systems geschaard en naarmate je project complexer wordt zul je zwaardere of andere tools nodig hebben. Bij kleinere projecten kun je scheduling met MS project doen, maar bij een grotere project zul je naar Primavera moeten gebruiken. In elk van de 9 categorieën wordt wel beschreven welke tool je kan gebruiken. Er zit hier veel in op het gebied van stakeholders, er wordt best veel over technieken besproken. Over groepsprocessen staat er bijvoorbeeld: hoe acteren groepen in een process, WBS wordt genoemd, time management, tools en techniques wordt expliciet beschreven met outputs, arrow diagrams etc. gant charts, milestones. Contracting zit er goed in, communication zit er ook in. Technieken worden wel in PMBOK genoemd, maar is niet voldoende om projecten van zeer grote complexiteit aan te pakken.

Context Dimensie:

Dit ontbreekt heel erg in de meeste technieken. Elk van de 9 hokjes draagt iets bij aan de context, maar het feit of het bijvoorbeeld het zegt niet dat er vooraf kritische succesfactoren gedefinieerd moeten worden, welke heel erg zou helpen. In PMBOK mist dan ook “define critical succesfactors” value drivers, drivers voor het project. Bij zoeken op critical succesfactors in de guide blijkt het helemaal niet voor te komen. Driver komt ook niet voor. Project goal zit er ook niet in. Context, (project context): wordt in 1.5.3 kort genoemd. Maar wordt niet uitgewerkt, staan ook geen tools of technieken genoemd. Context waarin het project plaatsvind ontbreekt heel erg bij de meeste methodieken en ook bij PMBOK. Checklist staat er wel bij, maar er staan geen methoden of technieken bij die je kunnen helpen om de context in kaart te brengen of te schetsen. (doelenboom of drivers ontbreken, wat is bijvoorbeeld the ultimate goal van het project). Dynamiek en onzekerheden kun je kwijt in het risk management hokje. Maar bijvoorbeeld als er in het context van het project iets verandert waardoor je in t project zelf moet aanpassen wordt niet herkend, zitten geen waarschuwingen hiervoor in de PMBOK.

Tekortkomingen:

Stakeholder management is zwak, heeft geen value driver (wat maakt een project tot een succes), contract management is naïef (geen incentives voor goed gedrag) dit kan verbeterd worden, kwantitatieve risk management kan ook verbeterd worden

Combinaties:

Voor elk van deze dingen zijn er goede methoden (bijvoorbeeld voor stakeholders heb je proces management of collaboration). Voor risk management heb je ATOM methodologie (Simon en hilson). Context blijft moeilijk, geen technieken hiervoor. Moeilijk om de grens hiervan te bepalen, weet niet wat ik ermee moet.

Sterke kanten PMBOK:

Heldere structuur, uitgewerkt in stappenplan, internationale standaarden (PMI en andere instituten certificeren, en allemaal werken met een zelfde soort structuur als PMBOK) checklists zijn ook handig. En als je de zwaktes ervan kent kan je dit met andere methodes indekken.

Appendix 4 Interview transcript Lean Six Sigma

Respondent: Dr. Zofia Lukso, Hoofddocent sectie Energie & Industrie, TBM

Onderzoeksgebied: leider van het deelproject Intelligent Infrastructures, deel van Next Generation Infrastructures, waarbij intelligente infrastructures worden bekeken en agent-based modelling wordt gebruikt om socio-technische systemen te modelleren. Lean Six Sigma behoort ook tot haar onderwijsstof.

Achtergrond: Wiskunde en filosofie, gepromoveerd bij de TU Eindhoven in regeltechniek (natuurkunde), vervolgens werkzaam bij de TU Delft.

Definitie van een project: Dat is een afgebakende set van activiteiten met als duidelijk doel iets te bereiken. Dat doel moet dan gespecificeerd worden en binnen het project wordt dan aangegeven wanneer dit bereikt moet worden en met welke middelen (resources, software).

Complexiteit van een project: Zegt eerder dat een project gecompliceerd is ipv complex. En een project is gecompliceerd als je bijvoorbeeld verschillende partijen hebt met verschillende belangen die met elkaar moeten communiceren. Een gezamenlijk doel moet vertaald worden naar de subdoelen binnen de organisatie en dat is niet makkelijk.

Lean Six Sigma:

Essentie: Is met six sigma begonnen, welke een filosofie is die gericht is op het verminderen van variatie in processen die herhaald worden. Wat je eigenlijk probeert met six sigma projecten is indruk te krijgen in de variatie van de output van de activiteiten en deze variatie beheerst te houden. Dat betekent zo klein mogelijk en de gemiddelde waarde is wat je ervan had verwacht. Dus met six sigma streeft je ernaar om herhaaldelijke processen op een zodanige manier te verbeteren dat de variatie beheerst wordt (of exact wat de klant wilt afhankelijk van de doelstelling).

Lean: is gericht op het minimaliseren van waste/afval. Dus als je praat over lean management, ga je altijd inventariseren wat de waste in de organisatie is die je kunt minimaliseren. Vaak kan de waste ook inhouden quality given away (je zou het je tijd beter kunnen gebruiken aan dingen die meer toegevoegde waarde hebben).

Een combinatie van beide approaches (lean en six sigma) biedt alles wat je zou wensen voor het beheersen van operaties. Beide methoden gebruiken tools die gebaseerd zijn op statistiek en data analyse, alleen is de focus anders. Six Sigma is gebaseerd op de normale verdeling en, waarbij je niet buiten de six sigma grenzen treedt. Op

het moment dat je de six sigma kwaliteit hebt, betekent dit dat je als organisatie zeker weet dat je producten gaat leveren die altijd aan de specificatie voldoen.

Lean Six Sigma is op zichzelf al genoeg om een project te managen. Er staat beschreven dat op het moment dat je een project selecteert je taken en rollen moet toewijzen. Hierin zijn er verschillende hiërarchieën, zoals master black belt, black belt, green belt met verschillende verantwoordelijkheden. Normaal gesproken is er in organisaties waarin six sigma wordt gebruikt een hele duidelijke infrastructuur ontwikkeld, waarin mensen zijn toegewezen voor bepaalde rollen. Zo is de master black belt verantwoordelijk voor het definiëren van het project (planning, schedule) en duidelijke aanwijzingen over de taken en verantwoordelijkheden.

Actor dimensie

Dus Lean Six Sigma gaat goed om met actoren in kaart brengen en rollen te verdelen. Alvorens je een six sigma project kan beginnen heb je eerst structuur in de organisatie nodig, zodat je weet wie, waarvoor ingezet kan worden. Dus Lean Six Sigma biedt de mogelijkheden om coördinatie en samenwerking te kunnen ondersteunen, bijvoorbeeld operators moeten meten, green belts schatten de modellen

Met dynamiek in de actor dimensie verstaat ze dat er een situatie kan ontstaan waarin actoren kunnen veranderen. Nogmaals in lean six sigma zijn er rollen gedefinieerd. Op het moment dat er een green belter niet inzetbaar is, dan wordt er een andere greenbelt ingezet. Dus veranderingen worden opgevangen op hetzelfde niveau. Wordt het echter bij het monitoren van een project geconstateerd dat een project niet bijdraagt aan het hoofddoel kan het project scope verandert worden. Dit is echter de verantwoordelijkheid van de master black belt, want hij wordt erop afgerekend. Ook kan er momenten van tussentijdse evaluaties ingebouwd worden of in het ergste geval het project gestopt worden.

Wat de onzekerheden in de actor dimensie betreft is six sigma te rigide om onzekerheden in de actor dimensie te hebben, want elk project start met duidelijke verantwoordelijkheden en taken van de deelnemers.

Lean Six Sigma wordt dan ook gebruikt in organisaties ipv netwerkmachtige situaties waarbij partijen van elkaar afhankelijk zijn (bijvoorbeeld HSL project, dan heb je meerdere partijen en zou je het project niveau opereren). Ook al wordt Lean Six Sigma niet toegepast in situaties met meerder partijen, wat bedrijven wel doen is bijv. suppliers ook leren wat de kracht van Six sigma is en zo samen te kunnen werken.

Support Dimensie

Volgens het oogpunt van complexiteitstheorie kan een complex systeem nooit door 1 paradigma beschreven worden. Je hebt verschillende invalshoeken (bijv. psychologie, sociologie, of regeltechniek) nodig om een complex systeem te beschrijven. En als je dat probeert te transformeren naar jou raamwerk, dan zou je kunnen zeggen dat een complex project door verschillende paradigma's beschreven moet worden. Je zou dan kunnen praten van verschillende tools die je nodig hebt voor interactie, voor het monitoren van data of voor economische factoren. Als je zegt dat je verschillende typen tools hebt, dan kan je zeggen dat het iets aan de complexiteit kan bijdragen.

Tools zijn altijd nodig, ze dragen dus niet bij aan de complexiteit, zo zijn actoren ook nodig. Ze suggereert om te kijken naar type tools in plaats van naar aantallen. Als je bijv. 100 administratie tools nodig hebt zou je het nog geen complex project noemen, maar wel bijvoorbeeld als je verschillende typen tools zoals, mathematische, conceptualisatie, simulatie, evaluatie tools. Six Sigma zegt zeker iets over het gebruik van support systemen, want het houdt van structuur, het zegt dan ook in welke situatie wat gebruikt moet worden. Over het omgaan met dynamiek en onzekerheden in de support dimensie: Six Sigma zegt ook hoe je moet experimenteren als je niet weet wat je kan verwachten. Over het algemeen worden de tools goed beschreven.

Context dimensie

Uitgelegd dat met context wordt bedoeld de omgeving waarbinnen Lean Six Sigma kan worden toegepast. Zij maakt een onderscheid hierin van de concrete omgeving (bijv. bouwprojecten) en een virtuele omgeving. Lean Six Sigma is toepasbaar op alle concrete projecten in alle sectoren, waar er een tastbaar resultaat uit te verkrijgen valt (bijv. bankwezen, procesindustrie, logistieke sector).

Op de vraag of de relaties tussen de fysieke elementen van de context waarbinnen six sigma gebruikt wordt gemanaged kan worden door Six Sigma: Six Sigma zegt alleen iets over de processen en afhankelijk van wat voor project en wat de grenzen zijn van het project moet je meer of minder interacties meenemen. Maar het is dus gericht op processen, bijv. sorteerprocessen bij TNT post en de interactie daartussen.

Dynamiek: processen zijn dynamisch

Onzekerheden: Wat voor onzekerheden? Je kijkt naar verschillende processen en er komen steeds andere resultaten uit. Eerst probeer je dit te begrijpen, wat is vervolgens verwacht gedrag en wat het gemiddeld gedrag moet zijn (in een bepaalde band). Hierin zijn er ook onzekerheden. Dan ga je onderzoeken waarom die band zo groot is, wat de oorzaak is van de afwijkingen. In een bouwbedrijf kan Lean Six Sigma ook gebruikt worden, maar op het moment dat het project gedefinieerd is, moet de master black belt het project verder bewaken en monitoren.

Tekortkomingen

Het is meer een bepaalde manier van denken; de Japanse naamgevingen worden soms belachelijk gevonden, of mensen willen niet met black belts of green belts geassocieerd worden. Het nadeel is verder dat het echt een projectmanagement techniek is waarvoor er formuleren ontwikkeld zijn, data moet bijgehouden worden en het wordt ook een zwaar administratieve klus gevonden. Verder is het alleen geschikt voor bepaalde typen processen (repeated processes) en gericht om toegepast te worden binnen individuele organisaties en minder geschikt voor netwerkachtige omgevingen met meerder partijen en belangen.

Appendix 5 Interview transcript Proces Management

Respondent: Dr. Wijnand Veeneman, Hoofddocent TU Delft bij de groep POLG.

Onderzoeksgebied: governance en public transport en hoe manage je grote technische projecten.

Managen van grote technische projecten, niet geïnteresseerd in al die methoden die daarin structurerend zijn (zoals prince 2, systems engineering) en of de ene aanpak beter is dan de andere, maar wat brengt een bepaalde aanpak voor het proces van het managen van een project. Geen aanhanger van een bepaalde aanpak, is moeilijk om te zeggen dat t ene beter is dan de andere.

Achtergrond: sociaal geograaf, maar doet er niks mee. Gepromoveerd op TB, transport geografisch perspectief, op aansturing van openbaar vervoer en zo kwam hij op governance van OV kant.

Definitie Project: allerlei mensen die nuttige definities geven wat een project is, maar heb geen voorkeur van een definitie. Een project kan heel goed lineair zijn als je goed kan voorspellen wat de scope is die je gaat bouwen.

Project Complexiteit: Als het heel proces van beton leggen, rails leggen, wat de kosten zijn, allemaal lineair en duidelijk is, kun je de muren gesloten houden en is het project niet complex. De truc is dat in veel projecten je dit niet zo kunt dicht timmeren, omdat je erachter zult komen dat je de kunstmatig opgebouwde muren zult moeten openbreken. Dus ben je voor een deel muren aan het afbreken en bouwen, bijvoorbeeld mag het meer kosten, was de originele project definitie niet goed, ligt het aan degene die het project moet uitvoeren. Er is geen duidelijke scheidslijn tussen een lineair project en een complex project, maar t kan complexer worden (afbakening van het project klopt niet).

Definitie Proces Management: Begint met definitie van project management (optrekken van grenzen waarbinnen het moet gebeuren met een predict and control achtige aanpak). Op gegeven moment zie je dat er vakjes daarbinnen als miniprojectjes met predict and control worden neergezet. Inkaderen van de kleine taken helpt met inkaderen van het hele taak. OP gegeven moment besef je dat er zaken niet meer passen binnen de grenzen. Dit is het moment dat proces management perspectief bij komt kijken. Op zo een moment dat er dingen anders moeten dan in de afgekaderde projecten (honingdraad structuur), dat het simpelweg afgebakend houden ervan niet meer werkt. Proces management zegt dat als je het verticaal escaleert, a la project management, dan mis je een deel van de oplossingsruimte. Dus onderscheid tussen de manier waarop probleem oplossen in complexe projecten gebeurt: enerzijds, hiërarchisch sturen, waarbij veel van de rijkheid van de mogelijkheden van de oplossingsruimte verloren gaat of anderzijds kun je kijken hoe je aan elkaar kunt aanpassen. Het voordeel van procesmatig

aansturen is dat samenwerking tussen partijen mogelijk wordt, waardoor meer oplossingsmogelijkheden naar voren komen. Aan de andere kan verzwakt de verantwoordelijkheden tussen de partijen, terwijl bij hiërarchische sturing mensen meer gecontroleerder te werk gaan, waardoor er minder gesteggel is.

De basisaanpak is projectmatig, ook als je iets tijdelijk procesmatig aanpakt. Bij proces management gaat het niet om het implementeren van een inhoudelijke oplossing, maar het implementeren van interactie tussen mensen die toewerken naar een duidelijke oplossing. Processen zijn: evenementen die je kan gebruiken, nadenken over wie/waarom/wel moet meedoen. Nadenken over exit rules en of je alles in 1 keer wilt of verschillende rondes. Bij proces management moet je als opdrachtgever blijven sturen, anders komen jou eigen belangen onder druk te staan, dus is het wel van belang dat je het proces wel managed.

Actor dimensie

In sommige proces management boeken worden de actoren wel in kaart gebracht, ze gaan heel erg in op het nadenken over wat de dilemma's zijn die tussen de verschillende actoren bestaan. Dilemma's moeten in kaart gebracht worden en vanuit de dilemma's probeer je deze te managen. Dilemma management: Hoe kan er gestuurd worden zodat partijen tot een oplossing komen, waarin ze elkaar niet in de weg zitten (bijv. de ene partij door toedoen van de andere partij extra kosten niet zelf hoeft te betalen). Basis van proces management is het zoeken naar actoren en hun dilemma's, de dilemma's in kaart brengen, vanuit de dilemma's redeneren en proberen oplossing te creëren, waarbij beide partijen samen tot een oplossing die een manager al planned niet had bewerkstelligen. Belangrijke factoren bij samenwerking tussen partijen: speed, openness, transparency.

Proces management vindt dynamiek goed, terwijl project management dynamiek juist vervelend vindt. Dynamiek gebruik je bij proces management om tot oplossingen te komen, vraagt wel veel onderlinge afstemming. De dynamiek zit erin dat partijen samen tot een oplossing komen, dat partijen bewegen en hun positie opschuiven. Als je onzekerheden hebt zorg ervoor dat je proces management achtige instrumenten ingebouwd hebt en dat je ermee aan de slag kunt.

Support Dimensie

Wat kan helpen is dat als partijen een gezamenlijk model hebben van de scope die je wilt realiseren en van de omvang van de scope die je wilt realiseren. Als actoren samen gemodelleerd hebben hoe de bouwprocessen moeten verlopen en de fijn afstemming daarmee scherper kunnen analyseren en als je daarmee snel inhoud kan voeren in afwegingen, dan kun je sneller tot oplossingen komen. Dus het kan helpen om d.m.v. simulaties of modelleringen de onzekerheden bij beide partijen te reduceren en samen keuzes te maken. Wonderlijk genoeg is inhoudelijk ondersteuning nodig om samen tot keuzes in het proces te kunnen komen. Dus support tools helpen, zo staat het in boek proces management hoe je commitment kan versterken of sense of urgency kan realiseren (ook wordt simulatie besproken).

Wat ik te vaak zie ik dat de tools de potentie hebben om de complexiteit op te lossen, maar het zeer matig doen en juist een soort van schijnzekerheid geven. (Dit is misschien niet het gezochte antwoord, maar geeft wel aan wat de rol van support kan zijn bij proces management).

Op de vraag of hoe meer support je nodig hebt, hoe complexer het project of proces? De respondent verandert de essentie van de vraag en zegt: geven support tools meer of minder zekerheid? Deze vraag is alleen te beantwoorden als de support tools werken, dan brengen ze minder complexiteit en meer zekerheid, maar dit weet je pas achteraf. De toepassing van de tool en het succes ervan is afhankelijk van het project zelf en wie erbij betrokken zijn en niet bijvoorbeeld van het maken van een risk analysis matrix. Er is ook wel een relatie tussen support en onzekerheid in de zin dat hoe onzekerder de situatie is, hoe minder bestaande tools succesvol zullen zijn. Het gaat er niet om of support tools goed zijn om toe te passen of niet, maar het gaat erom hoe je het uiteindelijk doet en hoe het past in de omgeving waarbinnen je het doet.

Context dimensie:

Project management kun je overal toepassen, van een simpele badkamer renovatie tot een boorinstallatie project van Shell. Beide hebben echter totaal geen inhoudelijke overeenkomsten, zelfde geldt ook bij proces management. Het is namelijk wat anders als je proces management doet op kleinschalige afstemming op een ding dat al staat, dan proces management voor besluitvorming over het hele project. Sommige projecten zijn in de basis al heel erg procesmatig, bijvoorbeeld waterhuishouding van het IJsselmeer. Dus het hangt heel erg van het niveau waarop je naar het project kijkt en wat voor type opgave het project heeft. Dus het is wel in elk context toepasbaar, maar zeker niet overal even belangrijk, het hangt af van de onzekerheid en complexiteit die je in het project hebt, als je een project hebt met heel weinig onzekerheid en complexiteit, moet je het project niet onnodig te compliceren door heel procesmatig om te gaan. Hoe complexer een project is, hoe meer gemak je hebt van het inruimen van procesachtige mechanismen. Elementen, korf met honingdraadjes, waarin de elementen aan elkaar verbonden zijn. Dynamiek: Bij dynamiek veranderen de relaties, de blokjes veranderen er gebeurt iets in het project, waarbij actoren het niet meer eens zijn met elkaar. Procesmatige aanpak zegt dat je interactie moet opzoeken, maar dit hoeft niet altijd want uitgangssituatie van een project is dat grenzen helder zijn, anders bestaat het project alleen maar uit stukjes proces. Bij onzekerheden weet je niet hoe de dynamiek zich zal afspelen en de dingen zullen veranderen. Bij projectmatig denken heb je muren opgetrokken voor het project en daarbinnen speelt het project zich af en de relatie tussen deze omgeving en het project heb je vastgelegd met randvoorwaarden.

Tekortkomingen van process management:

De grenzen die zijn neergezet bij een project is voorwerk, is mooi, maar als je de grenzen gaat openbreken levert het een hoop gesteggel en gedoe. Proces moet je zodanig neerzetten dat je het project niet zodanig laat beheersen door processen, dat het heel erg mis gaat, bijv. uitloop in tijd etc. Bij procesmatig denken moet je

streng zijn, anders krijg je teveel geneuzel over alles. Je moet voorzichtig zijn met dat een ieder leert, samenwerkt en snapt hoe het werkt, dus de kennis is er wel maar het systeem niet. En bij een projectmatig perspectief: het systeem is wel gerealiseerd, maar het werkt niet. Dus balans zoeken, zodat ze elkaar niet in de weg zitten.

Appendix 6 Interview transcript Collaboration Engineering

Respondent: Dr.ir. Gwendolyn Kolfschoten, Assistant professor bij Sectie Systems Engineering

Onderzoeksgebied: Onderzoek gaat over collaboration support of het ondersteunen van samenwerkingsprocessen in organisaties.

Achtergrond: TB, Phd in collaboration support. Na Phd, werkzaam op TBM.

Definitie van een project: Project is het standaardiseren van een proces met een collaboratieve aanpak.

Complexiteit van een project: Het complexe aan een project als er veel verschillende stakeholders met belangen bij betrokken zijn, kan ook inhoudelijk complex zijn met ingewikkelde materie, maar vooral als er conflicterende belangen, en of conflicterende informatie of kennis als experts het niet met elkaar eens zijn. Gefocust op actoren dimensie.

Collaboration engineering: een proces ontwikkelen om een groep te ondersteunen om een specifiek doel te bereiken. Idealiter is het proces dat herhaalbaar is voor datzelfde proces, onafhankelijk van de groep of onderwerp is waarvoor het gebruikt wordt. Het idee is dat je hetzelfde proces kan gebruiken voor meerdere samenwerkingstrajecten (projecten).

Actor dimensie:

In kaart brengen: Raamwerk om een inventarisatie te doen naar de mensen die naar de sessie komen, welke belangen, welke kenmerken, opleiding, relaties met elkaar ze hebben met elkaar (team, verschillende disciplines). Op die manier krijg je inzicht in wie er aan tafel zit en op basis hiervan wordt het proces ontworpen in eerste instantie en daarna wordt het gecheckt of de aannames kloppen voor de specifieke situatie. Bijvoorbeeld bij een sessie waarbij er net een reorganisatie is geweest, moet de groep veel aangestuurd worden om de dieper liggende integriteitsissues te bekijken. Ze moeten gefocust worden op het doel als er andere belangen in hun hoofd op dat moment spelen. Relaties: worden in kaart gebracht en ingegaan als het nodig is als er bijvoorbeeld conflicten zijn. Dus als het in de weg zit worden er wel tools geboden als het nodig is. Voor alle assessment wordt er een generieke studie gedaan. Default aanpak: wie komen er, welke belangen en hoe om te gaan, wordt in 1 keer vooraf gedaan voor alle sessies. Ijvoorbeeld in deze sessie was er een leermoment over integriteit, en hierna positioneren mensen zich anders. Dus standpunten en perspectieven worden op de situatie gewijzigd en niet iedere groep is hetzelfde als de basis assessment, dus daar zit ook wat dynamiek in. Inspelen tijdens de sessie, gebeurt niet altijd. Sommige dingen kunnen je niet vooraf voorspellen. Er zijn altijd dingen in die anders lopen dan

je had verwacht. De trainees hebben beperkte middelen om te gaan om processen te kunnen aanpassen om met grote veranderingen om te gaan, een facilitator heeft daarentegen wel de mogelijkheid. Dus het zit wel in de aanpak, maar niet toegankelijk voor een ieder.

Support Dimensie:

Bijvoorbeeld een Elektronisch vergadersysteem. Het systeem kan voor allerlei processen gebruikt worden zoals, risico analyse, requirements opstellen (GSS, GDR). Elementen: Netwerk van computers, iedere deelnemer heeft een eigen input device en een centraal scherm is er om feedback te kunnen geven. Je kan ook de elementen coördineren, leiderstation die beheert alle laptops en kan verschillende tools aan/uitzetten of configureren. Kan er veel in coördineren. Actoren kunnen ook onderling met elkaar communiceren. Dynamiek: Bedoeling is dat de tools steeds anders geconfigureerd worden, mate van flexibiliteit in. Dit is vooraf in collaboration engineering gedefinieerd, soms zijn er processen waarbij er flexibele stappen nodig zijn (als dit dan dit...). Onzekerheden: Het systeem kan anders ingezet worden, maar de practicioner heeft meestal niet de kennis de tool opnieuw te kunnen configureren. Om te kunnen omgaan met onzekerheden is er een bepaalde kennis nodig, welke vanuit de oogpunt van de facilitator het wel kan.

Context Dimensie:

Sessies gaan vaak over onderwerpen. Inhoud is eigenlijk een verandering in een actoren dimensie. Bijvoorbeeld zou ook sessies gedaan kunnen worden over ontwerpen of aanpassing van een systeem, biedt geen mogelijkheid om iets aan een systeem aan te passen. De aanpak biedt actoren mogelijkheid om erover na te denken. Je kan niets fysiek aan het systeem veranderen, slechts inventariseren hoe je dat gaat doen. Het is niet dat je daadwerkelijk dat je iets aan het context systeem gaat veranderen. Dus je kan het plan daarvoor kunnen maken. Je kan er fysiek iets mee doen. Je kan voor alle 4 complexiteits dimensies, kan je wel brainstormen of oplossingen voor bedenken, maar je kan het niet aanpassen. Context gebruik je wel om je sessie ontwerp aan te passen, op basis van wat 1 van de stakeholders zegt hierover. Bijvoorbeeld als er veel onzekerheden zijn in een context kan je een dit gebruiken om een risico analyse te doen.

Tekortkomingen: Scope waarvoor het toepasbaar is beperkt. Het gaat altijd over samenwerkingsprocessen, stukjes analyse of besluitvorming. Dit is maar een beperkt deel van alle analyse en besluitvorming die er in een organisatie plaatsvinden. Focus ligt meer op de actor dimensie, dus alleen geschikt voor projecten waarin echt alle actoren deel moeten nemen aan het bedenken van een oplossing.

Mogelijke combinaties: kan gecombineerd worden met decision support. Tools om meer inzicht te krijgen in de complexiteit van systemen of context, zodat je dat als input kan gebruiken voor besluitvorming, of verbeteringen of aanpassingen. Nu is het allemaal gebaseerd op basis van experience van de deelnemers. Resultaten van

simulaties, of analyses uit het context zou je beter kunnen gebruiken om processen erop in te stellen. Als je hebt over meerdere samenwerkingsvormen in de lijn van een groot project heb je bovenop heb je een project management aanpak nodig.

Factoren om te kiezen voor CE: als het om samenwerking tussen grote groepen van actoren te ondersteunen. Je hebt ook wel creativiteitstechnieken en andere vormen van workshops ondersteuning (teambuilding) etc., maar deze zijn niet gericht op een bepaalde uitkomst. CE is inmiddels een redelijk volwassen aanpak met tools, kan nog wel wat aan gedaan worden, maar in termen van flexibiliteit kan er wel wat aan ontwikkeld worden.

Appendix 7 Interview transcript Traditional Software Development

Respondent: Dr. Stephan Lukosch, Associate professor at Section of Systems engineering

Research Area: About collaboration support systems, how to improve collaborative design and engineering of complex projects. So how you enable people to collaborate and how you support them with the right technology. One of the ideas is to measure collaboration performance and use these kinds of measurements to forecast the effectiveness of collaboration processes. Background is PhD in computer science.

Project definition: A project has a beginning and normally should have an end, where you are trying to create something. If you want the project to end, there should be a clearly defined goal, which is normally a product.

Project Complexity: The complexity of a project increases when you have several different actors in the project that work together to achieve a goal, because then you have to align them. Another dimension could be the goal or the intended outcome of the project. If the product itself is technically complex, like a power plant. If you think in the line of multi actor systems, you should have in mind that even when you create a product it may involve multiple actors and you have to bring them on the table.

Approach: General software development approaches to handle the complexity: waterfall model, spiral model. Agile is one way to cope with complexity in projects.

Project complexity: Software project: When you have a lot of different classes with a lot of different relations, lot of inheritance, you could say the program is more complex. This is basically true for every software project, the more lines of code you have the more interdependencies you have, the more complex the product becomes.

Agile oriented modeling approaches are using object oriented modeling as well, so it is more a tool rather than an approach. From here on the focus will be more on software development instead of object oriented modeling, because as mentioned object oriented modeling is more a tool, instead of an approach.

Actor dimension:

In a Software project you normally have various different stakeholders. When e.g. Microsoft develops something, they do it for the market. There is always a prospective customer, who normally has some requirements. On the other side there are software developers who has to understand the requirements of the customer. There are also designer who are responsible for the design of the software and the interaction inside the tool. There are different software development approaches, for example when you consider the waterfall method. What this method does in the beginning is bring this people together and then writes more or less one specification of the future tool. But

they doesn't reconsider this requirements at a later stage, what you get is no feedback loop or iterations. This might be good in some projects, for example when you have to integrate your software in something else. Example, when you have to design software for a ship, where there are specifications that cannot be changed. But on the other hand there are Agile Approaches, where you try to bring all stakeholders together at one table, so developers, designers and future customers. What you try there is try on practice which is called custom on side?. With Agile approaches you have a customer who is joining the development team for the whole development time. Every day you have a product, a prototype, where the customer can have a look at it and reflect on it or identify new requirements. When the hypothesis is that the more relations there are between different actors in a software project, the complexer a project is, I agree on that. I think the roles and the relations between the actors in a software project are more or less stable (you have the customer who wants something from the producer, he has some developers who have to develop something based on the requirements of the customer and the developers have to interact to each other. On the other hand there are designers who have to talk to the developers and there are testers and persons that evaluate the product. But the relations between these actors is stable. For example when you have the waterfall method it is pretty simple, because you have one phase after another. Here you tackle each of this complexities one after another. This does not mean that you get a perfect product at the end, but the complexity of handling according to the waterfall method is less compared to handling the product with an Agile approach, because you have to take care of these relations every day. Agile methods cope better with dynamics than the waterfall method.

Support Dimension

You should use some kind of modeling language or specification in order to make sure that the product is developed in the end. For example UML is a means of enabling communication between the different actors or depending on the language you can use another programming language or specification description. But it depends on what the developer is familiar with. The formal specification of the software which you are going to use is the most important when you have a lot of different actors in the project, because then you have to enable communication en make everyone aware of it. If you want others to use the software in the end or allow them to develop it further , you need some kind of specification. What kind of specification you choose to specify your software depends on yourself. There are so many different possibilities in

When you consider software it does not depend on the type of approach, but on the programming language. So it does not make any sense to use UML when you are going to use a linear programming language. These types of languages don't know about objects, so you cannot model object because it cannot transfer your model. The approaches, like waterfall method, doesn't prescribe what kind of programming language you are going to use. As a developer you have the flexibility to choose which language of modeling tool you will use for your product.

Context dimension:

You can consider UML as a modeling language this is an approach for describing and visualizing different aspects. But of course it heavily depends on the project context itself, how many attributes you have. When you have a more complex project where you have a lot of different classes, objects and relations, the context gets more complex. UML helps you to deal with the context complexity, because it has different diagram types to model different objects or activities or relations. With UML you can specify entities, objects and relations and you can specify how this entities interact, but it does not tell you anything about running or managing the project. SYSML can be used to specify whole systems, it is used within systems engineering. Then you can model your chemical plant with SYSML. Dynamics: Well if you consider software development, there are different methods you can use. I would say that the waterfall method does not allow, because it has not iteration. But if we go more towards Agile approaches, they have something in between, which is called the iterative waterfall and SCRUM and extreme programming (Agile development methods). Normally, for ex. in the waterfall method, you have an analysis, design, implementation and test phase. In the waterfall approach you only go through these phases once, you test it and deliver it.

Depending on the project size or complexity each phase can take a certain period of time. When you consider a more Agile approach you basically run through every phase every day. SCRUM has iterations cycle of a month. My perception is that the shorter the iteration cycle is, the better you can cope with dynamics and uncertainties. The waterfall method is completely useless when you have to cope with dynamics and uncertainties and Agile approaches you can better cope with them. The whole idea is to identify or resolve uncertainties and to cope with dynamics. Especially when you consider software projects where you have a lot of feedback of the customer as the future user, that is basically the uncertainty, because it is very difficult to say to a software development company what you really would like to have. When you know that there will be more dynamics and uncertainties, I would say you need to choose a more Agile approach. And if the requirements are pretty clear and the customer really knows what he/she wants you can use a different method.

Shortcomings:

You cannot handle every software project with an Agile method, because when you go to a software company you normally have a fixed amount of time. So you don't know what in the end will be result. Waterfall method has the benefit that the requirements are already specified and the software company says that they are going to deal with the requirements within a certain time and money.

Combinations:

Software developers don't care about methodologies, they just want to develop software. All the approaches are basically a means to deal with the requirements of the customer. The most complex part is to get the

requirements right, based on the needs of the customer. Context plays a role in it, because if you develop something for the military there are some very specific requirements of what the software should do/not.

Appendix 8 Interview transcript Agile Development

Respondent: Dr. Marijn Janssen, Universitair hoofddocent sectie ICT, Directeur SEPAM

Onderzoeksgebied: ontwerpen om architectuur binnen organisatienetwerken van de overheid die willen gaan samenwerken, daarvoor moet je zowel de organisatie als ict gaan veranderen en hoe doe je dat nu: bijvoorbeeld met support: heel veel ontwerpaanpakken die erachter zitten.

Achtergrond: Technische bedrijfskunde in Eindhoven (afgestudeerd in administratieve logistiek). Administratieve processen die lopen, ICT is veel meer een hobby. Gepromoveerd bij systeemkunde op het ontwerpen van een architectuur die kopers en verkopers aan elkaar verbinden en simulatie voor gebruikt om dat met elkaar te simuleren en te verbinden. Daarna bij min van justitie gewerkt, aantal integratieprojecten gedraaid, daarna bij sectie ICT beland bij TBM.

Project: Wordt een verschil gemaakt tussen een programma en project. Programma is het totaal wat je wilt bewerkstelligen en dat ga je uitvoeren middels meerdere projecten. Elk project moet een bijdrage hebben en als je een programma hebt afgerond wil je dan de transformatie of bijdrage wil je dan bereikt worden. Een project kan heel klein zijn, maar als het een heel groot project betreft praat hij in zijn wereld over programma's. Er zijn veel losstaande projecten bij ministeries en overheden en die doen allemaal leuke dingen, maar op gegeven moment moet t bij elkaar gevoegd worden. Vaak zijn ze op TBM niet geïnteresseerd in kleine projecten, zoals het bouwen van digiD, maar het embedden in het grote geheel (hoe gebruik je authenticatie en identificatie mechanismes in het geheel en hoe zorg je dat die twee op elkaar aansluiten en gaan werken. Probleem in het begin was dat iedereen in het begin zijn eigen id dienst, en op gegeven moment heeft de belastingdienst geopperd om digiD te gebruiken (standaardisatievraag).

Complexiteit: Zit in het begrijpen van je omgeving. Dit is 1 van de moeilijkste factoren die vaak genegeerd wordt of vaak beperkt wordt door de overheid. Blijkt dat het negeren van private spelers dus bedrijven die de informatie moeten aanleveren voor belastingen, wetgeving komt ook bij kijken, veel weerstand, veel overheden zijn klein en kunnen niet met alle initiatieven van de centrale overheid omgaan. Verhouding centraal-decentraal spelen daar een grote rol in. Onzekerheid is vaak op langere termijn (is er financiering voor, wie gaat het beheren, gaat het betalingsmodel niet veranderen, dynamische omgeving verandert tijdens t project.)

Approach:

Agile betekent dat je vaak begint met een klein prototype achtige. Risico was dat er van tevoren niet aan de requirements gedacht was, design space waarbinnen je kan manoeuvreren was niet aangegeven. Men maakte allerlei keuzes die niet gericht waren op basis van de actoren die erin zaten, maar op basis van gebruikte technologie (omdat ze er goed in waren), i.p.v. de standaard van de organisatie. Dus creëer je producten die niet goed passen op. Ander typisch valkuil is dat als je een datamodel neerzet onafhankelijk van een andere datamodel, dat je niet goed kunt koppelen. Meestal heb je grote problemen daarin. Agile is helemaal niet erg, alleen moet je de kaders eromheen wel hebben, dus een ontwerpproces eromheen. Het gaat niet om het ontwerp, maar t moet binnen het kaders van het architectuur passen.

Actordimensie

Agile biedt wel inzicht in de actoren, want je wilt juist heel veel interactie. Als je bijvoorbeeld een agile afstudeerproject hebt, schrijf je dit nu op, kom je de week daarop, krijg je commentaar, etc.. Veel interacties en iteraties, korte cyclus met elkaar en binnen paar maanden opzetten. Als je uitgangspunten echter fout zijn, als je bijv. iemand inhuurt die niet bekend is met de organisatie, maar is dan duur om te onderhouden en te runnen. Met Agile kan je zeker de betrokkenheid van actoren coördineren, want Agile sluit helemaal niet uit dat er proces management is. Je moet nog steeds deadlines hebben, interacties met elkaar plannen, Het is alleen informeel alleen, soms zijn de gebruikers zelf onderdeel van het team. Hoofdgebruiker geeft soms input aan het team. Bij Agile is het veel opener, geen exacte deadlines. Boehm heeft het mooi gedefinieerd. Meestal is het minder op proces support gericht. Geen overkoepelende project management nodig, want het staat haaks op project management. Agile is meer prototyping (vs spiral). Met Agile kan je wel makkelijk met de dynamiek van de actoren omgaan, want Agile gebruik je juist wanneer je niet weet hoe de dingen in elkaar zitten. Je moet wel je randvoorwaarden creëren, modulariseren. Klein flexibel team. Agile gebruik je dus bij grote onzekerheid, je weet niet hoe het eruit komt te zien. Maar niet als er teveel stakeholders in zitten. Het zou kunnen dat je er een schil omheen legt, (dit het development) en daarbuiten project management mensen voor communicatie etc. Doel bij Agile is om het ontwerp neer te zetten en daarna eromheen te gaan kijken. Bij Agile moet je het klein houden, je kan niet een team van 100 mensen aansturen, dit gaat meer richting bureaucratie.

Support Dimensie

Is een kwestie van onafhankelijke en afhankelijke variabelen. Support wordt bepaald door jou aantal actoren en context variabelen. Het probleem is leidend, dan ga je nadenken hoe je het gaat aanpakken en dan bepaal je (resources, mensen, geld). Wat je wel kunt doen is te kijken naar wat voor type support is er aanwezig. Als je bijv. alleen mensen hebt die Agile kunnen werken, dan moet je projecten ook optuigen als dat soort projecten (geldt dan als randvoorwaarde). De mensen die erbij betrokken zijn halen de manier van tools vaak uit hun hoofden, ze kunnen ze vlug in brengen. Brainstormen over het ontwerp etc. Bij Agile modelleert men juist niet, is ook een kritiek hierop (bij de extreme versie). Het eindproduct is vaak ook niet duidelijk, je weet alleen dat je mensen blij

wilt maken met een aantal bepaalde eisen. Agile development is typisch iets wat in het voortraject zit, waar je nog niet helemaal precies weet wat je wilt. Als je dit eenmaal weet kan je t anders managen. In Agile zit er grote mate van onzekerheid in, afbakening en systeemgrenzen moeten goed gedefinieerd zijn, is de kennis van de mensen wel goed, zijn t creatieve mensen, doel wordt uit t oog verloren.

Context dimensie

Je moet dit wel hebben als leidraad voor je architectuur bijvoorbeeld, maar daarbinnen heb je wel een mate van vrijheid. Hangt af van het project. Bij sommige projecten kijk je niet naar wat je al hebt, terwijl bij andere het expliciet wordt gezegd dat het bij de organisatie moet passen. Omgeving wordt bij Agile meegenomen, want bij Agile is het doel om dicht bij de omgeving en de users te staan, alleen moet je omgeving niet te complex zijn, Anders lukt het niet op die manier. Als je bijvoorbeeld hier met 20 man had gezeten dan lukt het niet om een interview af te nemen. Project omgeving wordt ook wel vaak onderschat. Als bijvoorbeeld uncertainty heel veel actoren inhoudt, dan zou ik niet zeggen agile werkt niet meer. Maar als uncertainty inhoudt: we weten niet wat we willen, want die actoren hebben geen duidelijke requirements etc, dan wordt het een ander verhaal. Is het aantal actoren onzeker, of wat ze willen onzeker. Met Agile zou je zeggen, we weten niet precies wat we willen, maar de context is 1 van de redenen om het agile te doen. Dynamiek kan zijn externe invloeden, nieuwe technologie, andere actoren. Tijdens het project verandert het dynamiek, Uncertainty heeft te maken met proven technology. Agile zegt eigenlijk niet hoe je moet omgaan met veranderingen in de context, maar juist Agile, dus vlug op inspelen. Iets meteen inbouwen, morgen weer veranderen. Heel erg gericht dus op dynamiek.

Tekortkomingen: Voorbeeld project : soort portal waar overheden en bedrijven samenwerken,. En het begon met Agile, met leuke prototypes. Er werden ontwerpen gemaakt die niet echt gewaardeerd werden. Dat maakte interactie met agile developers op gegeven moment heel moeilijk, omdat de verwachtingen teveel uit elkaar liepen. Beter van tevoren nadenken en niet meteen Agile beginnen. Typisch probleem

Sterk: flexibel, heel snel inspelen op wensen, Het systeem krijgt snel vorm, iets tastbaars. Heel vlug, het werkt vanaf het begin, quick wins, grote user betrokkenheid.

Combinaties:

Ik denk dat Agile een klein element is van het totaal geheel. Bij kleine ondernemingen kan het wel werken als zij altijd agile werken, maar bij grote projecten kan het niet: hier gaat het samen met collaboration engineering. Maar dit wordt impliciet gedaan. Je gebruikt wel een development environment, waar je kunt samenwerken. Dit is eigenlijk al een group tool.

Appendix 9 Interview transcript Systems Engineering

Respondent: Dr. Alexander de Haan, Docent sectie beleidsanalyse.

Onderzoeksveld: Research gedaan in de luchtvaart sector en op dit moment meer bezig met beleid analytische, systeem denken, systems engineering of nieuwe methoden van effectief werken: Basisgedachtengoed in actoren en factoren denken, en niet in oplossingen maar in oplossingsruimten en het problematiek expliciet beschrijven. Het expliciteren en conceptualiseren van een probleem dient als basis om verder te gaan, voor de eerste jaars (zowel bsc+msc).. Onderzoeksveld ligt dus heel dicht bij het onderwijsgebied, systems engineering.

Achtergrond: L&R uit TU Delft, Psychologie in Leiden.

Project: Kan van alles zijn. Het start wanneer iemand zegt dat hij/zij niet tevreden is, dan pas kan je een project hebben. Hierna volgt de conceptualisatiebeschrijving (wie zijn er allemaal bij betrokken, wat kan je eraan doen, welke mogelijkheden zijn er om te meten of we bereiken wat we willen, welke mogelijkheid is er om te kijken of we gewenste doelen halen). Hierna komt project planning bij kijken, monitoring van resources etc., uitvoering en evaluatie (is er bereikt wat men wilde). Het moet dus ergens starten met een ontevredenheid, kan van ontwikkeling van vliegtuigvleugel zijn, of ontwikkelen van beleidstraject, maar kan ook bijv. zijn een research project.

Project Complexiteit: Als je bijv. op Utrecht centraal wissel a en wissel b moet aanleggen, moet je niet een heel rapport gaan schrijven met functionele eisen van hoe je de trein van de ene naar de andere rail brengt, maar het is een keuze tussen twee producten. Dus heb er geen standaard aanpak voor, maar wel een overweging van wat de eisen van het project zijn en wat voor aanpak past er dan bij. Bijvoorbeeld: veiligheidscultuur in de luchtvaart is van een hele andere dimensie als project dan twee wissels vervangen op Utrecht centraal.

Systems thinking/Systems Engineering:

Systems thinking is echt een denkmethode van hoe je alles samen moet krijgen en hoe je daar grip op krijgt. Systems engineering: is als je vanuit systems thinking het probleem benadert, als je vindt dat het probleem zo complex is dan is systems engineering een handige methode om ervoor te zorgen dat alle actoren, als ze met elkaar communiceren dat ze het op zo een manier doen dat ze er verder mee kunnen komen. Want de een specificceert wat de andere moet aanleveren en een andere het product van een ander wilt zien om iets toe te voegen eraan. Dit is een manier om alle verander management bij te houden en eisen die veranderen in de loop van de tijd. Dat kan je allemaal mooi bij houden met SE. Systems Engineering is meer een methode en Systems

thinking is meer een methode voor die manier hoe je naar de wereld kijkt (bril). Je past niet alle twee noodzakelijkerwijs gezamenlijk toe.

Als je kijkt naar de drie dimensies zou je met systems thinking geen enkele element mogen weglaten uit het raamwerk. Bij systems thinking moet je met allemaal rekening houden. Onzekerheid je per definitie al als je met mensen werkt. Bij relaties denk ik ook aan fysieke relaties tussen energie stromen, of natuurkundige wetten. Systeem denken is flauw, want het scoort overal op, het houdt met alles en overal rekening mee, Systems Engineering is wat concreter:

Actor dimensie:

SE brengt wel de actoren in kaart (actorenkaart), waarin ook duidelijk wordt hoe ze van elkaar afhankelijk zijn en de project manager ook weet hoe het zit en wie van elkaar afhankelijk is. Dynamiek en onzekerheid neem je indien nodig wel mee. Als je als project manager kijkt naar je systeem en weet je dat je een project hebt over 30 jaar gesmeerd (bijv. deltawerken) dan zullen actoren veranderen, dan weet je dat op voorhand, maar je kan dit niet op voorhand definiëren. Je komt het tegen in het systeem en je past het aan, maar is onduidelijk of systems engineering hier op gericht is, terwijl je met bijv. een context scenario zoals je dat bij beleidsanalyse doet al op voorhand gaat verkennen van wat mogelijke toekomsten zou ik allemaal zien en analyseren. Met SE kan dit eigenlijk niet. SE is more down to earth dan ST. In eerste instantie richt het zich tot actoren en relaties en in context nog wel iets meer, ook wel de dynamica ervan en qua tools is het juist erop gericht om onzekerheid juist te reduceren, doordat je steeds alle veranderingen steeds vastlegt in een retrievable systeem, waarbij je kan bijhouden hoe de eisen van een actor kan veranderen. En omdat het wordt vastgelegd kan je elk moment terug grijpen erop en zeggen dat je bijv. een miljoen had afgesproken, maar de eisen veranderden en werd het 2 miljoen. Doordat er dynamiek is, zorgt SE ervoor dat de uncertainty gemanaged blijft, maar niet dat het weg is. Bij dynamiek en onzekerheid lukt het niet om alleen met SE in kaart te brengen, maar andere approaches gebruiken wordt noodzakelijk.

Support dimensie:

Sommigen vinden dat dit niet een dimensie is om de complexiteit in kaart te brengen of te managen, maar juist om de complexiteit te reduceren. Bij SE reduceer je eigenlijk discussies en onwetendheid tussen actoren. Het is niet zozeer onzekerheidsreductie, maar meer het werkbaar maken van een bos spaghetti, waar alles aan mekaar kleeft. SE is op zich ook een support tool. Verder ziet hij geen probleem om een extra neer te zetten. Je hebt toch altijd een actor dimensie, een context dimensie en een dimensie als ik wat er mee wilt doen of veranderen, hoe doe ik dat dan. Dit zijn de 3 elementen van een project. Tool bepaalt ook heel erg hoe je naar de wereld kijkt. Als je alleen al kijkt naar wie de actoren zijn, kijk je in feite naar wat voor type tools er van te pas komen. Vindt er een prima bril, om zo naar een project te kijken.

De specifieke tools zijn dan causale diagrammen, systeemdiagrammen, actorenkaarten. Maar het zijn allemaal expliciete technieken die beschrijven bijv. wat jou rol is in het project en hoe je dat gaat uitvoeren en hoe er gecontroleerd gaat worden of je dat voor mekaar hebt en gekregen en wat de consequenties zijn als je het niet doet. SE biedt bijna een projectmanagement achtige structuur waarmee je op systeemniveau dat allemaal kan vastleggen (bijv. in de bouwwereld zijn er veel softwarepakketen ontwikkeld). In het boek van hem, inleiding TB, is het gedachtengoed beschreven waarop elke SE techniek is gebaseerd. Vaak laten bedrijven ook hun eigen tool ontwikkelen. SE is bij de overheid verplicht gesteld, een soort out of the box denken maar wel vanuit het TB bril.

Context dimensie:

waarin je zegt dat SE heel sterk is: Hoe zou je dynamiek in de context dimensie willen definiëren?

Je zegt bijv., we hebben een project dat over 10 a 20 jr loopt en uiteindelijk is het idee om een railverbinding te maken tussen twee steden. Maar daartussen in kunnen allerlei ontwikkelingen plaatsvinden, dus kunnen we niet nu al definiëren wat de precieze specificaties zijn van het hele bestek (Taluds, rails etc, of aantal sporten Kvolts), afhankelijk van factoren als reistijden enz. Denk dat SE net zo goed in staat om daarin in te spelen door bijv. keuzemomenten in te bouwen door elk 2 jaar waarbij je evaluatie doet van wat je allemaal gedaan hebt en of in het licht is van de ontwikkelingen. SE is hierin een heel expliciete methode en in staat om dit goed te doen. Mijn idee over dynamiek en onzekerheden is eigenlijk scenario denken. Actor dimensie kan ook flexibel zijn, want er kunnen nieuwe belangen komen of nieuwe relaties ontstaan. SE geeft wel de mogelijkheid om dat wel te managen, maar niet expliciet naar op zoek is. Terwijl het bij context het voortdurend op zoek is naar welke omgeving er wordt geopereerd en hoe verandert de omgeving en daar ook tools voor biedt. Bijv. context scenario's. Of dat je wijzigingen voortdurend kan bijhouden, er zijn technieken daarvoor.

Met SE zou je bijvoorbeeld de context wel goed kunnen specificeren als toevoeging van project management. SE is heel sterk in context, omdat SE wel degelijk context scenario's maakt, als je bijv. een spoorlijn wilt aanleggen tussen Groningen en Maastricht en je hebt er 10 jr. de tijd voor, kan er in die 10 jr. van alles gebeuren. Bijv. politieke veranderingen, veranderende vraag naar vervoer, en daar maken we een aantal scenario's van en gaandeweg de rit nemen we meer detail besluiten (context scenario's uit inleiding TB).

Tekortkomingen:

Absoluut, 1 ding is zeker een nadeel en dat is het van de overheid moet. Dit maakt het dus verplicht om bijvoorbeeld voor simpele projecten waarbij je moet kiezen tussen A en B, een hele specificatie schema van te maken. Er wordt dus ongelooflijk veel extra werk verricht, wat helemaal niet nodig is. Er wordt ook nergens gezegd in welke situatie het gebruikt moet worden, welke er zeker zou moeten zijn.

Grote tekortkoming van de methodiek is dat het noord europees, noord amerikaans of australisch is. Maar voor een italiaanse promovenda die het maar niks vond. Het is gewoon typisch positivistisch, van we knippen de wereld op in stukjes en plakken het weer bij elkaar. Het is wel handig voor bruggenbouwers en vliegtuigbouwers, maar

vaak is het ook niet. Maar het houdt helemaal geen rekening met hoe mensen met elkaar omgaan, het systeem houdt geen rekening met andere factoren die ook van waarde zijn. Je kan geen grote infrastructurele projecten anders doen, want dan wordt het een chaos. Maar je kan het niet domweg toepassen, processen moet begeleid worden.

Combinaties suggesties:

Methodiek van systems engineering houdt geen rekening met de complexiteit van mensen, maar alleen de complexiteit dusdanig behapbaar/reducen dat je een werkbare situatie krijgt. In principe ben je in staat om te meten en om te kijken of er voortgang geboekt wordt. Of je nou met de methodiek kan nadenken over de energievoorziening van Europa is nog maar heel erg de vraag. Daarmee moet je wel stappen naar andere methodieken, bijv. het proces management, waarbij je wel moet oppassen dat je niet zegt dat alles proces is. De rijkheid is menselijk gedrag wordt daar wel geprobeerd duidelijk te maken, wat SE niet voor mekaar krijgt. Heb geen idee hoe je die twee combineert, je hebt wel allebei nodig.

Bij het combineren van proces management met collaboration engineering gaat het wat makkelijker, omdat ze dezelfde basis hebben. Maar bij SE is het ook wat moeilijker omdat het wat verder van mensen af staat, het maakt het analytisch en exact. Intermenselijke contacten zijn helemaal niet goed verwerkt in systems engineering. Dit moet je eerst hebben en dan zou je de weerslag hiervan wat explicieter kunnen opstellen.

Waarom gekozen voor SE:

Vond in systeemdenken een middenweg, die negeert niet allerlei complexiteit en voorspelbaarheid van mensen. Systeem denken en system engineering heeft wel iets in zich van er iets iemand of een groep ontevreden. Dan gaan we daar wat aandoen en zorgen ervoor dat er ook een bruikbaar product ligt. Als er een nieuwe theorie komt die nog meer rekening zou houden met de rijkheid en creativiteit van mensen dan zou ik de eerste zijn die zou overstappen op die methode. Heb nu al de neiging om binnen SE coaching elementen in te brengen, omdat SE wel erg kil is.

Appendix 10 Interview transcript System Dynamics

Respondent: Dr.ir. Els van Daalen, Hoofddocent beleidsanalyse.

Onderzoeksveld: methoden en technieken om beleidsmaker te ondersteunen

Achtergrond: Werktuigbouwkunde

Definitie Project: Zou kunnen zijn een vraagstuk, waar men op zoek is naar een aantal maatregelen om een project aan te pakken en het moet binnen een bepaalde tijd en scope. Bij systeem dynamica projecten gaat het om het inzichtelijk maken van het systeem en kijken of je aangrijpingspunten te vinden voor beleid binnen dat systeem.

Complexiteit van het project: In systeem dynamica is dit het omvang van het probleem dat je moet oplossen (hoeveel levels en flows) en de onzekerheden die er in het systeem zitten. Het probleem is om juiste data te vinden, want alles moet gekwantificeerd worden (hiervoor veel data nodig en het vinden hiervan is moeilijk)

Essentie Systeem Dynamica: Om een systeem te representeren op een dusdanige manier dat je inzicht in het systeem kunt krijgen en aangrijpingspunten voor beleid kunt vinden. Inzicht in complexiteit: Dit is juist 1 van de bedoelingen van SD, maar ook om die complexiteit dusdanig behapbaar te maken om zodoende tot een oplossing te kunnen komen.

Actor Dimensie:

Je kan de actoren wel in kaart brengen, maar niet in de losse actoren. Een actor representatie gaat het meer wat voor maatregelen ze kunnen doen of hoeveel resources. Wat ze doen kan je wel modelleren, maar wat hun belangen zijn is wat moeilijker. Bijv. Hoeveel boetes geeft de politie per maand/per week. Belangen moet je vertalen naar activiteiten in het systeem. Het zijn rationeel handelende actoren. Relaties kun je eigenlijk niet in kaart brengen. Dynamiek: De betrokkenheid van andere actoren zou je moeten doorvertalen in belangen of kijken naar veranderende belangen. Onzekerheden: je kan op 2 manieren met onzekerheden omgaan: je weet niet zeker hoe een systeem in elkaar zit en je modelleert het op verschillende manieren om te kijken wat voor consequenties dat heeft.

Support Dimensie:

Er zijn 3 of 4 verschillende soorten software, allemaal zelfde essentie. Er zijn 2 fases: kwalitatieve (vertalen naar diagrammen) en kwantitatieve (stocks en flows). Dit hangt af van je project af, soms heb je genoeg aan kwalitatieve model. Bij een volledig SD project gebruik je zowel kwantitatief als kwalitatief. SD zegt niks over het aantal elementen. Dit is een keuze, want je kijkt vanuit de context want er zijn zeker elementen en relaties. Wat voor soort relaties en types die staan wel vast. Die zijn op een bepaalde manier met elkaar gerelateerd. Als je dat goed doet, zie je een bepaalde dynamiek. Er is wel gedefinieerd : welke elementen, maar niet hoeveel. SD over de

Support dimensie is interessant omdat het inzicht biedt in welke softwarepakketten je kan gebruiken. Het hele punt van SD is dat als je de elementen en de relaties in kaart brengt, dat daar automatisch de dynamiek van het systeem uit volgt. Dus de dynamiek is niet anders dan de combinatie van elementen en relaties. De structuur van het systeem (elementen en relaties) bepaalt het gedrag (dynamiek) van het systeem. Dit is 1 van de uitgangspunten van SD. Als je de elementen en relaties hebt, komt dan uit wat het dynamiek van het systeem is. Onzekerheden gaan bij SD via gevoeligheidsanalyses, maar SD kan niet zo goed omgaan met onzekerheden. Je kan een model met onzekerheidsfactor modelleren en eentje zonder, (gevoeligheidsanalyse) en kijk je dan wat het verschil is. Het is oorspronkelijk niet voor bedoeld voor systemen met onzekerheden.

Context Dimensie:

Je kan met SD de fysieke elementen in kaart brengen, zodra je het in levels en flows kan beschrijven. Dus als er voldoende elementen zijn om het in levels en flows te beschrijven. Inzicht krijgen doe je door een model te maken en als je wilt weten hoe het werkt, doe je het door verschillende maatregelen door te werken en te kijken wat er dan gebeurt. Dynamiek: het draait allemaal om het inzichtelijk maken van dynamiek. Daarom heet het ook SD. Het is bedoeld om de dynamiek in kaart te brengen door je elementen en relaties te modelleren. Onzekerheden: gevoeligheidsanalyse. Je gaat met de actor en context dimensie in feite op dezelfde manier om, je probeert alles te vertalen (maakt niet uit of het een mens of ding is) naar levels en flows.

Tekortkomingen: Als je kijkt naar de complexiteit dan kun je zeggen dat omgaan met onzekerheden is niet zo goed, maar daar worden er dingen op verzonnen. Oorspronkelijk was SD zo gemaakt dat de analist het model maakt en verteld aan de beleidsmaker dat hij dit en dat heeft doorberekend...en dan wordt het opgelost, maar dit kreeg weinig commitment van de probleemeigenaar. Wat je nu hebt is group model building, dat je met een hele groep bij elkaar zit. Dus 1 van de tekortkomingen van de methode op zich is dat het niet zo goed kan omgaan met conflicterende belangen. Daar moet er iets anders op verzonnen worden, misschien door die mensen een gezamenlijk model te laten bouwen. Dit hebben ze gedaan door CE te combineren met SD.

Sterke punten: Omdat de notatie vrij eenvoudig is, is het goed te begrijpen voor mensen die weinig ervaring hebben met modelleren. Je kan vrij snel verschillende beleidsmaatregelen doorrekenen die je in de praktijk niet zou kunnen doen.

Mogelijke combinaties: Meestal wordt SD vergeleken met discreet en agent based modelling. Het verschil met discreet is dat je bij SD op een wat hoger abstractie plaatvind en langere tijdschalen en bij AB modelling kijk je naar hoe gedragen individuele personen of actoren zich (is ook een manier van discreet modelleren). Dit wordt eigenlijk in het begin al vastgesteld. Combinatie met CE voor het beter omgaan met conflicterende belangen tussen actoren.

